

UNIVERSITY OF
BIRMINGHAM

MULTI-SCALE ENERGY MANAGEMENT FOR MULTI-MODE HYBRID VEHICLES USING MULTI-AGENT REINFORCEMENT LEARNING

BY
MIN HUA

A THESIS SUBMITTED TO
THE COLLEGE OF ENGINEERING AND PHYSICAL SCIENCE OF
THE UNIVERSITY OF BIRMINGHAM

DOCTOR OF PHILOSOPHY

Department of Mechanical Engineering
School of Engineering
College of Engineering and Physical Science of
The University of Birmingham

January 2025

UNIVERSITY OF
BIRMINGHAM

University of Birmingham Research Archive

e-thesis repository

This unpublished thesis/dissertation is copyright of the author and/or third parties. The intellectual property rights of the author or third parties in respect of this work are as defined by The Copyright Designs and Patents Act 1988 or as modified by any successor legislation.

Any use made of information contained in this thesis/dissertation must be in accordance with that legislation and must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the permission of the copyright holder.

Abstract

The automotive industry is undergoing a transformative shift driven by advancements in connectivity, intelligence, and electrification. While the rise of connected and autonomous vehicles (CAVs) and plug-in hybrid electric vehicles (PHEVs) presents new opportunities, it also introduces challenges including maintaining efficient inter-vehicle communication in dynamic traffic environments, coordinating energy flow among multiple propulsion sources under varying operating conditions, and ensuring the control effectiveness and practical applicability of hybrid powertrains. This research addresses these challenges by developing a reinforcement learning (RL) framework that integrates macro-level cooperative adaptive cruise control (CACC) and micro-level energy management strategies (EMSs) for multi-mode PHEVs, enabling intelligent and scalable energy optimization.

At the macro (traffic) level, CACC is critical for ensuring the stability and energy efficiency of CAV platoons. A fully-decentralized multi-agent reinforcement learning (MARL) framework is introduced to enhance communication efficiency by incorporating the quantified stochastic gradient descent (QSGD) and a binary differential consensus (BDC) method. Compared to existing state-of-the-art MARL approaches, BDC-MARL achieves a 5.8% improvement in energy efficiency, maintains an average velocity of 15.26 m/s, and ensures consistent inter-vehicle spacing (IVS) of 20.76 m. Validation using real-world OpenACC data demonstrates its ability to balance platoon stability with energy optimization effectively.

At the micro (vehicle) level, multi-mode PHEVs face the challenge of managing complex multi-input and multi-output (MIMO) energy flows, often leading to oversimplified and suboptimal solutions. To overcome this, a novel MARL-based framework is proposed by incorporating a hand-shaking strategy with a relevance ratio to improve coordination among energy control agents. By fine-tuning parameters such as learning rates and network configurations, this system achieves up to 4% energy savings compared to single-agent systems and up to 23.54% compared to traditional rule-based methods, validated in real-world driving conditions.

From a theoretical perspective, this research further refines RL methods in EMSs for PHEVs by addressing overestimation bias. An ensemble strategy with in-target minimization and an update-to-data (UTD) mechanism reduces the bias from 1.18 to 5.2×10^{-4} while achieving up to 15.59% energy savings compared to conventional methods. Through theoretical analysis, bias visualization, and extensive testing on software-in-the-loop (SiL) and hardware-in-the-loop (HiL) platforms under real-world driving conditions, the practical applicability in improving energy savings is validated.

By integrating macro and micro perspectives and aligning applications with theoretical analyses, this research provides efficient solutions for improving energy efficiency in CACC and PHEVs, contributing to the development of a sustainable and intelligent future in mobility.

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my supervisors, Professor Hongming Xu and Professor Quan Zhou, whose wisdom and unwavering faith in my potential have illuminated every step of this doctoral journey. Their intellectual rigor, patient guidance, and compassionate mentorship transcended academic instruction—they taught me to think boldly, persevere through uncertainty, and embrace the beauty of discovery. This thesis stands as a testament to their transformative influence, not only on my work but on my identity as a scholar.

To my extraordinary colleagues, Dr. Ji Li, Dr. Gengxin Zhang, Dr. Yinglong He, Dr. Zeyu Sun, Dr. Shuai Bin, Dr. Cetengfei Zhang, Mr. Fanggang Zhang, Mr. Xu He, Mr. Chengqing Wen, Mr. Hongyu Sun, Mr. Abdul Abdillah, Mr. Zhaofeng Gao, Miss Shiyi Yin, and Miss Shuo Wang, thank you for being both collaborators and kindred spirits. Your brilliance sparked countless "eureka" moments in late-night lab sessions, and your camaraderie turned challenges into shared triumphs. This journey would have been far lonelier without your laughter, insights, and the quiet solidarity that only fellow travelers on this path can understand. A special note of thanks goes to Dr. Lun Hua and Dr. Mike Bassett for their invaluable support to the industrial projects that shaped this work.

I am profoundly indebted to the visiting scholars Dr. Liping Xie, Dr. Zhong Ren, Dr. Jinhai Wang, Mr. Kun Jiang, Miss Yiqi Jia, and Miss Dandan Wu—for broadening my horizons with their global perspectives. Your generosity in sharing expertise, questioning assumptions, and nurturing interdisciplinary dialogue breathed new life into my research.

To the unsung heroes of the School of Engineering IT team: your tireless technical support was the invisible scaffolding holding this work together. Whether rescuing data from digital oblivion or optimizing code at midnight, your professionalism turned potential crises into mere footnotes. A special tribute to Dr. George Brinklow, whose editorial genius and scholarly

generosity elevated this thesis beyond its raw form. Your meticulous feedback not only honed my arguments with surgical precision but also infused clarity into every chapter.

To my cherished friends, Dr. Wei Liu, Dr. Dong Chen, and Dr. Xinda Qi, who stood as my emotional compass through this odyssey. To Dr. Wei Liu, my academic confidant and steadfast ally: your belief in me never wavered, even when mine faltered.

Finally, to my family, my parents, whose sacrifices crossed oceans; my brother, who became my cheerleaders across time zones; and my nephews and nieces, whose innocent smiles grounded me in joy, this thesis is your victory as much as mine. You weathered my absences, celebrated small wins with disproportionate enthusiasm, and loved me through every moment of doubt. Your faith was the foundation upon which every equation, every simulation, and every word here rests.

To all who walked this path with me: these pages carry not just ideas, but fragments of your kindness, wisdom, and humanity. For that, I am eternally grateful.

Contents

Abstract iii

Acknowledgements v

List of Publications xix

Nomenclature xxii

1 Introduction **1**

1.1 **Background** 1

1.1.1 The connectivity of vehicles 1

1.1.2 The intelligence of vehicles 3

1.1.3 The electrification of vehicles 6

1.1.4 Current development of powertrains for electrified vehicles 8

1.2 **Motivation** 13

1.3 **Aims and Objectives** 14

1.4 **Thesis Outline** 15

2 Literature Review **18**

2.1 **Cooperative adaptive cruise control of CAVs** 18

2.1.1 General problem formulation 19

2.1.2 Control approches 20

2.1.3 Challenges 28

2.2 **Deep reinforcement learning of EMSs for electrified vehicles** 29

2.2.1 General Problem Formulation 30

2.2.2 Applications of DRL in EMSs 31

2.2.3 Challenges 39

2.3 **Q learning of EMSs for electrified vehicles** 40

2.3.1 General problem formulation in Q learning 40

2.3.2 Applications of Q-learning in EMSs . . . 41
2.3.3 Challenges in Q-Learning for EMSs . . . 49
2.4 **Summary and research gap** . . . 50

3 Methodology and Facility **52**
3.1 **Methodologies** . . . 52
3.1.1 Research target . . . 52
3.1.2 Preliminaries of RL . . . 53
3.1.3 Preliminaries of MARL . . . 56
3.1.4 Training and execution strategies in MARL . . . 57
3.1.5 MARL algorithm variants . . . 59
3.2 **Research facilities and resources** . . . 64
3.2.1 ETAS HiL testing platform . . . 64
3.2.2 Speedgoat HiL testing platform . . . 65
3.2.3 Driving cycles and OpenACC data . . . 66
3.3 **Multi-mode PHEV modeling and modes** . . . 75
3.3.1 Vehicle model . . . 76
3.3.2 Modeling of main components . . . 78
3.3.3 Working modes . . . 80
3.4 **Chapter Summary** . . . 83

4 Communication-Efficient MARL for Platoon Stability and Energy-efficiency Co-optimization in Cooperative Adaptive Cruise Control of CAVs **84**
4.1 **Architecture of MARL-based CACC for CAVs and the Optimization Problem** . . . 85
4.1.1 Decentralized MARL framework for CACC of CAV . . . 85
4.1.2 Co-optimization of platoon stability and energy efficiency . . . 88
4.2 **Communication-efficient MARL for CACC** . . . 90
4.2.1 Decentralized Multi-agent MDP problem . . . 90
4.2.2 Quantified communicative information-sharing update design . . . 92
4.3 **Experiment setups** . . . 93

4.3.1 General setups for the learning agents . . . 93
4.3.2 Real-world driving scenario extraction . . . 94
4.4 **Results and discussion** . . . 94
4.4.1 Co-optimization Performance with State-of-the-art MARL . . . 94
4.4.2 Communication Efficiency of MARL on CACC . . . 97
4.4.3 Impact of Platooning Size on the Scalability on CACC . . . 100
4.4.4 Co-optimization Performance Validation with Real-World Data . . . 102
4.5 **Chapter summary** . . . 103

5 Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning 106
5.1 **MIMO control of a multi-mode PHEV** . . . 106
5.1.1 The energy flow model . . . 107
5.1.2 The multiple-input and multiple-output (MIMO) energy management controller . . . 110
5.2 **Hand-shaking multi-agent learning for MIMO control** . . . 111
5.2.1 Multi-agent learning with the DDPG algorithm . . . 111
5.2.2 DDPG-based learning process . . . 112
5.2.3 States and actions . . . 115
5.2.4 Reward functions and hand-shaking design . . . 116
5.3 **Results and Discussion** . . . 118
5.3.1 Impacts of learning agent design on MIMO control performance . . . 119
5.3.2 Hand-shaking of the multi-agent systems with different R_{rel} value . . . 124
5.3.3 Performance comparison with different methods . . . 127
5.4 **Chapter summary** . . . 130

6 Efficient Energy Management of Plug-in Hybrid Electric Vehicles through Ensemble with In-target Minimization Q-learning 132
6.1 **The Hybrid Electric Vehicle System** . . . 133
6.1.1 Vehicle modelling . . . 134
6.1.2 RL-based energy management . . . 136

6.2 **Ensemble Q-learning with in-target minimization and UTD ratio** . . 137
6.2.1 Single Q-learning . . . 137
6.2.2 Ensemble Q-learning design . . . 138
6.2.3 Ensemble learning with in-target minimization and UTD . . . 139
6.2.4 Normalized Bias Estimation . . . 142
6.3 **Theoretical analysis** . . . 143
6.3.1 Preliminaries . . . 143
6.3.2 Convergence analysis . . . 143
6.3.3 **UTD impact analysis** . . . 145
6.3.4 Bias control . . . 146
6.3.5 Sample complexity analysis . . . 146
6.3.6 Stability Analysis . . . 149
6.4 **Results and discussion** . . . 150
6.4.1 Validation setup . . . 150
6.4.2 Comparative analysis with the single and double Q learning systems . 152
6.4.3 Improvement in learning performance with the ensemble with in-target minimization and UTD ratio . . . 154
6.4.4 Real-time performance of DQM-UTD algorithm . . . 157
6.4.5 Robustness testing of DQM-UTD algorithm . . . 159
6.5 **Chapter summary** . . . 162

7 Conclusion and future work **164**
7.1 **Conclusion** . . . 164
7.2 **Research Impact** . . . 166
7.3 **Future work** . . . 167

Reference **170**

List of Figures

1.1 Global connected vehicles market size by communication type, 2023-2033 [5]. 2

1.2 Four main factors that accelerate the need for the intelligence of vehicles [23]. 5

1.3 Vehicle intelligence system market growth [23]. 5

1.4 Evolution of vehicle powertrains from conventional ICE to fully electric vehicles [26]. 6

1.5 Global electrified car stock trends, 2010-2023 (passenger cars only) [29]. . . 7

1.6 Configuration of series HEVs [41]. 9

1.7 Configuration of parallel HEVs [41]. 10

1.8 Configurations of subtypes P0 to P4 in parallel HEVs [41]. 11

1.9 Configurations of power-split HEVs [41]. 12

1.10 Configurations of multi-mode HEVs [41]. 12

1.11 Illustration of the macro-micro energy management system control framework. 15

2.1 Cooperative adaptive cruise control scenario. 19

3.1 The overview framework of the whole PhD research. 53

3.2 Illustration of RL. 54

3.3 Illustration of MARL. 57

3.4 Illustration of CTDE in MARL. 58

3.5 Illustration of DTDE in MARL. 59

3.6 ETAS HiL testing platform. 65

3.7 Speedgoat HiL testing platform. 66

3.8 The standard driving cycles: (a) NYCC driving cycle, (b) Manhattan driving cycle, (c) UDDS driving cycle, (d) Artemis Rural driving cycle. 67

3.9 The standard driving cycles are (a) RTS95 driving cycle, (b) FTP75 driving cycle, (c) NEDC driving cycle, (d) WLTC-Class3 driving cycle, and (e) CLTC-P driving cycle. 69
3.10 The standard driving cycles are (a) US06 driving cycle, (b) HWFET driving cycle, and (c) IM240 driving cycle. 70
3.11 APTA driving cycle. 70
3.12 Training driving cycle. 72
3.13 The combined driving cycles for the validation are (a) Driving Cycle I, (b) Driving Cycle II, (c) Driving Cycle III, and (d) Driving Cycle IV. 73
3.14 Zalazone Test Platform. 74
3.15 Dataset for dynamic_part1. 75
3.16 Dataset for dynamic_part2. 75
3.17 Dataset for dynamic_part9. 76
3.18 Dynamics within 60s: (a) Specific datasets from dynamic_part1; (b) Specific datasets from dynamic_part2; (c) Specific datasets from dynamic_part9. . . 76
3.19 Backward-facing vehicle model 78
3.20 BSFC map of engine. 79
3.21 Efficiency map of the electric motor. 80

4.1 The multi-agent system for CAVs 86
4.2 Contour plot of the fitted energy consumption model with respect to vehicle velocity and acceleration. 90
4.3 The comparison with different methods 95
4.4 The performance comparison of all vehicles with typical MARL algorithms. . 96
4.5 The comparison with three different sharing methods 97
4.6 The performance process of different sharing methods. 98
4.7 The statistical comparison of different communicative methods 99
4.8 The statistical comparison of different communicative bits 100
4.9 The comparison with different sizes 101
4.10 The comparison with different size 101

4.11 Comparison of CACC strategies across different groups and corresponding OpenACC data. 103

5.1 Configuration of the multi-mode PHEV powertrain 107
5.2 MIMO control architecture 110
5.3 DDPG-based EMS with multi-agent learning 111
5.4 Control policy optimization using the DDPG algorithm 113
5.5 OU noise process under different parameters. 115
5.6 A profile of the learning cycle. 118
5.7 Impact of critic network layers on the control performance. 120
5.8 The results of different learning rates 122
5.9 The average rewards of different policy noise 124
5.10 Learning performance of different ratios (SoC initial value is 0.28) 125
5.11 Learning performance of $a)R_{rel} = 0; b)R_{rel} = 0.2; c)R_{rel} = 0.6$ 126
5.12 Scatter plots of fuel consumption versus final SoC for four control strategies. 130

6.1 The schematic of the studied PHEV 133
6.2 Ensemble learning with in-target minimization and UTD updates in end-to-end EMS. 140
6.3 The driving cycles for proposed strategy validation: (a) Driving Cycle I, (b) Driving Cycle II, (c) Driving Cycle III, (d) Driving Cycle IV 151
6.4 Hardware-in-the-loop testing platform for real-time validation and testing. . . 152
6.5 Comparative analysis of overestimation bias between single and double Q-learning systems 152
6.6 Q-value distributions across different control ratios (a=0.1, 0.5, 0.8, 1.0) in single Q-learning system 153
6.7 Q-value distributions between Q1 and Q2 at control ratio a=0.5 154
6.8 Performance comparison of average reward and system efficiency between single and double Q learning systems. 154
6.9 Average and std of normalized bias under different UTD ratios 155
6.10 Average reward and system efficiency under different UTD ratios 155

6.11 Average and Std of normalized bias when $G = 15$ 157
6.12 Real-time performance comparison in HiL testing platform 158
6.13 Correlation between driving cycle characteristics and DQM-UTD energy savings161

List of Tables

2.1 Comparison of value-based methods in EMSs for typical works 34
2.2 Comparison of actor-critic based methods in EMSs for typical works 37
2.3 Several works in Q-learning Based EMSs 43
2.4 Detailed comparison of enhanced Q-learning methods 46
2.5 Summary of hybrid methods in EMSs 49

3.1 An overview of the primary works for MARL algorithms. 60
3.2 Summary of driving cycle categories 71
3.3 Four combined driving cycle statistics 73

4.1 Reward comparison with different algorithms 96
4.2 Test performance comparison with different algorithms 96
4.3 Extraction of real-world scenarios from OpenACC 102

5.1 Comparison of single-agent and the proposed multi-agent systems 112
5.2 Main hyperparameters and their values. 119
5.3 Comparison of SoC and Fuel consumption 121
5.4 Policy noise combination 123
5.5 Comparison of SOC and Fuel consumption 123
5.6 Comparison of importance levels 125
5.7 Average values and standard deviation. 127
5.8 Learning performance of different methods. (– represents the baseline performance.) 129

6.1 Vehicle specifications of the studied PHEV. 134
6.2 Performance comparison under different UTDs 156
6.3 Statistics analysis of driving cycles 159

6.4 Performance comparison of DQM-UTD and other methods under various driving cycles 160

List of Publications

Journal papers

1. **Min Hua**, Dong Chen, Kun Jiang, Fanggang Zhang, Jinhai Wang, Bo Wang, Quan Zhou, and Hongming Xu. Communication-Efficient MARL for Platoon Stability and Energy-efficiency Co-Optimization in Cooperative Adaptive Cruise Control of CAVs. IEEE Transactions on Vehicular Technology, pp.1-12, 2024. (IF=6.1, JCR Q1)

2. **Min Hua**, Cetengfei Zhang, Fanggang Zhang, Zhi Li, Xiaoli Yu, Hongming Xu, and Quan Zhou. Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning. Applied Energy 348, pp. 121526, 2023. (IF=10.1, JCR Q1)

3. Shuai Bin†, **Min Hua**†, Yanfei Li, Shijin Shuai, Hongming Xu, and Quan Zhou. Optimal Energy Management of Plug-in Hybrid Electric Vehicles Through Ensemble Reinforcement Learning With Exploration-to-Exploitation Ratio Control. IEEE Transactions on Intelligent Vehicles, 2024. (†Equal contribution) (IF=14 , JCR Q1)

4. **Min Hua**, Bin Shuai, Quan Zhou, Jinhai Wang, Yinglong He, and Hongming Xu. Recent Progress in Energy Management of Connected Hybrid Electric Vehicles Using Reinforcement Learning. International Journal of Automotive Manufacturing and Materials, 2023.

5. **Min Hua**, Xinda Qi, Dong Chen, Kun Jiang, Zemin Eitan Liu, Quan Zhou, and Hongming Xu. Multi-Agent Reinforcement Learning for Connected and Automated Vehicles Control: Recent Advancements and Future Prospects. IEEE Transactions on Automation Science and Engineering, 2024. (Accept, IF=5.9, JCR Q1)

6. **Min Hua**, Bin Shuai, Fanggang Zhang, Jinhai Wang, Cetengfei Zhang, Quan Zhou, and Hongming Xu. Efficient Energy Management of Plug-in Hybrid Electric Vehicles

through Random Ensemble with In-target Minimization Q-learning. IEEE Transactions on Transportation Electrification, 2024. (Round 1, IF=7.2, JCR Q1)

7. Kun Jiang, **Min Hua**, Xu He, Lu Dong, Quan Zhou, Hongming Xu, and Changyin Sun. Improving String Stability in Cooperative Adaptive Cruise Control Through Multi-Agent Reinforcement Learning With Potential-Driven Motivation. IEEE Transactions on Artificial Intelligence, 2024.

8. Wang, Jinhai, Changqing Du, Fuwu Yan, **Min Hua**, Xiangyu Gongye, Quan Yuan, Hongming Xu, and Quan Zhou. Bayesian optimization for hyper-parameter tuning of an improved twin delayed deep deterministic policy gradients based energy management strategy for plug-in hybrid electric vehicles. Applied Energy 381 (2025): 125171.

9. Cetengfei Zhang, Quan Zhou, **Min Hua**, Hongming Xu, Mike Bassett, and Fanggang Zhang. Cuboid equivalent consumption minimization strategy for energy management of multi-mode plug-in hybrid vehicles considering diverse time scale objectives. Applied Energy 351, pp. 121901, 2023.

10. Jinhai Wang, Changqing Du, Fuwu Yan, Xuexue Duan, **Min Hua**, Hongming Xu, and Quan Zhou. Energy Management of a Plug-in Hybrid Electric Vehicle Using Bayesian Optimization and Soft Actor-Critic Algorithm. IEEE Transactions on Transportation Electrification, 2024.

11. Fuwu Yan, Jinhai Wang, Changqing Du, and **Min Hua**. Multi-objective energy management strategy for hybrid electric vehicles based on TD3 with non-parametric reward function. Energies 16, no. 1, pp. 74, 2022.

12. Zemin Eitan Liu, Yanfei Li, Quan Zhou, Yong Li, Bin Shuai, Hongming Xu, **Min Hua**, Guikun Tan, and Lubing Xu. Deep reinforcement Learning based Energy Management for Heavy Duty HEV considering Discrete-Continuous Hybrid Action Space. IEEE Transactions on Transportation Electrification, 2024.

13. Zemin Eitan Liu, Yong Li, Quan Zhou, Bin Shuai, **Min Hua**, Hongming Xu, Lubing Xu, Guikun Tan, Yanfei Li. Real-time energy management for HEV combining naturalistic

driving data and deep reinforcement learning with high generalization, Applied Energy, Vol. 377, Part A, 2025.

Conferences

1. **Min Hua**, Zhi Li, and Quan Zhou. Energy Management of Multi-mode Hybrid Electric Vehicles based on Hand-shaking Multi-agent Learning. The 8th Applied Energy Symposium: Low carbon cities and urban energy systems, 2022.

2. **Min Hua**, Quan Zhou, Chongming Wang, Cetengfei Zhang, Bin Shuai, Hongming Xu. Surrogate Modelling for Battery State-of-Charge Estimation in Electric Vehicles Based on Pseudo-2-Dimensional Model and Gradient Boosting Machines. The 13th International Conference on Applied Energy, 2021.

3. Mo, Longfei, **Min Hua**, Hongyu Sun, Hongming Xu, Bin Shuai, and Quan Zhou. Study on the Impacts of Hazardous Behaviors on Autonomous Vehicle Collision Rates Based on Humanoid Scenario Generation in CARLA, 2023 7th CAA International Conference on Vehicular Control and Intelligence (CVCI), pp. 1-6, 2023.

4. Cetengfei Zhang, Quan Zhou, **Min Hua**, Chongming Wang, Hongming Xu. Online Double-layer System Identification Scheme for Battery State-of-Health Prediction. The 13th International Conference on Applied Energy, 2021.

Abbreviations

CACC Cooperative Adaptive Cruise Control

CAV Connected and Autonomous Vehicle

V2V Vehicle-to-Vehicle

V2I Vehicle-to-Infrastructure

V2P Vehicle-to-Pedestrian

V2X Vehicle-to-Everything

4G/5G Fourth/ Fifth Generation of Mobile Networks

OEM Original Equipment Manufacturers

AI Artificial Intelligence

WHO World Health Organization

ADAS Advanced Driver-Assistance System

ICE Internal Combustion Engine **HEV** Hybrid Electric Vehicle

PHEV Plug-In Hybrid Electric Vehicle

BEV Battery Electric Vehicle

EMS Energy Management System

MG Motor Generator

PG Planetary Gear

PSD Power Split Device

RL Reinforcement Learning

DRL Deep Reinforcement Learning

MARL Multi-Agent Reinforcement Learning

MADRL Multi-Agent Deep Reinforcement Learning

MIMO Multi-Input, Multi-Output

SiL Software-in-the-Loop

HiL Hardware-in-the-Loop

QSGD Quantized Stochastic Gradient Descent

BDC Binary Differential Consensus

DDPG Deep Deterministic Policy Gradient

(MISO Multiple-input and Single-output

(MIMO Multiple-input and Multi-output

SoC State-of Charge

UTD Update-to-Data

E2E Exploration-to-Exploitation

OCV Open Circuit Voltage

DP Dynamic Programming

MPC Model Predictive Control

MOOP Multiobjective Optimization

eCACC ecological CACC

NMPC nonlinear MPC

LSTM Long Short-Term Memory

I-CACC Intent-sharing mechanism in CACC

DQN Deep Q Learning

MTCC Multitimescale Control and Communication

ESs Expert Systems

MAMPS Multi-agent Model Predictive Shielding

DMPO Decentralized Model-based Policy Optimization

TD3 Twin Delayed Deep Deterministic Policy Gradient

CommPPO Communication Proximal Policy Optimization

NeurComm Differentiable Communication Protocol

MAHARL Multi-agent Hierarchical Attention RL

GNN Graph Neural Networks

MA-DDQN Multi-Agent Double Deep Q-Network

DPPO Distributed Proximal Policy Optimization

MADDPG Multi-Agent Deep Deterministic Policy Gradient

HDV Human-driven Vehicles

ECMS Equivalent Consumption Minimization Strategy

DPR Driving Pattern Recognitio

PMP Pontryagin's minimum principle

AMPC Adaptive Model Predictive Control

UDDS Urban Dynamometer Driving Schedule

NEDC New European Driving Cycle

DDQN Double Deep Q-learning/ Dueling DQN

AC Actor-Critic

DDRL Double DRL

TL Transfer Learning

A2C Advantage Actor-Critic

A3C Asynchronous Advantage Actor-Critic

PER Prioritized Experience Replay

LSH Locality Sensitive Hashing

SAQ-learning Self-adaptive Q-learning

GA Genetic Algorithm

SDP Stochastic Dynamic Programming

PSO Particle Swarm Optimization

QMOPSO Q-learning Multi-Objective PSO

TPM Transition Probability Matrix

IMN Induced Matrix Norm

HER Hindsight Experience Replay

SAC Soft Actor-Critic

IQL Independent Q-learning

CTDE Centralized training with decentralized execution

DTDE Decentralized training with decentralized execution

VDN Value Decomposition Network

QMIX Monotonic Value Function Factorization for Deep Multi-Agent Reinforcement Learning

QTRAN Q-Learning with Transformed Value Function

WQMIX Weighted QMIX

AVD-Net Attention-Based Value Decomposition Network

IMAC Information Bottleneck Multi-Agent Communication

FHM Feudal Hierarchical Model

HQMIX Hierarchical QMIX

RODE Role-Oriented Decomposition with Action Clustering

HSD Hierarchical Skill Decomposition

HAVEN Hierarchical Agent View for Efficient Navigation

CIR Causal Influence Reward

ICL Individual Causal Learning

LAIES Learning Agent Influence on External States

FD-MARL Causal Factorized Multi-Agent Reinforcement Learning

DVD Deconfounded Value Decomposition

IA2C Independent Advantage Actor-Critic

FPrint Fingerprint

DIAL Differentiable Inter-Agent Learning

CommNet Learning Multiagent Communication with Backpropagation

OpenACC Open Database of car-following experiments involving vehicles with Adaptive Cruise Control (ACC) systems

WAC-MARL MARL with Weight Averaging Consensus

DCEA-MARL MARL with Differential Consensus-enhanced Adjustment

BDC-MARL MARL with Binary Differential Consensus

CHAPTER 1

Introduction

1.1 | Background

The automotive industry is undergoing a profound transformation driven by three major trends: connectivity, intelligence, and electrification [1]. These developments are reshaping not only how vehicles operate individually but also how they interact with each other and their environment, leading to significant improvements in safety, efficiency, and environmental sustainability [2].

1.1.1 The connectivity of vehicles

Vehicle connectivity represents a transformative leap in modern transportation, revolutionizing how vehicles interact with their environment and each other [3, 4]. As illustrated in Fig. 1.1, the global connected vehicles market is projected to grow significantly, from USD 75.2 billion in 2023 to USD 446.6 billion by 2033, reflecting a robust compound annual growth rate (CAGR) of 19.5% [5]. This growth is driven by various communication technologies, including vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), vehicle-to-pedestrian (V2P), and vehicle-to-device (V2D) communications, collectively forming the vehicle-to-everything (V2X) ecosystem [6]. The emergence of 5G technology has been pivotal in enabling connected vehicles, providing faster connectivity and reduced latency crucial for real-time communication and decision-making. This advancement is particularly significant for V2V communication, which accounted for a dominant market share of 41.1% in 2023 [7]. V2V communication plays a critical role in enhancing road safety and traffic management by facilitating real-time data exchange between vehicles. Similarly, V2I communication facilitates interaction between vehicles and road infrastructure, such as traffic lights and signs, optimizing traffic flow and

improving overall transportation efficiency. Moreover, the integration of V2P communications has become increasingly important in urban environments, where it enhances the safety of vulnerable road users through advanced alert systems [8].

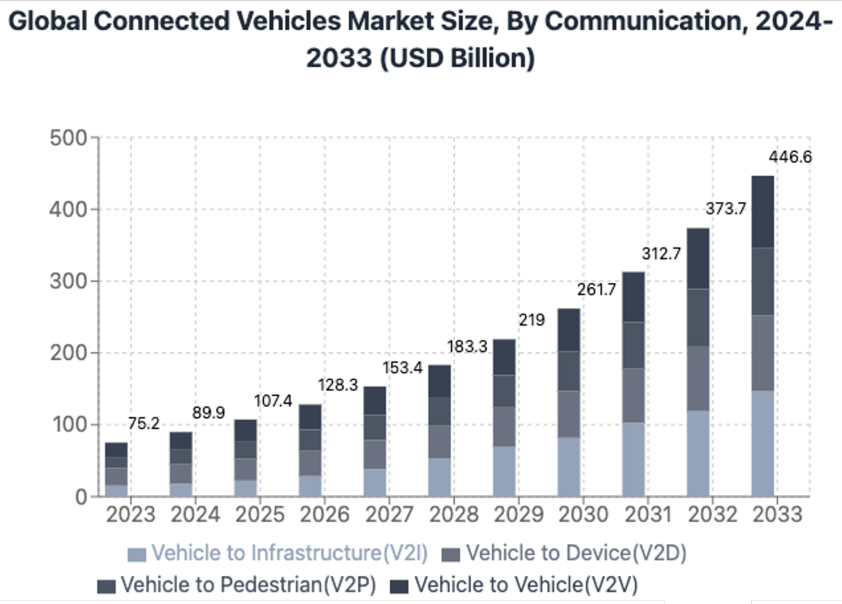


Fig. 1.1. Global connected vehicles market size by communication type, 2023-2033 [5].

Significant investments and policy support highlight the growing importance of vehicle connectivity. Governments worldwide are actively promoting the development of V2X technologies through substantial funding aimed at improving road safety and enhancing mobility [9]. These efforts increasingly emphasize cellular connectivity, which dominated the market with a 40.3% share in 2023, supported by widespread 4G Long-Term Evolution (LTE) networks and the ongoing deployment of 5G technology [10]. The integration of these advanced communication systems enables real-time vehicle diagnostics, automatic software updates, and enhanced infotainment services, fundamentally transforming the driving experience and laying the foundation for the future of connected and autonomous vehicles (CAVs) [11].

Original equipment manufacturers (OEMs) are playing a pivotal role in accelerating the adoption of connected vehicle technologies, accounting for 81.2% of the total market share within the connected vehicle ecosystem [12]. This market dominance reflects the growing integration of advanced V2X communication capabilities directly into vehicle platforms. OEMs are increasingly investing in embedded solutions that provide enhanced cybersecurity, real-time diagnostics, and over-the-air (OTA) updates. These technological advancements are not merely

conceptual but have already been translated into tangible deployments. Leading manufacturers such as Toyota, Audi, and Ford have launched V2X-equipped production vehicles capable of supporting critical safety applications, including intersection movement assist, left-turn assist, and real-time hazard warnings [13].

Simultaneously, the Asia-Pacific region is emerging as a global leader in this domain, capturing 46.6% of the market (USD 35.0 billion in 2023) driven by strategic investments from countries such as China, Japan, and South Korea in both automotive innovation and smart city development [14]. In parallel, smart mobility initiatives in countries like China, the United States, and Germany are actively integrating vehicle-to-infrastructure (V2I) communications into urban traffic management systems [15]. These systems enable adaptive signal control and congestion mitigation based on real-time data exchange. Furthermore, in commercial transportation and logistics, V2V-enabled coordinated platooning is being adopted to reduce fuel consumption, improve aerodynamic efficiency, and optimize highway capacity, marking a significant shift toward cooperative and efficient vehicular operation [16].

Despite these advancements, several challenges persist in the widespread deployment of connected vehicle technologies. First, interoperability remains a major issue, as different OEMs and infrastructure providers adopt heterogeneous communication protocols and standards. Second, ensuring data privacy and cybersecurity is critical, given the sensitive and real-time nature of vehicle communication. Third, maintaining communication reliability and low latency in dense urban or high-speed environments is technically challenging, particularly under variable network conditions. Moreover, the integration of connectivity with control algorithms introduces additional complexity, requiring real-time synchronization and resilience to packet loss or delays.

1.1.2 The intelligence of vehicles

The intelligence of vehicles has emerged as a critical solution to address the mounting challenges in the automotive field [17]. With rapid urbanization and growing vehicle populations, cities worldwide face severe issues including traffic congestion, accidents, parking difficulties, and environmental pollution. For instance, studies reveal that average urban commute times have surged by 20% globally over the past decade, exacerbating both economic and social challenges [18]. Traffic accidents also remain a leading cause of fatalities, with approximately 1.3 million

lives lost annually and millions more injured, as reported by the World Health Organization (WHO) [19]. The environmental impact is equally concerning, with vehicle emissions of SO_2, NO_x, CO, CO_2, and particulate matter contributing significantly to air quality degradation, accounting for nearly 24% of global CO_2 emissions from energy use [20].

Artificial intelligence (AI) has emerged as a transformative technology to address these challenges through various applications. The integration of AI in vehicles enables intelligent traffic management systems that can optimize traffic flow and reduce congestion [21]. Advanced driver assistance systems (ADAS) and autonomous driving technologies help prevent accidents by enhancing vehicle safety capabilities [22]. AI-powered parking systems assist in efficient parking spot allocation and navigation, addressing the time-consuming challenge of finding parking spaces in urban areas. Additionally, AI optimization of vehicle powertrain operations and route planning helps reduce emissions and environmental impact.

As shown in Fig. 1.2, the development of vehicle intelligence is driven by four main factors: (1) The strategic needs of the economy emphasize the importance of integrating advanced manufacturing processes with intelligent systems. (2) The society needs of automobiles reflect the growing demand for efficient and safe transportation solutions in increasingly congested urban environments. (3) The advancement of AI, particularly in areas such as machine learning and computer vision, provides the technological foundation for intelligent vehicles. (4) New business models are emerging that leverage vehicle intelligence to create innovative transportation services and mobility solutions [23]. These intelligent systems utilize multiple AI technologies, including computer vision for environmental perception, deep learning for decision-making, and reinforcement learning for adaptive behavior. As these technologies continue to evolve, they promise to transform urban mobility by creating safer, more efficient, and environmentally sustainable transportation systems [24].

The vehicle intelligence system market demonstrates robust growth trajectories, with projections indicating significant expansion from USD 9.57 billion in 2024 to USD 21.50 billion by 2032, representing a CAGR of 11.2% [23], as shown in Fig. 1.3. This remarkable growth reflects increasing demand for ADAS, rising safety concerns in the automotive sector, and the growing integration of AI-powered systems in modern vehicles. These trends align with the widespread

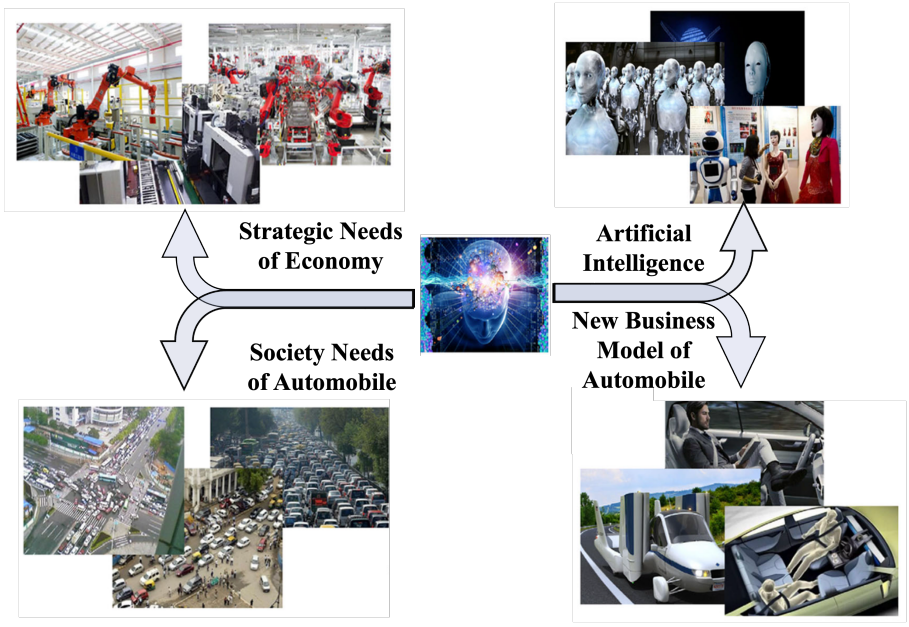


Fig. 1.2. Four main factors that accelerate the need for the intelligence of vehicles [23].

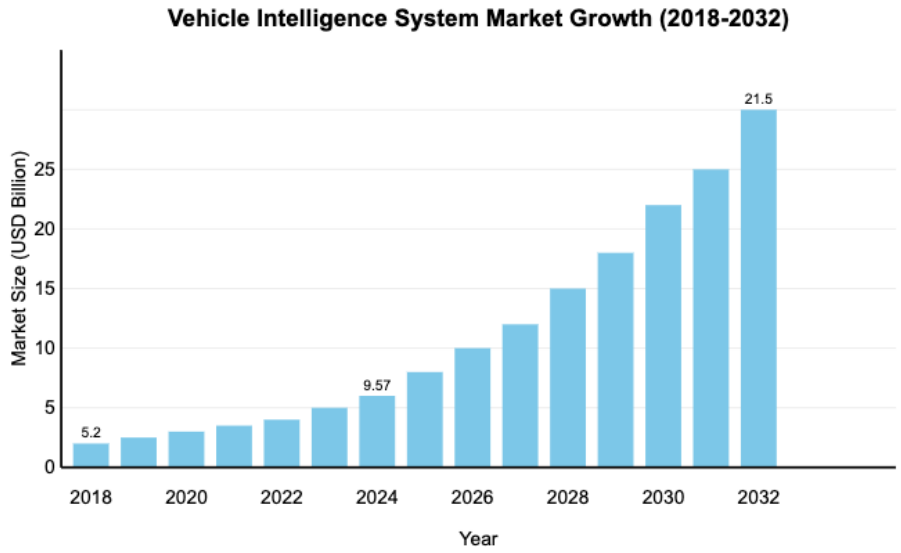


Fig. 1.3. Vehicle intelligence system market growth [23].

adoption of vehicle intelligence technologies and the implementation of stringent safety regulations, driving the industry's rapid development [25].

AI-driven vehicle intelligence is transforming modern transportation by addressing critical challenges such as congestion, safety, and environmental impact. While these advancements have greatly enhanced mobility systems, significant challenges persist in achieving scalability, real-time adaptability, and integration with other technologies to enable safer and more energy-efficient transportation solutions.

1.1.3 The electrification of vehicles

The automotive industry has undergone significant transformation in its drive toward sustainable transportation, marked by the evolution of vehicle powertrains [26], as shown in Fig. 1.4. This evolution began with traditional internal combustion engine (ICE) vehicles powered solely by fossil fuels. As environmental concerns grew, manufacturers developed hybrid electric vehicles (HEVs) that combine fossil fuel engines with electric motors and batteries, improving overall efficiency while reducing emissions [27]. This technology evolved further with plug-in hybrid electric vehicles (PHEVs), which feature larger battery capacity and the ability to charge from external power sources, offering the flexibility to run on electricity or fossil fuel. The latest advancement is the battery electric vehicle (BEV), which charges from renewable energy sources to eliminate fossil fuel dependency by operating exclusively on electric power through batteries and motors [28].

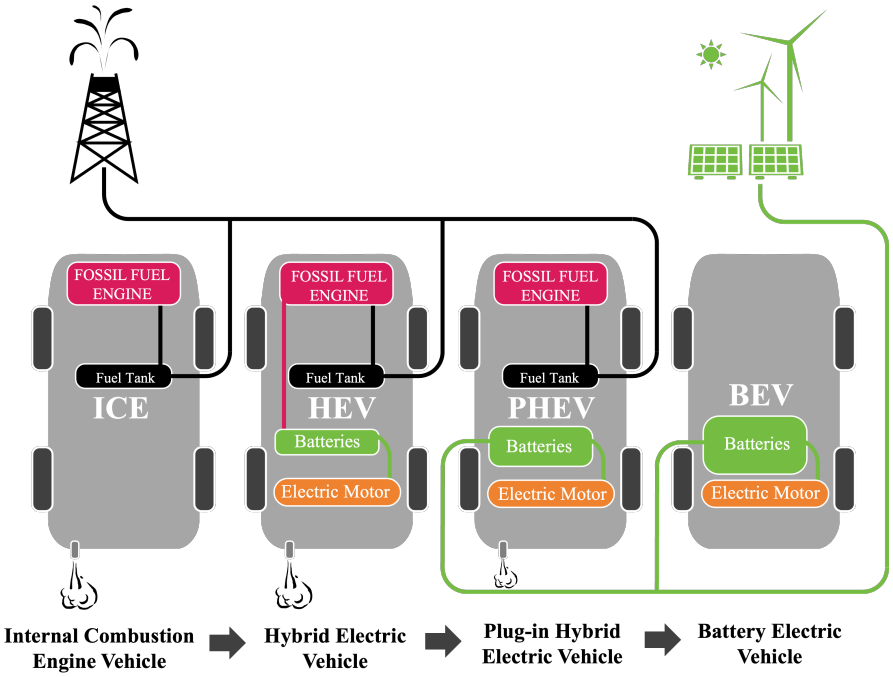


Fig. 1.4. Evolution of vehicle powertrains from conventional ICE to fully electric vehicles [26].

The adoption of electrified vehicles has accelerated globally, driven by technological progress and policy support. As shown in Fig. 1.5, the global electric vehicle stock has grown exponentially from 2010 to 2023, with particularly strong growth in China, Europe, and the United States. By the end of 2023, the global electric car fleet had reached approximately 40 million vehicles. This remarkable growth is evidenced by the fact that over 250,000 new electric cars were sold every

week in 2023, surpassing the total annual sales from just a decade ago. China has maintained its position as the world's largest electric vehicle market, with BEVs and PHEVs together accounting for nearly 60% of global electric car sales in 2023 [29]. The country's success in electric vehicle adoption can be attributed to comprehensive policy support and a maturing industrial ecosystem. Chinese manufacturers have demonstrated increasing competitiveness, producing more than half of all electric cars sold globally in 2023, despite accounting for just 10% of global ICE vehicle sales [30]. Europe has emerged as the second-largest market for electric vehicles, with electric cars representing approximately 25% of all car sales in the region in 2023. Countries like Norway have shown exceptional progress, with electric vehicles accounting for nearly 95% of new car sales. This success has been driven by strong regulatory frameworks, including emissions standards and incentive programs, alongside growing consumer acceptance of electric mobility solutions [31].

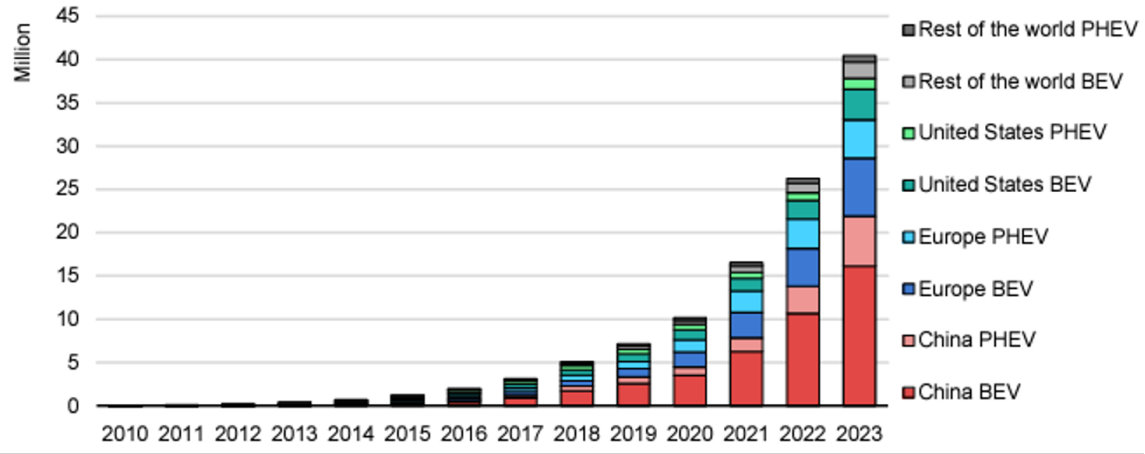


Fig. 1.5. Global electrified car stock trends, 2010-2023 (passenger cars only) [29].

The rapid growth of vehicle electrification is driven by continuous advancements in battery technology and powertrain systems, alongside supportive government policies and incentives globally [32]. Stricter emission regulations and rising environmental awareness have further accelerated this transition, while the steady decline in battery costs and significant improvements in charging infrastructure have made electrified vehicles increasingly accessible to mainstream consumers [33].

However, the transition to electrified mobility also introduces several technical and economic challenges. A central concern is powertrain efficiency, particularly in hybrid configurations. While series hybrid systems are theoretically capable of maintaining the engine at its optimal

operating point, their real-world efficiency is often compromised by transient engine behavior, multiple energy conversion stages (from engine to generator, then to battery and motor), and suboptimal energy management strategies [34, 35]. For example, a comparative study of hybrid systems found that the series configuration exhibited higher fuel consumption (4.4 g/s) than the parallel setup (3.45 g/s) under peak load conditions, suggesting that series hybrids may underperform in high-demand scenarios despite their theoretical advantages [36].

Another persistent challenge lies in drivetrain adaptability across diverse driving conditions. BEVs and PHEVs may exhibit reduced efficiency under high-speed or low-temperature conditions, where battery performance and thermal management become critical [37]. Moreover, battery degradation, range anxiety, and long charging times remain concerns for many consumers, especially in regions lacking fast-charging infrastructure.

From a systems perspective, integrating electrification with connected and intelligent technologies (e.g., V2X-enabled EMS) adds further complexity. These require real-time coordination between powertrain control and external data sources, which imposes strict demands on computation, communication latency, and algorithm robustness [38]. Addressing these challenges is essential for unlocking the full potential of electrified mobility.

1.1.4 Current development of powertrains for electrified vehicles

Building on this foundation, the development of powertrains for electrified vehicles has progressed significantly, driven by global environmental regulations and the growing demand for sustainable transportation [39]. Traditional ICE powertrains, which rely solely on fossil fuels, served as the starting point for this evolution. To address increasing concerns over fuel economy and CO_2 emissions, the automotive industry introduced HEVs, which integrate ICEs with electric motors and batteries. These systems improve efficiency, reduce emissions, and offer a transition toward electrification [40]. Generally, HEVs are categorized into four key configurations based on their mechanical architecture and power flow: **series**, **parallel**, **power-split**, and **multi-mode** [41].

Series hybrids

Series HEVs primarily rely on traction motors to drive the vehicle, while the ICE is connected solely to a generator, as illustrated in Fig. 1.6. The traction motors are powered by both the battery and the generator and can be installed on the front, rear, or both axles, enabling electric all-wheel-drive capability.

A key feature of series hybrids is the absence of a mechanical connection between the ICE and the drive axle. This allows the ICE to operate independently of vehicle speed or driver power demands, consistently running within its optimal efficiency range. The traction motors, with their broader operating range and higher efficiency compared to the ICE, further enhance overall system performance. Additionally, the simplified powertrain design eliminates the need for a conventional transmission, reducing mechanical complexity and improving energy management efficiency [42].

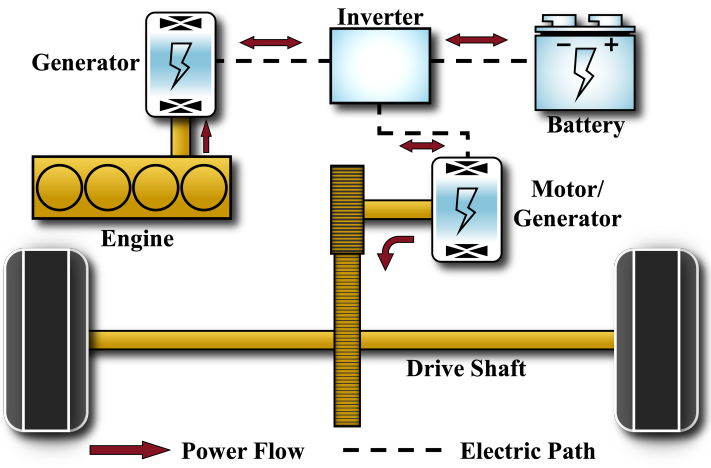


Fig. 1.6. Configuration of series HEVs [41].

Parallel hybrids

Parallel HEVs in Fig. 1.7 combine the ICE and motor-generator (MG) in a mechanically connected system, allowing them to work together or independently based on driving conditions. This design enhances power delivery flexibility and improves energy efficiency by redistributing power demands. The MG operates as a generator during low power demand to recover energy and as a motor during high power demand to assist the ICE, enabling it to run more efficiently. This results in reduced fuel consumption and emissions compared to conventional vehicles.

Additionally, a transmission system is incorporated to align the high-speed operation of the ICE with the lower speed requirements of the vehicle, ensuring smooth power transfer [43].

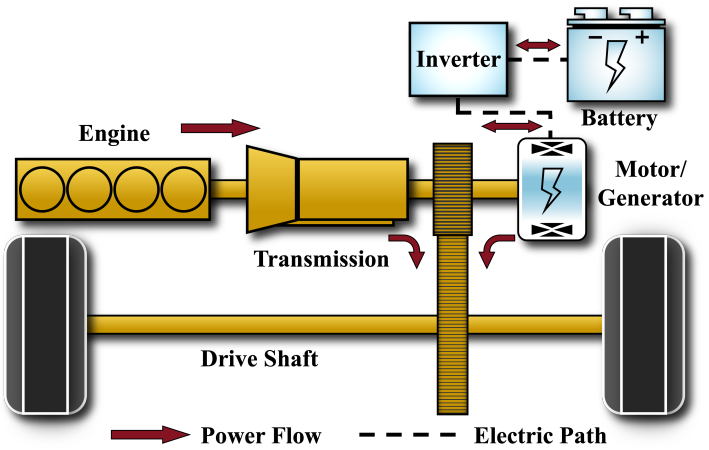


Fig. 1.7. Configuration of parallel HEVs [41].

Parallel hybrid powertrains can be categorized into five subtypes based on the location and size of the MG: P0, P1, P2, P3, and P4, as shown in Fig. 1.8. The P0 uses a belt-driven starter/generator located before the ICE, which supports only basic functions like start-stop due to its torque limitations. In the P1, the motor is integrated into the engine's crankshaft, allowing for regenerative braking and acceleration assistance, though its size limits torque output. P2 and P3 with motors placed at the transmission input and output, respectively, support higher speeds and better regenerative braking, offering superior efficiency. P4 integrates the motor into the drive shaft or wheel hubs, often combined with P2 or P3 for four-wheel drive applications. P2 and P3 are the most widely adopted due to their balance of performance and efficiency [44].

Power-split hybrids

Power-split HEVs utilize one or more planetary gear (PG) sets, also known as power split devices (PSDs), to connect the ICE, two MGs, and the driveshaft, as illustrated in Fig. 1.9. These PG sets are central to the power-split hybrid system, acting as an electronic continuously variable transmission (E-CVT). This decoupling of the ICE from vehicle speed enables efficient engine operation across various driving conditions, particularly in urban environments, where power-split HEVs often outperform series and parallel HEVs in fuel economy [45].

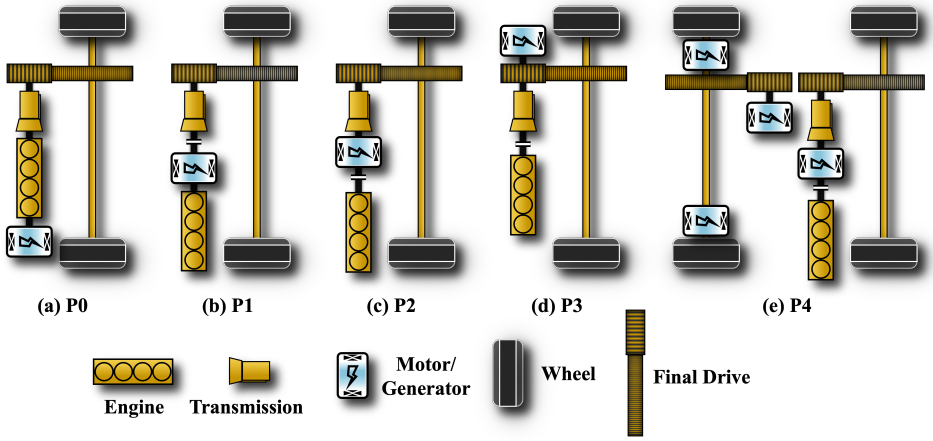


Fig. 1.8. Configurations of subtypes P0 to P4 in parallel HEVs [41].

The PSD allows power to flow through two main paths: the mechanical path and the electrical path. In the electrical path, the system functions like a series hybrid, where part of the ICE power is converted into electricity via a generator to drive the motor or charge the battery. In the mechanical path, the system operates like a parallel hybrid, with the ICE directly powering the driveshaft. This dual capability allows power-split HEVs to combine the strengths of series and parallel hybrids for improved efficiency and performance [46]. A key advantage of power-split architectures is their operational flexibility. The system can operate in multiple modes depending on driving conditions and power demands. For instance, at low speeds or light loads, the vehicle runs solely on electric power, with the motor using energy from the battery while the ICE remains off. When higher power is needed at low speeds, the ICE operates at its optimal efficiency point, generating electricity through the generator to power the motor. During highway cruising, the ICE directly drives the wheels through the mechanical path, which is more efficient for steady-state operation [47].

Multi-mode hybrids

Multi-mode HEVs enhance flexibility and efficiency by incorporating clutches into parallel or power-split systems, enabling transitions between series, parallel, and power-split configurations [48], as shown in Fig. 1.10. This adaptability allows the system to optimize energy efficiency and performance across varying driving conditions by leveraging the strengths of each configuration. There are two main configurations, including **series-parallel** and **PG coupling**. **Series-parallel** systems, like Honda's i-MMD, use clutches to switch between EV, series, and

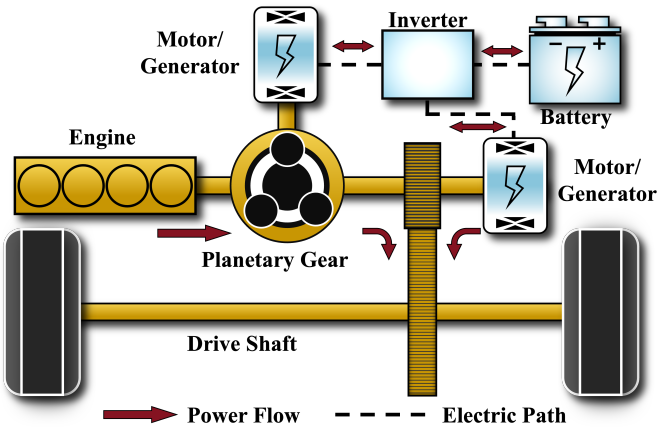


Fig. 1.9. Configurations of power-split HEVs [41].

parallel modes, employing two motor generators to optimize efficiency [49]. **PG coupling** configurations, such as GM's Advanced Hybrid System (AHS), expand power-split systems by introducing clutches between planetary gear nodes, allowing high torque at low speeds (input-split) and better efficiency at high speeds (compound-split). GM's Voltec system and Toyota's multi-stage hybrids further enhance these designs with advanced gear arrangements and multi-speed transmissions for improved performance. While multi-mode hybrids offer significant benefits in efficiency and flexibility, their increased system complexity and cost present design challenges. Despite this, they remain a crucial innovation in hybrid technology, combining the strengths of multiple configurations for sustainability and performance [50].

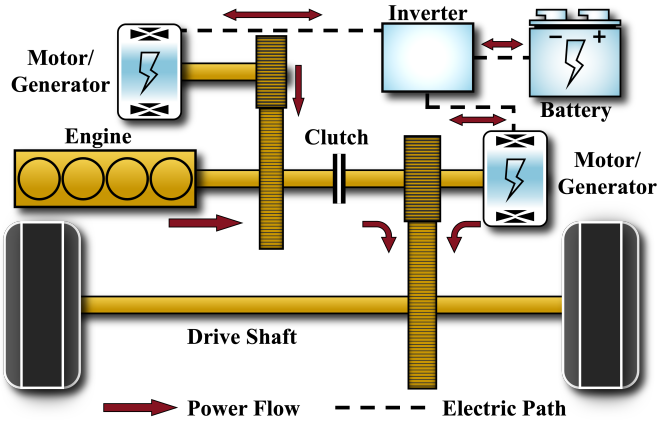


Fig. 1.10. Configurations of multi-mode HEVs [41].

Electrified powertrains have evolved from hybrid systems to sophisticated multi-mode configurations, achieving significant improvements in energy efficiency and emissions reduction. When combined with connectivity and intelligent technologies, these innovations pave the way for

cleaner and more efficient mobility. However, fully realizing their potential requires overcoming key challenges in energy optimization, particularly in integrating these technologies across multiple vehicles and diverse power sources.

1.2 | Motivation

This PhD research addresses energy optimization challenges in the modern automotive industry, driven by the transformative trends of connectivity, intelligence, and electrification. The growing adoption of CAVs with electrified powertrains is fundamentally reshaping the transportation landscape, driving the need for more efficient and sustainable mobility solutions [51]. As these technologies continue to evolve, there is a critical demand for advanced energy management strategies (EMSs) that effectively integrate these three trends and leverage reinforcement learning (RL) to operate efficiently across both macro (traffic) level and micro (vehicle) levels. At the macro-level scenario, multiple vehicles are coordinated to enhance stability and minimize energy waste. At the micro-level scenario, they optimize individual powertrain operation and driving strategies for maximum efficiency. Bridging these two levels is essential for achieving a scalable and intelligent energy optimization framework. Therefore, the motivation for this research stems from three interconnected areas:

1. **Energy Management for Multiple Vehicles:** Connectivity is enabling vehicles to interact with one another and with infrastructure through technologies such as V2V communication. These advancements create opportunities for vehicles to collaborate in reducing energy consumption, improving traffic flow, and enhancing safety. However, many existing methods fail to fully exploit the potential of connectivity for cooperative energy optimization. This research is motivated by the need to leverage connectivity for joint energy and stability optimization in multi-vehicle systems. It introduces a multi-agent reinforcement learning (MARL)-based decentralized control framework that enables vehicles to learn cooperative driving policies that minimize energy consumption while maintaining platoon performance.

2. **Energy Management in Multi-Mode Powertrains:** Electrification has introduced hybrid architectures that combine ICEs, electric motors, and batteries into multi-mode

systems. Managing energy flow across these modes is inherently complex, especially under dynamic driving conditions. However, efficiently managing energy across these modes is a key challenge, as traditional methods often treat these components independently, leading to inefficiencies. This research is motivated by the need for a more adaptive and integrated energy management solution. It proposes a MARL-based EMS that dynamically allocates energy across modes through a multiple-input multiple-output (MIMO) control framework, improving efficiency under driving variability.

3. **Energy Management in Specified-Mode Powertrain:** As EMSs evolve, intelligence becomes essential for enabling adaptive, generalizable, and robust control. Studying specified-mode powertrains allows for focused analysis of core RL challenges such as overestimation bias and training stability. This research is motivated by the need to establish a practical and robust RL-based EMS, combining theoretical analysis and real-time validation. It supports the broader objective of embedding learning-based intelligence into practical EMS design, complementing the more complex scenarios in earlier chapters.

The overall motivation for this research is to create a unified framework that integrates energy optimization at both the macro (traffic) and micro (vehicle) levels, as illustrated in Fig. 1.11. By addressing energy optimization for multi-vehicle systems, multi-mode PHEVs, and specified-mode PHEVs, this research contributes the development of the automobile industry to becoming more energy-efficient, sustainable, and intelligent. Ultimately, this research aims to support global efforts to reduce energy consumption and environmental impact while enhancing the performance of future connected, intelligent, and electrified technologies.

1.3 | Aims and Objectives

The aim of this PhD research is to develop advanced energy optimization frameworks that improve the efficiency and sustainability of the modern automobile industry. By addressing both macro-level and micro-level energy management challenges, this research seeks to contribute to the advancement of intelligent, connected, and electrified vehicles. To achieve this aim, the research focuses on the following specific objectives:

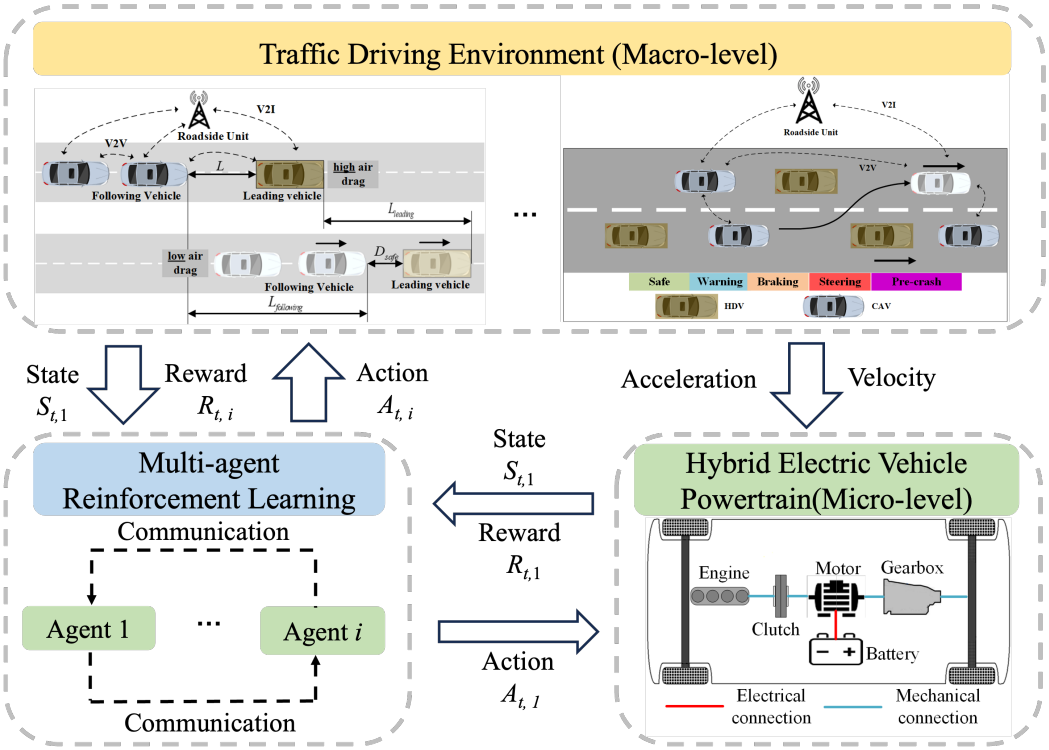


Fig. 1.11. Illustration of the macro-micro energy management system control framework.

1. To develop energy-efficient and scalable decentralized cooperative systems by integrating energy efficiency and improving communication strategies, enabling robust and sustainable operations in the CACC scenario.
2. To develop advanced EMSs for multi-mode PHEVs that address the complexity of multi-input, multi-output (MIMO) control. The research aims to optimize energy allocation across diverse driving scenarios, improving fuel efficiency and bridging the gap between traffic coordination and powertrain optimization.
3. To develop ensemble RL frameworks that enhance self-learning stability, adaptability, and reliability by addressing challenges such as overestimation bias and the exploration-to-exploitation trade-off, ensuring practicality and robustness through validation using software-in-the-loop (SiL) and hardware-in-the-loop (HiL) platforms in dynamic driving conditions.

1.4 | Thesis Outline

This thesis is organized into six main sections: a literature review, a methodology and platform, three core chapters presenting the primary research contributions, and a concluding chapter that summarizes findings and discusses future directions.

Chapter 2 provides a comprehensive review of the advancements in cooperative control and EMSs for electrified vehicles. It begins with an exploration of strategies for the CACC of CAVs, focusing on optimizing traffic flow, energy efficiency, and safety through coordinated behaviors. The chapter then examines the application of RL techniques, including deep reinforcement learning (DRL) and Q-learning, in EMS for electrified vehicles. DRL is highlighted for its ability to operate effectively in complex and dynamic environments, while Q-learning offers a practical, model-free solution for real-time energy optimization. Finally, the chapter summarizes the key research gaps, providing a foundation for further investigation.

Chapter 3 outlines the research methodology and experimental framework, beginning with the preliminaries of RL and extending to MARL. It provides a detailed description of the development of multi-vehicle testing platforms, data collection methods, and vehicle modeling approaches, which form the foundation for the proposed strategies. The chapter introduces simulation environments for the optimization control, providing the foundation for testing the proposed strategies. Additionally, it describes the experimental facilities, including a HiL system, which enables real-time validation and ensures the practical applicability of the methods under realistic driving conditions. By integrating simulation and real-world testing, this chapter establishes a robust framework for developing and validating advanced EMSs.

Chapter 4 focuses on integrating energy optimization into CACC frameworks for CAVs, addressing gaps such as the lack of energy efficiency considerations and communication optimization. The research makes significant contributions by developing a decentralized MARL approach with a multi-objective reward function to balance energy efficiency, stability, and safety. It also enhances communication efficiency through quantized stochastic gradient descent (QSGD) and binary differential consensus (BDC) methods, validated across diverse platoon sizes and conditions. Finally, the research validates its practical effectiveness using real-world OpenACC data, demonstrating the model's robustness and applicability in actual CACC operations, with a particular focus on achieving improved energy efficiency while maintaining stability and safety.

Chapter 5 investigates the application of multi-agent deep reinforcement learning (MADRL) in EMS for multi-mode PHEVs, addressing the challenges of MIMO control. The research first

conducts extensive parametric studies to optimize deep deterministic policy gradient (DDPG) agent configurations, including network layers, learning rates, and policy noise parameters. Building on this foundation, it develops an innovative hand-shaking strategy that introduces a relevance ratio to facilitate effective collaboration and synchronization among DDPG agents toward achieving global objectives. Finally, the proposed system successfully manages multiple objectives, including minimizing fuel consumption and maintaining battery state of charge (SoC), establishing a robust framework for applying MADRL to PHEV energy optimization.

Chapter 6 introduces advanced ensemble learning methods to improve energy management for hybrid powertrains. Two main approaches are developed. The first is a supervisory ensemble system with a learning automata module, which integrates distinct ensemble methods (maximum-based, random-based, and weighted-based) for final action determination. The second is an ensemble learning framework designed to address overestimation bias in Q-learning. This framework introduces an update-to-data (UTD) mechanism, improving computational efficiency and convergence speed. Additionally, the research proposes different decay functions for exploration-to-exploitation (E2E) trajectory management in RL, enabling more efficient energy optimization. The study also includes theoretical proofs to validate the effectiveness of the proposed framework in mitigating overestimation bias and provides visualization of the reduced bias, offering insights into the robustness of the improved policy. These methods are validated through extensive SiL and HiL testing under real-world driving scenarios, demonstrating significant advancements in hybrid powertrain energy management.

Chapter 7 summarises the key findings and implications of this research, discusses the results in depth, and proposes directions for future investigation.

CHAPTER 2

Literature Review

This chapter primarily presents some recent advances in intelligent EMSs, from the viewpoints of connectivity, intelligence, and electrification of electrified vehicles. The literature review follows a recent progression from macro-level cooperation to micro-level powertrain optimization. The review begins by examining CACC, and exploring how CAVs leverage communication technologies to optimize stability and energy savings. This macro-level coordination directly influences vehicle energy consumption patterns, naturally leading to the subsequent focus on microscopic energy management. At this micro level, EMSs for various electrified powertrain configurations have been studied, supported by mathematical foundations and theoretical analysis of learning-based control methods. By bridging V2V cooperation with individual powertrain optimization, the chapter offers a thorough understanding of both the broader traffic dynamics and the specific challenges in electrified vehicle energy management. This chapter has been accepted in *International Journal of Automotive Manufacturing and Materials* and has been submitted in *IEEE Transactions on Automation Science and Engineering*.

2.1 | Cooperative adaptive cruise control of CAVs

The development of CAVs has emerged as a promising solution to enhance transportation efficiency, safety, and environmental sustainability. Operating within networked environments, CAVs engage in interactions with multiple vehicles, promoting cooperative driving [52, 53]. Through these connections, CAVs can coordinate their behaviors for merging, intersection management, and other driving scenarios, leading to improved traffic safety and efficiency [54].

2.1.1 General problem formulation

CACC represents a fundamental application of vehicle cooperation, focusing on longitudinal control where CAVs maintain close formations to improve efficiency, safety, and energy consumption by optimizing platoon formation, speed, and spacing [55], thereby contributing to safer and more sustainable transportation systems [56, 57]. As shown in Fig. 2.1, the optimal problem aims to ensure a safe, fuel-efficient, and smooth vehicle following operations, while also maximizing the advantages of driving in close formation [58].

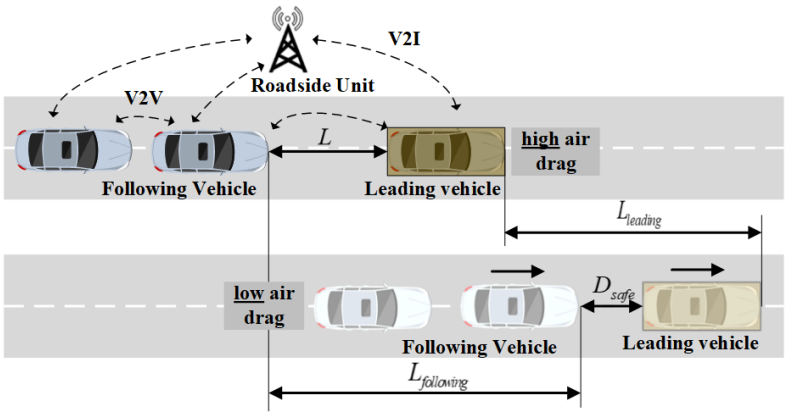


Fig. 2.1. Cooperative adaptive cruise control scenario.

Given N vehicles in a platoon, the control objective is to maintain a desired spacing L_i between vehicle i and its preceding vehicle $i-1$, while ensuring velocity matching. The state vector [59] for each vehicle i at time t is defined as $s_i(t) = [x_i(t)\ v_i(t)]^T$, where $x_i(t)$ and $v_i(t)$ represent the position and velocity of vehicle i at time t, respectively. This state vector captures the longitudinal motion of the vehicle and serves as the basis for the control design. Therefore, the dynamics of each vehicle can then be expressed as Eq. 2.1:

$$\dot{\mathbf{s}}_i(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{s}_i(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_i(t), \tag{2.1}$$

where $u_i(t)$ is the control input, defined by a feedback control law that accounts for both position and velocity errors relative to the preceding vehicle $i-1$: $u_i(t) = k_p\left(x_{i-1}(t) - x_i(t) - L_i\right) + k_d\left(v_{i-1}(t) - v_i(t)\right)$ with k_p and k_d being the proportional and derivative gains for the spacing and velocity errors, respectively.

To optimize the platoon behavior over a time horizon T, a cost function J is defined to penalize both the spacing error and the velocity mismatch:

$$J = \sum_{i=1}^{N} \int_{0}^{T} \left(\alpha e_{x,i}^{2}(t) + \beta e_{v,i}^{2}(t) \right) dt, \tag{2.2}$$

where $e_{x,i}(t) = (x_{i-1}(t) - x_i(t)) - L_i$ and $e_{v,i}(t) = v_{i-1}(t) - v_i(t)$ denote the position and velocity errors, and α, β are weighting coefficients that balance their respective contributions in the objective. In cooperative vehicle control, these parameters are crucial, as position optimization ensures safety and aerodynamic benefits, while velocity optimization contributes to ride comfort and energy efficiency. Thus, cooperative strategies would be developed to achieve the collective goal, adapting their behaviors in response to the actions of other vehicles within the network.

2.1.2 Control approches

Control approaches for CACC in CAVs can be broadly categorized into three major groups: **classical control methods**, **learning-based methods**, and **hybrid methods**.

Classical control methods

Classical control methods include well-established approaches that are typically deterministic and analytically grounded. These methods are subdivided into the following categories:

a) Rule-based methods for CACC in CAVs

Ding et al. proposed a rule-based adjusting algorithm to efficiently and safely coordinate CAVs in CACC. Simulation results demonstrated its effectiveness and robustness across various conditions, outperforming traditional rule-based and optimization-based methods in throughput, delay, computational cost, and fuel consumption [60]. He et al. compared five cooperative driving strategies for CACC in CAVs, recommending dynamic programming (DP)-based, grouping-based, and rule-based methods for their efficiency and practicality, while noting a trade-off between traffic efficiency and fairness [61]. Gao et al. proposed a fuzzy rule-based control strategy selection approach for CAVs, utilizing fuzzy reasoning Petri nets for system modeling and parallel reasoning. The method enables quick selection of the most dependable sensor data for control in dynamic environments, with case studies validating its effectiveness

[62]. Rule-based methods are characterized by their simplicity and ease of implementation. These approaches rely on predefined rules and heuristics to manage vehicle behaviors, making them computationally efficient. However, they often lack the adaptability required to handle dynamic and complex traffic environments, which limits their effectiveness in scenarios with high variability or uncertainty.

b) Optimization-based methods for CACC in CAVs

Optimization-based methods leverage sophisticated mathematical formulations to achieve optimal solutions for multiple control objectives in CACC systems. These approaches, primarily founded on optimal control theory (e.g., model predictive control (MPC)), systematically calculate optimal velocities for platoon vehicles while balancing multiple objectives such as energy efficiency, safety maintenance, and travel time reduction [63–65].

Recent advances in optimization-based CACC have demonstrated promising results across various applications. In terms of multi-objective optimization, Zhong et al. proposed a comprehensive multi-objective optimization (MOOP)-CACC framework that effectively balances mobility, safety, driver comfort, and fuel consumption. Their simulations showed remarkable improvements, including 98% reduction in time headway deviation, 93% reduction in unsafe conditions, 42% reduction in jitter, and 33% reduction in fuel consumption compared to conventional CACC systems [66]. Building on this multi-objective approach, Zhao et al. developed a data-driven MPC framework that leverages V2V communication to ensure stability, safety, and efficiency under uncertain traffic conditions [67]. Energy efficiency has been a particular focus of optimization-based methods. Park et al. introduced a predictive eco-cruise control system that achieved up to 15% fuel savings by incorporating roadway topographic data from high-resolution digital maps [68]. Similarly, Ma et al proposed an ecological CACC (eCACC) strategy, based on nonlinear MPC (NMPC), demonstrated up to 16.1% energy savings while maintaining improved stability and comfort compared to existing controllers [69]. In the context of emission reduction, Wang et al. developed an NMPC framework that effectively stabilized traffic flow and reduced emissions in congested platoons, with benefits scaling with the penetration rate of CAVs [70].

Despite these impressive achievements and theoretical elegance, optimization-based methods face several practical challenges. A fundamental limitation lies in their reliance on simplified

assumptions, such as constant or predetermined lead vehicle velocities, which rarely hold in dynamic real-world conditions. Moreover, the computational complexity of solving high-dimensional optimization problems in real-time poses significant implementation challenges for vehicle onboard control units [71], particularly in scenarios involving large-scale platoons or rapidly changing traffic conditions. As the system scales, this computational burden can compromise real-time performance, making implementation increasingly difficult. These limitations have motivated research into alternative approaches, such as learning-based methods, which offer greater computational efficiency while maintaining robust performance in dynamic environments.

Learning-based methods for CACC in CAVs

Learning-based methods, leveraging advanced AI techniques such as deep learning (DL) and RL, have emerged as promising solutions for CACC systems due to their superior adaptability and self-optimization capabilities [72]. These approaches can learn optimal control strategies directly from experience, enabling them to handle complex and dynamic traffic scenarios more effectively than traditional methods.

a) Single-agent learning methods

Recent research has demonstrated significant achievements in single-agent learning approaches, which can be generally categorized into **DL**, **RL**, and **hybrid approaches**.

DL-based methods leverage neural networks to achieve end-to-end vehicle control and enable sophisticated prediction and planning capabilities, allowing vehicles to learn directly from raw sensory inputs. Long Short-Term Memory (LSTM) networks have shown particular promise in CACC applications. Tian et al. proposed an LSTM-based method for CACC platoons that predicts preceding vehicle acceleration using radar data, effectively addressing the impact of wireless communication delays on string stability. Their simulations with NGSIM data demonstrated that LSTM control successfully compensates for delays exceeding 0.115 seconds while maintaining stability [73]. Similarly, Hao et al. developed an advanced framework combining LSTM networks with risk prediction to enhance platoon stability and safety. Their approach incorporates innovative strategies, such as platoon splitting and additional leader identification, to enhance smoothness and safety during abnormal driving conditions [74]. **RL**-

based methods enable vehicles to learn optimal control policies through continuous interaction with the environment, demonstrating superior adaptability in dynamic traffic scenarios. A notable example is the DDPG-based controller from Lin et al., which achieved remarkable improvements in control performance, reducing episode costs by up to 5.8% compared to conventional MPC approaches [75]. This improvement highlights the potential of DL and RL in optimizing vehicle control strategies. Alves Neto et al. also proposed the DDPG approach for vehicular platoons that generalizes training across different network topologies and incorporates integral action in acceleration commands to address disturbance rejection. Experiments demonstrated improved robustness to uncertain parameters and external forces, outperforming state-of-the-art methods in reducing steady-state errors and overshoot, offering insights into the generalization and resilience of RL-based platoon control [76]. Kanipakam et al. further advanced the field by introducing an innovative intent-sharing mechanism in CACC (I-CACC) using RL, demonstrating superior performance across multiple metrics including safety, comfort, string stability, and gap-keeping behavior [77]. Ke et al. developed an integrated CACC system combining lidar and V2V communication with deep Q network (DQN) learning. Their experimental results showed a substantial 44.74% reduction in inter-vehicle spacing compared to conventional ACC, with DQN contributing an additional 40.19% improvement while maintaining minimal communication latency of 1 ms through WiFi implementation [78]. Liu et al. introduced a multi-timescale control and communication (MTCC) framework for platoon control, addressing observation delays with an augmented Markov decision process (MDP) and training in realistic V2X delay environments. The approach outperformed baseline DRL methods in multi-timescale control and communication optimization [79]. Building on these foundations, **hybrid approaches** combine multiple paradigms to leverage their complementary strengths. Fu et al. integrated DRL with expert systems (ESs) to enhance control accuracy, adaptability, and interpretability for CAVs while leveraging blockchain-based driving rule sharing to ensure privacy and security [80]. These hybrid solutions excel in balancing the data-driven advantages of learning methods with the established robustness of conventional control strategies, representing a promising direction for future CACC development.

Despite these promising results, single-agent learning methods face several challenges. DL-based methods often require large amounts of training data and may struggle with generalization

across different traffic scenarios. RL-based approaches frequently encounter issues with sample efficiency and training stability, particularly in complex real-world environments. Even hybrid approaches, while addressing some limitations, still face challenges in balancing computational efficiency with real-time performance requirements. Additionally, these methods typically focus on individual vehicle performance without fully considering the collective behavior of multiple vehicles in dynamic traffic environments.

b) Advanced learning methods

Advanced learning methods in CACC systems represent cutting-edge approaches that address specific challenges in real-world deployment. **Transfer learning (TL)** enables knowledge sharing across different traffic scenarios and conditions, allowing control strategies learned in one environment to be effectively adapted to new situations. Albeaik et al. developed a gray-box DL model and DRL framework for CACC in heavy-duty trucks, achieving stable, accurate performance with zero-shot field transferability and improved speed and time-gap error variance compared to benchmarks [81]. **Meta learning** techniques enhance the system's adaptability by enabling rapid adjustment to novel driving scenarios and facilitating the learning of optimal control strategies with minimal data. Wang et al. developed an implicit quantile network and meta-learning to learn continuous control and to enhance bus system efficiency and reliability by addressing uncertainties and extreme events, outperforming existing models in simulations [82]. **Federated learning** addresses privacy and security concerns in connected vehicle networks by enabling distributed learning across multiple vehicles while keeping sensitive data local, allowing collaborative model training without compromising individual vehicle data privacy [83]. Parvini et al. addressed decentralized subchannel allocation and power control in CACC systems, leveraging federated learning to enhance string stability, and V2V link reliability, outperforming centralized methods [84].

These advanced methods collectively enhance the practicality and robustness of learning-based CACC systems, particularly in handling diverse traffic conditions and scaling to large vehicle networks. However, as vehicle platoons grow in size and complexity, the need for coordinated decision-making and collaborative learning becomes increasingly critical. This necessitates moving beyond single-agent and advanced learning paradigms to approaches that can explicitly model and optimize multi-agent interactions.

c) Multi-agent learning

Among learning-based methods, MARL has demonstrated superiority over single-agent RL and other advanced approaches in various domains, including industrial robotics and autonomous driving [85]. MARL enables multiple agents to learn collaborative or competitive behaviors in shared environments, significantly enhancing their ability to handle complex tasks and improve overall system performance [86]. Recent developments in MARL-based CACC strategies have primarily focused on two key aspects: efficient learning algorithms and communication protocols.

Regarding **learning algorithms**, researchers have made significant strides in enhancing traffic efficiency and safety. Han et al. further advanced this field by implementing a MARL framework with a safe actor-critic algorithm that leverages bounded-error neighbor data to enhance both traffic efficiency and safety [87]. Zhang et al. proposed a MARL framework by using Markov games and LSTM networks to capture platoon dynamics, while asynchronous RL enables knowledge sharing among vehicles to enhance collaboration and efficiency. Simulations showed a 19.2% reduction in fuel consumption compared to rule-based methods [88]. Zhang et al. introduced multi-agent model predictive shielding (MAMPS), an algorithm that ensures safety for learned policies by switching to a backup policy when safety cannot be guaranteed. Simulations demonstrated that MAMPS maintains safety while achieving strong performance in multi-agent environments [89]. Du et al. proposed a decentralized model-based policy optimization framework (DMPO) to improve data efficiency in multi-agent control tasks. Each agent learns a dynamic model for state prediction and communicates locally, with myopic rollouts reducing model bias. The approach maintains policy independence with an extended value function and achieves comparable performance to model-free methods while using significantly fewer samples [90]. Chen et al. proposed a GuidDDPG algorithm for vehicle platoon collaborative control, integrating ACC guidance to improve convergence speed and efficiency. Simulations demonstrated superior performance compared to DDPG and twin delayed deep deterministic policy gradient (TD3) [91]. Gangopadhyay et al. proposed a counterexample-guided refinement method for MARL policies to ensure safety in critical applications. The algorithm identifies and eliminates counterexamples through targeted gradient updates, refining policies to meet safety specifications. Empirical validation showed improved

safety in various cooperative multi-agent tasks [92]. These advances in learning algorithms demonstrate a clear trend toward integrating safety mechanisms, efficiency improvements, and collaborative learning strategies, with particular emphasis on practical implementations that can handle real-world driving scenarios while maintaining robust performance guarantees.

Efficient communication among CAVs in platooning control also poses a fundamental challenge in achieving seamless coordination and maintaining the desired following properties, such as precise inter-vehicle spacing and synchronized maneuvers. MARL has shown promising results in addressing this challenge through various innovative approaches [93]. Peake et al. introduced a MARL system utilizing a robust LSTM-based communication protocol for stable CACC control [94]. Song et al. proposed a state-shared MADRL framework that enables CAV platoons to share and periodically update state information between neighboring vehicles, leading to enhanced travel efficiency and reduced environmental impact through coordinated decision-making [95]. Chen et al. proposed a fully decentralized MARL framework for CACC, enhancing efficiency and scalability without relying on centralized communication. A quantization-based scheme reduces communication overhead by transmitting only quantized non-zero information [96]. Li et al. developed the communication proximal policy optimization (CommPPO) algorithm, which incorporates a predecessor-leader-follower communication protocol to manage both global and local state information. Notably, CommPPO's novel reward communication channel effectively addresses common MARL challenges such as spurious rewards and lazy agents [97]. Chu et al. approached the platooning control problem by formulating it as a spatiotemporal MDP, introducing both a spatial discount factor for local agents and a differentiable communication protocol (NeurComm) that improves learning efficiency by mitigating non-stationarity [98]. Communication-based approaches can also leverage V2V information sharing and graph neural networks (GNN) to enhance vehicle coordination. For instance, Liu et al. introduced a multi-agent hierarchical attention RL (MAHARL) framework, which incorporates the GNN at each hierarchical level to account for the reciprocal influences between agents. The hierarchical architecture of MAHARL empowers agents with foresight, enabling them to make favorable decisions, even in the absence of immediate incentives, and consider long-term rewards [99]. These methods enable sophisticated information exchange and processing, where each vehicle can learn optimal strategies while

considering the states and actions of neighboring vehicles. The integration of MARL with advanced communication architectures has proven especially effective in handling complex scenarios requiring synchronized movements and collective decision-making in vehicle platoons.

Hybrid methods

While MARL has shown promising results in CACC applications, the complexity of real-world scenarios has motivated researchers to explore hybrid approaches that combine MARL with other methodologies to leverage their complementary strengths. These hybrid methods aim to address challenges that single approaches, including MARL, struggle to handle effectively, particularly in situations involving incomplete information, multiple objectives, or complex vehicle interactions. Among these hybrid approaches, game-theoretic methods and mixed platooning control strategies have emerged as particularly promising directions.

a) Game-theoretic methods

Additionally, in scenarios where complete information is lacking, Li et al. [100] developed a game-theoretic framework to capture the strategic interactions between the leading vehicle and the following vehicles. To overcome the absence of full information, the equilibrium solution in platooning control is identified through backward induction, coupled with information collection from the following vehicles. Zohdy et al. developed a game-theoretic heuristic optimization algorithm for CACC-equipped vehicles, minimizing total delay through collaboration with a 70% average delay reduction compared to traditional stop control [101].

While game-theoretic approaches provide effective solutions for strategic decision-making in vehicle platoons, the increasing complexity of real-world traffic scenarios demands consideration of more diverse vehicle interactions [102]. Particularly, the coexistence of multiple platoons and different vehicle types in practical transportation systems introduces new challenges that require specialized control strategies.

b) Mixed platooning control methods

Mixed platooning control involves the simultaneous presence of multiple vehicle platoons within a network, necessitating a sophisticated level of coordination and communication both within (intra-platoon) and between (inter-platooning) these platoons [103]. Parvini et al. introduced two advanced MARL algorithms specifically designed to address the challenges

of mixed platooning control. These include the modified multi-agent DDPG (MADDPG) and its variant, the modified MADDPG with task decomposition. In these frameworks, the leaders of the platoons are treated as independent agents interacting with the environment to determine the most effective policies for platoon management. Notably, these algorithms employ multiple critics to estimate both global and local rewards, thereby fostering cooperative behavior among the agents. In the second algorithm, individual rewards are further decomposed into task-specific sub-reward functions, offering insights into their work as detailed in [104].

Even though CAVs have made significant advancements, there will be an extended period during which CAVs will coexist with human-driven vehicles (HDVs), a situation commonly referred to as mixed-traffic scenarios [105]. Different penetration rates can be assessed to understand the impact of mixed traffic. In the study presented in [106], the impact of the penetration rate of CAVs on the energy efficiency of the traffic network is examined, with a specific emphasis on a cooperative eco-driving system that utilizes longitudinal control. A noteworthy aspect of this research is the introduction of a role transition protocol for CAVs, enabling them to smoothly transition between leading and following positions within a vehicle string. Lu et al. enhanced the evaluation by integrating several key elements, including altruism control, a quantitative car-following strategy, a refined platoon reward function, and a collision avoidance method, as outlined in their work [107]. In [108], the authors introduced the concept of characterizing consecutive HDVs as a collective entity, referred to as AHDV, to minimize stochastic variability and leverage macroscopic characteristics for the control of following CAVs. They employ a control strategy built on distributed proximal policy optimization (DPPO) to anticipate disturbances and downstream traffic conditions in mixed traffic scenarios.

The field of CACC for CAVs continues to evolve rapidly, with learning-based approaches, particularly MARL, showing promising results. Future research should focus on addressing the challenges of energy efficiency, scalability, and robust performance while maintaining safety guarantees.

2.1.3 Challenges

Despite significant advancements in CACC, several key challenges remain unresolved, limiting its practical deployment in various scenarios [56]. The reliance on V2V and V2I communication

is compromised by delays, bandwidth constraints, and cybersecurity risks, affecting reliability and safety [109]. Ensuring control stability in heterogeneous traffic is challenging due to nonlinear vehicle dynamics, inter-vehicle delays, and uncertain maneuvers [110]. Additionally, centralized control frameworks struggle with scalability in large platoons, leading to high computational overhead and vulnerability to single-point failures [111, 112]. Furthermore, while energy optimization is a critical goal in CACC, achieving fuel efficiency and emissions reduction in dynamic and uncertain environments remains challenging, particularly in the presence of frequent accelerations and decelerations caused by platoon instability [106].

These challenges highlight the need for solutions ensuring robust communication, stable coordination, adaptability, and energy efficiency. Traditional methods, such as rule-based and optimization-based approaches, provide foundational insights but often fall short in addressing real-world complexities, particularly in balancing energy consumption with safety and performance. Learning-based approaches, especially MARL, show promise by enabling decentralized, adaptive decision-making and optimization across multiple objectives [113, 114]. However, MARL approaches still face challenges such as inefficient communication protocols, scalability limitations, and energy efficiency in dynamically changing conditions [98, 115, 116]. Addressing these gaps requires a decentralized MARL framework that minimizes reliance on global observations, ensures robust communication, and integrates energy optimization alongside other control objectives. By incorporating advanced learning algorithms and communication-efficient protocols, the framework addresses the limitations of centralized and existing learning-based methods. This approach aims to bridge the gap between theoretical advances and the practical demands of sustainable and energy-efficient transportation systems.

2.2 | Deep reinforcement learning of EMSs for electrified vehicles

Traditional EMSs for electrified vehicles mainly rely on rule-based and optimization-based approaches. While computationally efficient, rule-based methods depend on predefined thresholds, limiting adaptability to dynamic conditions [117, 118]. Optimization-based methods, such as DP and MPC, provide greater adaptability by solving mathematical optimization

problems, enhancing fuel economy and energy distribution [119]. However, their computational complexity and reliance on accurate system models make real-time implementation challenging, particularly in large-scale or rapidly changing environments [120].

To overcome these limitations, DRL has emerged as the transformative approach for EMSs in electrified vehicles, effectively overcoming the constraints of traditional control methods [121]. By integrating the decision-making capabilities of RL with the powerful function approximation of deep neural networks, DRL excels in handling high-dimensional state spaces and complex energy management configurations, making it well-suited for dynamic driving scenarios.

2.2.1 General Problem Formulation

The energy management problem using DRL can be formulated as a MDP defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$:

- $\mathcal{S}$: The state space, representing the system's current status, such as battery state-of-charge (SoC), vehicle speed, and power demand.
- $\mathcal{A}$: The action space, defining the control actions, such as power split ratio, motor torque, or gear position.
- $\mathcal{P}(s'|s,a)$: The transition probability, capturing the system's dynamics and describing the probability of transitioning to state s' from state s after taking action a.
- $\mathcal{R}(s,a)$: The reward function, quantifying the immediate benefit or cost of taking action a in state s. Common objectives include minimizing fuel consumption, emissions, or battery degradation.
- $\gamma \in [0,1]$: The discount factor, reflecting the trade-off between immediate and future rewards.

The objective of the DRL agent is to learn a policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$ that maximizes the expected cumulative discounted reward, defined as:

$$J(\pi) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t)\right], \tag{2.3}$$

where s_t and a_t denote the state and action at time step t. The policy $\pi(a|s)$ determines the probability of selecting action a given state s.

DRL approximates the optimal policy π^* or the corresponding state-action value function $Q^*(s,a)$ using neural networks. The value function $Q^\pi(s,a)$ is defined as:

$$Q^\pi(s,a) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t) \middle| s_0 = s, a_0 = a \right]. \tag{2.4}$$

The optimal value function satisfies the Bellman optimality equation:

$$Q^*(s,a) = \mathcal{R}(s,a) + \gamma \mathbb{E}_{s'} \left[\max_{a'} Q^*(s',a') \right], \tag{2.5}$$

where s' represents the next state, and a' denotes possible future actions.

In DRL, neural networks parameterized by θ are used to approximate $Q^*(s,a)$ or $\pi^*(a|s)$. The training process involves minimizing a loss function $\mathcal{L}(\theta)$, such as the mean squared error for Q-learning:

$$\mathcal{L}(\theta) = \mathbb{E}_{(s,a,r,s')} \left[\left(Q_\theta(s,a) - y \right)^2 \right], \tag{2.6}$$

where the target value y is defined as:

$$y = \mathcal{R}(s,a) + \gamma \max_{a'} Q_{\theta^-}(s',a'), \tag{2.7}$$

where θ^- denotes the parameters of a target network, used to stabilize training.

This formulation enables DRL to handle complex, dynamic EMS tasks such as optimizing power split, managing battery usage, and leveraging regenerative braking, all while adapting to variable driving conditions.

2.2.2 Applications of DRL in EMSs

The applications of DRL in EMSs can be categorized into three main approaches based on the algorithmic developments:

Value-based methods

The initial applications of value-based methods primarily focused on DQN. DQN utilizes neural networks to estimate Q-values for different actions in given states, allowing it to process continuous state inputs. This approach reduces storage requirements and eliminates the need for state discretization. As an off-policy algorithm, DQN could utilize experience replay to accelerate convergence. Studies have shown that DQN-based EMSs with experience replay can achieve better optimization results than traditional Q-learning when applied to unfamiliar driving environments. Wu et al. developed a DQN-based strategy for energy management in hybrid electric buses, effectively addressing state dimensionality challenges and outperforming Q-learning in efficiency and convergence. The DQN approach achieved 89% of the fuel economy attained by DP and successfully managed battery SoC under varying conditions [122]. Qi et al. pioneered the application of DQN to optimize EMSs for power-split PHEVs. Their approach utilized fixed commuting routes for both training and application, leveraging historical driving data collected from these predetermined paths. The implementation yielded impressive results, achieving fuel consumption reductions of 12.0% and 16.5% compared to conventional rule-based strategies [123]. Building upon this success, the team further enhanced their methodology by introducing a Dueling DQN architecture, which significantly accelerated convergence rates. This improvement stemmed from the innovative decomposition of the Q-value function into two components: a state-value function (V-value) estimating the inherent value of each state, and an advantage function quantifying the marginal benefit of specific actions within given states. This architectural refinement enabled more stable training and improved sample efficiency [124]. Hu et al. demonstrated the adaptability of DQN-based EMSs for parallel HEVs. Their study showed that EMSs trained on the Urban Dynamometer Driving Schedule (UDDS) could effectively transfer to the New European Driving Cycle (NEDC), achieving a 10.29% fuel reduction compared to rule-based strategies. This work proved the neural network's capability to generalize across different driving cycles in real-time applications [125]. Han et al. proposed a double deep Q-learning (DDQN)-based EMS for a series hybrid tracked vehicle, incorporating lateral dynamics in the vehicle model. Results showed that the DDQN-based EMS achieved a 7.1% reduction in fuel consumption compared to DQN and

reached 93.23% of the optimization performance of DP. The DDQN framework addresses the overoptimistic policy value estimation in conventional DQN, achieving faster convergence and improved optimization performance [126]. He et al. developed a DQN-based EMS for series HEVs, utilizing 1000-second natural driving profiles for both training and testing. Their approach achieved a 5.96% fuel consumption reduction compared to traditional RL methods while demonstrating superior generalization to new driving scenarios [127]. Following this work, Du et al. conducted a comparative study between Dyna and DQN algorithms, revealing DQN's superior performance in approximating DP results and better adaptability. The team further enhanced DQN's performance through experience replay and NAG-Adam optimization techniques. Their final improvement employed DDQN to address Q-value overestimation, yielding an additional 2.52% reduction in fuel consumption compared to standard DQN [128–131]. Zhang et al. developed a hierarchical EMS for a power-split PHEV. The upper layer used DP to plan engine modes and SoC trajectories, while the lower layer applied DQN for real-time control. Training was conducted on standard driving cycles, and testing used real-world road data. Results demonstrated real-time control capability, with fuel consumption reductions of 4% and 10% compared to adaptive equivalent consumption minimization strategy (A-ECMS) and rule-based control strategies, respectively [132]. Several examples of DRL in EMSs of electrified vehicles are summarized in Table 2.1. The table categorizes works by vehicle type, DRL method, state variables, actions, and rewards, offering a concise overview of diverse approaches in EMSs.

Value-based methods, particularly DQN and its variants (double DQN and dueling DQN), have demonstrated significant success in EMS optimization across various HEV architectures. These approaches have consistently achieved impressive results, including fuel consumption reductions of 10-16% compared to rule-based strategies and performance levels reaching up to 93% of DP. The evolution from basic DQN to advanced architectures like DDQN and Dueling DQN has addressed key challenges such as Q-value overestimation and training efficiency. Notable strengths include the ability to handle high-dimensional state spaces, generalization to unfamiliar driving cycles, and real-time control capabilities. However, while these methods have proven effective, their fundamental limitation of requiring discrete action spaces has motivated

the development of alternative approaches for continuous control applications in HEV energy management.

Table 2.1
Comparison of value-based methods in EMSs for typical works

Work	Type	Method	State	Action	Reward
Wu et al. [122]	Power-split	DQN	SoC, engine current power, velocity, acceleration	engine power increment	fuel consumption, SoC stability
Qi et al. [124]	Power-split	dueling DQN (DDQN)	power demand, distance to destination, SoC	engine power output	reciprocal of fuel consumption
Hu et al. [125]	parallel	DQN	total torque demand, SoC	torque-split ratio between engine and battery	reciprocal of fuel consumption
Han et al. [126]	series	double deep Q-learning (DDQN)	power demand, velocity, acceleration, SoC, angular speed	engine rotating speed	fuel consumption, SoC stability
Du et al. [131]	series	DQN	SoC, power demand, generator speed	engine throttle	fuel consumption, SoC stability

Actor-critic methods

While value-based methods show promise, they are limited to discrete action spaces. Actor-critic (AC) methods emerged to handle continuous control variables common in HEV control (e.g., engine speed, torque) [122]. These methods combined the advantages of both policy-based and value-based approaches. Li et al. addressed the discretization problem in DQN by employing the AC algorithm to optimize the EMS of a simplified P1 PHEV. Results showed that fuel consumption increased by only 2.76% compared to DP, while computational load decreased by 88.7% [133]. This study marked the first use of the AC framework for EMSs in continuous action spaces, achieving significant results and establishing the AC framework as a focus for future EMS research.

Despite their advantages, AC methods, as on-policy algorithms, are limited in their ability to utilize historical sampling data and are prone to value overestimation, similar to DQN. To overcome these limitations, advanced algorithms such as proximal policy optimization (PPO) and DDPG were developed. PPO leverages importance sampling to enhance sample efficiency [134], while DDPG combines the strengths of AC and DQN, offering higher efficiency in utilizing historical data [135]. Building on these advancements, Zhang et al. introduced a PPO-LSTM-based EMS for a parallel electric-hydraulic HEV, optimizing mode switching

and balancing performance with Shannon entropy evaluation. The strategy reduced energy consumption by 18.51% (WLTC) and 15.74% (NEDC) while maintaining dynamic performance, showcasing its efficiency and robustness [136]. Wu et al. used DDPG as the optimization algorithm, while considering battery thermal safety and degradation. The total driving energy consumption was further reduced by 23.9% compared to DQN [137]. Zhang et al. applied DDPG with prioritized experience replay (PER) to optimize the EMS of a series HEV. The resulting EMS achieved 93.9% of DP performance and enabled real-time online control [138]. Furthermore, Zhang et al. developed enhanced DDPG-based EMSs for HEVs, integrating locality sensitive hashing (LSH) for better state exploration and PER for improved learning efficiency. It reduced lithium battery power consumption by 8.5%, increased supercapacitor recovery by 6.25%, and cut battery current fluctuations by 15.7% compared to DDPG, DQN, and Q-Learning EMSs [136]. Li et al. proposed a DDPG-based EMS for a series HEV, achieving a 96.5% fuel economy of DP and outperforming MPC-based EMS. It reduced engine start times and demonstrated real-time applicability with a 0.001 s computation speed, adaptable to various HEV types [139]. Hu et al. combined DDPG with an equivalent consumption minimization strategy (ECMS) for a parallel HEV. DDPG optimized the equivalent fuel consumption factor in ECMS while the primary control remained ECMS-based. This approach achieved near-DP optimization results, outperforming standalone ECMS and RL methods while ensuring safe exploration during early training with ECMS's prior knowledge [140]. Chen et al. used DDPG to optimize the EMS of a series-parallel HEV, initializing DDPG with DP results and real-world engine operation data. This approach significantly reduced exploration time during training. Tests using UDDS and real-world driving data showed DDPG outperformed DQN and achieved results close to DP [141].

Additionally, Lian et al. proposed a transfer learning (TL) method [142] to enhance the efficiency of developing DDPG-based EMSs for different HEV types. Knowledge from the DRL-trained EMS for a Prius was transferred to the EMS for a power-split bus, a series vehicle, and a series-parallel bus, followed by fine-tuning on small datasets. Simulations showed TL improved convergence efficiency by 70% and enabled cross-type knowledge transfer, significantly accelerating EMS development for structurally diverse HEVs [143].

While DDPG marked a substantial improvement [144], it is sensitive to hyperparameters and prone to training instability due to Q-value overestimation. To address these challenges, twin delayed DDPG (TD3) was developed, incorporating dual critic networks, delayed updates, and robust noise addition, making it a superior alternative [145]. Zhou et al. used TD3 to optimize the EMS of a P2 parallel HEV, enriching training environments with speed, mass, and gradient variations. Results showed TD3 outperformed DDPG, DQN, Double DQN, and Dueling DQN, achieving 94% of DP optimization performance [146]. Huang et al. utilized TD3 to optimize an EMS for a power-split hybrid electric bus, incorporating battery health considerations. Their approach reduced energy consumption by 9.58% compared to DDPG and 17.29% compared to DDQN, while improving computational efficiency by 10.98% [147]. Liu et al. developed a DRL-based EMS for a power-split HEVs using TD3 with a novel reward function that penalizes irrational actions based on a rough vehicle model. Simulations showed the TD3-based EMS outperformed DDPG, reducing computation time by 10% and fuel consumption by 7.28% in complex tasks involving physical constraints, while achieving comparable results in simpler tasks [148]. Niu et al. proposed TD3-based EMS for a HEV with a novel experience replay mechanism that selects samples via an evaluation network to enhance learning efficiency and convergence. Compared to traditional TD3-based EMS, the proposed method reduced training time by 8.73% and improved fuel economy by 2.14% [149].

To tackle the simultaneous optimization of mode switching and power distribution in series-parallel HEVs, Tan et al. proposed a novel AC framework for energy management in a HEV, addressing challenges like large search space and complex driving conditions. The framework employs an actor network to generate continuous control signals and a critic network to evaluate them, optimizing directly in the continuous space and reducing computation load. Simulation results showed the AC method outperformed discretization-based strategies, surpassing the baseline DP by 94.5%, 97.1%, and 90.5% in CTUDC, WVUCITY, and WVUSUB driving cycles, respectively, with only one-tenth of the computational cost [150]. However, the discrete clutch control in series-parallel systems involves only a single-dimensional action. When multiple discrete actions coexist with continuous actions, two alternative approaches have been developed, Li et al. proposed a terrain-informed EMS for power-split hybrid electric buses using the DDPG algorithm. The method handles hybrid action spaces, combining two

continuous actions for the engine and four discrete actions for powertrain mode selection [151]. Tang et al. introduced a DRL-based EMS and a double DRL (DDRL) framework for HEVs, combining DQN for gear-shifting and DDPG for throttle control. The DDRL framework achieved multiobjective synchronization and demonstrated strong potential for optimizing fuel economy and ensuring real-time performance [152].

Table 2.2
Comparison of actor-critic based methods in EMSs for typical works

Work	Type	Method	State	Action	Reward
Li et al. [133]	Power-split	Deep AC network	Velocity, acceleration, SoC	Engine torque, engine speed, power split factor	Fuel consumption
Zhang et al. [136]	Parallel	PPO	Velocity, accelerator pedal opening, brake pedal opening, SoC, voltage, motor speed, motor torque, HPA pressure, LPA pressure	Typical working modes	Velocity deviation, SoC deviation
Li et al. [139]	Series	DDPG	Velocity, acceleration, SoC, engine power, traveled distance, SoC deviation	Increment of engine power	Fuel consumption rate, SoC deviation
Lian et al. [143]	Four types of structures	DDPG-TL	Velocity, acceleration, SoC	Engine torque, engine speed, motor torque	Fuel consumption rate, SoC deviation
Zhou et al. [146]	Parallel	TD3	Velocity, acceleration, SoC	Engine torque	Fuel consumption, SOC stability
Liu et al. [148]	Power-split	TD3	Velocity, acceleration, SoC	Engine power	Fuel consumption rate, SoC deviation

Several works are summarized in Table 2.2. The table categorizes these studies by vehicle type, method, state variables, actions, and rewards, providing a comprehensive overview of their approaches. It demonstrates the versatility of AC methods such as Deep AC, PPO, DDPG, and TD3 in optimizing EMSs for different vehicle powertrain configurations. Actor-critic methods, particularly DDPG and its advanced variant TD3, have revolutionized EMS optimization by effectively addressing continuous action space challenges. These algorithms have demonstrated remarkable improvements in fuel economy and computational efficiency, with TD3 achieving up to 94% of DP performance. The progression from basic AC to DDPG and TD3 has systematically resolved challenges such as sample efficiency, training stability, and hyperparameter sensitivity. Integrating these methods with classical techniques (e.g., ECMS),

incorporating battery health considerations, and extending their applicability across diverse HEV architectures highlight their transformative potential in EMS optimization.

Distributed policy optimization methods

Distributed policy optimization methods emerged as a breakthrough solution to address the computational bottlenecks in EMS development and optimization processes. These approaches fundamentally revolutionized the training efficiency of DRL algorithms through parallel computing architectures. The two prominent algorithms, named asynchronous advantage actor-critic (A3C) [153] and DPPO [154], demonstrated remarkable improvements in both computational efficiency and optimization performance. Zhou et al. leveraged A3C to enhance simulation efficiency, reducing the computational load to just one-eighth of that required by DDPG, while maintaining superior optimization performance and enabling real-time control in simulation environments [155]. Biswas et al. applied A3C to develop an online EMS for a power-split HEV, using Markov Chain models and historical data for future driving cycle prediction. The A3C-trained EMS achieved near-DP performance on familiar cycles and 94% of DP performance on unfamiliar ones, outperforming PPO [156]. Huang et al. developed distributed a DRL-based EMS for a fuel cell HEV (FCHEV) using the A3C with multi-process parallel computation. It improved convergence speed by over 87% and reduced operating costs by up to 44%, optimizing resource use and hydrogen efficiency [157]. Similarly, Tang et al. developed an EMS for a parallel HEV, combining rule-based shift strategies with DQN, A3C, and DPPO. A3C and DPPO achieved 94% of DP optimization performance with four times the computational efficiency of DQN, demonstrating the benefits of distributed computing [158].

These distributed methods proved particularly valuable in simulation platform optimization, where rapid iteration and development cycles are crucial. The success of these distributed approaches not only accelerates the development process but also opens new possibilities for implementing more complex and sophisticated EMSs that would have been computationally prohibitive with traditional methods [159]. Their ability to maintain high performance while significantly reducing computational overhead has made them increasingly popular in both research and practical applications, especially in scenarios requiring rapid prototyping and testing of EMSs.

DRL-based EMSs have evolved significantly through multiple generations of algorithms, each addressing specific challenges in EMSs of electrified vehicles. Early AC methods demonstrated strong adaptability in unfamiliar environments, while DDPG emerged as a breakthrough by combining the advantages of both DQN and AC frameworks to improve historical sample efficiency. The development of TD3 further refined these capabilities by addressing the stability issues and hyperparameter sensitivity inherent in DDPG. Meanwhile, distributed approaches like A3C have shown promise in reducing computational overhead while maintaining optimization quality. However, fundamental trade-offs persist between algorithms: stochastic policy methods (AC, A3C, PPO) offer superior environmental adaptability but lower sample efficiency, while deterministic approaches (DDPG, TD3) excel in sample efficiency but may struggle with environmental generalization. Some hybrid approaches that effectively balance these competing objectives may leverage advanced techniques such as TL, multi-agent systems, or the integration of domain knowledge. These methods can enhance adaptability and efficiency, enabling superior performance across diverse driving scenarios.

2.2.3 Challenges

The demand for decarbonization and enhanced vehicle performance has accelerated the shift towards electrification, with EVs like PHEVs, BEVs, and FCEVs leading this transition [160]. EMSs are vital for optimizing energy efficiency and maintaining powertrain health [161]. Multi-mode PHEVs, capable of operating in pure battery, series hybrid, or parallel hybrid modes, introduce unique challenges due to their complex MIMO control requirements, unlike traditional single-input single-output (SISO) systems [162, 163]. DRL has shown promise for EMS optimization, effectively managing high-dimensional inputs such as SoC, power demand, and vehicle velocity [164]. However, its deployment faces significant challenges, including high computational complexity, real-time decision-making constraints, and limited generalization to dynamic and uncertain environments. Models often require substantial resources, struggle to adapt to unfamiliar driving conditions and fail to ensure optimal performance in varying scenarios.

Existing approaches using independent DRL agents for tasks like gear shifting or throttle control [152] lack communication and coordination, limiting global optimization. Recent

MARL frameworks show potential for multi-objective EMSs [165, 166], but challenges in agent collaboration and robustness remain. To address these limitations and build on previous MARL applications in CACC, a MADRL framework can be applied to multi-mode PHEVs, enabling cooperative agent interaction in MIMO control to enhance to improve energy efficiency and control performance in electrified vehicles.

2.3 | Q learning of EMSs for electrified vehicles

Q-learning has emerged as a promising EMS approach for electrified vehicles due to its adaptability, efficiency, and practicality. First, when the hybrid powertrain structure is not overly complex and operates in a specific mode, Q-learning serves as a viable alternative to traditional optimization methods, delivering competitive performance. Second, its discrete action space aligns naturally with power split decisions, where control actions involve selecting from a finite set of power distribution options. Third, the simple architecture of Q learning facilitates straightforward implementation and verification in real vehicle systems, making it practically appealing for automotive applications. Moreover, Q-learning's ability to learn optimal policies through direct interaction with the environment allows it to adapt to varying driving conditions, while its low computational requirements make it suitable for real-time control applications. Combined with its proven convergence properties, Q-learning is particularly effective for energy management in HEVs, where system complexity remains within manageable bounds.

2.3.1 General problem formulation in Q learning

Problem formulation

Based on MDP defined by $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \gamma)$, the core objective of Q-learning in EMSs is to estimate the optimal action-value function $Q^*(s,a)$, representing the long-term reward of taking action a in state s and following the optimal policy.

Q-learning learns $Q^*(s,a)$ iteratively using the Bellman equation:

$$Q^*(s,a) = \mathcal{R}(s,a) + \gamma \max_{a'} Q^*(s',a'), \tag{2.8}$$

where s' is the next state and a' represents possible future actions.

The Q-value update rule is:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[\mathcal{R}(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a) \right], \tag{2.9}$$

where α is the learning rate. This iterative approach refines the Q-values based on environmental feedback, gradually converging to the optimal policy. Q-learning converges to the optimal Q-function Q^* with probability 1 under three key conditions:

1) The state and action spaces are finite, which naturally aligns with the discrete control nature of power split decisions in PHEVs.

2) Every state-action pair is visited infinitely often, ensuring comprehensive exploration of all possible vehicle operating conditions.

3) The learning rate $\alpha_t(s,a)$ follows a specific decay pattern: $\sum_{t=0}^{\infty} \alpha_t(s,a) = \infty$ and $\sum_{t=0}^{\infty} \alpha_t^2(s,a) < \infty$.

This condition ensures sufficient learning while gradually stabilizing the learned policy.

The convergence of Q-learning in EMS applications can be established through theoretical analysis, demonstrating its ability to reliably learn optimal energy management policies for electrified vehicles. This analysis validates its effectiveness while also offering insights into hyperparameter tuning, enabling performance optimization and stability in real-world implementations. In the application-specific chapters, we will further present corresponding theoretical analyses that build on this foundation.

2.3.2 Applications of Q-learning in EMSs

Following the theoretical foundation established above, the practical applications of Q-learning in EMSs have demonstrated its effectiveness across diverse electrified vehicle scenarios. These implementations have evolved from basic table-based approaches to sophisticated variants, each addressing specific challenges in energy management optimization. The following sections examine this evolution, beginning with **basic Q-learning methods** that established the fundamental framework for RL in EMSs, progressing through **enhanced Q-learning variants**

that addressed early limitations, and culminating in **hybrid approaches** that combine Q-learning with other control strategies.

Basic Q-learning methods

The foundational implementation of Q-learning in EMSs utilized a traditional state-action table approach, providing a straightforward solution for discrete action spaces in hybrid vehicle control by mapping system states (e.g., battery SoC and power demand) to control actions through a Q-table. This tabular RL algorithm stores Q-values for different actions in each state and updates them iteratively through interactions between the agent and environment, using a greedy approach to select actions with the highest reward. While this method demonstrats the feasibility of RL in EMSs and achieves real-time optimization control with adaptability to unfamiliar driving scenarios, it faces limitations in handling complex state spaces and significant storage requirements [167]. Despite these constraints, these early applications established a critical foundation for more advanced Q-learning-based EMS implementations, which have since proven superior to traditional EMS methods in both optimization performance and real-time capabilities [168].

Relevant studies have examined the effectiveness of Q-learning in EMSs by addressing four key aspects: *state and action selection*, *state and action discretization*, *action exploration*, and *learning experience selection*. These early works primarily focused on improving system performance by refining fundamental aspects such as the choice of states and actions, discretization techniques, and learning rates, providing valuable insights into the intrinsic properties of Q-learning for EMSs. Such studies have significantly influenced subsequent advancements in Q-learning methods and their optimization for more complex hybrid vehicle systems.

The *selection of states and actions* plays a pivotal role in Q-learning-based EMSs as it directly impacts the performance of the learning algorithm. The summarized selections in Table 2.3 highlight the most commonly used state variables, including SoC, power demand, vehicle speed, and acceleration.

The *discretization of states and actions* is another critical aspect of early Q-learning applications. Tabular RL methods, as highlighted by Sutton et al. [174], are effective for small state-action spaces, where exact optimal value functions and policies can be determined. However, as

Table 2.3
Several works in Q-learning Based EMSs

Work	Type	States	Action	Reward
Xiong et al. [169]	Power-split	SoC, ultracapacitor voltage, power demand	current of the battery pack	Total energy loss
Liu et al. [170]	Parallel	SoC, power demand, remaining distance to travel	engine torque, gear shifting	fuel consumption, battery energy usage cost
Zou et al. [171]	Series	SoC, engine speed	SoC,engine torque	Fuel consumption
Fang et al. [172]	Parallel	SoC, velocity, acceleration, vehicle location	Motor torque	Fuel consumption, SoC deviation
Qi et al. [173]	Power-split	SoC, power demand	engine power	fuel consumption

state and action spaces grow, tabular methods face scalability issues, requiring approximate RL methods to handle the complexity. Xu et al. conducted a parametric study to examine the impact of state-action discretization, and learning experience selection on RL-based EMS performance. The study revealed that finer action discretization improves fuel economy, while excessive state discretization may increase fuel consumption [168]. Yuan et al. demonstrated the trade-off between finer discretization and computational efficiency, adopting 200 action values to optimize EMS performance in HEVs [175]. Cao et al. emphasized the importance of balancing exploration and exploitation, particularly in discretized action spaces, to achieve better learning outcomes [176].

For *action exploration* and *learning experience selection*, advancements in these areas have significantly improved the efficiency and adaptability of Q-learning-based EMSs. These foundational developments have paved the way for numerous enhanced Q-learning methods, addressing the limitations of basic Q-learning in handling complex state-action spaces and achieving better real-time performance, which will be discussed in the following section.

Enhanced Q-learning methods

Based on the basic Q-learning review, enhanced Q-learning methods have emerged to address the limitations of basic implementations, particularly in handling complex state-action spaces, improving convergence, and balancing exploration-exploitation trade-offs. These advancements can be broadly categorized into three key areas: *exploration strategies*, *learning process optimization*, and *ensemble methods*, each focusing on improving the efficiency, robustness, and adaptability of Q-learning-based EMSs.

Exploration strategies is a fundamental challenge in Q-learning, requiring a balance between discovering new policies (exploration) and optimizing known policies (exploitation). Early studies used simple strategies such as fixed or decaying ϵ-greedy approaches, which control the probability of random action selection. Xu et al. implemented Q-learning with a fixed exploration-to-exploitation ratio (E2E) of 0.25 in a 48V mild parallel HEV, demonstrating its efficacy in mid-sized systems [177]. Qi et al. used an E2E ratio of 0.7 to prioritize exploration for real-time fuel consumption optimization in PHEVs [173]. Liu et al. adopted an E2E ratio of 0.1 for RL-based EMSs with trip information, achieving significant fuel economy improvements by leveraging GPS-derived travel distances [170]. Adaptive exploration strategies, such as exponential decay functions, were later introduced to dynamically adjust E2E ratios throughout the learning process, improving efficiency across different learning stages [178]. Fayyazi et al. proposed a self-adaptive Q-learning (SAQ-learning) EMS to optimize energy consumption and emissions in real-time. The method introduces a novel exploration strategy, replacing traditional decayed ϵ-greedy, enabling the system to adapt to unknown situations efficiently. Simulations demonstrated that SAQ-learning improves operational time by 23% over standard Q-learning and outperforms DP and fuzzy logic methods [179].

Optimizing the learning process has been a focus of enhanced methods, targeting issues such as overestimation bias, slow convergence, and suboptimal updates [124, 168]. The bias estimation due to the maximization in Q-learning has long been recognized as a significant challenge to effective learning. This issue was first brought to attention by [180], and [181] later introduced double Q-learning to address this bias in the tabular case, demonstrating that it generally leads to an under-estimation bias. Hasselt et al. introduced double Q-learning to address the overestimation bias in traditional Q-learning. By decoupling action selection and evaluation during Q-value updates, double Q-learning reduces bias and enhances policy stability, particularly in environments with high uncertainty [181]. Meng et al. proposed a double Q-learning-based EMS for FC hybrid systems to address the overestimation issues in traditional Q-learning and DQN algorithms, which cause instability and poor performance. Simulations demonstrated the effectiveness of the proposed strategy in improving energy efficiency and reducing overall energy consumption [182]. Han et al. proposed an EMS for a HEV based on double Q-learning to optimize energy distribution between the engine and battery, ensuring the

engine operates in a high-efficiency zone. By addressing the maximization bias in Q-learning, the double Q-learning EMS achieved better control performance. Simulations comparing double Q-learning, rule-based, and Q-learning strategies showed that the proposed method improved fuel economy and maintained SoC stability effectively [183]. Second strategies, such as weighted Q-value estimates and correction terms, further refine the Q-value updates [184]. These approaches mitigate error propagation and improve convergence rates, making them suitable for dynamic EMS applications [185, 186]. The third method is the multi-step method, multi-step temporal difference (TD) learning improves learning efficiency by propagating rewards over multiple steps, enhancing the agent's ability to learn long-term dependencies in complex environments [187]. Zhou et al. developed model-free predictive EMS for electrified off-highway vehicles using a multi-step RL algorithm. Tests showed a 71% longer prediction horizon and 7.8% energy savings compared to model-based methods, with efficiency improving from 34% to 44% in 5 hours [188]. The fourth one is prioritized experience replay (PER). By prioritizing transitions with high TD errors during training, PER ensures that the agent focuses on critical experiences, accelerating convergence and improving learning efficiency in EMSs [189]. Du et al. proposed a novel RL framework for the EMS in a series HEV using double Q-learning with modified prioritized experience replay (MPER). It achieved higher efficiency, reduced energy consumption, and near-optimal fuel economy while demonstrating robustness across driving cycles [131].

Ensemble methods have further enhanced Q-learning-based EMSs by leveraging multiple agents or policies to optimize performance [190, 191]. For Q-learning-based approaches [192] demonstrated that averaging multiple Q estimates can reduce variance [193] introduced random ensemble mixture, which enforces optimal Bellman consistency by combining random convex mixtures of multiple Q estimates. Xu et al. proposed an ensemble RL strategy to enhance fuel economy in parallel hybrid electric vehicles. The method combines multiple Q-learning agents with different state combinations and integrates thermostatic and equivalent consumption minimization strategies as individual agents. Actions are determined by a weighted average of decisions from all agents, with Q-values updated after each driving cycle. Compared to a single RL agent, the ensemble strategy improved fuel economy by 3.2% [194]. Yang et al. proposed a multi-objective EMS for a HEV using multi-agent Q-learning to optimize fuel

economy, maintain SoC, reduce battery degradation, and constrain ultracapacitor SoC. Viewing the engine–generator set and hybrid energy storage system as intelligent agents in a stochastic game, MARL ensures a balance of objectives by achieving Nash equilibrium. Simulations showed that the MARL-based EMS outperformed single-agent RL and DP in optimizing multiple objectives [195]. Lan et al. introduced Maxmin Q-learning, which significantly improved learning stability by maintaining multiple Q-value estimates and using their minimum value for action selection. This method proved particularly effective in mitigating overestimation bias in discrete action spaces [196]. While subsequent research has shown that the update-to-data (UTD) ratio method can substantially enhance performance, the theoretical foundations and experimental validations explaining these improvements remain limited [197].

Table 2.4
Detailed comparison of enhanced Q-learning methods

Method Category	Key Techniques	Main Advantages	Research Outcomes
Exploration Strategies	· Fixed/Decaying ϵ-greedy · Adaptive E2E ratios · SAQ-learning	· Balance exploration and exploitation · Dynamic adaptation to varying conditions · Enhanced capability in handling unknown scenarios	· Fuel economy improvements · Better torque generation · Route-based optimization improvements
Learning Process Optimization	· Double Q-learning · Bias correction · Multi-step TD · Prioritized Experience Replay (PER) · Clipped-double Q-learning	· Mitigates overestimation bias · Faster convergence · Improved long-term dependencies · Focused learning on critical experiences · Enhanced stability for EMS control	· Enhanced energy efficiency · Reduced error propagation · Convergence acceleration
Ensemble Methods	· Q-value averaging · Random ensemble mixtures · Multi-agent Q · Maxmin Q-learning · UTD methods	· Variance reduction · Improved Bellman consistency · Optimization across multiple objectives · Stabilized learning in complex environments	· Fuel economy improvement · Superior SoC maintenance · Reduced battery degradation · Enhanced stability in discrete action spaces · Superior to single-agent RL frameworks

Table 2.4 summarizes key advancements in enhanced Q-learning methods for EMSs, highlighting techniques, advantages, and outcomes in hybrid vehicle applications. These advancements remain within the framework of traditional Q-learning, addressing its key challenges such as overestimation bias, inefficient learning from experiences, and slow convergence. By refining these aspects, Q-learning has been made more reliable and adaptable for EMSs in HEVs, particularly in scenarios with moderate state-action spaces and discrete control tasks.

Hybrid methods

The integration of Q-learning with complementary techniques represents a significant advancement in EMSs. By combining the adaptive learning capabilities of Q-learning with established control strategies, these hybrid methods address the limitations of pure Q-learning [198], such as slow convergence and limited adaptability. This approach enhances real-time performance, fuel efficiency, and operational stability, making it highly effective for HEVs. Key strategies include the integration of Q-learning with rule-based methods, genetic algorithms (GA), MPC, and optimization techniques, all of which have demonstrated significant performance improvements in various applications.

Li et al. developed a Q-learning-based EMS for a HEV, incorporating deterministic rules for real-time power allocation. This integration reduced fuel consumption and minimized fuel cell current fluctuations, showcasing its effectiveness for improving operational stability in hybrid systems [199]. Similarly, Ghaderi et al. proposed a three-layer EMS for a recreational vehicle, combining Q-learning with online estimators and rule-based activation. This method achieved up to a 13.08% reduction in trip costs compared to other online strategies, emphasizing its efficiency in cost-sensitive applications [200].

In another approach, Tao et al. introduced a novel fuzzy EMS combining Q-learning with GA optimization for a HEV. The GA optimized initial Q-table values, enabling better adaptability without requiring prior knowledge of driving patterns. Simulation results demonstrated significant reductions in current fluctuations (6.9%–41.5%) and hydrogen consumption (0.35%–6.08%), while maintaining the SoC within safe limits, thus ensuring system reliability [201]. Chen et al. integrated double Q-learning with MPC in an EMS for a PHEV, leveraging an Elman neural network for vehicle speed prediction. This hybrid method enabled rolling optimization

of battery output, achieving fuel economy comparable to stochastic dynamic programming (SDP) while maintaining adaptability to various SoC reference trajectories [202]. Yin et al. further enhanced a Q-learning-based EMS by developing a hierarchical MPC strategy for HEV platoons. Using V2V communication, this approach maintained safe spacing and low-speed errors while reducing computation time by 23% compared to SDP, making it suitable for dynamic driving conditions [203]. Similarly, Shen et al. combined double-delay Q-learning with MPC and a CNN-based speed predictor for a PHEV, achieving near-optimal fuel consumption with a computation time under 23 ms, demonstrating its real-time feasibility [204].

Zhang et al. applied a Q-learning multi-objective particle swarm optimization (QMOPSO) algorithm for a connected FCHV, addressing high-dimensional optimization challenges. The QMOPSO method improved population initialization and subpopulation division, leading to enhanced tracking stability (50.20%), fuel economy (6.10%), and reduced power source degradation (2.04%) compared to standard PSO approaches [205]. Mortabit et al. integrated ECMS with Q-learning for torque estimation in the P4 HEV. The integration achieved torque estimates comparable to ECMS under new driving cycles while highlighting the potential of combining machine learning with traditional methods to improve control efficiency [206]. Lin et al. developed an online recursive EMS for PHEV using Q-learning with a cosine similarity-based update mechanism. The strategy adopted to varying driving conditions and outperformed rule-based and equivalent consumption minimization strategies, achieving significant energy savings over long distances [207]. Xiong et al. developed a RL-based EMS for a PHEV. Using a Markov-based power transition probability matrix (TPM) and KL divergence for adaptive updates, the strategy reduced energy loss by 16.8% compared to rule-based methods. It limits battery discharge, reduces charging frequency, and adapts to varying conditions, enhancing system efficiency [169]. Zou et al. developed a RL-based EMS for a hybrid tracked vehicle, optimizing energy allocation between the engine-generator and battery. Using a Markov chain algorithm with KL divergence for real-time updates, the strategy improved fuel efficiency and demonstrated robust performance under field-tested driving schedules [171]. Liu et al. proposed transfer RL-based adaptive EMSs for parallel HEVs. The bi-level approach involved driving cycle transformations using TPM and introduced matrix norm (IMN) to adapt Q-value tables and determine control strategies [208].

Table 2.5
Summary of hybrid methods in EMSs

Works	Performance	Powertrain types	Hybrid techniques
[202, 203, 209]	Optimization effect	HEV, PHEV	MPC
[201]	Optimization effect	FCHEV, PHEV	GA
[199, 200]	Optimization effect	FCHEV, HEV	RB
[210]	Optimization effect	PHEV	Neuro-DP
[206]	Optimization effect	HEV	ECMS
[205]	Optimization effect	PHEV	PSO
[169, 208]	Convergence rate	PHEB	MC & TPM
[208]	Convergence rate	HEV	TL
[171]	Real-time performance / adaptability	HEV	MC & KL divergence
[204]	Real-time performance / adaptability	Plug-in FCEV, PHEV	NNs
[207, 211]	Real-time performance / adaptability	PHEV	Cosine similarity and forgetting factor

These hybrid Q-learning methods demonstrate their effectiveness in enhancing EMSs for HEVs, as summarized in Table 2.5. By leveraging hybrid techniques, these approaches address critical challenges, such as improving real-time adaptability, optimization efficiency, and convergence rate, while maintaining system reliability and stability in dynamic driving conditions.

2.3.3 Challenges in Q-Learning for EMSs

Q-learning offers great potential for EMSs in electrified vehicles, but several challenges limit its practical application. One key issue is the overestimation problem in value function approximation, which often leads to suboptimal policies. Although techniques like double Q-learning [181] and clipped-double Q-learning [145] have mitigated these biases, ensuring robust performance across diverse driving scenarios remains difficult. Scalability is another major limitation, as traditional Q-learning struggles with high-dimensional state-action spaces in complex vehicle models, such as those involving SoC, power demand, and traffic conditions. Additionally, convergence issues caused by the exploration-exploitation trade-off and adaptability to real-world uncertainties, such as sensor noise and dynamic traffic conditions, further hinder its effectiveness.

Advancements like ensemble methods [192], weighted Bellman updates [187], and Maxmin Q-learning [196] have improved robustness and stability, while end-to-end learning approaches simplify control pipelines [212]. However, these methods often lack generalization to multi-objective, real-world EMS problems. To address these gaps, hybrid Q-learning frameworks incorporating ensemble learning, bias correction, and sample efficiency are needed to ensure adaptability, real-time, and robust energy management for electrified vehicles.

2.4 | Summary and research gap

This chapter reviewed recent advancements in intelligent energy management for electrified vehicles, focusing on CACC for CAVs and RL-based EMS methods, including DRL and Q-learning. CACC strategies, ranging from rule-based to optimization-based and learning-based approaches were shown to improve traffic stability, safety, and energy efficiency. DRL demonstrates exceptional flexibility in optimizing complex EMS tasks in dynamic environments, while Q-learning offers efficient solutions for simpler problems. The existing literature highlights the key projected trends and associated research gaps in these areas, which can be summarized as follows:

(1) While significant advancements have been made in decentralized cooperative frameworks, such as MARL, current approaches predominantly focus on improving traffic efficiency and safety. However, critical aspects such as energy consumption and communication efficiency have been largely overlooked. The lack of energy-aware strategies and effective communication protocols hinders the potential of these systems to deliver holistic benefits, particularly in scenarios involving high vehicle density or real-time dynamic interactions. Addressing these gaps is essential to ensure that decentralized cooperative systems are both energy-efficient and scalable in dynamic environments.

(2) Current EMSs for hybrid powertrains, especially multi-mode PHEVs, struggle to address the challenges of MIMO control. Traditional single-agent strategies cannot adequately handle the complex interactions required to balance multiple objectives, such as fuel economy, and emissions reduction. Additionally, research on RL-based EMSs for multi-mode PHEVs remains limited, with most studies focusing on simpler powertrain configurations. The lack of robust

frameworks for simultaneous optimization of multiple objectives in complex hybrid systems represents a significant gap that must be addressed to fully exploit the potential of these advanced powertrains.

(3) Ensemble methods in RL have shown promise for enhancing stability and robustness by leveraging multiple base agents to represent and learn value functions. However, a critical challenge lies in determining the optimal coordination and distribution of responsibilities among these agents, as traditional ensemble frameworks often assume equal separation power for all agents. Additionally, the role of the E2E ratio within ensemble configurations remains poorly understood, limiting the effectiveness of these methods in addressing complex, real-time energy management tasks. Overestimation bias, a persistent issue in Q-learning-based approaches, further complicates the design of ensemble frameworks, as it can degrade policy reliability in real-world applications. These gaps underscore the need for ensemble RL frameworks that not only address coordination and bias mitigation but also ensure robust performance across diverse driving conditions.

(4) While many intelligent control strategies demonstrate promising results in simulation environments, their practical deployment faces significant challenges. Moreover, the scalability and adaptability of these strategies in diverse driving scenarios and traffic networks remain underexamined. Comprehensive validation frameworks are needed to bridge the gap between theoretical development and practical application, ensuring the reliability, scalability, and effectiveness of these systems.

CHAPTER 3

Methodology and Facility

3.1 | Methodologies

In this section, we begin with a comprehensive overview of the fundamental principles of RL. We then delve into various prominent MARL algorithms, establishing the context to enhance the understanding of our methodologies.

3.1.1 Research target

The primary research target of this PhD study is to develop an energy-efficient and stable control in CAVs and PHEVs, as illustrated in Fig. 3.1. This study is structured both top-down (from macro-level cooperative control to micro-level energy management) and left-to-right (from application to theory analysis), reflecting a progressive transition from fleet-level coordination to component-level optimization and real-world deployment.

The overall research is divided into three interconnected parts, each following a consistent methodology pipeline including vehicle modelling, optimization problem formulation, control strategy design, and validation: 1) At the macro level, this part first focuses on the co-optimization of platoon stability and energy efficiency within CACC systems. This is achieved through the development of a MARL framework that models traffic scenarios, formulates decentralized decision-making problems, and designs a communication-efficient mechanism; 2) building upon this, this part transitions to the micro level, where it explores EMS for PHEVs. For multi-mode PHEVs, a MARL-based EMS is proposed, incorporating multiple-input multiple-output (MIMO) optimization and a hand-shaking mechanism to manage mode transitions and interaction between different power sources. This design addresses the inherent complexity of multi-mode operation and enables more coordinated energy control; 3) finally, for

specified-mode PHEVs, RL is employed to optimize EMS under various operating conditions. This part focuses on resolving algorithmic limitations, such as overestimation bias and inefficient learning, through theoretical analysis and HiL validation. These methods are validated through real-world data, theoretical analysis, and HiL testing, enhancing the practical implementation and efficiency of EMSs in electrified vehicles.

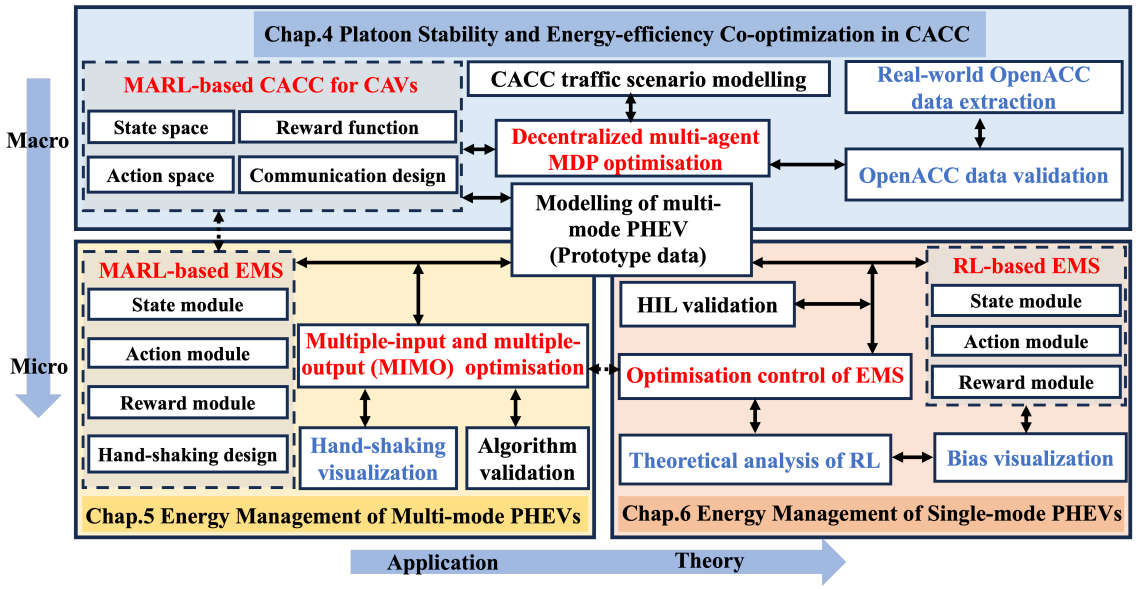


Fig. 3.1. The overview framework of the whole PhD research.

3.1.2 Preliminaries of RL

RL, frequently modeled through a mathematical framework known as a MDP, has risen as an effective method for data-driven sequential decision-making [213]. Recently, deep neural networks (DNN) have substantially elevated the capability of RL to manage complex problems [214]. Key developments include sophisticated algorithms such as the DQN[214], DDPG [135], and A2C [215]. For instance, the AlphaStar, which operates on principles similar to DQN, marked a significant milestone by outperforming professional esports players in StarCraft II, illustrating the power of RL in strategic game environments [216]. Moreover, the development of the autonomous driving system by leveraging DDPG exemplifies the potential of RL, showcasing how vehicles can autonomously navigate through environments with dynamic obstacles [217].

In the context of a RL framework (as shown in Fig. 6.1), a learner known as the **agent** navigates through the **environment** using a process of trial and error. This agent assesses

situations, undertakes specific **actions** in the environment, and consequently receives feedback in the form of a **reward** signal and a **new state**. The feedback, provided by the environment, serves as an indicator to the agent about the effectiveness of its actions, signaling whether they have a positive or negative impact. Many RL problems are framed as **MDP**, which provides a mathematical framework for modeling decision-making where outcomes are partly random and partly under the control of a decision-maker.

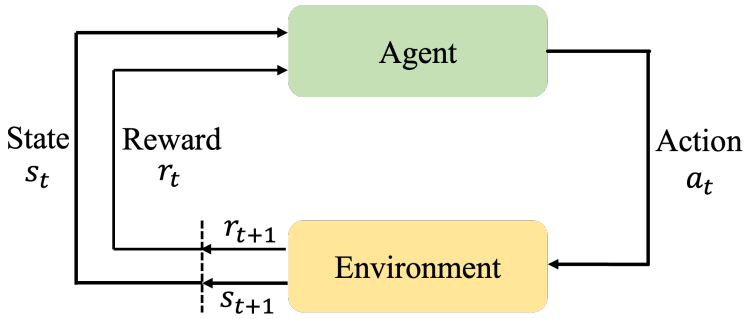


Fig. 3.2. Illustration of RL.

In the following subsections, we will explore three prevalent RL algorithms, including **value-based methods**, **policy-based methods**, and **actor-critic methods**.

Value-based methods

In the domain of value-based methods within RL, the Q-function, represented as Q_θ, is usually parameterized by a set of parameters denoted as θ. This parameterization facilitates using different function approximators for estimating Q-values, including traditional Q-tables [218] and the complex DNN [214]. The core mechanism for updating these parameters relies on the temporal difference (TD), expressed as $(\mathcal{T}Q_{\theta^-} - Q_\theta)(s_t, a_t)$, where $\mathcal{T}$ signifies the dynamic programming operator and Q_{θ^-} represents a version of the model parameters θ^- that is temporarily fixed to stabilize updates [72]. To mitigate the variance in Q-value estimates and enhance the exploration ability of the algorithm, strategies such as the ϵ – greedy method and experience replay mechanism are frequently employed within deep Q-learning frameworks [219]. The optimal action $a^*(s)$ can be described as:

$$a^*(s) = \arg\max_a \{Q^*(s_t = s, a_t = a)\}, \tag{3.1}$$

Some widely recognized deep Q-learning-based algorithms include DQN [220], DDQN [221], and hindsight experience replay (HER) [222].

Policy-based methods

Diverging from the value-based Q-learning approach, policy-based methods prioritize direct manipulation of the policy itself, denoted as π_θ, via a distinct set of parameters θ. The primary goal of adjusting θ revolves around amplifying the probability of selected actions and the overall accumulation of rewards. This process is guided by a specifically defined loss function, which is mathematically represented as:

$$\nabla_\theta \mathcal{L}(\pi_\theta) = \mathop{\mathbb{E}}_{\tau \sim \pi_\theta}[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) R_t], \tag{3.2}$$

Subsequently, the parameters of the policy network are updated by employing stochastic gradient ascent, facilitating an iterative improvement of the policy, as detailed by

$$\theta_{k+1} = \theta_k + \alpha \nabla_\theta \mathcal{L}(\pi_\theta), \tag{3.3}$$

In contrast to Q-learning methods, policy gradient techniques demonstrate enhanced resilience to the non-stationary dynamics characterizing individual trajectories, albeit at the cost of potentially higher variance in outcomes [223]. Among the most prominent algorithms that embody the principles of policy gradients are DDPG [135], Soft Actor-Critic (SAC) [224], and Asynchronous Advantage Actor-Critic (A3C) [215].

Actor-critic methods

Actor-critic methods aim to mitigate the high variability in outcomes often seen with policy gradient techniques by integrating an advantage function. This function enhances the policy gradient methodology by utilizing both the policy update (Actor) and a value estimation function (Critic) [215]. The advantage function, a pivotal element in this approach, is defined as:

$$A^\pi(s_t, a_t) = Q^{\pi_\theta}(s_t, a_t) - V_w(s_t), \tag{3.4}$$

where the update of parameters θ is guided by a policy loss function articulated as:

$$\nabla_\theta \mathcal{L} = \mathop{\mathbb{E}}_{\pi_\theta}[\sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t) A_t], \tag{3.5}$$

Concurrently, the critic value function is refined through:

$$\mathcal{L} = \min_w \mathop{\mathbb{E}}_{\mathcal{D}}[(R_t + \gamma V_{w^-}(s_t) - V_w(s_t))^2], \tag{3.6}$$

with $\mathcal{D}$ denoting the experience replay buffer, a repository for history experiences. This buffer, in collaboration with parameters from preceding iterations often employed within a target network, facilitates the learning process [225].

Despite considerable advancements, single-agent RL algorithms frequently encounter scalability challenges, especially in complex, real-world physical environments that involve multiple agents. These difficulties stem from inherent non-stationarities and the partial observability characteristic of multi-agent scenarios [98].

3.1.3 Preliminaries of MARL

In systems with multiple agents in Fig. 3.3, not only the reward agents themselves but also the reward of their neighbors will be influenced by their actions. MARL has been widely used in various complex systems, including managing traffic [223], strategic gameplay [226], optimizing wireless network resources [227], and configuring power grids [228], to highlight a few areas. A fundamental approach within this field is independent Q-learning (IQL), where each agent's Q-function is primarily developed based on local actions, as $Q_i(s, a) \approx Q_i(s, a_i)$, simplifying the learning process. Similarly, the independent advantage actor-critic (IA2C) [72] represents a variant of MARL, indicating the breadth of strategies explored in this space. However, both IQL and IA2C face challenges in applications due to the partial observability and dynamic environment, as they operate under the assumption that all other agents' actions are part of the environment, which complicates learning when agents' strategies continually evolve [223]. Multi-agent systems are often treated as networks $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ without a central controller [223, 228], where agents $i \in \mathcal{V}$ communicate with nearby peers $\mathcal{N}_i := \{j|\varepsilon_{ij} \in \mathcal{E})\}$ via the edge connections $\varepsilon_{ij}, i \neq j$. This decentralized framework, in which each agent has a

limited perception of the environment, closely mirrors real-world scenarios. This structure effectively simulates the dynamic environment as a partially observable MDP (POMDP) $\mathcal{M}_{\mathcal{G}} = (\{\mathcal{A}_i, \mathcal{S}_i, \mathcal{R}_i\}_{i\subseteq\mathcal{V}}, \mathcal{P})$, with the system's behavior determined by the collective actions ($\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_N$) of all agents and the state transitions (with the state: $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_N$) governed by probability distributions $\mathcal{P}(s'|s, a)$. Rewards play a crucial role in guiding agents towards achieving goals, which vary across *cooperative*, *competitive*, and *mixed* scenarios. In cooperative environments, agents work together towards a common goal, possibly sharing a uniform reward ($\mathcal{R}_1 = \mathcal{R}_2 = \cdots = \mathcal{R}_N$) or aiming to optimize an average reward that accommodates differences among agents (*team-average* reward [229]). Conversely, competitive scenarios feature agents with individualistic goals, often modeled as zero-sum games ($\bar{\mathcal{R}} = \frac{1}{N}\sum_{i\in\mathcal{V}} \mathcal{R}_i(s, a, s')$), where one agent's gain is the loss of another. Mixed environments (also known as *general sum* game settings) allow for both cooperation and competition, with agents pursuing individual yet not directly conflicting objectives.

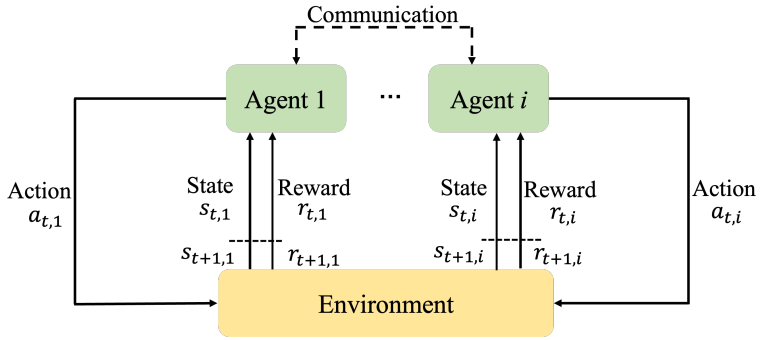


Fig. 3.3. Illustration of MARL.

3.1.4 Training and execution strategies in MARL

In MARL settings, different approaches and frameworks are employed to tackle the complex challenges of coordinating and training agents in decentralized environments. This subsection explores two fundamental paradigms: Centralized Training with Decentralized Execution (CTDE) and Decentralized Training with Decentralized Execution (DTDE), each offering distinct insights and trade-offs in the pursuit of effective multi-agent learning [230–232].

Centralized training with decentralized execution (CTDE)

In this framework, the decentralized problem is initially transformed into a centralized one, solvable by a central controller, which gathers essential training information, e.g., observation,

reward, and global state information, for the training process. Following this data collection, the centralized value functions are learned based on the information of all agents, and then the gradient from the centralized value function is used to train the policy of each agent. But in the process of execution, each agent outputs action according to its individual observation (see Fig. 3.4).

It is important to acknowledge the inherent trade-offs of the CTDE. One significant strength is the enhanced learning stability enabled by centralized training, which can result in more robust policies [233]. However, it is crucial to recognize that maintaining a centralized controller comes with its own set of challenges. Firstly, it can be prohibitively expensive and infeasible in certain scenarios, particularly in large-scale or resource-constrained environments. Additionally, the centralized controller introduces privacy concerns, as it necessitates the sharing of sensitive information among agents. Moreover, the central controller becomes a single point of failure, rendering the whole system vulnerable to disruptions. These trade-offs highlight the need for careful consideration of the CTDE framework's suitability in specific MARL applications, taking into account the advantages and drawbacks it presents.

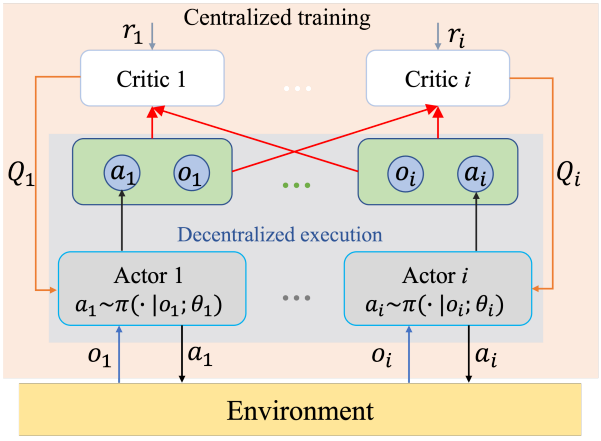


Fig. 3.4. Illustration of CTDE in MARL.

Decentralized training with decentralized execution (DTDE)

In decentralized settings, individual agents acquire knowledge independently, devoid of direct interactions. Each agent operates with its distinct set of observations, policies, and algorithms, utilizing the environment as the exclusive channel for engagement. Commonly employed algorithms within this framework encompass independent Q-learning (IQL) [234] and IA2C [72],

frequently adopted as foundational benchmarks. Although these algorithms enable autonomous learning by agents, they may not fully harness the collaborative potential [235]. To address non-stationarity concerns, [236] proposes a fully decentralized MARL framework. In this framework, each agent makes decisions based on both its local observation and the messages exchanged with neighboring agents via the network, offering convergence guarantees. Fig. 3.5 shows the DTDE pattern based on the actor-critic framework.

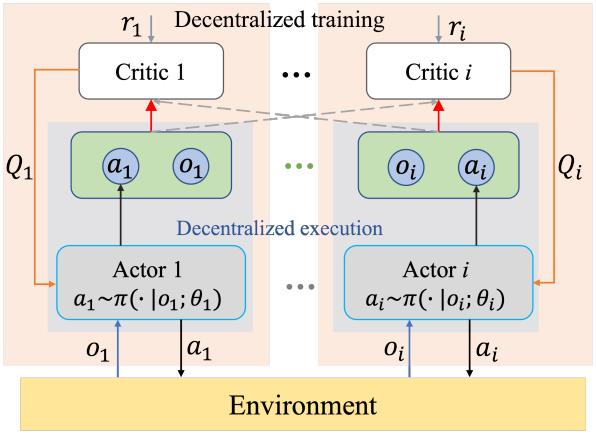


Fig. 3.5. Illustration of DTDE in MARL.

3.1.5 MARL algorithm variants

In this subsection, the MARL algorithms are divided into four categories: **value function decomposition**, **learning to communicate**, **hierarchical structure**, and **causal inference**, which are rooted in the inherent complexity and diverse requirements of multi-agent environments. Table 3.1 offers an extensive overview of key works from four distinct perspectives within various settings.

Value function decomposition

The challenge of credit assignment in cooperative settings has emerged as a significant area of research interest. The shared rewards in a fully cooperative environment make it difficult to distinguish the contribution of each agent; some agents tend to be lazy [255]. To solve the above problem, some studies learn different value functions to distinguish the contribution of each agent. Foerster et al. [256] and Guo et al. [257] introduced the counterfactual baseline principle to learn different value functions by centralized learning methods in the context of a shared team reward environment. Additionally, Hou et al. have developed a credit allocation

Table 3.1
An overview of the primary works for MARL algorithms.

Categories	Work	MARL algorithm	Novelty
Value Function Decomposition	[237], 2018	VDN	Decomposing value functions using linear summation
	[238], 2018	QMIX	Enforcing monotonicity constraints between joint and individual value functions
	[239], 2019	QTRAN	Relaxing the non-negative weight network in Qmix
	[240], 2020	WQMIX	Introducing weighting functions and non-monotonic mixing networks
	[241], 2021	AVD-Net	An attention-based approach to capitalize on the coordination relations between agents
	[242], 2023	PER	Both the reward of an agent acting on its own and cooperating with other observable agents
Learning to Communicate	[243], 2017	FingerPrint	Merging low-dimensional policy fingerprints in the state space of neighboring agents
	[113], 2016	DIAL	The agent continuously encodes information and sends it to other recipients
	[98], 2020	NeurComm	Encoding and concatenating communicating signals rather than aggregating them
	[244], 2020	IMAC	Learning an efficient communication protocol based on the information bottleneck principle
Hierarchical Structure	[245], 2019	FHM	Establishing a feudal hierarchy structure
	[246], 2018	HQMIX	Including hierarchical QMIX and hierarchical communication network
	[247], 2019	HSD	Training a cooperative decentralized policy for high-level skill selection
	[248], 2019	RODE	Introducing a method to clearly divide the action space through action clustering
	[249], 2023	HAVEN	Designing a dual coordination mechanism within a two-level hierarchical structure
Causal Inference	[250], 2018	CIR	Agents receive a reward based on their causal influence on the actions of others
	[251], 2023	ICL	Evaluating the causal effect of each agent's observations on team collaboration performance
	[252], 2023	LAIES	Calculating the causal effect of their actions on external states using the do-calculus process
	[253], 2022	FD-MARL	Constructing continuous communication protocols based on causal analysis
	[254], 2022	DVD	Deconfounded value function decomposition based on causal effects

mechanism based on role attention, which facilitates the learning process of role policies in multi-agent systems by managing how credit is allocated among agents [258]. Liu et al. construct the causal effect of the agent's actions on the external state from the perspective of a causal diagram, solving the problem of lazy agents [252].

Among the above methods to solve credit assignment, value function decomposition is considered to be an important research direction. The main concept and feature of value function decomposition is to decompose a joint value function into individual components to identify the distinct contributions of each agent. In the process, many useful value-function methods were proposed to solve increasingly complex problems. Value Decomposition Networks (VDN) decomposes the joint action-value function by breaking it down into a sum of the action-value functions for individual agents. This linear decomposition method yields favorable results in certain simple problems [237]. Different from the linear value function decomposition method of VDN, Monotonic Value Function Factorization (QMIX) enhances the representational capacity of the joint value function by enforcing monotonicity constraints between the joint and the individual value functions. This approach allows for handling increasingly complex scenarios effectively [238]. Q-value Transforms (QTRAN) further improves the representation ability of joint functions by relaxing the non-negative weight network in QMIX [239]. Weighted QMIX (WQMIX) extends the value function method to complex scenarios of non-monotonic rewards by introducing weighting functions and non-monotonic mixing networks [240]. After that, some variant algorithms based on VDN and QMIX were used to solve different specific problems [241, 242].

Learning to Communicate

Research in MARL can be categorized into two distinct groups according to their methods of communication [98]. The first group operates without communication among agents and centers its efforts on stabilizing training by employing advanced value estimation methods. For example, MADDPG [259] extends DDPG to the MARL setting by employing a centralized critic network to enable the global value estimate. Similarly, Counterfactual Multi-Agent (COMA) [260] modifies the actor-critic method for use in MARL, calculating the advantage function for each agent by employing a centralized critic in conjunction with a counterfactual baseline. The

second group investigates heuristic communication protocols, which may involve either direct message sharing or learnable communication protocols. For instance, in Fingerprint Policy Stabilization (FingerPrint) [243], the low-dimensional policy fingerprints are directly shared and incorporated into the state space of neighboring agents. In Differentiable Inter-Agent Learning (DIAL) [113], each DQN agent concurrently produces a communicated message while assessing action values. This message is encoded and subsequently combined with additional input signals at the receiving end. Neural Communication Protocols (NeurComm) proposed in [98] argues that encoding and concatenating communicating signals, rather than aggregating them, offers distinct advantages in mitigating information loss during communication. Information Maximization via Agent Communication (IMAC) learns an efficient communication protocol as well as scheduling based on the information bottleneck principle [244]. Their method involves the development of a novel differentiable communication protocol, which incorporates information related to state, policy, and the encoded data from neighboring agents. Subsequently, this information is encoded and concatenated into the state space of the neighboring agents.

Hierarchical Structure

In the realm of multi-agent systems, a plethora of challenges are encountered, encompassing issues like sparse rewards, sequential decision-making, and limited transfer capabilities. To address these challenges effectively, the adoption of a hierarchical approach is unequivocally advantageous. The fundamental tenet of hierarchical multi-agent reinforcement learning revolves around the acquisition of hierarchical strategies for decision-making across varying levels of temporal abstraction. Current research in this domain broadly categorizes hierarchical multi-agent reinforcement learning into two paradigms: option-based hierarchical structures [261, 262] and goal-based hierarchical structures [263, 264], both of which have found applications in diverse multi-agent scenarios.

Feudal Multi-agent Hierarchies (FMH) establish a feudal hierarchy within a multi-agent context, yielding noteworthy results. However, a primary limitation of this approach is its inherent challenge in tackling fully collaborative tasks with shared rewards [245]. To address the predicament of multi-agent scenarios characterized by sparse and delayed rewards, several hierarchical structures with temporal abstraction have been proposed, including hierarchical

QMIX and hierarchical communication network methods [246]. Furthermore, recognizing the limitations of traditional approaches, which typically rely on manually defined high-level action spaces, the Hierarchical Learning with Skill Discovery (HSD) model was conceived. This model aims to autonomously train decentralized policies for high-level skill selection while simultaneously developing independent low-level policies for the execution of these skills, thereby enhancing adaptability to changing scenarios [247]. On the other hand, learning ROles to DEcompose (RODE) is introduced for clear action space partitioning through action clustering, where each action corresponds to a distinct subspace, this innovation substantially mitigates the challenge of extensive parameter tuning associated with HSD [248]. Moreover, HierArchical Value dEcompositioN (HAVEN) introduces a dual coordination mechanism, encompassing inter-layer and inter-agent strategies, facilitated by the formulation of a reward function within a two-level hierarchical structure. This approach effectively addresses the instability issue through the concurrent optimization of strategies at all levels and inter-agent coordination [249]. Hierarchical MARL is recognized as a promising avenue for addressing collaborative decision-making challenges in complex, large-scale scenarios. Nonetheless, it faces the challenge of difficulty in designing complex dynamic hierarchical structures and difficult migration of policies.

Causal Inference

There are multiple complex variables in MARL, and it is usually difficult to directly discover their internal relationships. Therefore, some recent research has begun to combine MARL with causal inference to discover the causal relationships between agents or variables, and further understand the operating mechanism of agents through intervention and inference, thereby motivating the agent to conduct more targeted learning [265]. Pearl's three-level causal model is considered a powerful tool for constructing causal relationships between variables [266]. Causal influence reward methods promote the collaborative performance of agents by rewarding those agents that have a causal influence on the actions of other agents, where this causal influence is evaluated by counterfactual reasoning [250]. Rafael Pina et al proposed the Independent Causal Learning (ICL) algorithm to evaluate the causal effect of each agent's

observations on team collaboration performance and solve the credit allocation problem in the independent learning framework [251].

There are also other works that combine causal inference from different perspectives under the framework of MARL. Lazy Agents Intervention and Exploitation Strategy (LAIES) mathematically define the concept of fully lazy agents and teams by calculating the causal effect of their actions on external states using the do-calculus process, which solved the sparse reward problem in MARL [252]. FD-MARL allows agents to modify communication messages by choosing the counterfactual that bears the most significant influence on others; continuous communication based on causal analysis enables efficient information transformation in a fully decentralized manner [253]. The Deconfounded Value Decomposition (DVD) method investigates value function decomposition from the perspective of causal inference, which cuts off the backdoor confounding path from the global state to the joint value function [254]. In addition, causal inference has also been applied to some practical application scenarios, such as traffic signal control [267] and human-computer interaction decision-making [268].

3.2 | Research facilities and resources

This study utilized two prominent platforms, ETAS and Speedgoat, for HiL testing to ensure the feasibility and efficiency of the developed control algorithms. Additionally, various driving cycles were introduced, and OpenACC data was extracted to validate the control strategies under real-world conditions.

3.2.1 ETAS HiL testing platform

The ETAS HiL testing system provides a robust and flexible environment for real-time validation of control strategies, as illustrated in Fig. 3.6. It consists of the LABCAR real-time computer and the ES910 prototype controller, interconnected through a controller area network (CAN) bus to mimic real-world communication between a vehicle control unit and its physical systems. A host PC, equipped with high-performance hardware, is used for developing and deploying vehicle models and control algorithms. The PC operated on a Windows 10 64-bit system with an Intel Core i7 processor, ensuring smooth execution of complex simulations.

MATLAB/Simulink is utilized for designing control strategies, which are subsequently compiled into C-code using ETAS software. The vehicle model is uploaded to the LABCAR via Ethernet, while the control algorithm is implemented in the ES910 controller. This integrated workflow allows real-time validation of the developed algorithms, ensuring their practical applicability within a time frame of 1 second.

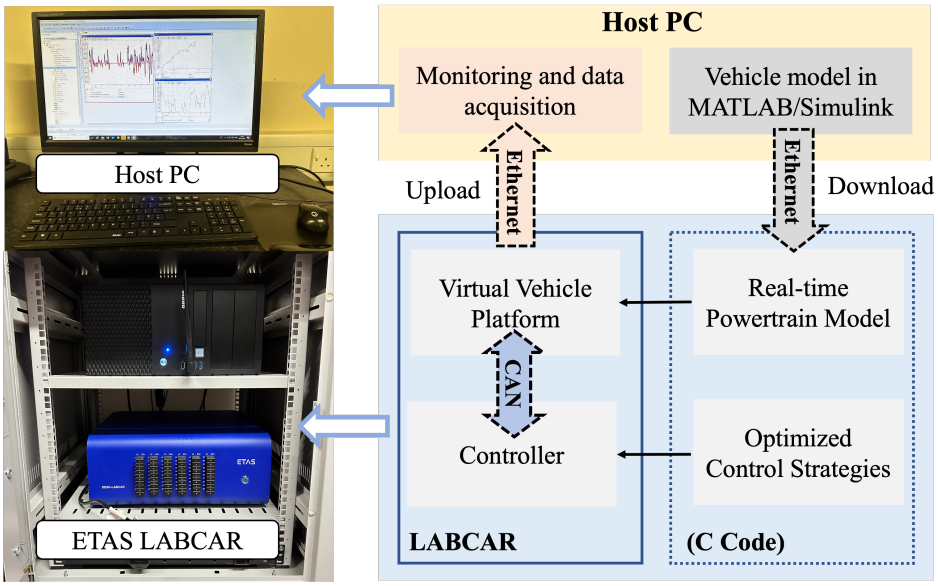


Fig. 3.6. ETAS HiL testing platform.

3.2.2 Speedgoat HiL testing platform

The Speedgoat platform offers an alternative real-time testing environment with integration into MATLAB/Simulink workflows, as shown in Fig. 3.7. This system is configured with an Intel Core i7 processor, 16GB RAM, two CAN interfaces, and four EtherCAT/Ethernet ports, ensuring flexibility and high-speed performance. Similar to the ETAS setup, the virtual vehicle and control strategies are developed in MATLAB/Simulink and integrated into the Speedgoat target machine via Ethernet.

During real-time HiL testing, the virtual vehicle model and designed controllers communicate through CAN protocols within the Speedgoat system. The compatibility of Speedgoat with Simulink Real-Time software closely emulates real-world vehicle operations. As a result, this platform provides a reliable testing environment for validating control strategies under realistic conditions.

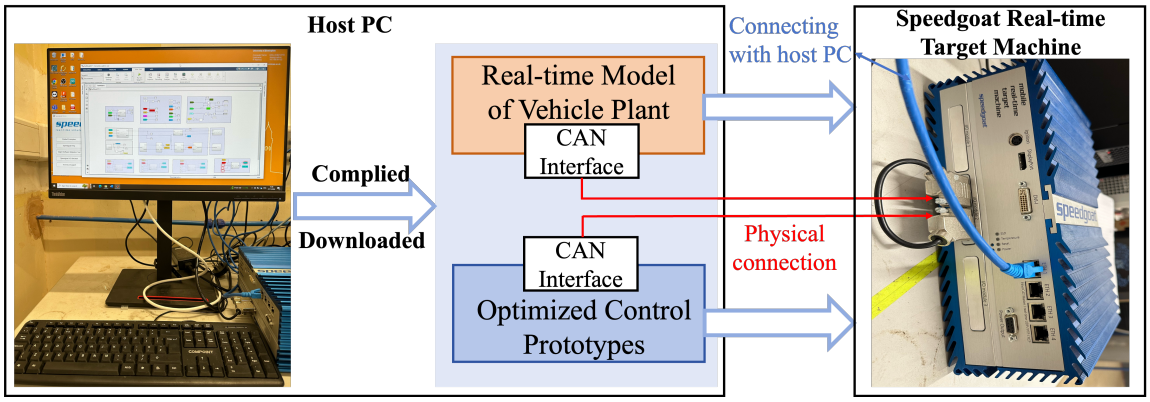


Fig. 3.7. Speedgoat HiL testing platform.

3.2.3 Driving cycles and OpenACC data

Standard driving cycles

Standard driving cycles are predefined velocity profiles used to evaluate the performance, energy consumption, and emissions of vehicles under controlled conditions. They provide a consistent and repeatable framework for testing and comparison across different vehicle types, ensuring uniformity in performance assessments [269]. Below are descriptions of commonly used standard driving cycles.

Low-speed cycles (Urban driving)

Low-speed cycles are designed to replicate driving conditions in densely populated urban areas. These cycles typically feature low average speeds, frequent stops and starts, and significant idling times, which are characteristic of urban traffic congestion and short commutes, as described in Fig. 3.8.

New York City Cycle (NYCC) represents low-speed, stop-and-go driving conditions typical of inner-city traffic. It includes frequent idling, slow accelerations, and short cruising periods. The maximum speed is approximately 45 km/h, and the duration of the cycle is around 10 minutes. This makes it a suitable test for evaluating emissions and fuel consumption in heavy traffic.

The Manhattan cycle is similar to the NYCC, focusing on dense metropolitan traffic conditions, but it is longer, with a duration of approximately 20 minutes. It includes a maximum speed of around 40 km/h and emphasizes high idling periods and frequent stops, simulating traffic congestion in large urban centers like Manhattan.

Urban Dynamometer Driving Schedule (UDDS) is originally designed for light-duty vehicles; the UDDS is widely used in emissions and fuel economy testing. It represents a mix of city driving scenarios with numerous stops, slow cruising, and gradual accelerations. Although its maximum speed reaches 91 km/h, the average speed is much lower, making it ideal for evaluating urban performance.

Artemis Rural Driving Cycle is a low-speed cycle that reflects rural-urban transitional driving conditions often encountered in suburban or less densely populated areas in Europe. While it includes slightly higher speeds compared to typical urban driving cycles, it maintains characteristics such as moderate accelerations, decelerations, and idling. This cycle bridges the gap between urban and rural scenarios, making it relevant for testing emissions and fuel efficiency in areas with mixed traffic conditions and less congestion than in city centers.

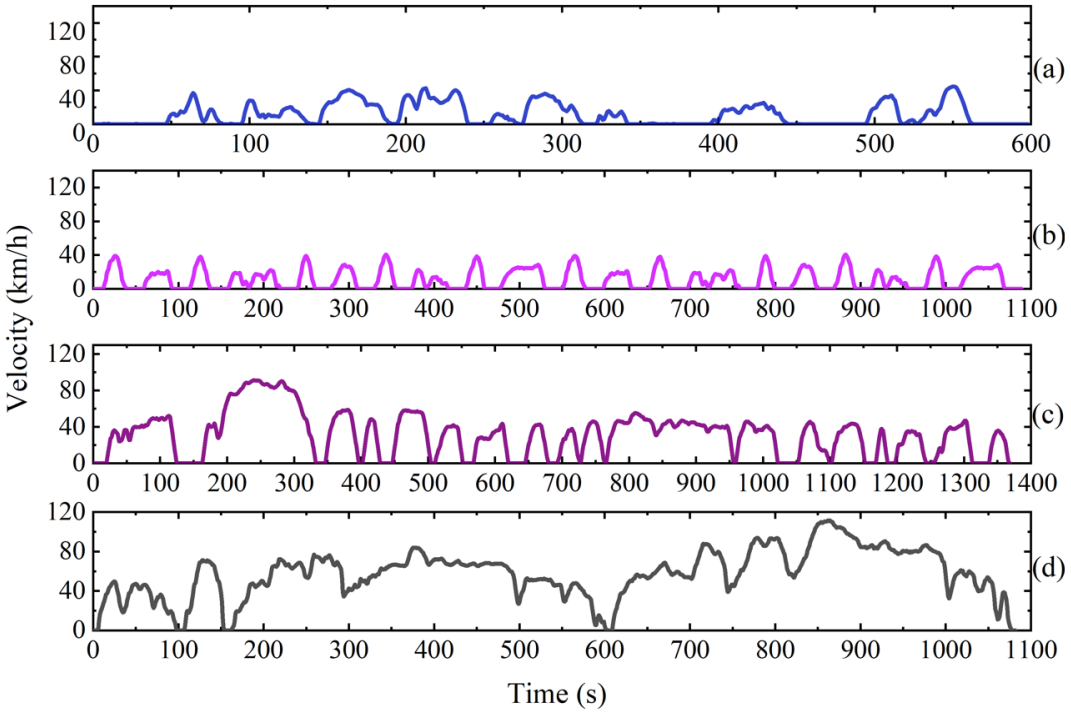


Fig. 3.8. The standard driving cycles: (a) NYCC driving cycle, (b) Manhattan driving cycle, (c) UDDS driving cycle, (d) Artemis Rural driving cycle.

Medium-speed cycles (Mixed driving)

Medium-speed cycles capture the characteristics of both urban and suburban driving, providing a balanced test for vehicles that operate in mixed environments. These cycles include moderate speeds, smoother transitions, and fewer stops compared to urban driving in Fig. 3.9.

Real Traffic Simulation 95 Driving Cycle (RTS95) is designed to represent realistic traffic scenarios that span urban and suburban environments. With a maximum speed of approximately 95 km/h and a duration of around 22 minutes, it includes moderate accelerations, decelerations, and cruising periods. This makes it ideal for evaluating vehicle performance, emissions, and fuel efficiency in diverse mixed driving conditions.

Federal Test Procedure (FTP75) is one of the most commonly used driving cycles in the United States, designed to simulate a combination of city and suburban driving. It has a maximum speed of around 91 km/h and a duration of 31 minutes, making it ideal for evaluating emissions and fuel economy under real-world mixed conditions.

New European Driving Cycle (NEDC) is widely used in Europe, the NEDC consists of two phases: an urban phase with stop-and-go driving and an extra-urban phase for suburban and highway driving. It has a maximum speed of 120 km/h and a duration of 20 minutes, reflecting typical European driving behavior.

Worldwide Harmonized Light Vehicles Test Cycle-Class 3 (WLTC-Class3) globally harmonized driving cycle includes phases for low, medium, high, and extra-high speeds, making it a comprehensive test for diverse driving conditions. It has a maximum speed of 131 km/h and a duration of 30 minutes, covering a wide range of real-world scenarios.

China Light-Duty Test Cycle-Passenger (CLTC-P) specifically developed for light-duty vehicles in China, the CLTC-P reflects mixed urban and suburban driving conditions typical in Chinese traffic. It has a maximum speed of 114 km/h and a duration of 20 minutes, emphasizing smooth transitions between stop-and-go city driving and suburban cruising.

High-speed cycles (Highway driving)

High-speed cycles are intended to simulate highway driving conditions where vehicles operate at higher speeds with fewer stops, as shown in Fig. 3.10. These cycles are critical for testing fuel efficiency, emissions, and high-speed performance.

US06 represents aggressive highway driving with sharp accelerations, rapid decelerations, and steady high-speed cruising. It has a maximum speed of 129 km/h and a duration of 10 minutes, making it one of the most demanding cycles for energy consumption and emissions testing.

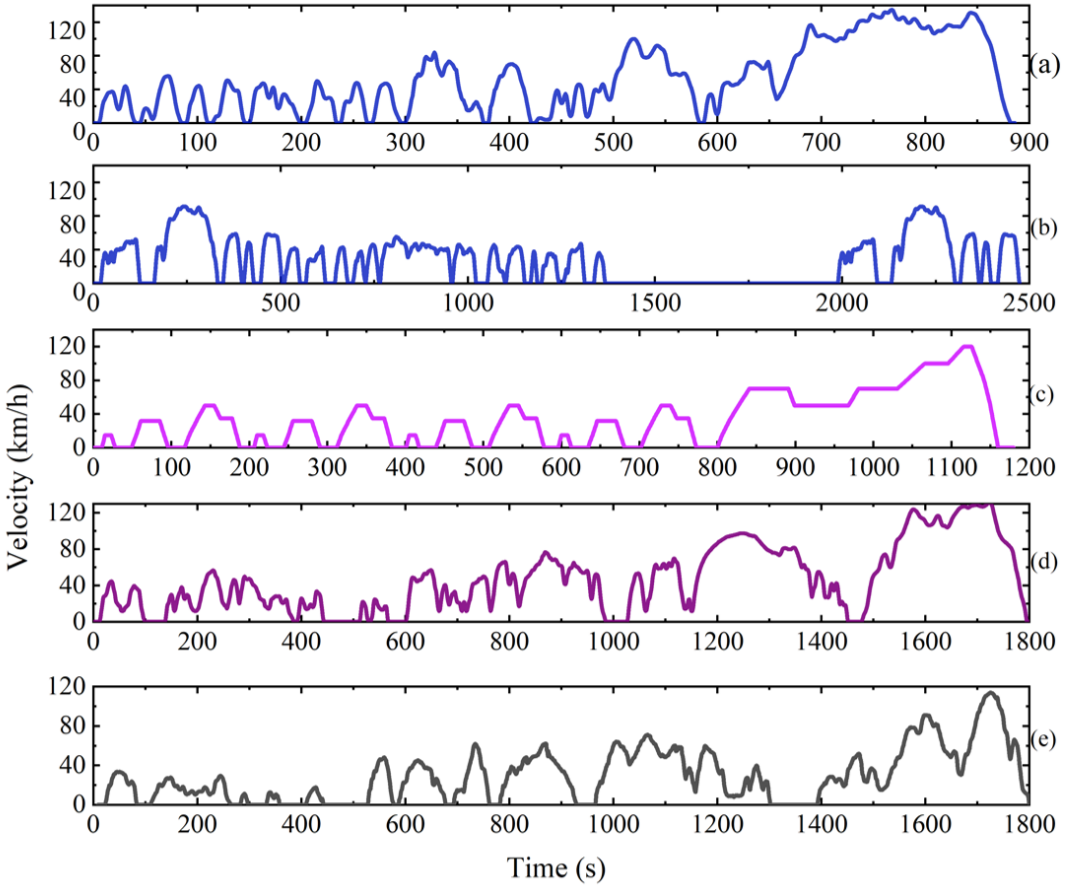


Fig. 3.9. The standard driving cycles are (a) RTS95 driving cycle, (b) FTP75 driving cycle, (c) NEDC driving cycle, (d) WLTC-Class3 driving cycle, and (e) CLTC-P driving cycle.

Highway Fuel Economy Test (HWFET) is used to measure fuel economy under steady conditions, focusing on smooth and efficient highway cruising. It has a maximum speed of 97 km/h and a duration of 12.5 minutes, providing a realistic assessment of highway performance.

Inspection and Maintenance Test-240 seconds (IM240) is used primarily for emissions compliance for short-duration tests (240 seconds). Despite its shorter length, the IM240 evaluates high-speed emissions performance with a maximum speed of 91 km/h, making it suitable for both highway driving and regulatory testing.

Specialized Cycle

Specialized cycles are designed for specific applications, focusing on unique driving conditions or vehicle types outside the standard urban, mixed, or highway classifications.

The American Public Transportation Association Cycle (APTA) captures the stop-and-go nature of transit systems with varying loads, low-to-moderate speeds, and frequent idling in Fig.

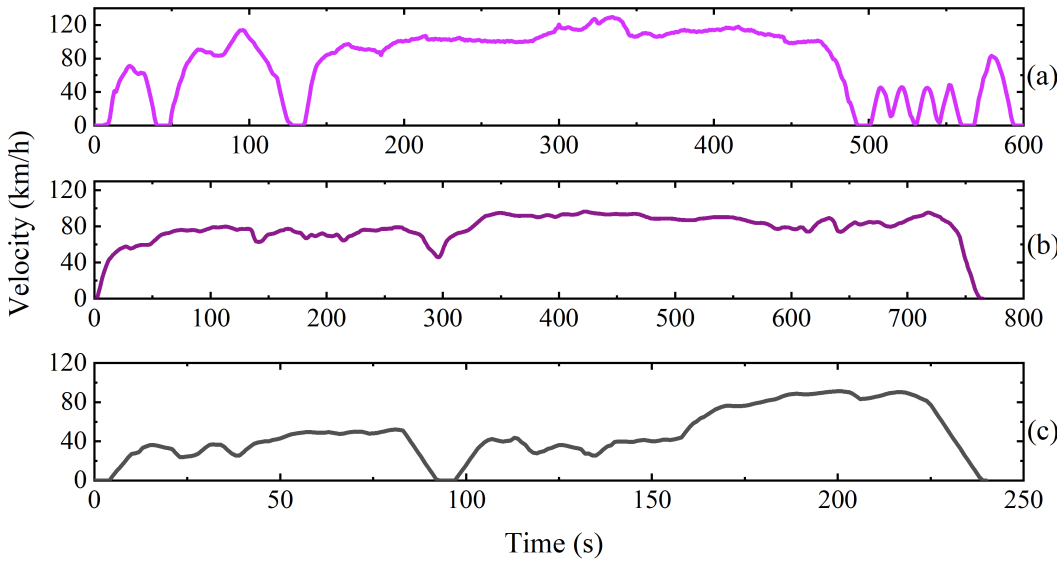


Fig. 3.10. The standard driving cycles are (a) US06 driving cycle, (b) HWFET driving cycle, and (c) IM240 driving cycle.

3.11. The APTA cycle is primarily used to assess emissions and fuel efficiency under real-world public transit conditions, reflecting the operational challenges faced by these vehicles.

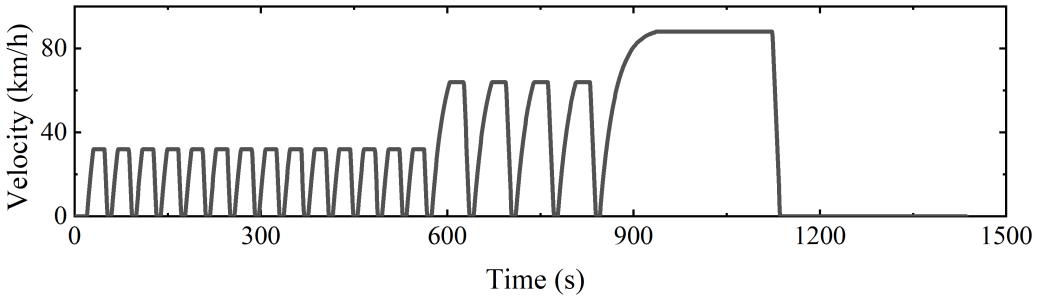


Fig. 3.11. APTA driving cycle.

As shown in Table 3.2, driving cycles are categorized into four types, ranging from low-speed cycles like NYCC that simulate dense urban environments to high-speed cycles such as US06 that model highway driving conditions, with additional specialized cycles for public transit applications.

Combined driving cycle

A combined driving cycle is also essential for evaluating the performance, energy consumption, and emissions of vehicles in a manner that reflects a wide range of real-world driving conditions. Standard driving cycles, while useful, are often designed for specific conditions like urban or highway driving, they often fail to capture the complexity of actual patterns. By integrating

Table 3.2
Summary of driving cycle categories

Category	Driving Cycles	Description
Low-Speed Cycles	NYCC, Manhattan, UDDS, Artemis Rural	Simulates dense urban driving with frequent stops, low speeds, and high idling.
Medium-Speed Cycles	RTS95, FTP75, NEDC, WLTC-Class3, CLTC-P	Represents mixed urban and suburban driving, with moderate speeds and reduced stops.
High-Speed Cycles	US06, HWFET, IM240	Models highway driving with high speeds, steady cruising, and rapid accelerations.
Specialized Cycles	APTA	Designed for public transit systems, focusing on emissions and fuel efficiency in buses.

multiple cycles, a combined driving cycle provides a comprehensive and realistic representation, including transitions between environments such as city streets and highways. To achieve this, the combined driving cycle is divided into two parts: **training cycles** and **validation cycles**.

Training cycles: When employing the RL algorithms, the training driving cycle, shown in Fig. 3.12, is constructed using elements from four standard driving cycles—Artemis Rural, RTS95, UDDS, and WLTP—to provide a diverse and comprehensive representation of real-world driving conditions. The cycle is divided into four phases: Phase 1 captures the low-speed characteristics of the Artemis Rural cycle, Phase 2 highlights the maximum acceleration from the RTS95 cycle, Phase 3 represents the medium-speed conditions typical of the UDDS cycle, and Phase 4 incorporates the high-speed segment from the WLTP cycle. To enhance the robustness of the learning process, the sequence of these four phases is randomly reorganized in each learning episode, introducing variability and learning noise to simulate a wide range of driving scenarios. This approach ensures a thorough evaluation of vehicle performance under dynamic and unpredictable conditions.

Validation cycles: The combined cycles are validated on three comprehensive driving conditions (Cycle I, II, III), as shown in Fig. 3.13. These driving conditions were synthesized from eleven standard driving conditions, representing a variety of road features and driving

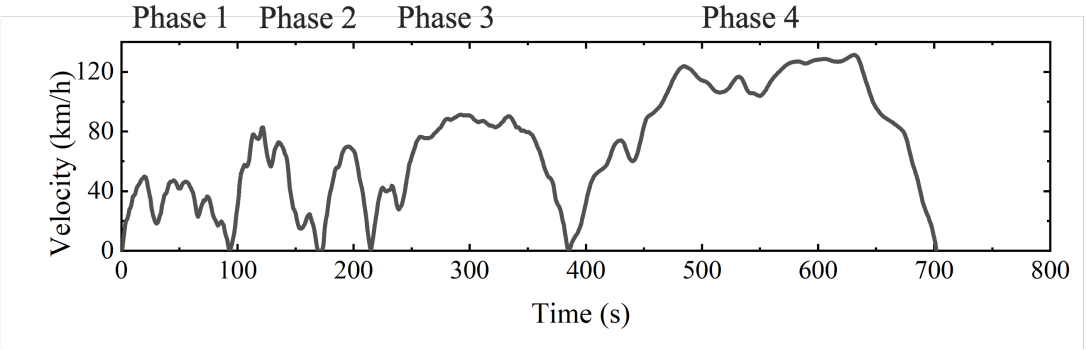


Fig. 3.12. Training driving cycle.

behaviors. This approach allows for a more granular representation of real-world traffic patterns, increasing the reliability and applicability of vehicle testing results.

To further enhance the complexity and representativeness of the driving conditions, the Ministry of Industry and Information Technology (MIIT) collaborated in providing road traffic data for passenger cars in Wuhan, sourced from the China Automotive Traffic Condition (CATC) information system. The data went through a preparation process, including removing errors, smoothing fluctuations, and filling in gaps, to ensure the data quality. Subsequent feature extraction focused on key metrics such as speed, acceleration, and distance, which were validated and adjusted to align with real-world driving behavior. The final processed dataset was synthesized into a highly realistic driving condition (Cycle IV in Fig. 3.13).

The four combined driving cycles were carefully selected to capture a wide range of real-world driving conditions. The statistical analysis in Table 3.3 highlights distinct characteristics, such as mean and maximum speeds, as well as acceleration patterns, ensuring comprehensive coverage of diverse traffic conditions. Cycle I and II reflect mixed urban and suburban driving, while Cycle III focuses on stop-and-go traffic typical of busy urban areas. Cycle IV, derived from real-world traffic data in Wuhan, represents low-speed and steady driving in densely populated city centers. By doing so, we can evaluate vehicle performance under varying traffic scenarios, ensuring that the test results are robust and applicable across diverse geographic regions and driving features.

OpenACC data from real-world scenarios

An open-access database of car-following experiments involving vehicles equipped with ACC systems has been developed by the Joint Research Centre (JRC) of the European Commission

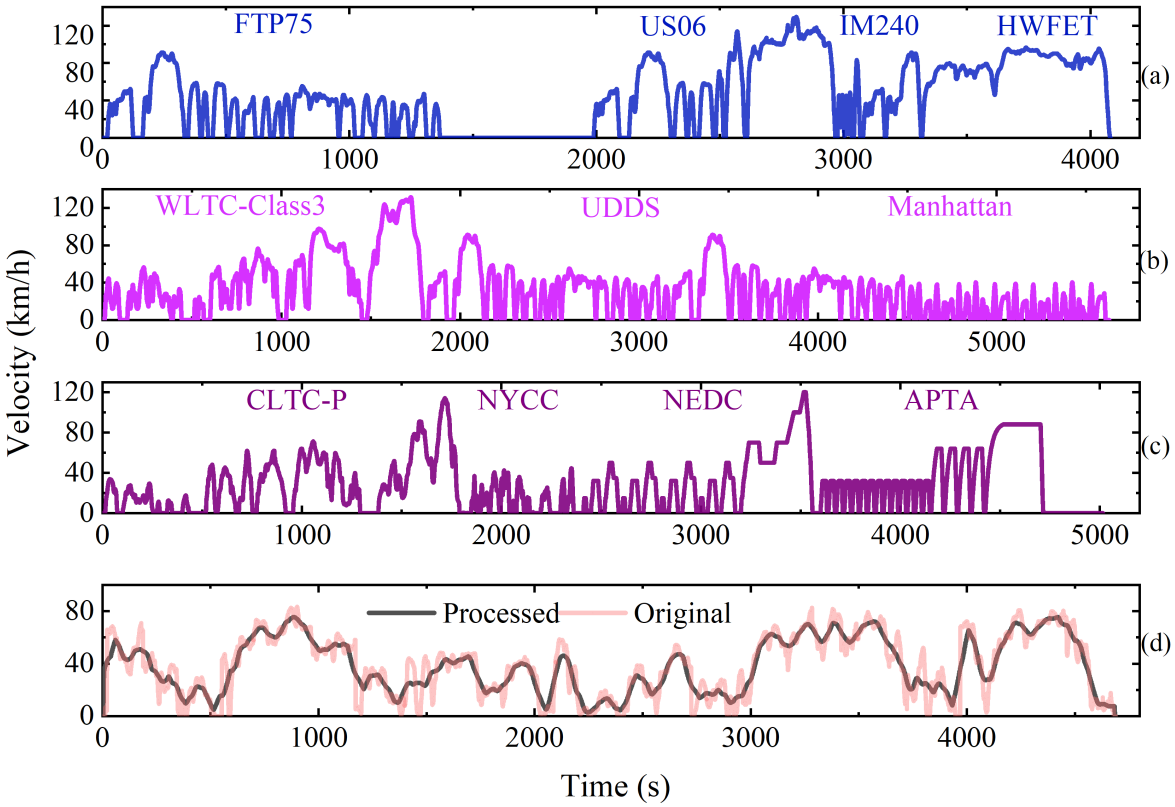


Fig. 3.13. The combined driving cycles for the validation are (a) Driving Cycle I, (b) Driving Cycle II, (c) Driving Cycle III, and (d) Driving Cycle IV.

Table 3.3
Four combined driving cycle statistics

Cycle	Mean Speed	Max Speed	Std Speed	Mean Acc	Max Acc	Min Acc	Std Acc
I	12.32	35.87	10.11	2.12×10^{-18}	3.76	-3.08	0.61
II	8.98	36.47	8.18	5.36×10^{-18}	2.04	-2.49	0.59
III	8.01	33.35	7.99	3.54×10^{-18}	2.68	-2.64	0.53
IV	10.713	21.00	5.65	0	2.99	-1.04	0.11

(EC) [270]. One of these datasets features a detailed test campaign conducted at the ZalaZONE, involving 10 commercially available ACC-equipped vehicles. The tests were carried out on two distinct tracks: the Dynamic Platform and the Handling Course, examining ACC performance and string stability in low-speed scenarios (30–60 km/h) under different ACC settings and vehicle sequences in Fig. 3.14. Data collection employed advanced equipment such as the INVENTURE Race Logic VBOX, U-Blox 9, and the Tracker App, with sensor data processed and interpolated to 10 Hz using 2nd-degree B-splines [271]. The dataset includes time, raw speed, and inter-vehicle spacing (IVS).

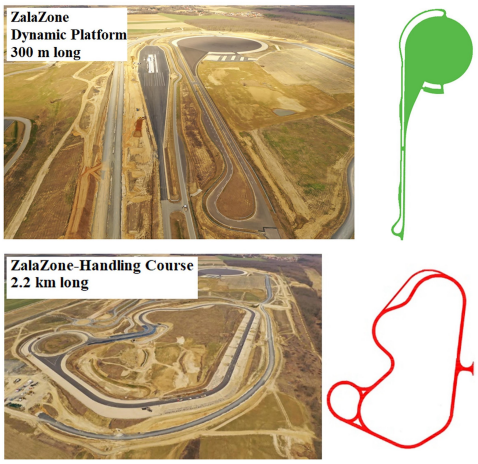


Fig. 3.14. Zalazone Test Platform.

On the Dynamic Platform, the experiments focused on how ACC systems responded to controlled disturbances. These involved sudden decelerations followed by accelerations back to the original speed within a vehicle platoon. On the Handling Course, ACC performance was tested on different road geometries, with the lead vehicle maintaining a constant speed via cruise control without any external disturbances. Each test included up to 11 vehicles (a leader and 10 ACC-equipped vehicles), although not all tests featured data from all vehicles due to factors like vehicles leaving the platoon or incomplete data collection.

In my research, I focused on the dynamic behavior of ACC systems and analyzed the dataset from the Dynamic Platform, which includes 26 test groups. To avoid redundancy and ensure the dataset's suitability as validation scenarios for algorithms, I conducted a thorough analysis and comparison of all available data, along with an evaluation of data quality. Based on this process, I selected three representative groups from the datasets (dynamic_part1, dynamic_part2, dynamic_part9), as shown in Fig. 3.15 to Fig. 3.17. These groups were chosen for their reliability and their ability to capture sudden deceleration events, making them ideal for reflecting the dynamic characteristics of ACC systems. Notably, these datasets represent consistent speed settings but varying IVS, providing diverse scenarios for the analysis. Based on the three selected representative datasets, which capture the same optimal velocity settings but differ in IVS, I identified a specific 60-second interval within each selected dataset that represents moments of dynamic change, focusing on events such as sudden deceleration or acceleration. The selected intervals are as follows: timestamps $316s - 376s$ for dynamic_part1, $450s - 510s$ for dynamic_part2, and $473s - 533s$ for dynamic_part9. These intervals, shown

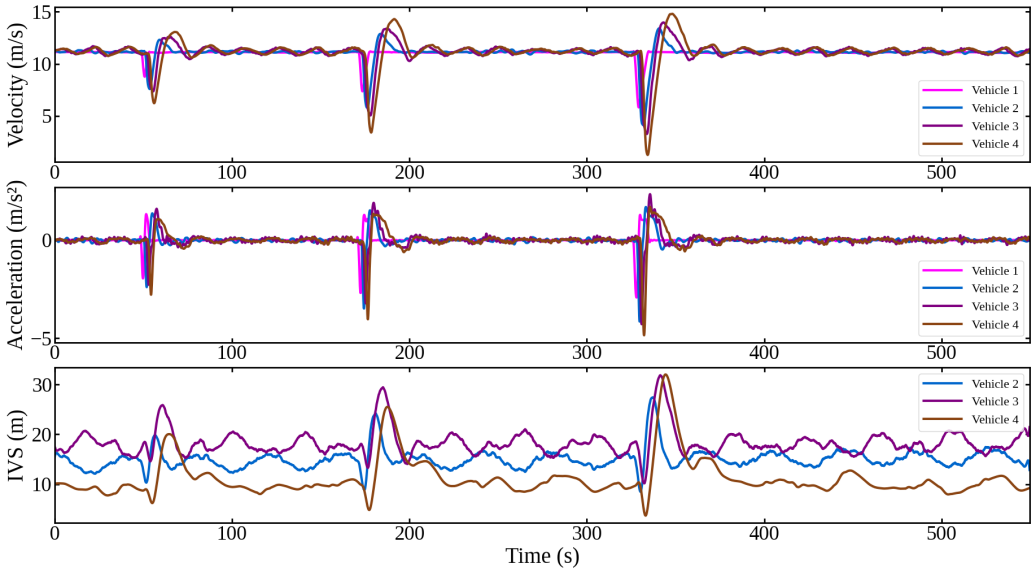


Fig. 3.15. Dataset for dynamic_part1.

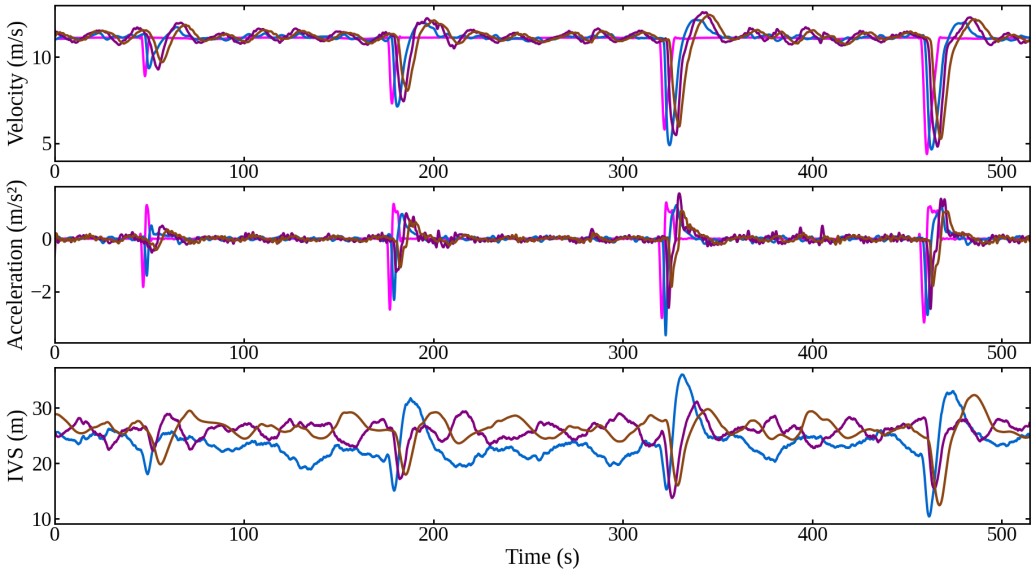


Fig. 3.16. Dataset for dynamic_part2.

in Fig. 3.18, were chosen because they highlight critical dynamic transitions and provide challenging conditions for evaluating ACC system performance. By focusing on these dynamic processes, they serve as benchmark scenarios for algorithm validation, closely reflecting the real-world operational characteristics of ACC systems.

3.3 | Multi-mode PHEV modeling and modes

Vehicle modeling is a critical component of automotive engineering, allowing for the simulation and analysis of vehicle dynamics and energy flow for PHEV [272].

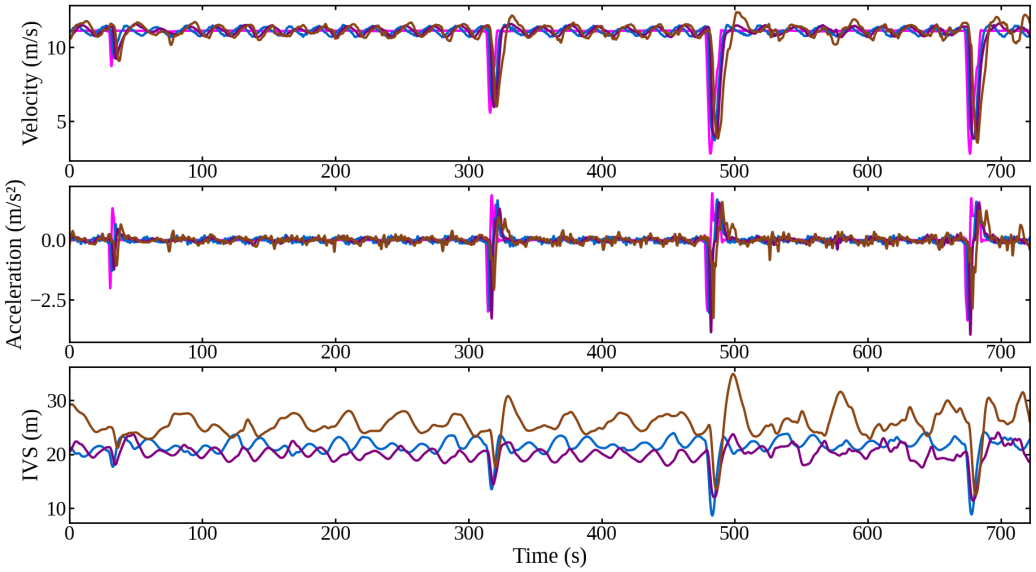


Fig. 3.17. Dataset for dynamic_part9.

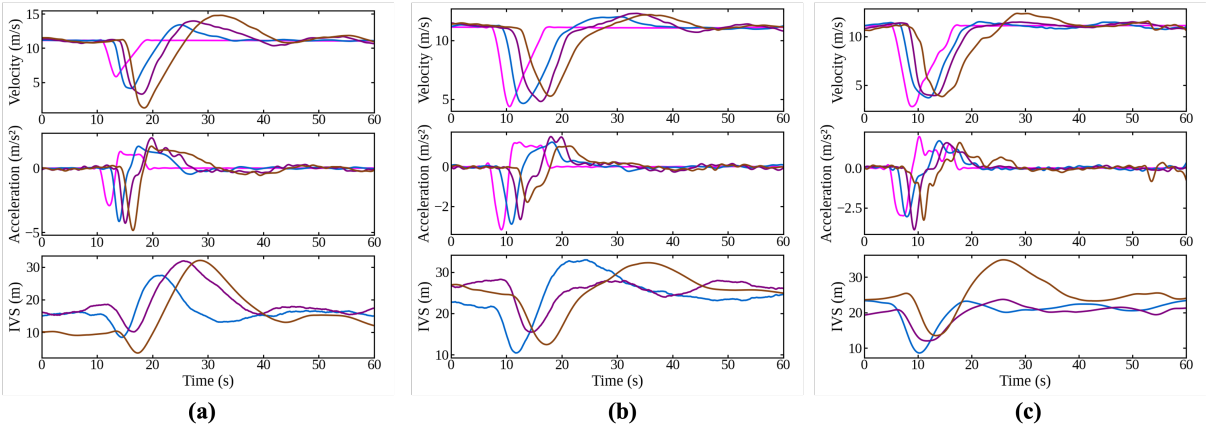


Fig. 3.18. Dynamics within 60s: (a) Specific datasets from dynamic_part1; (b) Specific datasets from dynamic_part2; (c) Specific datasets from dynamic_part9.

3.3.1 Vehicle model

Two primary modeling approaches are considered for simulating and analyzing PHEV powertrains: *forward-facing* and *backward-facing* models [273]. Each approach has unique advantages depending on the application focus.

The *forward-facing vehicle model* simulates the powertrain starting from the driver's input (torque or pedal demand) to the resulting vehicle motion. This approach is used to track a desired speed profile from a drive cycle, with a driver model providing the required torque for the ICE and brakes. The EMS processes the driver's torque commands and distributes power flow between the ICE and the motor. The torque generated by the powertrain is then

transmitted through the gearbox and the final drive ratio to the wheels, determining the vehicle's speed. Feedback from the resulting vehicle speed is used to adjust power distribution in real-time. This method is particularly useful for analyzing instantaneous power flows and vehicle responses to dynamic inputs.

In contrast, the *backward-facing vehicle model* starts from the vehicle's power demand (typically at the wheels) and traces energy flow back to the power sources, such as the ICE and battery. This approach is widely used for estimating energy consumption and evaluating the efficiency of powertrain components. By calculating the forces needed at the wheels to overcome rolling resistance, aerodynamic drag, and other resistive forces, the power demand at each time step is determined. The EMS then allocates power between the powertrain components accordingly. Performance feedback is used to refine control strategies, ensuring accurate energy allocation and system efficiency [274].

In this research, the *backward-facing approach* is adopted to analyze energy requirements and powertrain efficiency for multi-mode PHEVs, incorporating powertrains capable of operating in various operational configurations [48]. As illustrated in Fig. 3.19, the backward-facing model begins with the power demand calculated by the vehicle longitudinal model based on a given driving cycle. This power demand is processed by the EMS control unit, which optimally distributes power among the powertrain components, such as the transmission, ICE, motor, and battery [275]. The vehicle longitudinal model [276] can be calculated by

$$\begin{aligned} F_{tot} &= mgf\cos\alpha + \frac{1}{2}\rho A_f C_d v^2 + mg\sin\alpha + ma; \\ P_{dem} &= F_{tot}\cdot v; \end{aligned} \tag{3.7}$$

whereF_{tot} is the total force of the vehicle demand; m is the vehicle mass; g is the gravity acceleration; f is the rolling resistance coefficient; ρ is the air density; A_f is the front area of the vehicle; C_d is the aerodynamic drag coefficient; v is the vehicle speed; α is the road slope; in this paper, the road slope should not be considered; P_{dem} is the power demand of the vehicle.

Importantly, EMS control is formulated as an optimization task aimed at minimizing the cost function J, which accounts for factors such as energy consumption E_{cons}, power losses P_{loss},

and battery SoC SoC(t): $\min_{\mathbf{u}(t)} \int_{t_0}^{t_f} J\left(E_{\text{cons}}, P_{\text{loss}}, \text{SoC}(t)\right) dt$. The optimization is subject to the power balance constraint and operational limits for the components from the engine P_{ICE}, and the motor P_{motor}: $P_{\text{ICE}}(t) \leq P_{\text{ICE,max}}, P_{\text{motor}}(t) \leq P_{\text{motor,max}}, and \text{SoC}_{\text{min}} \leq \text{SoC}(t) \leq \text{SoC}_{\text{max}}$. These constraints ensure efficient and safe operation of the powertrain components while meeting the vehicle's power demand.

This framework facilitates a detailed analysis of energy consumption, powertrain efficiency, and performance under various driving conditions. Since this research emphasizes various powertrain configurations, detailed data and specifications from MAHLE Powertrain will be presented in the relevant chapters.

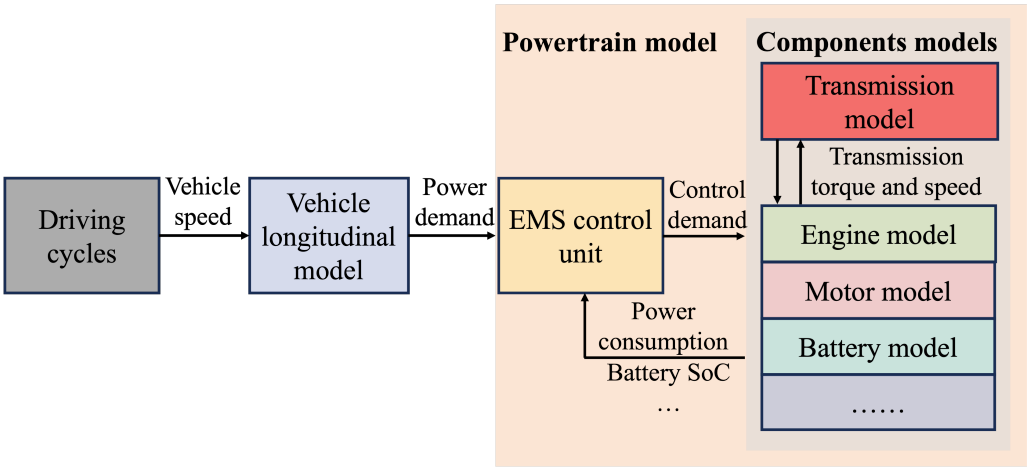


Fig. 3.19. Backward-facing vehicle model

3.3.2 Modeling of main components

Engine modeling

Map-based engine modeling represents a fundamental methodology in powertrain development, particularly for HEVs. This approach relies on experimental data obtained from dynamometer testing, where key performance parameters such as engine torque, power output, and brake-specific fuel consumption (BSFC) are measured across a range of operating conditions. These data are compiled into three-dimensional performance maps, as shown in Fig. 3.20, which illustrates the BSFC distribution across the engine's speed-torque range. Such maps are critical tools for optimizing powertrain performance and developing EMS. In HEVs, they facilitate precise coordination between the ICE and the electric motor by identifying high-efficiency

operating regions and minimizing fuel consumption. Additionally, these models support the evaluation of performance trade-offs and the implementation of advanced control strategies to maximize overall system efficiency under varying driving conditions [277].

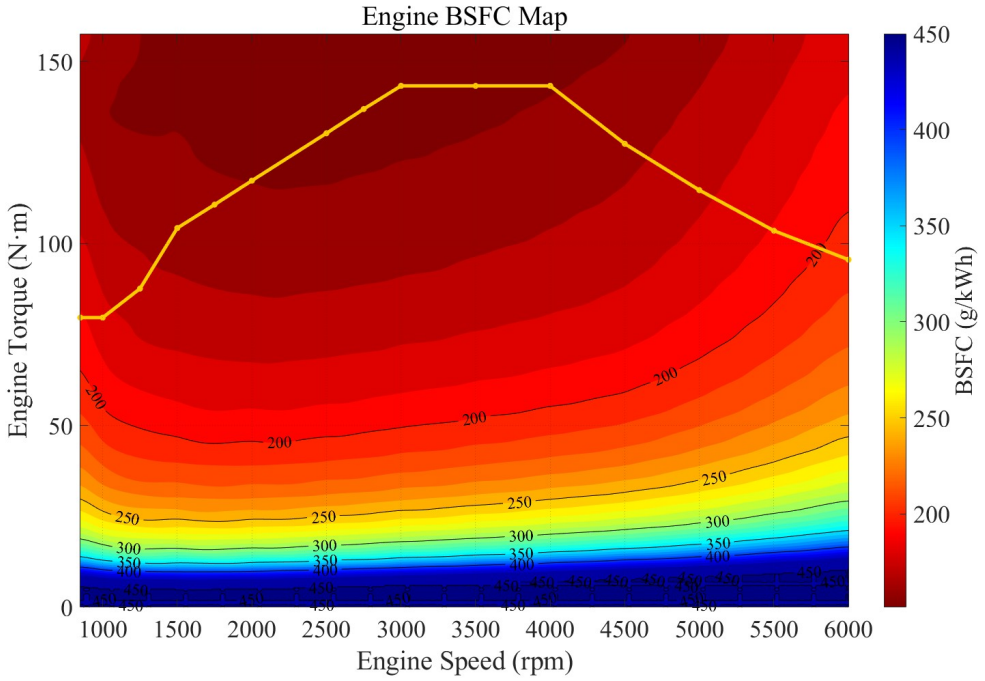


Fig. 3.20. BSFC map of engine.

Motor modeling

Electric motor modeling utilizes three-dimensional performance maps derived from experimental data, correlating motor speed, torque, and efficiency. Dynamometer testing measures key performance metrics across various operating conditions, including motor speeds and motor torques [278]. The efficiency maps represent motor speed and torque as primary axes, with efficiency levels shown as a color gradient. The maximum torque curve, often marked as a boundary, defines the motor operational limits across its speed range, as detailed in Fig. 3.21. This map-based approach is highly effective for real-time control and energy management in electric and hybrid powertrains, balancing computational efficiency with accuracy.

Battery modeling

Battery modeling for hybrid electric vehicles commonly uses an equivalent circuit model (ECM) approach, which provides a good balance between accuracy and computational efficiency [279]. A widely used configuration is the second-order RC model, which consists of an open-circuit

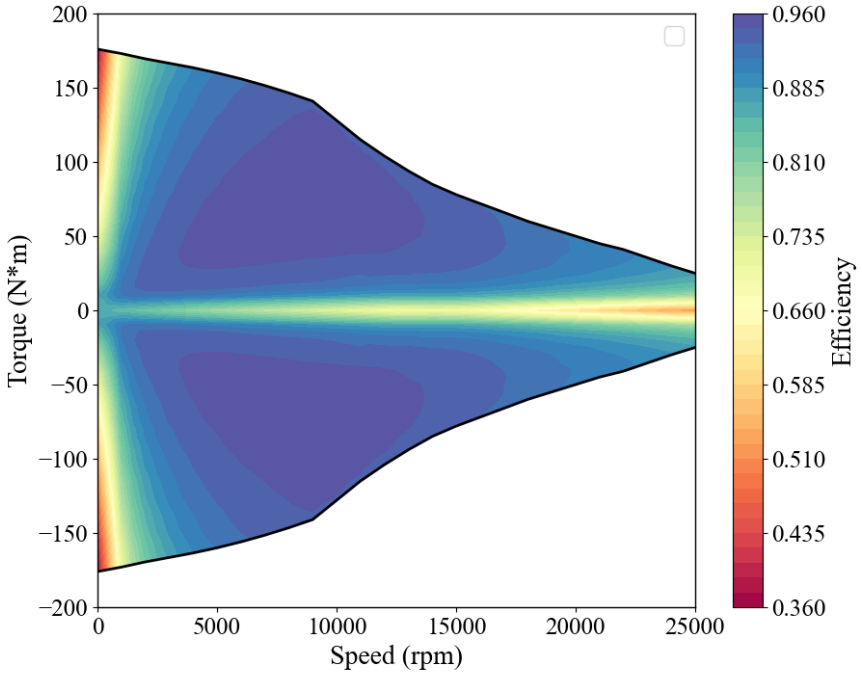


Fig. 3.21. Efficiency map of the electric motor.

voltage (OCV) source, an internal resistance, and two RC pairs [280]. This structure captures both the fast and slow dynamic responses of the battery during charge and discharge cycles. The general terminal voltage of the battery, U_{batt}, is expressed as:

$$U_{\text{batt}} = U_{\text{oc}} - I_{\text{batt}} R_{\text{int}} - \sum_{i=1}^{2} U_{R_i}, \tag{3.8}$$

where U_{oc} is the open-circuit voltage as a function of the SoC, $U_{\text{oc}} = f(\text{SoC})$; I_{batt} is the battery current; R_{int} is the internal resistance; and U_{R_i} represents the voltage drops across the RC pairs, capturing transient effects. The SoC is updated using the following equation:

$$\text{SoC}_{t+1} = \text{SoC}_t - \frac{I_{\text{batt}} \Delta t}{Q_{\text{max}}}, \tag{3.9}$$

where Δt is the time step, and Q_{max} is the maximum battery capacity.

3.3.3 Working modes

The architecture of a multi-mode PHEV powertrain, as depicted in Fig. 1.10, integrates several key components: an ICE, two motor/generators (MG1 and MG2), a battery pack, and a series

of mechanical and electrical connections. Solid black lines represent the mechanical connections within the system, while dashed green lines indicate electrical connections, including power flow between the battery, motors, and plug-in charging interface. The green dotted lines represent energy recovery during regenerative braking.

A clutch system is implemented to manage the connection between MG1 and the gearbox, allowing the powertrain to seamlessly switch between operating modes for optimal efficiency and performance. The PHEV can operate in three primary modes: pure EV, series hybrid, and parallel hybrid, each tailored for different driving conditions.

Pure EV mode

In pure EV mode, the ICE and MG1 are completely shut down, leaving MG2 as the sole power source. The energy required for propulsion is taken entirely from the battery pack, which provides torque to the wheels through MG2, a gearbox, and a differential. This mode is typically used during low-speed urban driving or short trips, maximizing energy efficiency and minimizing emissions.

The power balance in pure EV mode is given by:

$$P_{\text{Batt}} = P_{\text{MG2}}, \tag{3.10}$$

where P_{Batt} represents the power supplied by the battery. The power provided by MG2 is expressed as:

$$P_{\text{MG2}} = \frac{T_{\text{MG2}} \cdot \omega_{\text{MG2}}}{\eta_{\text{MG2}}}, \tag{3.11}$$

where T_{MG2} denotes the torque generated by MG2, ω_{MG2} is the rotational speed of MG2, and η_{MG2} represents the efficiency of MG2. The torque output of MG2 is transmitted to the wheels through the gear ratios i_{R2} in the gearbox and i_{RD} in the differential, ensuring smooth and efficient operation.

Series hybrid mode

In hybrid series mode, the system activates the ICE and MG1 to operate as a range extender. ICE drives MG1, generating electricity to supplement the battery in powering MG2, which continues to propel the vehicle. The clutch is disengaged in this mode, and the energy flow

depends on the SoC of the battery. This mode is optimal for extended-range driving when the battery charge is low.

The energy flow in series hybrid mode is represented as:

$$P_{\text{MG2}} = P_{\text{Batt}} + P_{\text{ICE}} \cdot \eta_{\text{MG1}}, \tag{3.12}$$

where:

$$P_{\text{MG1}} = P_{\text{ICE}} \cdot \eta_{\text{MG1}}, \tag{3.13}$$

$$P_{\text{ICE}} = \frac{T_{\text{ICE}} \cdot \omega_{\text{ICE}}}{\eta_{\text{ICE}}}, \tag{3.14}$$

and

$$P_{\text{MG2}} = \frac{T_{\text{MG2}} \cdot \omega_{\text{MG2}}}{\eta_{\text{MG2}}}. \tag{3.15}$$

Here T_{ICE} and ω_{ICE} represent the torque and rotational speed of the internal combustion engine (ICE), respectively. MG1 and ICE operate at the same rotational speed during series hybrid mode, ensuring efficient power conversion and proper synchronization between the components.

Parallel hybrid mode

In parallel hybrid mode, both the ICE and MG2 contribute to the propulsion of the vehicle. The clutch is engaged to mechanically link the ICE to the gearbox, enabling it to provide additional driving torque. This mode is well-suited for highway driving or scenarios where higher power is needed.

The power flow in parallel hybrid mode is described by:

$$P_{\text{Drive}} = P_{\text{MG2}} + P_{\text{ICE}}, \tag{3.16}$$

where:

$$P_{\text{MG2}} = \frac{T_{\text{MG2}} \cdot \omega_{\text{MG2}}}{\eta_{\text{MG2}}}, \tag{3.17}$$

and

$$P_{\text{ICE}} = \frac{T_{\text{ICE}} \cdot \omega_{\text{ICE}}}{\eta_{\text{ICE}}}. \tag{3.18}$$

In this configuration, the combined torque of MG2 and ICE is transmitted to the wheels through the gearbox and differential, ensuring efficient and flexible operation. This setup allows the powertrain to adapt to various driving requirements, using both electrical and combustion power sources effectively.

The multi-mode PHEV powertrain combines the benefits of electric and hybrid vehicles. It offers the ability to adapt to varying driving conditions, optimizing fuel efficiency, and reducing emissions. Using multiple modes, the vehicle can achieve higher energy efficiency, extended range, and improved performance while maintaining operational flexibility in urban, suburban, and highway scenarios.

3.4 | Chapter Summary

This chapter provides an overview of the methodologies and facilities utilized in the research, focusing on key techniques and tools essential for this study. The primary studies in this chapter are summarized as follows:

(1) Fundamental methodologies are introduced in RL, including value-based, policy-based, and actor-critic methods. Additionally, the chapter explores the basics of MARL, highlighting the challenges and strategies of coordinating multiple agents in complex environments.

(2) The training and execution strategies for MARL are discussed, with a focus on CTDE and DTDE approaches. Furthermore, MARL algorithm variants, such as value function decomposition, communication learning, hierarchical structures, and causal inference, are reviewed to demonstrate their role in optimizing agent behaviors and interactions.

(3) Research facilities and resources, such as the ETAS and Speedgoat HiL testing platforms, are described in detail. The significance of driving cycle datasets, including OpenACC and simulated driving cycles, is emphasized for validating methodologies under various scenarios.

(4) The multi-mode PHEV powertrain system is modeled with a detailed analysis of key components, including vehicle power flow, engine modeling, and battery modeling. Various operational modes, such as pure EV mode, series hybrid mode, and parallel hybrid mode, are analyzed to support the optimization of energy management for improved performance and efficiency.

CHAPTER 4

Communication-Efficient MARL for Platoon Stability and Energy-efficiency Co-optimization in Cooperative Adaptive Cruise Control of CAVs

This chapter introduces energy optimization into the CACC problem, enhances communication efficiency, and validates the proposed approach using real-world scenarios, which have been published in *IEEE Transactions on Vehicular Technology* [281]. The main contributions of this chapter are as follows:

1) We introduce energy efficiency into short-term CACC problems as a decentralized MARL framework, designing a multi-objective reward function for energy consumption, stability, and safety. We benchmark the performance of our proposed algorithm against several non-communicative and communicative MARL approaches.

2) To improve the effectiveness of communicative information-sharing, we integrate QSGD and a BDC method into the proposed framework, enhancing the training process across various CACC scenarios. Comparisons are conducted to evaluate information-sharing efficiency, including sensitivity and scalability tests with varying platoon sizes.

3) The practical effectiveness of our approach is further demonstrated through real-world scenarios using open-source OpenACC data. By analyzing these real-world scenarios, we validate our model's applicability and robustness in achieving improved energy efficiency, stability, and safety in real-world CACC operations.

4.1 | Architecture of MARL-based CACC for CAVs and the Optimization Problem

The MARL system for CACC of CAVs is illustrated in Fig. 4.1. The system includes $\mathcal{N}$ CAVs that are connected through a V2V network. We assume that all CAVs are plug-in hybrids equipped with a 70 kW ICE, two motor/generators (MG1 with a nominal power of 21 kW and MG2 with a nominal power of 135 kW), and a 20 kWh battery [282, 283]. Details of the vehicle modeling and vehicle specification can be found in [283]. However, when considering the dynamics of plug-in hybrids without focusing on energy management between two power sources, the electric motor is utilized for propulsion in a short-term condition without engaging the full capabilities of the hybrid powertrain. Within an undirected graph $\mathcal{G}_t = (\mathcal{N}, \mathcal{E}_t)$ at each step t, agent i takes into account not only their current state but also the information $\mathcal{E}_t$ from their neighbors ($\mathcal{N}_{nei}$ represents the front and rear vehicle), and the states of its neighbor vehicles $\mathcal{N}_{nei}$. To capture and represent temporal dependencies in the shared environment more effectively, a state representation module is designed by incorporating the LSTM network with a fully-connected (FC) neural network for the actor networks and critic networks of all learning agents. To efficiently update the $\mathcal{N}$ critic networks effectively, the quantized policy update and binary differential consensus method are developed, enabling more effective learning and reducing communication overhead during training. Details of the system are described as follows:

4.1.1 Decentralized MARL framework for CACC of CAV

In this section, we introduce the CACC problem as a model-free multi-agent MDP framework through a decentralized MARL algorithm to maintain a predefined and adjustable inter-vehicle spacing (IVS) d^* and longitudinal velocity v^*. Since centralized approaches, though capable of managing all vehicle states, face significant real-world limitations, including scalability issues due to communication bottlenecks, single points of failure, and privacy concerns from aggregating data in a central node. These challenges make centralized solutions impractical for large-scale, real-time coordination. The decentralized framework provides agents with only

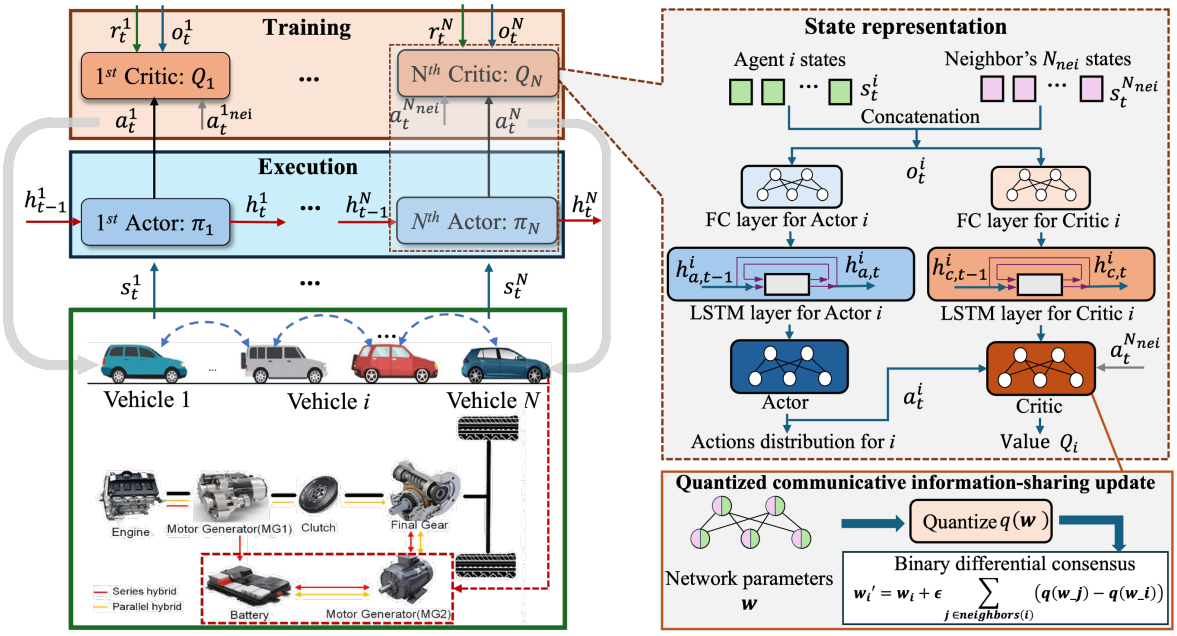


Fig. 4.1. The multi-agent system for CAVs

partial observations of the environment, concentrating mainly on the vehicles directly ahead and behind. This reflects the practical constraints for each vehicle, which are limited in their ability to communicate with other vehicles. Consequently, a partially observable Markov decision process (POMDP) has been formulated by the tuple $\left(\mathcal{S}, \{\mathcal{A}_i\}_{i\in\mathcal{N}}, P, \{R_i\}_{i\in\mathcal{N}}, \{\mathcal{G}_t\}_{t\geq 0}\right)$. More details [96] are given as follows:

State Space

To achieve an optimal balance between accurately modeling vehicle behavior and avoiding overcomplexity, the state space $\mathcal{S}$ is defined as $[v, v_{\text{diff}}, vh, d, u]$, specifically,

- The first state is the current normalized vehicle velocity $v = \left(v_{i,t} - v_{i,0}\right)/v_{i,0}$ from time 0 to current time t to reflect the relative change in a scaled manner, where v_i is longitudinal velocity.

- The second state is the clipped vehicle velocity difference with its leading vehicle $v_{\text{diff}} = \text{clip}\left(\left(v_{i-1,t} - v_{i,t}\right)/5, -2, 2\right)$ to ensure that vehicles can respond appropriately to the leader's behavior in the controllable range, with v_i and the corresponding preceding vehicle $(i-1)$.

- The third state is the IVS-based velocity according to vehicle behavior modeling, $vh = \mathrm{clip}\left(\left(v^{\circ}(d_i) - v_{i,t}\right)/5, -2, 2\right)$, where $v^{\circ}(d_i)$ represents the behavior of the i th vehicle with optimal velocity model (OVM) [284] and d_i is the IVS of each vehicle i. The formulation of OVM for the behavior of the ith vehicle is defined as follows:

$$
\begin{aligned}
u_i &= \alpha_i \left(v^{\circ}\left(d_i; d^s, d^g\right) - v_i\right) + \beta_i \left(v_{i-1} - v_i\right) \\
v^{\circ}(d_i) &\triangleq \begin{cases} 0, & \text{if } d_i < d^s, \\ \frac{1}{2} v_{\max}\left(1 - \cos\left(\pi \frac{d_i - d^s}{d^g - d^s}\right)\right), & \text{if } d^s \le d_i \le d^g, \\ v_{\max}, & \text{if } d_i > d^g \end{cases}
\end{aligned}
\tag{4.1}
$$

  where α_i and β_i are the IVS gain and relative velocity gain, respectively, highlighting how both the IVS and the relative velocity contribute to adjustments in acceleration u_i of vehicle i. Specifically, $d^s = 5$ m and $d^g = 35$ m denote the stop IVS and the IVS at full velocity of the traffic flow, respectively, reflecting the behavior of vehicles under congested conditions and free-flowing traffic [96]. Additionally, v° represents the IVS-based velocity policy. This policy function is continuous and differentiable, which is beneficial for computational models and simulations in the environment of the MARL framework.

- The fourth state is the normalized IVS $d = \left(d_{i,t} + \left(v_{i-1,t} - v_{i,t}\right)\Delta t - d^*\right)/d^*$, where Δt is the sampling time. The discretized **vehicle longitudinal kinematics** for each vehicle i are given by:

$$
\begin{aligned}
d_{i,t+1} &= d_{i,t} + \left(v_{i-1,t} - v_{i,t}\right)\Delta t \\
v_{i,t+1} &= v_{i,t} + u_{i,t}\Delta t
\end{aligned}
\tag{4.2}
$$

- The last one is the normalized acceleration $u = u_{i,t}/u_{\max}$, where $u_{\max}$ is the maximum acceleration.

Normalizing and clipping the velocity, IVS, and acceleration ensure that the state variables remain within practical bounds to enhance the model's stability and predictability across diverse driving scenarios. Additionally, to meet safety, comfort, and power requirements, each vehicle

must adhere to the longitudinal modeling constraints [72]:

$$\begin{aligned} d_i &\geq 1 \text{ m} \\ 0 \leq v_i &\leq 30 \text{ m/s} \\ -2.5 \text{ m/s}^2 \leq u_i &\leq 2.5 \text{ m/s}^2 \end{aligned} \tag{4.3}$$

Action Space

In the defined CACC problem, the action space $\{\mathcal{A}_i\}_{i\in\mathcal{N}}$ is straightforwardly related to the longitudinal control. According to Eq. 4.1, the OVM control behavior is affected by two critical hyperparameters: IVS gains α, and relative velocity gains β. Thus, these values are chosen from a predefined set comprising four levels: $\{(0,0),(0.5,\ 0),(0,0.5),(0.5,0.5)\}$. Subsequently, the longitudinal action is calculated by Eq. 4.1. The selection of four discrete sets of parameters for OVM within a MARL framework is driven by the goal of balancing simplicity, computational efficiency, and the capacity to model a range of driving behaviors for longitudinal vehicle control. These distinct parameter sets simplify the exploration space for the RL, enabling an efficient search through potential actions to optimize driving performance.

4.1.2 Co-optimization of platoon stability and energy efficiency

Reward function

The collective objective is to train the agents toward these desired behaviors. So the multi-objective reward $\{R_i\}_{i\in\mathcal{N}}$ for the i th agent is designed as follows:

$$\begin{aligned} R_i = w_1 \left(d_i - d^*\right)^2 + w_2 \left(v_i - v^*\right)^2 + w_3 u_i^2 \\ + w_4 \left(2d^s - d_i\right)_+^2 + w_5 P_i \end{aligned} \tag{4.4}$$

where w_1, w_2, w_3, w_4, and w_5 are the weighting coefficients. This multi-objective reward design is conducted to achieve a trade-off between performance efficiency, driving comfort, safety, and energy savings in the CACC system. P_i is the power consumption of the electric motor of each vehicle, where a polynomial-based, differentiable approximation of an energy

consumption model [285] is calculated as follows:

$$\begin{aligned} P_i &= T_i \cdot \omega_i \cdot \eta_i^{-k} \\ T_i &= F_i \cdot R \end{aligned} \tag{4.5}$$

where T_i represents the motor torque of each vehicle, ω_i is the rotational velocity of the motor for each vehicle, and η_i is the efficiency of electric-mechanical conversion; R is the wheel radius. The k indicates the operational mode. $k = 1$ denotes the driving model while $k = -1$ denotes regenerative braking mode. According to **vehicle longitudinal dynamics**, the driving force of each vehicle i, F_i, can be calculated by

$$\begin{aligned} F_i &= mgf\cos\alpha + \frac{1}{2}\rho A_f C_d v_i^2 + mg\sin\alpha + mu_i \\ P_i &= F_i \cdot v_i \end{aligned} \tag{4.6}$$

where m is the vehicle mass; g is the gravity acceleration; f is the rolling resistance coefficient; ρ is the air density; A_f is the front area of the vehicle; C_d is the aerodynamic drag coefficient; α is the road slope. The critical parameters can be found in [283]. To simplify the calculation and calibration, the vehicle energy consumption, $P_i(v_i, u_i)$, is formulated as a function of vehicle velocity v_i and acceleration u_i as in [286],

$$P_i(v_i, u_i) = \sum_{k=0}^{4}\sum_{j=0}^{4} p_{kj} \cdot v_i^k \cdot u_i^j \tag{4.7}$$

where p_{kj} is the polynomial coefficient determined by fitting as shown in Fig. 4.2. In this work, each vehicle equipped with the CACC is considered to have identical capabilities and physical characteristics. The inclusion of the real energy model reveals that the CACC not only regulates vehicle velocity and IVS but also actively manages the powertrain to improve fuel economy and energy efficiency.

Further, if the IVS is less than twice the stop IVS d^s, preventing collisions and ensuring the safety of the vehicle platoon. Here, the "+" in the reward function indicates that a penalty is applied only when the d_i falls below twice the d^s, functioning similarly to a Rectified Linear Unit (ReLU) by activating the penalty under specific conditions; if the IVS $d_i \leq 1$ m, the

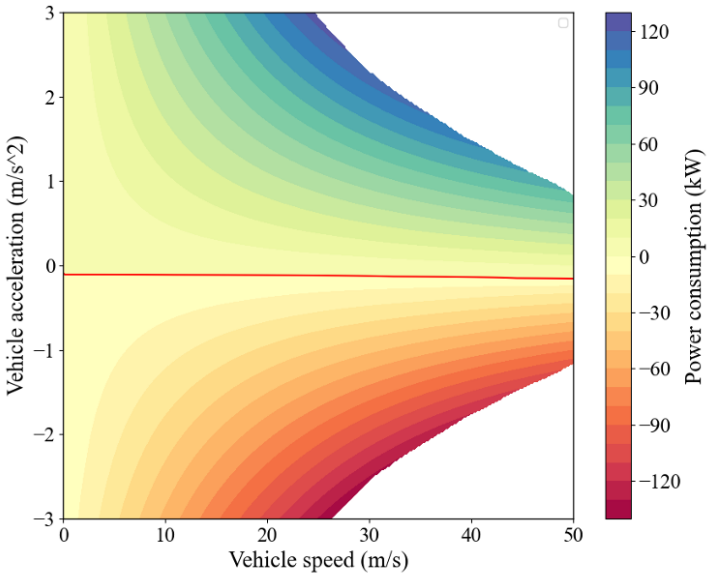


Fig. 4.2. Contour plot of the fitted energy consumption model with respect to vehicle velocity and acceleration.

substantial penalty of 1000 serves as a strong deterrent against behaviors that could lead to collisions, effectively terminating the training episode to guarantee safety.

Transition Probabilities

Based on the model-free MARL, we do not assume any prior knowledge of this transition probability P while developing our MARL algorithm.

4.2 | Communication-efficient MARL for CACC

4.2.1 Decentralized Multi-agent MDP problem

The multi-agent MDP is defined by the tuple $\left(\mathcal{S}, \{\mathcal{A}_i\}_{i\in\mathcal{N}}, P, \{R_i\}_{i\in\mathcal{N}}, \{\mathcal{G}_t\}_{t\geq 0}\right)$, where $\mathcal{S}$ is the global state space and $\mathcal{A}_i$ is the action space for each agent i. The local reward function for each agent i is $R_i : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$, and $P : \mathcal{S}\times\mathcal{A}\times\mathcal{S} \to [0,1]$ is the state transition probability function. Each agent i chooses its action a_i based on the local policy $\pi^i : \mathcal{S}\times\mathcal{A}_i \to [0,1]$, leading to a joint policy $\pi(s,a) = \prod_{i\in\mathcal{N}} \pi^i(s,a_i)$. A fully decentralized framework is characterized by the fact that the reward is received locally and the action is executed individually by each agent. The policy for each agent i is parameterized by $\pi^i_{\theta^i}$ with θ^i over Θ^i, where $\Theta = \prod_{i=1}^N \Theta^i$ by packing the parameters together as $\theta = \left[\left(\theta^1\right)^T, \cdots, \left(\theta^N\right)^T\right]^T \in \Theta$. Thus, the joint policy then becomes $\pi_\theta(s,a) = \prod_{i\in\mathcal{N}} \pi^i_{\theta^i}(s,a_i)$.

We assume that the Markov chain $\{s_t\}_{t\geq 0}$ is irreducible and aperiodic under any π_θ, with the stationary distribution denoted by d_θ. Additionally, the Markov chain of the state-action pair $\{(s_t, a_t)\}_{t\geq 0}$ has a stationary distribution $d_\theta(s) \cdot \pi_\theta(s, a)$ for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$. Furthermore, the collective objective of the agents is to collaboratively find a policy π_θ that maximizes the globally averaged long-term return over the network based solely on local information, formalized as:

$$\begin{aligned} \max_\theta J(\theta) &= \lim_T \frac{1}{T}\mathbb{E}\left(\sum_{t=0}^{T-1}\frac{1}{N}\sum_{i\in\mathcal{N}} r_{t+1}^i\right) \\ &= \sum_{s\in\mathcal{S}, a\in\mathcal{A}} d_\theta(s)\cdot\pi_\theta(s,a)\cdot\bar{R}(s,a), \end{aligned} \tag{4.8}$$

where $\bar{R}(s, a) = N^{-1} \cdot \sum_{i\in\mathcal{N}} R^i(s, a)$ is the globally averaged reward function. The global relative action-value function $Q_\theta(s, a)$ and state-value function $V_\theta(s)$ are defined as $Q_\theta(s, a) = \sum_t \mathbb{E}\left[\bar{r}_{t+1} - J(\theta) \mid s_0 = s, a_0 = a, \pi_\theta\right]$ and $V_\theta(s) = \sum_{a\in\mathcal{A}} \pi_\theta(s, a)Q_\theta(s, a)$ respectively. Furthermore, the advantage function can be defined as $A_\theta(s, a) = Q_\theta(s, a) - V_\theta(s)$.

Despite each agent learning independently in this framework, the overall goal is to maximize the average global reward $\bar{R}(s, a)$. This means that while each agent (each vehicle in the CACC) learns its own strategy, they collectively aim to improve the overall performance. Non-stationarity can occur; thus, some collaborative approaches have been conducted to deal with this. For example, the policy network (the part of the agent that decides what action to take) is updated independently by each agent. This means each agent doesn't need to guess what other agents are doing; it just focuses on its own learning, such as Independent Advantage Actor-Critic (IA2C) [259]. However, for the critic network (the part that estimates the value of the chosen action), a collaborative approach is used. Each agent shares its value function estimates with its neighboring agents (those close by in the network). These estimates are averaged to jointly evolve the overall performance. This is achieved with weight averaging consensus (WAC)-MARL, where each agent's weights are averaged with its neighbors. Alternatively, directed information sharing, which communicates with neighbors, efficiently relays information in both directions within a platoon. This approach uses differential consensus-enhanced adjustment (DCEA)-MARL, adjusting each agent's weights by adding a

weighted difference between the neighbors' weights and its own, as described:

$$\boldsymbol{w}_i' = \boldsymbol{w}_i + \epsilon \sum_{j \in \mathcal{N}_{\text{nei}}} \left(\boldsymbol{w}_j - \boldsymbol{w}_i \right). \tag{4.9}$$

This term captures the essence of this update mechanism, where an epsilon factor ϵ is the scale factor.

4.2.2 Quantified communicative information-sharing update design

The decentralized framework operates without the need for a centralized controller. However, it operates under the assumption that all agents are homogeneous, possessing identical characteristics. While this assumption simplifies the structural complexity of the problem, it fails to capture the inherent diversity among individual agents, especially for CACC. To tackle this issue, we introduce a difference update strategy that promotes a balance between individual learning and the collaborative influence by neighboring agents. Significantly, quantified stochastic gradient descent (QSGD) method is a family of compression schemes with convergence guarantees and practical performance [287]. Then, a binary differential consensus (BDC) method is presented, where agents first quantize their weights and then update their weights based on the quantized differential weights between themselves and their neighbors. Therefore, the update rule is described as:

$$\boldsymbol{w}_i' = \boldsymbol{w}_i + \epsilon \sum_{j \in \mathcal{N}_{\text{nei}}} \left(q(\boldsymbol{w}_j) - q(\boldsymbol{w}_i) \right) \tag{4.10}$$

where w_i' is the updated weight for agent i. w_i is the current weight of the agent i before the update. ϵ is a small positive scalar that scales the update magnitude. $q\left(w_j\right)$ and $q\left(w_i\right)$ are the quantized weights of neighbor agent j and agent i, respectively. The sum is taken over all neighbors j of agent i, as defined by the neighbor mask.

We utilize quantization to potentially reduce communication overhead while still achieving consensus through differential updates. To mitigate the impact of quantization errors, QSGD often incorporates mechanisms like error accumulation (also known as error feedback), which

means the quantization error from one iteration (i.e., the difference between the quantized gradient and the actual gradient) is carried over to the next iteration. This error feedback helps compensate for the quantization errors over time, allowing for more accurate updates despite the initial loss of information. The update strategy is given in **Algorithm** 1.

Algorithm 1: The gradient descent algorithm with gradient encoding

Data: Parameter vector w
Procedure: Gradient Descent
for *each iteration* t **do**
    Quantize gradient: $q(\nabla f(\boldsymbol{w})) \leftarrow \text{Quantize}\ (\nabla f(\boldsymbol{w})))$;
    Apply gradient $\boldsymbol{w} \leftarrow \boldsymbol{w} - \epsilon_t q(\nabla f(\boldsymbol{w}))$
end

In the CACC scenario, smaller messages lead to faster transmissions, which is critical for real-time applications like CACC, where timely updates directly impact system performance and safety. By minimizing bandwidth reliance, the system can scale more efficiently, supporting a larger number of vehicles without a proportional increase in communication overhead. Although this approach poses challenges regarding convergence and accuracy, these can be mitigated through techniques like error feedback.

4.3 | Experiment setups

In this section, we evaluate the BDC-MARL framework across various CACC scenarios. Initially, our approach is compared with several state-of-art MARL strategies. Then we explore the efficiency of diverse information-sharing methods. Concurrently, we investigate the effects of different platoon sizes. Finally, the proposed algorithm is applied to scenarios derived from real-world OpenACC data.

4.3.1 General setups for the learning agents

The network architecture comprises a single fully connected input layer for processing state information and an LSTM layer tasked with message extraction, all with hidden layers containing 64 neurons. An orthogonal initializer is utilized to optimize the network parameters during training. Training is conducted over 6×10^5 steps, utilizing a discount factor of $\gamma = 0.99$, an actor learning rate of 5.0×10^{-4}, and a critic learning rate of 2.5×10^{-4}. In the reward function, hyperparameters w_1, w_2, w_3, w_4, and w_5 in the reward function are set to $-1.0, -1.0, -0.1$,

-5.0, and -10, respectively, with significant emphasis on penalizing scenarios that feature inadequate safe IVS distances. To enhance generalization, each model undergoes training three times using distinct random seeds on a Ubuntu 18.04 server with a NVIDIA RTX 3080 and i7-12700 CPU.

4.3.2 Real-world driving scenario extraction

The real-world scenarios are from an open-access database involving vehicles with ACC systems provided by the JRC of the EC [270]. The dataset used in this paper is collected at the ZalaZONE from 10 commercially available vehicles with tunable ACC settings.

The real-world driving scenarios are generated for the testing in this paper based on the Dynamic Platform. The Dynamic Platform focuses on the vehicles' response to such perturbations in a controlled environment. This includes a rapid deceleration followed by an acceleration back to the original velocity. Three groups are selected and extracted respectively from three typical datasets (dynamic_part1, dynamic_part2, dynamic_part9) from timestamps $(316s - 376s, 450s - 510s, 473s - 533s)$. Additionally, an energy model was incorporated into the analysis to evaluate the energy consumption resulting from adjustments in velocity and acceleration. Consequently, data collected from experiments using actual vehicles is utilized to assess ACC performance under various driving conditions.

4.4 Results and discussion

4.4.1 Co-optimization Performance with State-of-the-art MARL

To demonstrate the effectiveness of the proposed method, we compare it with several state-of-the-art MARL algorithms, which include both non-communicative and communicative approaches. Specifically, **IA2C** [259] follows an independent learning strategy and is an A2C implementation of multi-agent deep deterministic policy gradient (MADDPG). IA2C employs a decentralized actor network and a centralized critic to evaluate the value function without explicit communication between agents. While **FPrint** [243] integrates neighboring agents' policies into its input (fingerprints) but, like IA2C, lacks an explicit communication protocol. Each agent has an independent policy network, and the value function is still evaluated

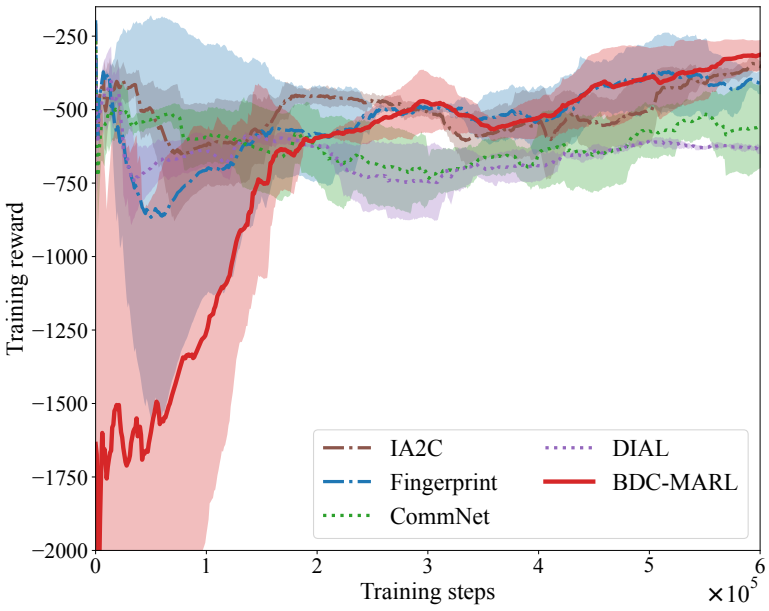
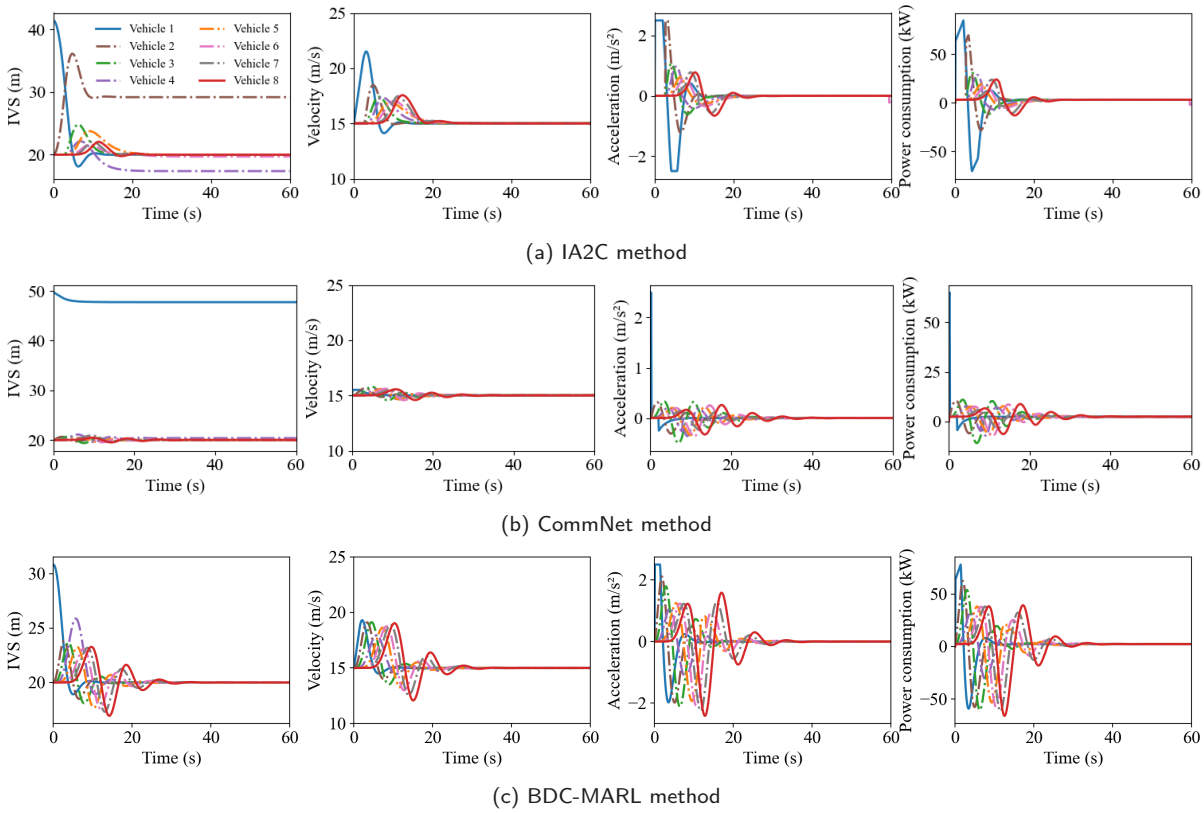


Fig. 4.3. The comparison with different methods

centrally with a global critic. In contrast, **DIAL** [113] and **CommNet** [288] implement learnable communication protocols that allow for the inclusion of additional information from neighboring agents, such as states or policy details, which necessitates a higher communication bandwidth, both of which use decentralized actor and decentralized framework. In particular, the local message sizes are $O(|s^i| + |\pi_i| + |h^i|)$for DIAL, $O(|s^i| + |h^i|)$ for CommNet, $O(|\pi_i|)$ for FPrint, $O(|w_i|)$ for BDC-MARL, and zero for IA2C. All the algorithms compared utilize the same deep neural network architecture.

A comparative learning progress through training curves for the various MARL algorithms is described as shown in Fig. 4.3 and Table 4.1. The proposed BDC-MARL outperforms other algorithms, starting with a higher initial reward and steadily increasing performance. Despite early fluctuations, it maintains a higher reward throughout training, indicating extensive strategy exploration before stabilizing. This thorough exploration can lead to more optimal policies. After the training phase, the evaluation for each algorithm was conducted using 50 distinct test seeds, each with different initial conditions to ensure comprehensive testing, as shown in Table 4.2. The BDC-MARL method achieved strong overall performance, particularly in maintaining low platoon power consumption without compromising other key metrics such as IVS, velocity, and acceleration. Although BDC-MARL shows slightly higher standard deviations in some



(a) IA2C method

(b) CommNet method

(c) BDC-MARL method

Fig. 4.4. The performance comparison of all vehicles with typical MARL algorithms.

metrics, this variability indicates its adaptive strategy under different conditions. Additionally, it balances efficiency with competitive computation time compared to other algorithms.

Table 4.1
Reward comparison with different algorithms

Algorithms	IA2C	Fingerprint	Commnet	DIAL	BDC-MARL
Reward value	-351.07	-409.12	-559.45	-636.19	**-314.99**

Table 4.2
Test performance comparison with different algorithms

Algorithms	IA2C		Fingerprint		Commnet		DIAL		BDC-MARL	
	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std
IVS (m)	**20.75**	0.81	21.68	0.31	22.38	0.03	22.08	0.2	20.76	0.78
Velocity (m/s)	15.28	0.47	15.11	0.26	**15.01**	0.03	15.06	0.09	15.26	0.69
Acceleration (m/s^2)	0.11	0.18	0.08	0.13	**0.01**	0.02	0.04	0.05	0.15	0.21
Power (kW)	17.97	0.31	18.28	0.42	18.94	0.04	18.89	0.15	**17.85**	0.46
Comp. time (h)	**1.39**		2.16		1.93		2.06		1.83	

Fig. 4.4 compares the performance of 8 vehicles using different MARL algorithms, including I2AC, CommNet, and BDC-MARL, based on IVS, velocity, acceleration, and power consumption over time. Divergences in some vehicles with I2AC (e.g., Vehicle 2 and 4) and CommNet (e.g.,

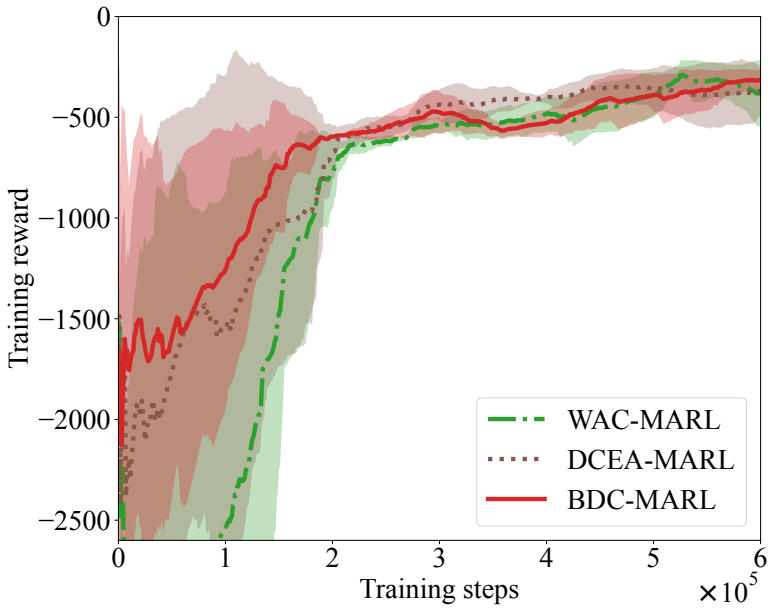


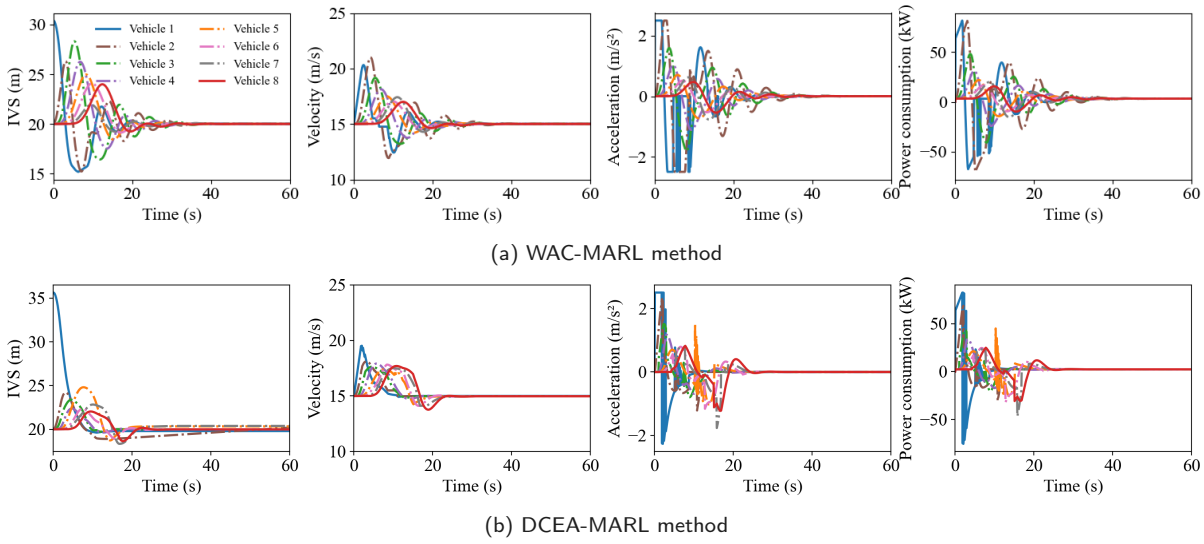
Fig. 4.5. The comparison with three different sharing methods

Vehicle 1) suggest challenges in information sharing and coordination. Effective communication is crucial for synchronizing actions and maintaining uniform IVS and velocity in MARL systems like CACC. The I2AC method's divergences may indicate lagging communication channels, while CommNet's velocity fluctuations suggest inefficient use of communicated information. These issues imply potential communication latency or protocols preventing agents from aligning actions effectively in dynamic tasks. Conversely, BDC-MARL's convergence towards the desired IVS and velocity with less variability indicates a more efficient communication strategy, enabling better coordination and faster adaptation to achieve goals.

4.4.2 Communication Efficiency of MARL on CACC

Using different information-sharing methods with fully decentralized framework, including **WAC-MARL**, **DCEA-MARL** and **BDC-MARL** is crucial in CACC scenario. Fig. 4.5 presents a comparison of learning curves, highlighting the superior performance of the BDC-MARL method against the other two. While BDC-MARL exhibits some mid-training variability, it ultimately achieves the highest rewards, making it the most effective method for maximizing training rewards. DCEA-MARL shows good initial learning velocity and stability, ideal for scenarios where early performance is critical. WAC-MARL demonstrates steady improvement and stability, suitable for environments requiring consistent performance gains over time. After

the training phase, each algorithm was evaluated on 50 seeds under varied starting conditions. The performance process with all vehicles is illustrated in Fig. 4.6. Comparing Fig. 4.4 (c) with Fig. 4.6, all three methods eventually converge towards a pre-designed IVS and velocity. The WAC method shows initial over-corrections in acceleration and power consumption before stabilizing. The DCEA method reaches stability quicker than the WAC method, despite initial inconsistencies in acceleration and velocity. The BDC-MARL approach quickly converges in terms of IVS and maintains a more stable velocity and acceleration pattern, due to its efficient use of limited bandwidth and information processing capabilities, preventing data overflow and response lag.



(a) WAC-MARL method

(b) DCEA-MARL method

Fig. 4.6. The performance process of different sharing methods.

The radar chart in Fig. 4.7 evaluates the performance of three methods using five metrics: acceleration (Acc), velocity (Vel), Power consumption (Power), Computation time (Time), and IVS. This chart effectively visualizes comprehensive performance through the aggregate area of each method's plot. A smaller area within the radar chart signifies a more robust overall performance, indicating a method's efficiency and effectiveness across the evaluated dimensions. Therefore, the balance across all five metrics for BDC-MARL can imply a more robust overall system performance. For acceleration and velocity, all three methods show comparable performance, with their lines relatively close to the outer edge. However, in IVS and power consumption, the BDC-MARL method is noticeably closer to the edge, indicating superior performance in these critical areas. While WAC-MARL saves a bit of computation

time due to its simpler information-sharing approach, BDC-MARL offers a more balanced and robust overall performance across all metrics.

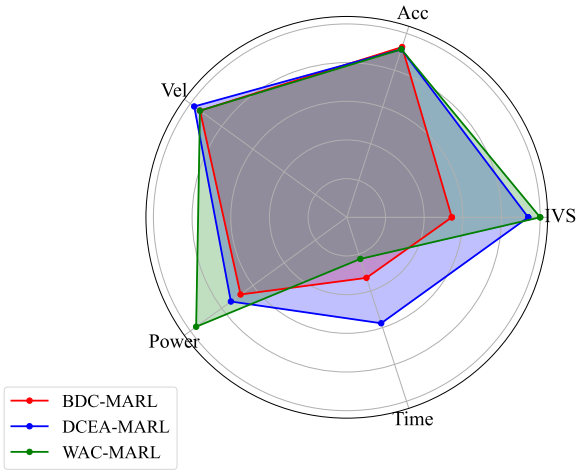


Fig. 4.7. The statistical comparison of different communicative methods

Fig. 4.8 presents a statistical comparison of different communicative bit sizes for the quantified communicative information-sharing update method. The graph represents the normalized communication bits, with lower values indicating a reduction in data transfer compared to the full 16-bit precision benchmark. As the number of bits decreases from 16 (DCEA-MARL) to 2 ((BDC-MARL), there is a significant reduction in communication overhead. As for the normalized computation time, showing how the decrease in bit size reduces computation time compared to the 16-bit baseline. The lower bit sizes, particularly 2 and 4 bits, result in a noticeable improvement in computational efficiency. On the other hand, the normalized evaluated reward is also benchmarked against the full 16-bit precision. While there is a slight drop in evaluated reward as the bit size decreases, especially at 2 bits, the reduction is minimal, demonstrating that lower bit precision can maintain robust performance while significantly reducing communication and computation costs. Therefore, the BDC-MARL achieves an excellent balance between computational efficiency and performance.

In summary, while the WAC and DCEA methods initially face oscillations, they eventually achieve consensus, promoting collective efficiency. However, BDC method stands out for its quick convergence and stability, demonstrating its effectiveness in environments with limited communication resources.

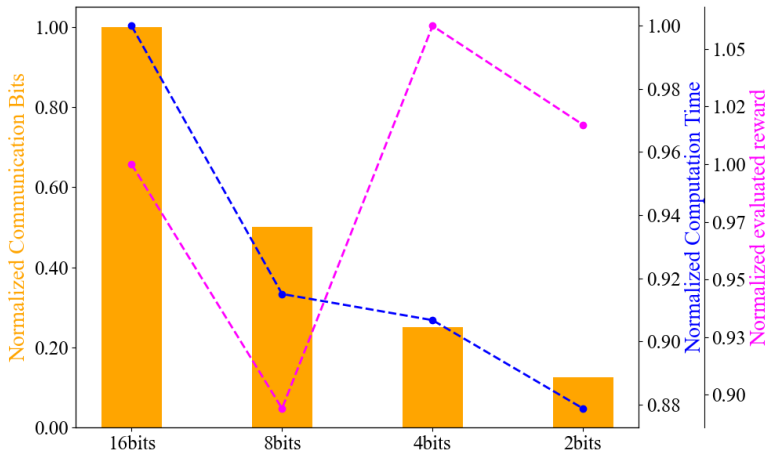


Fig. 4.8. The statistical comparison of different communicative bits

4.4.3 Impact of Platooning Size on the Scalability on CACC

In CACC, the overall performance including energy efficiency, safety, and the effectiveness of communication will be affected by different vehicle sizes, since there is the propagation phenomenon in the response and communication.

Fig. 4.9 shows the training curves with different sizes, ranging from a minimal arrangement (i.e., 2 AVs) to more populated configurations (i.e., 6 and 8 AVs), thereby providing a comprehensive overview of the scalability and adaptability in a variety of urban traffic scenarios, since larger vehicle sizes experience more complex effects, such as increased air resistance or more intricate behavior in a platoon. In Fig. 4.9, we can find that Size 8 needs more sophisticated models to handle these effects effectively, which could explain a slower but steady increase in training process. In contrast, Size 2 experiences less interaction and therefore might find an optimal policy faster.

The overall test performance under 50 different trials is displayed in Fig. 4.10. The comprehensive analysis of varying platoon sizes suggests that while larger platoons achieve fuel savings due to decreased aerodynamic drag, this does not result in a proportional increase in power consumption. Interestingly, a platoon size of 6 emerges as the most advantageous, exhibiting consistent IVS, moderate velocities, and the lowest variation in power consumption, indicative of an efficient balance between fuel efficiency and operational stability. Although larger platoons benefit from aerodynamic efficiency, they also confront the challenge of maintaining stable and safe following distances amidst dynamic traffic conditions, which is less of a concern

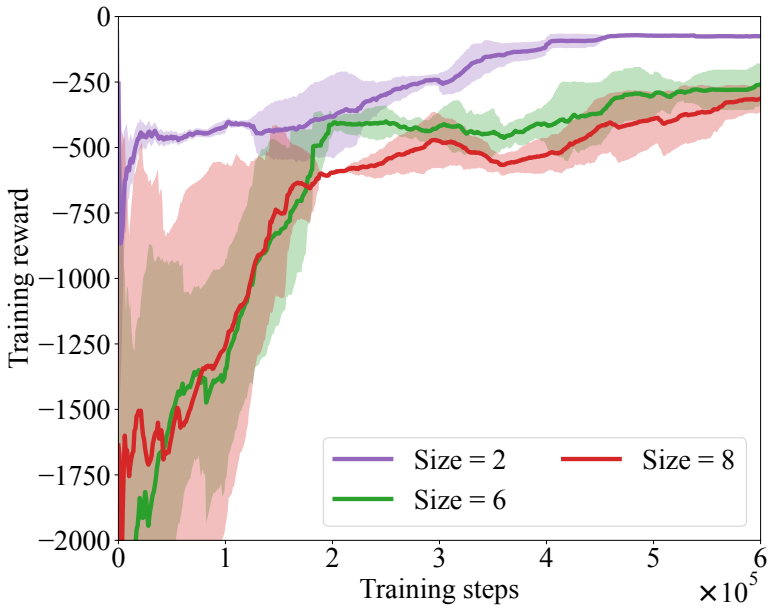


Fig. 4.9. The comparison with different sizes

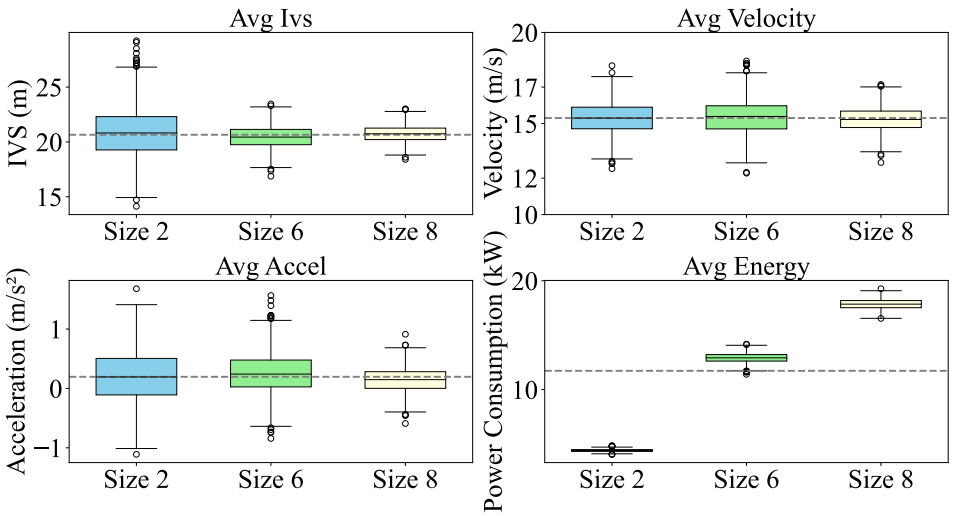


Fig. 4.10. The comparison with different size

with smaller platoons. Therefore, a platoon size of 6 is potentially optimal for achieving a harmonious balance between power efficiency, vehicle spacing stability, and the complexities inherent in managing larger numbers of closely following vehicles. More importantly, it also indicates that the presented algorithms effectively optimize platoon dynamics, achieving fuel economy improvements without a corresponding escalation in power consumption across varying platoon sizes.

4.4.4 Co-optimization Performance Validation with Real-World Data

In this part, the different targets (the target IVS and velocity) on the performance of the presented algorithm has been conducted from the real-data scenarios extracted from OpenACC. The examination is primarily focused on analyzing three different IVS settings with the target velocity 11 m/s and the Size 4, ranging from IVS setting S $(12m)$, M $(16m)$, and L $(20m)$, labelled as Group 1, Group 2, and Group 3.

Table 4.3 presents average and standard deviation values for velocity, IVS, acceleration, and total energy for vehicles in the proposed BDC-MARL algorithm and real-world OpenACC data. Here, the average velocities are close to the target velocity of 11 m/s, indicating that all strategies and groups are performing well in terms of pre-designed setting. However, there are noticeable differences in the standard deviation of IVS, showing greater variability in the OpenACC. The total energy consumption is fairly consistent across groups, since the analysis has been conducted over a short period, which might not be sufficient to capture the long-term benefits of BDC-MARL in terms of energy savings but the stability of the vehicle dynamics is improved.

Table 4.3
Extraction of real-world scenarios from OpenACC

Group	Statistical Values	Method	Velocity (11 m/s)	IVS (m)	Acceleration	Total energy for all vehicles (kWh)
1 (S)	Average	OpenACC	10.81	16.96	0.01	0.11
		BDC-MARL	10.75	**11.99**	**0**	0.11
	Std	OpenACC	2.01	5.36	0.77	/
		BDC-MARL	**0.73**	**1.37**	**0.23**	/
2 (M)	Average	OpenACC	10.28	21.89	0.11	0.09
		BDC-MARL	10.2	**16.99**	**0**	0.09
	Std	OpenACC	**2.08**	3.68	**0.68**	/
		BDC-MARL	2.23	**2.27**	0.71	/
3 (L)	Average	OpenACC	10.6	25.12	0.02	0.10
		BDC-MARL	10.47	**19.54**	**0**	0.10
	Std	OpenACC	1.68	4.3	**0.55**	/
		BDC-MARL	**1.47**	**2.15**	**0.55**	/

As shown in Fig. 4.11, the performance comparisons between the BDC-MARL strategy and OpenACC data across three groups is conducted, the BDC-MARL strategy maintains the target IVS with minor deviations, especially in the setting with the largest IVS (Group 3). Since larger IVS could introduce greater variances due to increased reaction time and distance, making the system slightly less accurate in maintaining a consistent gap. The test performance in the real-data scenario for different groups is relatively close in metrics velocity and acceleration, the BDC-MARL algorithm with less delay effectively balances acceleration and braking to match the target velocity despite different spacing settings.

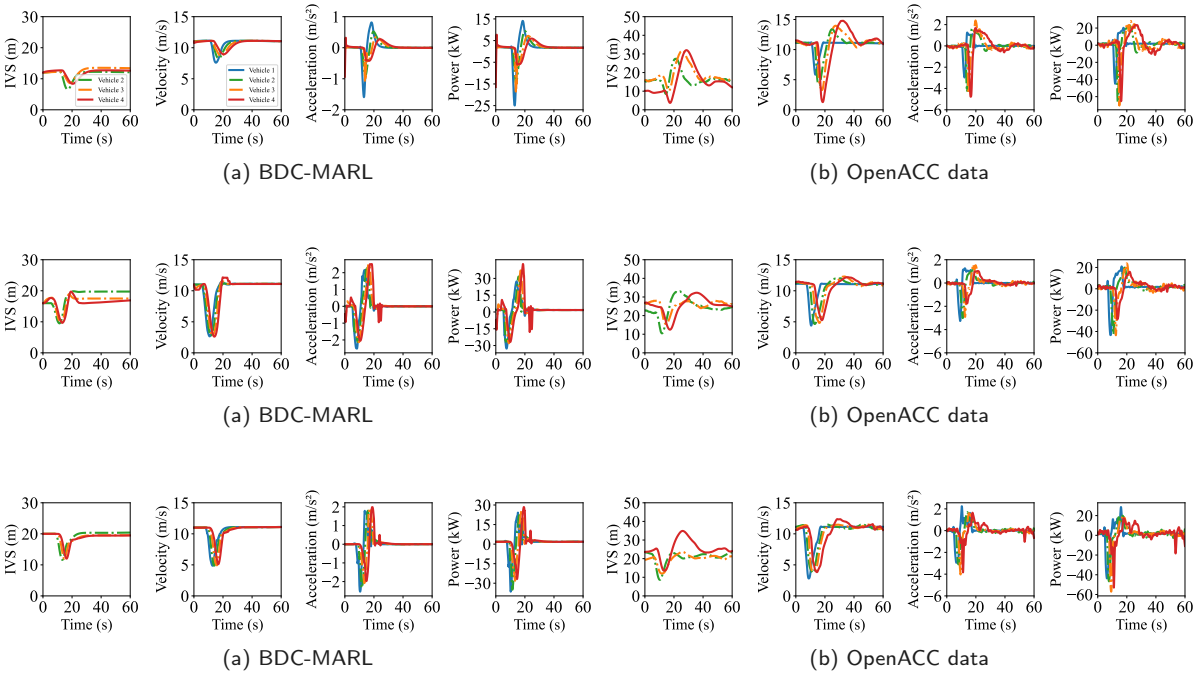


Fig. 4.11. Comparison of CACC strategies across different groups and corresponding OpenACC data.

4.5 | Chapter summary

The chapter introduces a BDC-MARL approach to enhance system robustness and efficiency. By minimizing dependency on potentially delayed or unreliable information exchanges between vehicles, this approach optimizes individual vehicle responses to dynamic conditions. The main conclusions are summarized as follows:

(1) Compared with state-of-the-art algorithms, the BDC-MARL achieves the largest improvement in energy savings, up to 5.8%, with an average velocity of 15.26 m/s and an IVS of 20.76 m. In CACC scenarios, delayed messages in real-time CACC tasks hinder perfor-

mance, as observed with IA2C and CommNet. In contrast, BDC-MARL demonstrates efficient communication that effectively mitigates this issue.

(2) By integrating QSGD into the communication protocol, BDC-MARL achieves stable performance across different information-sharing methods, rapidly converging to a stable IVS. It maintains a more consistent velocity and acceleration, utilizing limited bandwidth efficiently to avoid data overflow and reduce communication delays.

(3) A 6-vehicle platoon is optimal for balancing efficiency and stability, demonstrating the scalability in optimizing platoon dynamics for energy efficiency. Larger platoons save fuel through reduced aerodynamic drag without increasing power use proportionally.

(4) While both BDC-MARL and the algorithm behind the OpenACC dataset achieve similar average speeds, the OpenACC-based algorithm shows higher IVS variability due to less effective vehicle coordination. Despite short-term consistency in energy consumption, BDC-MARL enhances vehicle dynamics stability by leveraging robust communication protocols and optimized response strategies.

CHAPTER 5

Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning

While MADRL has been explored in various domains, including the previous chapter, its application to EMSs for PHEVs remains limited. This chapter proposes MARL as a promising solution for MIMO control in multi-mode PHEVs and focuses on the development of an EMS based on MADRL. This work, which has been published in *Applied Energy* [283] and a conference [289] with three new contributions:

1) The best setting for the DDPG agents has been obtained through a parametric study concerning network layers, learning rate, and policy noise.

2) A hand-shaking strategy has been developed by introducing a relevance ratio that facilitates collaboration among the DDPG agents and synchronizes their learning processes towards the global goal.

3) Multiple optimization objectives, including minimizing fuel consumption and minimizing battery SoC sustaining error, have been studied by implementing the proposed multi-agent learning system in a multi-mode PHEV.

5.1 | MIMO control of a multi-mode PHEV

The architecture of the multi-mode PHEV studied in this paper is shown in Fig. 5.1. The motor generator (MG1) and the engine work together to maintain the battery's SoC at a certain level for safety. The other motor (MG2) and engine are the power sources for driving.

The multi-mode PHEV can work in different modes, which are controlled by engaging or disengaging the clutch, as illustrated in Fig. 5.1. In series mode (red lines), the clutch is disengaged, and the engine powers MG1 to generate electricity, which is then used by MG2 to drive the wheels. MG1 does not directly drive the wheels, as it is optimized for generation rather than traction. In parallel mode (yellow lines), the clutch is engaged, and the engine will provide the part of driving torque as the supplement to the MG2.

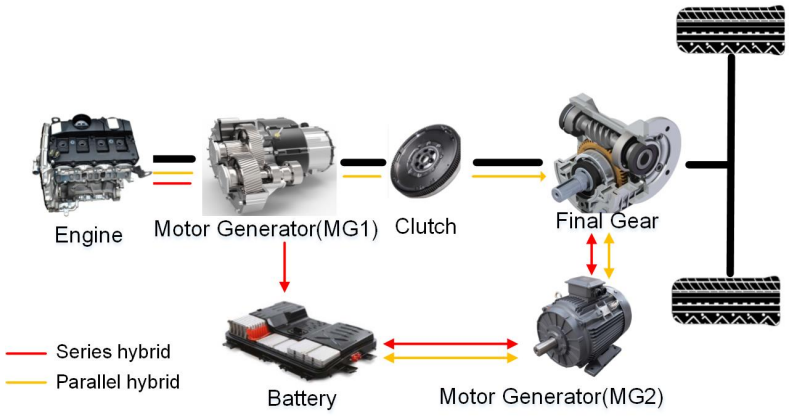


Fig. 5.1. Configuration of the multi-mode PHEV powertrain

5.1.1 The energy flow model

The energy flow of the vehicle is modeled based on longitudinal vehicle dynamics, and the force demand, $F_{dem}(t)$, the power demand, $P_{dem}(t)$, and the torque demand, $T_{dem}(t)$, can be calculated by

$$F_{dem}(t) = mgf\cos a + \frac{1}{2}\rho A_f C_d v^2(t) + mg\sin a + ma \tag{5.1}$$

$$P_{dem}(t) = F_{dem}(t) \cdot v \tag{5.2}$$

$$T_{dem}(t) = F_{dem}(t) \cdot R \tag{5.3}$$

where m is the vehicle mass, a is the vehicle acceleration, g is the gravity acceleration; f is the rolling resistance coefficient; ρ is the air density; $A_f$is the front area of the vehicle; C_d is

the air resistance coefficient; v is the longitudinal velocity; R is the wheel radius; a is the road slope, and the road slope should not be considered.

In the series mode, the energy flow is described as:

$$\left.\begin{aligned} P_{dem}(t) &= P_{mot2}(t) \\ T_{dem}(t) &= i_2 \cdot T_{mot2}(t) \\ P_{eng}(t) &= \frac{n_{eng}(t) \cdot T_{eng}(t)}{9550} \\ P_{eng}(t) &= P_{mot1}(t) \\ n_{eng}(t) &= n_{mot1}(t) \end{aligned}\right\} \tag{5.4}$$

In the parallel mode, the energy flow is described as:

$$\left.\begin{aligned} P_{dem}(t) &= P_{mot2}(t) + \left(P_{eng}(t) - P_{mot1}(t)\right) \\ T_{dem}(t) &= i_1 \cdot \left(T_{eng}(t) - T_{mot1}(t)\right) + i_2 \cdot T_{mot2}(t) \\ P_{eng}(t) &= \frac{n_{eng}(t) \cdot T_{eng}(t)}{9550} \\ n_{eng}(t) &= n_{mot1}(t) \end{aligned}\right\} \tag{5.5}$$

where, $P_{mot1,2}(t)$ and $P_{eng}(t)$ are the electric power of MG1 and MG2, respectively; $n_{mot1}(t)$,$n_{mot2}(t)$, and $n_{eng}(t)$ are the rotational speed of MG1, MG2, and the engine, respectively; the MG1 and MG2 torque are separately $T_{mot1,2}(t)$ and $T_{eng}(t)$ is the engine torque; i_1 is the transmission ratio of the gearbox to MG1, i_2 is the final ratio of the gearbox from MG2 to the wheels.

The energy of a multi-mode PHEV is from the battery and the engine (fuel tank), and the total power loss mainly consists of the engine loss $Loss_{eng}(t)$, and battery loss $Loss_{batt}(t)$, which can be calculated by:

$$\left.\begin{aligned} P_{loss}(t) &= Loss_{eng}(t) + Loss_{batt}(t) \\ Loss_{eng}(t) &= \dot{m}_f(t) \cdot H_f - \frac{n_{eng}(t) \cdot T_{eng}(t)}{9550} \\ Loss_{batt}(t) &= R \cdot I_{batt}(t)^2 \end{aligned}\right\} \tag{5.6}$$

where P_{loss} is the total power loss, H_f is the heat value of fuel (H_f = 43.5 KJ/g), and R is the equivalent internal resistance in the battery model.

Engine model

The engine model is used to determine the fuel consumption rate $\dot{m}_f$ (g/s) based on a 2D look-up table, which is a function of the engine speed, $n_{eng}(t)$, and the engine torque $T_{eng}(t)$ as

$$\dot{m}_f(t) = f\left(n_{eng}(t), T_{eng}(t)\right) \tag{5.7}$$

Motor models

The power demands of MG1 and MG2 are modeled on the basis of two quasistatic energy efficiency maps η_1 and η_2, respectively, as follows:

$$P_{mot1}(t) = \frac{n_{mot1}(t) \cdot T_{mot1}(t)}{9550} \cdot \eta_1\left(n_{mot1}(t), T_{mot1}(t)\right) \tag{5.8}$$

$$P_{mot2}(t) = \begin{cases} \frac{n_{mot2}(t) \cdot T_{mot2}(t)}{9550} \cdot \eta_2\left(n_{mot2}(t), T_{mot2}(t)\right); T_{mot2}(t) \leq 0 \\ \frac{n_{mot2}(t) \cdot T_{mot2}(t)}{9550} \cdot \frac{1}{\eta_2\left(n_{mot2}(t), T_{mot2}(t)\right)}; T_{mot2}(t) > 0. \end{cases} \tag{5.9}$$

Battery model

The battery model is established based on an equivalent circuit as follows:

$$\left.\begin{aligned} P_{batt}(t) &= P_{mot2}(t) - P_{mot1}(t) \\ P_{batt}(t) &= U \cdot I_{batt}(t) - R \cdot I_{batt}^2(t) \\ I_{batt}(t) &= \frac{U - \sqrt{U^2 - 4 \cdot R \cdot P_{batt}(t)}}{2 \cdot R} \\ SoC(t) &= SoC(0) - \frac{\int_0^t I_{batt}(t)dt}{Q_{batt}} \end{aligned}\right\} \tag{5.10}$$

where $P_{batt}(t)$ is the output power of the battery pack during charging and discharging; $SoC(0)$ is the initial SoC value; $I_{batt}(t)$ is the current of the battery at time t; and Q_{batt} is the nominal battery capacity ($Q_{batt} = 54.3\ Ah$). Since this research focuses on the energy flow in different driving cycles (up to 1800 seconds), the battery temperature and aging dynamics are ignored. The open-circuit voltage (OCV) U ($U = 350\ V$) and the battery internal resistance R ($R = 0.15\ \Omega$) are both constant.

5.1.2 The multiple-input and multiple-output (MIMO) energy management controller

A MIMO controller, as shown in Fig. 5.2, is developed to manage the energy flow of the studied vehicle. By observing the battery SoC and the overall torque demand as the control inputs, the MIMO controller calculates the torque demands for MG1, MG2, and the engine, respectively, to provide sufficient general torque to drive the vehicle while maintaining the battery SoC.

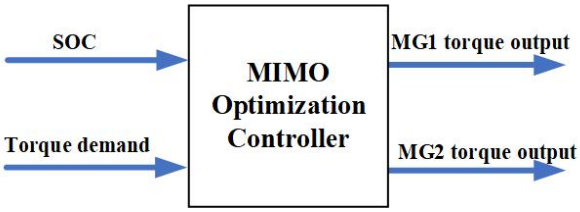


Fig. 5.2. MIMO control architecture

The core of the MIMO control is to resolve an optimization problem defined as follows:

$$
\begin{aligned}
&\textbf{Minimize } P_{loss}\left(u_{mot1}, u_{mot2}, T_{dem}\right) \text{ and } \Delta SoC\left(u_{mot1}, u_{mot2}, T_{dem}\right) \\
&Loss_{eng}(t) = m_f(t) \cdot H_f - \frac{n_{eng}(t) \cdot T_{eng}(t)}{9550} \\
&\textbf{s.t.} \quad Loss_{batt}(t) = R \cdot I_{batt}(t)^2 \\
&SoC(t) = SoC(0) - \frac{\int_0^t I_{batt}(t)\, dt}{Q_{batt}} \\
&SoC^- \leq SoC(t) \leq SoC^+ \\
&\text{and other physical constraints,}
\end{aligned}
\tag{5.11}
$$

where the overall power loss, P_{loss}, and the SoC difference, ΔSoC, are two objectives that need to be minimized; the MG1 torque command, $u_{mot1}(t)$and the MG2 torque command, $u_{mot2}(t)$, are the optimization variables to be determined during the real-time control. The optimization should be subjected to the vehicle energy flow models and other physical constraints of the powertrain system and subsystems.

5.2 | Hand-shaking multi-agent learning for MIMO control

5.2.1 Multi-agent learning with the DDPG algorithm

To resolve the optimization problem defined in Eq. 5.11, this paper proposes a hand-shaking multi-agent learning scheme, as shown in Fig. 5.3, in which two DDPG agents are involved to minimize the fuel consumption and battery usage simultaneously through the torque control of MG1 and MG2. Each learning agent has an actor-network and a critic-network. In each time interval, the agent starts learning with an observation of the state variables, and it uses the actor-network to generate a control action for the vehicle system followed by a reward evaluation from the system feedback. The critic-network is trained to update the actor-network using a policy gradient algorithm based on the recorded variables of state, action, and reward. It is worth noting that the two rewards are obtained through a hand-shaking manner module, where they are considered as two joint rewards. Further details regarding these rewards will be provided later.

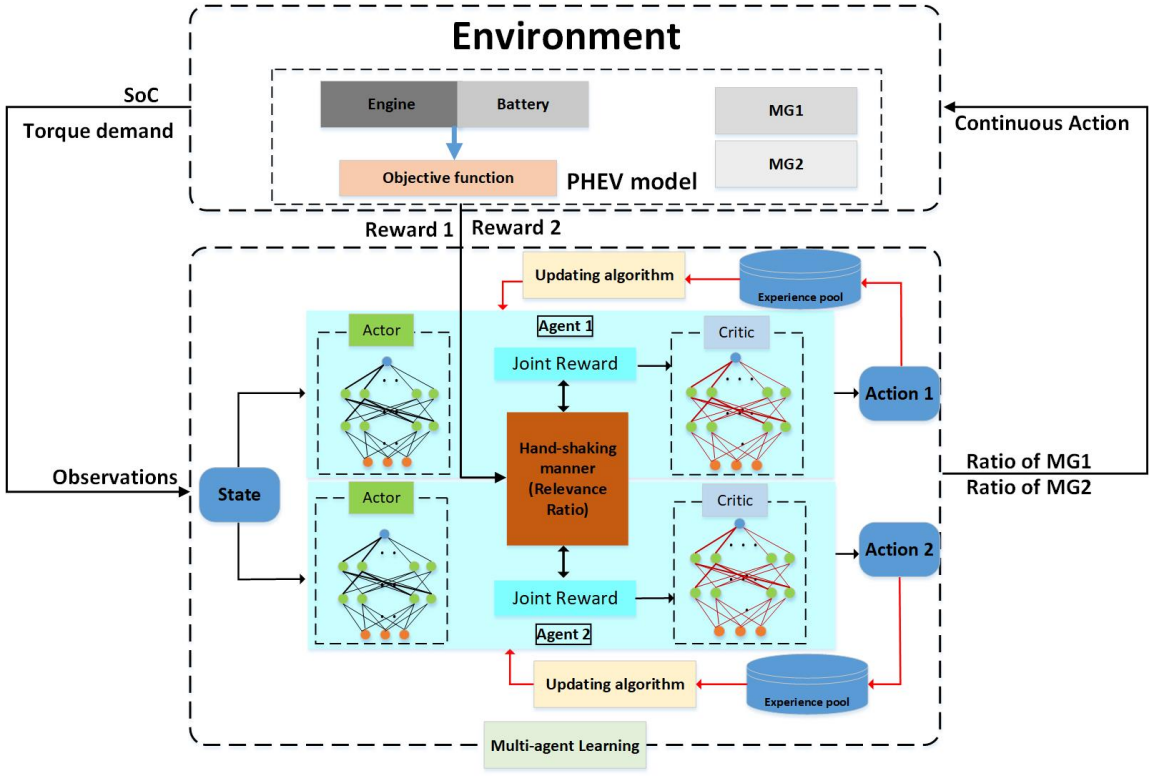


Fig. 5.3. DDPG-based EMS with multi-agent learning

Table 5.1
Comparison of single-agent and the proposed multi-agent systems

	Single-agent	The proposed multi-agent
No. of Agents	1	2
States	$T_{dem}(t), SoC(t)$	$T_{dem}(t), SoC(t)$
Action(s)	$u_{mot1}(t)$	$u_{mot1}(t), u_{mot2}(t)$
Reward	Weighted sum	Two unique functions for different preferences

The differences between the multi-agent system and the conventional single-agent system (baseline) are summarized in Table 5.1. Details of the main components of both learning systems are described as follows. The main difference between the single-agent system and the multi-agent system is the number of learning agents, i.e., the single-agent system only has one learning agent while the multi-agent system has more than two agents. The learning agent is a multi-input and single-output (MISO) control model that has the capability of self-learning for the development of control policy. It can be developed based on Q-learning, deep Q-learning, or other RL algorithms. In this study, the environment states and action variables are continuously varying. Therefore, the DDPG agent has been developed. The proposed multi-agent system includes two agents with different reward preferences for the two optimization objectives, and each agent generates the control signal for MG1 and MG2, respectively.

5.2.2 DDPG-based learning process

This paper implements the DDPG algorithm to optimize the control policy with an outer loop and an inner loop as shown in Fig. 5.4. In the inner loop shown with the red line, the agent interacts with the vehicle driving in real-world in with a sampling time of 1 s. Once the vehicle finished driving within a certain time defined in a driving cycle, the agent will update the policy in the outer loop as illustrated with the cyan line [152]. (See Fig. 5.4)

By defining a policy, $\pi(s)$ as

$$a = \pi(s) = \text{arg}maxQ(s, a) \tag{5.12}$$

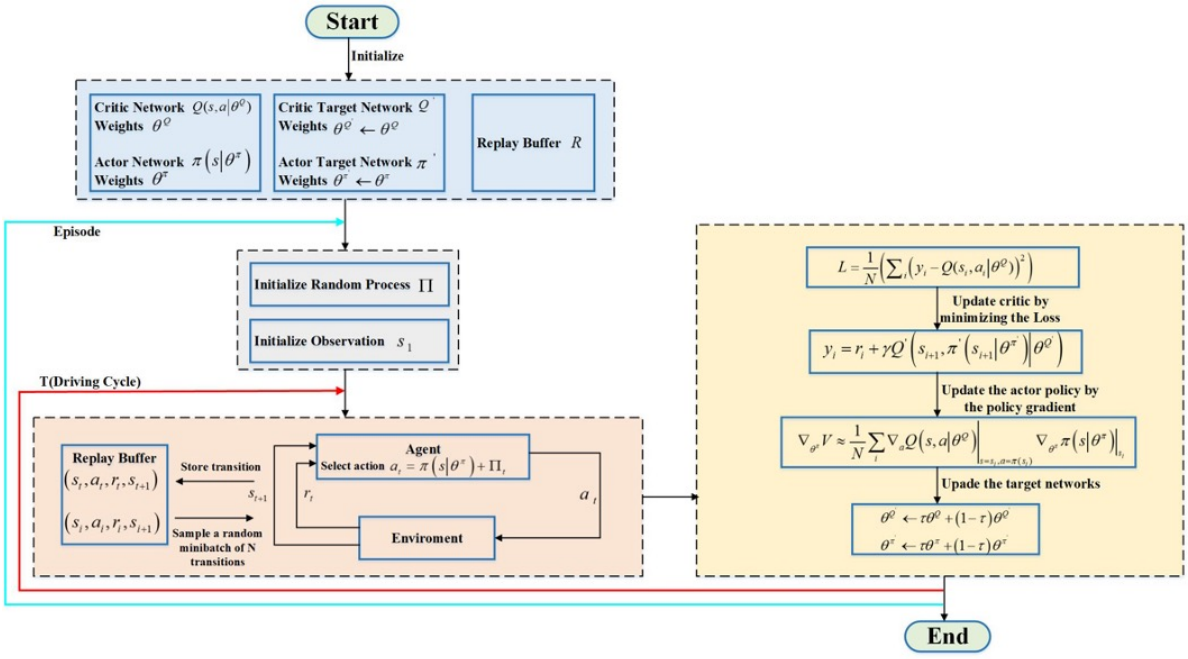


Fig. 5.4. Control policy optimization using the DDPG algorithm

where a is the control action; and s is the state variables. If the policy, $\pi : S \leftarrow A$, is deterministic with the actor-network, the policy update process can be formulated as statistical learning of the critic-network, $Q(s,a)$, by

$$Q^\pi(s_t, a_t) \leftarrow E_{rt, s_{t+1\sim E}}\left[r(s_t, a_t) + \gamma Q^\pi(s_{t+1}, \pi(s_{t+1}))\right] \tag{5.13}$$

where r is the reward variable, and γ is the discount factor. In the statistical learning process, an experience replay buffer with size $3R$ is used to store the transitions, which is a time-series batch of states, actions, and rewards in a form like $s_t, a_t, r_t, s_{t+1}, r_{t+1}, \cdots a_{t+R}, r_{t+R}$. Since DDPG is an off-policy algorithm, the buffer size can be very big to allow the system to learn from a large number of unrelated transitions. To refactor? computation load, this paper implements a minibatch method that randomly selects N samples ($N \ll R$) from the experience buffer to train the actor and critic networks at different times.

Once a new batch of data is collected from real-world driving, this paper implements the temporal difference (TD) method to estimate the hyperparameters of the critic network,θ^Q, by minimizing a loss function, defined by

$$L\left(\theta^{Q}\right)=E_{s_{t\sim\pi},a_{t\sim\pi},r_{t\sim E}}\left[\left(Q\left(s_t,\,a_t|\,\theta^{Q}\right)-r\left(s_t,a_t\right)+\gamma Q\left(s_{t+1},\pi\left(s_{t+1}\right)\mid\theta^{Q}\right)\right)^{2}\right] \tag{5.14}$$

Then, by implementing a Deterministic Policy Gradient (DPG) method, the actor-network can be updated by

$$\nabla_{\theta^{\pi}}V\approx\frac{1}{N}\sum_{i}\nabla_{a}Q\left(s,a\mid\theta^{Q}\right)\Big|_{s=s_i,a=\pi(s_i)}\nabla_{\theta^{\pi}}\pi\left(s\mid\theta^{\pi}\right)\Big|_{s=s_t} \tag{5.15}$$

Since the critic network $Q(s,a\mid\theta^{Q})$ is updated and used to estimate the target value, the Q update is prone to being unstable in many environments. Target networks, $Q^{'}(s,a\mid\theta^{Q'})$ and $\pi'(s\mid\theta^{\pi'})$, are employed for the actor and critic networks respectively to calculate the target values to provides momentum to the learning process by introducing a factor $\tau\ll 1$ to weights the parameters by $\theta'\leftarrow\tau\theta+(1-\tau)\theta'$.

Exploration is one of the most challenging problems of the learning process in continuous action domains. An exploration policy π' by combining the actor policy with a noise process Π, which is described as:

$$\pi^{'}\left(s_t\right)=\pi^{'}\left(s_t|\theta_t^{\pi}\right)+\Pi \tag{5.16}$$

An Ornstein-Uhlenbeck (OU) process [290] is chosen as a noise process:

$$da_t=-\beta a_t\,dt+\sigma\,dW_t \tag{5.17}$$

where W_t is a Wiener process with normally distributed increments, the decay rate $\beta>0$ (how "strongly" the system reacts to perturbations), and the variation $\sigma>0$ of the noise should be set and tuned, as shown in Fig. 5.5, since the OU process is time-series related and can be used to generate temporally correlated exploration in the action selection process of the previous step and the next step of RL to improve the exploration efficiency of control systems.

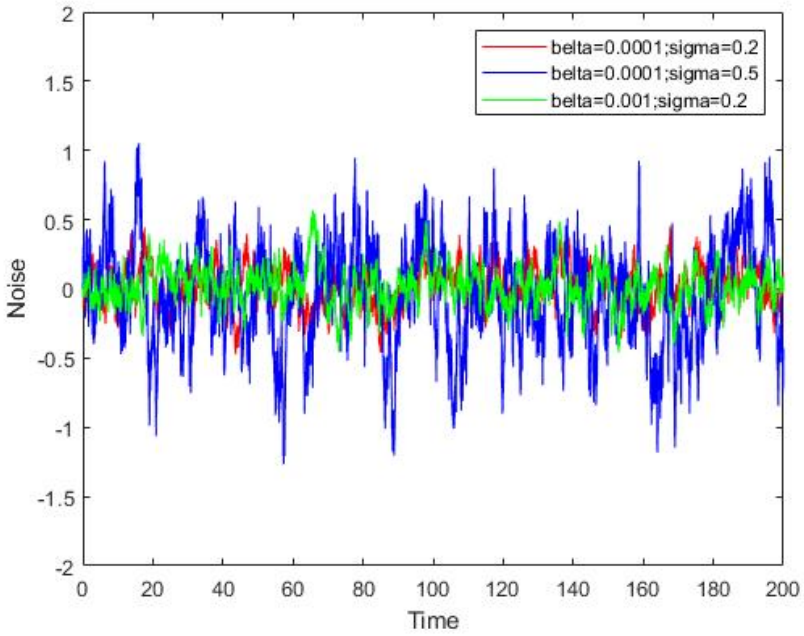


Fig. 5.5. OU noise process under different parameters.

5.2.3 States and actions

In this study, both the single-agent system and the multi-agent system monitor vehicle torque demands and battery SoC values as the state variables in a two-dimensional vector space, $s(t)$, as

$$s(t) = \left[T_{dem}(t), SoC(t)\right] \tag{5.18}$$

where $T_{dem}(t)$ is the power demand of the vehicle and $SoC(t)$ is the battery SOC.

As shown in Table 5.1, the single agent system can only output a single control action, $a_s(t)$, the control command of MG2 $u_{mot2}(t)$ keep the constant, and this study uses the DDPG algorithm to compute the control command of MG1, $u_{mot1}(t)$, based on a deterministic policy μ_{sa} :

$$a_s(t) = u_{mot1}(t) = \pi_{sa}(s(t) \mid \theta^{\pi_{sa}}) \tag{5.19}$$

where $\theta^{\pi_{sa}}$ is a hyper-parameter matrix representing the control policy that is updated over time. And the control commands of MG1 and MG2, i.e., $u_{mot1}(t)$ and $u_{mot2}(t)$, were dynamically

optimized through multi-agent learning. Based on $u_{mot1}(t)$ and $u_{mot2}(t)$, the control command $u_{eng}(t)$ of the engine can be calculated by

$$\begin{aligned} T_{eng} &= u_{mot1}(t) \cdot T_{mot1_max} + T_{GB} \\ T_{dem} &= u_{mot2}(t) \cdot T_{mot2_max} + T_{eng} \\ u_{eng}(t) &= \frac{u_{mot1}(t) \cdot T_{mot1_max} + T_{GB}}{T_{eng_max}} \end{aligned} \tag{5.20}$$

where T_{mot1_max} is the maximum torque of MG1; T_{eng_max} is the maximum torque that can be supplied by the engine, and T_{dem} is the torque demand for driving and braking the vehicle; T_{GB} is the torque of the output shaft provided by the engine when MG2 output torque cannot meet the requirement of the total torque output.

For the proposed multi-agent system that has two DDPG agents, the actions as the output of the multi-agent system can be expressed:

$$\left.\begin{aligned} a_m(t) &= \left[u_{mot1}(t), u_{mot2}(t)\right] \\ u_{mot1}(t) &= \pi_{ma1}(s(t) \mid \theta^{\pi_{ma1}}) \\ u_{mot2}(t) &= \pi_{ma2}(s(t) \mid \theta^{\pi_{ma2}}) \end{aligned}\right\} \tag{5.21}$$

where $u_{mot1}(t)$ is the output of the first DDPG agent for the control of MG1, and $u_{mot2}(t)$ is the output of the second DDPG agent for MG2. The engine control command can also be calculated using Eq. 5.20. Both $a_s(t)$ and $a_m(t)$ are calculated following a rolling process of exploration and exploitation [291].

5.2.4 Reward functions and hand-shaking design

The single-agent system implements a weighted sum method to incorporate the optimization objectives by

$$r_s(t) = -\alpha P_{loss}(t) - \beta \mid SoC_{ref} - SoC(t) \mid \tag{5.22}$$

where α is a scaling factor; SoC_{ref} is the target battery SoC value to be maintained during the driving; and β is a conditional weight factor, which yields:

$$\beta = \begin{cases} 0, & \text{if } \mathrm{SoC}(t) \geq \mathrm{SoC}_{\mathrm{ref}} \\ 2, & \text{if } \mathrm{SoC}(t) < \mathrm{SoC}_{\mathrm{ref}} \end{cases} \tag{5.23}$$

The conditional weight factor, β, will allow the DDPG agent to have a higher priority in minimizing fuel consumption when the SoC level is high [279].

A hand-shaking strategy is developed for the multi-agent system by introducing a relevance ratio R_{rel} that facilitates collaboration among the DDPG agents in the MADRL system. It ensures that the agents synchronize their learning processes and maintain stability during the training process. The relevance ratio aims to allow the agents to exchange information effectively through joint rewards for optimal control decisions. It incorporates global reward (r_{global}), and local rewards ($r_{local,1}$ and $r_{local,2}$) by

$$\begin{cases} r_{m1} = R_{rel} * r_{global} + r_{local,1} \\ r_{m2} = R_{rel} * r_{global} + r_{local,2} \end{cases} \tag{5.24}$$

where r_{m1} and r_{m2} are the joint rewards for the first DDPG agent and the second DDPG agent, respectively. Since minimizing the power loss is the main optimization objective, the power loss value is the element for the global reward function,

$$r_{\mathrm{global}} = -P_{\mathrm{loss}}(t) \tag{5.25}$$

Two local reward functions are designed to balance the usages of the ICE engine and the battery with two DDPG agents, $r_{local,1}$ and $r_{local,2}$, are allocated for the first DDPG agent and the second DDPG agent, respectively, and they can be calculated by:

$$\begin{cases} r_{local,1} = -\beta \mid SoC_{ref} - SoC(t) \mid \\ r_{local,2} = -\alpha \, Loss_{eng}(t) \end{cases} \tag{5.26}$$

where α is the scaling factor, and β is the weighting factor. In this research, α and β in the multi-agent system are set the same as in the single-agent system.

5.3 | Results and Discussion

A software-in-the-loop (SiL) testing platform is built for testing and validation on a workstation with an i7-7600U CPU and 64GB RAM. The models including the vehicle plant model and MIMO control model for the SiL test are developed using MATLAB/Simulink version 2022a, where the solver was ODE1 and the sample time is 1 s. The impacts of the learning agent design on MIMO control performance are firstly studied with sensitivity ranked. A parametric study on the relevance ratio, R_{rel}, is conducted to attain the best handshaking strategy for multi-agent control optimization. The performance of the single-agent system and multi-agent system are compared on a multi-mode PHEV driving under a training cycle and two testing cycles. The training driving cycle is built with elements generated from four standard driving cycles, including Artemis Rural, RTS95, UDDS, and WLTP, as illustrated in Fig. 5.6. It has four phases, where Phase 1 represents the low-speed region of the Artemis Rural cycle; Phase 2 involves the maximum acceleration in the RTS95 cycle; Phase 3 represents the medium-speed in UDDS Driving Cycle; and Phase 4 involves the high-speed region in the WLTP Cycle. In each learning episode, the four phases will be reorganized randomly to provide the learning noise for the robustness evaluation of the learning.

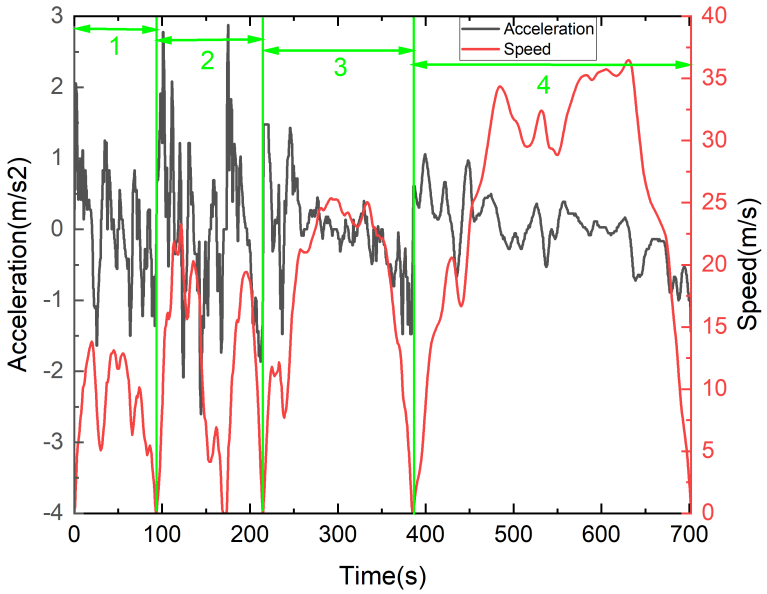


Fig. 5.6. A profile of the learning cycle.

5.3.1 Impacts of learning agent design on MIMO control performance

In this Section, the impacts of design parameters including the number of network layers, learning rate values, and different policy noise levels on the MIMO control performance are analyzed. The analysis is conducted based on the studied vehicle driving under the training cycle with an initial SoC value of 0.28. The main hyperparameters, which are vital to the learning dynamics and performances, are listed shown in Table 5.2.

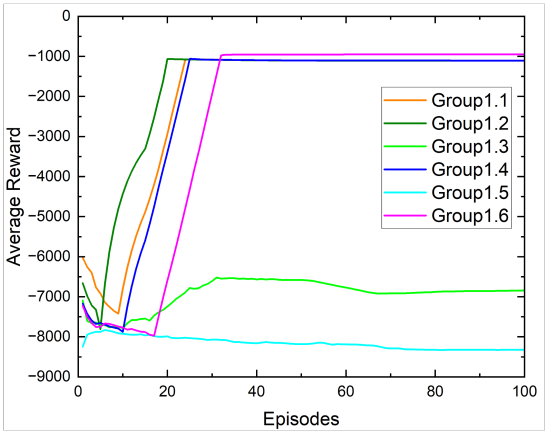
Table 5.2
Main hyperparameters and their values.

Hyperparameter	Values
Discount factor	0.99
Batch size	64
Experience buffer	1×10^5
Regulation factor for both networks	1×10^{-4}
Policy noise with a decay rate	0.2, 0.5
Policy noise with a variation	$1 \times 10^{-3}, 1 \times 10^{-4}$
Optimizer	Adam
Critic networks layers	2–7
Actor networks layers	3
Learning rate	$1 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-5}$

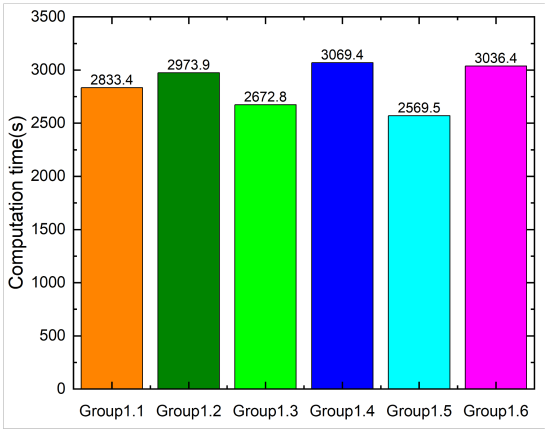
Networks layers

Since the critic network is to obtain an accurate estimate of the Q function value for the evolution of the actor-network, this paper first focuses on the design of the critic network. We investigate the critic networks with different numbers of layers from 2 to 7 with an interval of 1 set for Group 1.1 to Group 1.6 while maintaining the number of actor networks as a constant of 3. The tests are conducted with six individual groups to testify how the number of layers can affect MIMO control performance. The control performance including the average reward, the computer time, and the fuel economy are compared in Fig. 5.7.

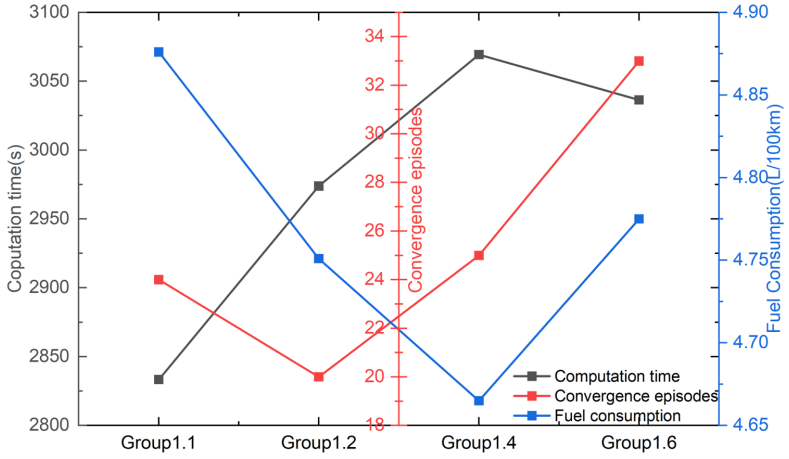
Theoretically, the number of network layers is related to the capability of representing nonlinear models and affects the complexity of the trained objects. This is why the learning performance tends to be improved while the computing time increases. From the results shown in 5.7 (a) and (b), Group 1.3 and Group 1.5 perform much worse than others although their computation



(a) The average rewards



(b) Computation time



(c) Comparison with effective groups

Fig. 5.7. Impact of critic network layers on the control performance.

Table 5.3
Comparison of SoC and Fuel consumption

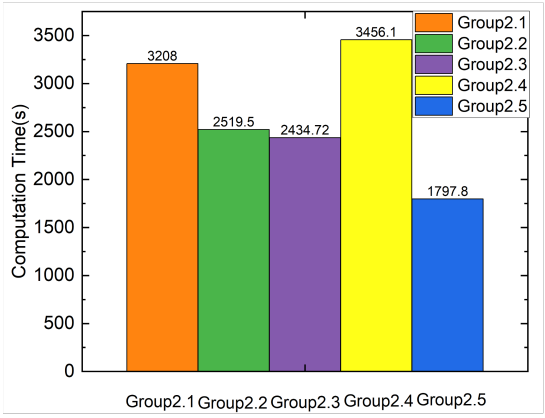
Group	2.1	2.2	2.3	2.4	2.5
Initial SoC	0.28	0.28	0.28	0.28	0.28
End SoC	0.311	0.294	0.303	0.316	—
Fuel Consumption(L/100km)	4.620	4.547	4.589	4.726	—

time is lower than others. Groups 1.1, 1.2, 1.4, and 1.6 all received acceptable performance during the training process with different computation efforts. Group 1.6 increases performance less with more costs of time to achieve stability, the possible reason would be the overfitting of the critic network due to its deep architecture. According to a comprehensive comparison as illustrated in Fig. 5.7 (c), this paper suggests the critic network with parameters in Group 1.2 is the best concerning three aspects including convergence episodes, computation time, and fuel consumption.

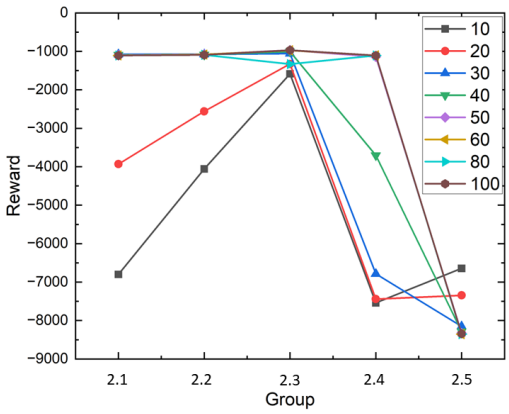
Learning rate

Five groups of parameters are designed to compare the impact of the learning rates: Group 2.1 is that the learning rate of actor and critic are both 1×10^{-4}, Group 2.2 is that the learning rate of actor and critic are both 1×10^{-3}, Group 2.3 is the learning rate of actor 1×10^{-4} and critic 1×10^{-3}, Group 2.4 is the learning rate of actor 1×10^{-3} and critic 1×10^{-4}, Group 2.5 is that the learning rate of actor and critic are both 1×10^{-5}. The network architecture of Group 1.2 is adopted to validate the influence of the learning rate. Then, by comparing five groups of learning rates of the actor and the critic networks under the learning driving cycle, the average reward, the computer time, the SoC of the battery, and the fuel consumption (L/100 km) are obtained in Fig. 5.8 and Table 5.3.

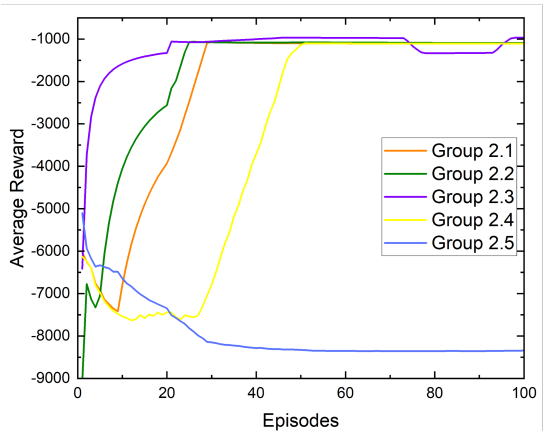
From Fig. 5.8 (a) and (b), it is apparent the learning rate is crucial for MIMO control optimization and produces a distinguished impact on learning through the update of both networks. Theatrically, the learning agent with a higher learning rate achieves better learning performance, this is the reason why Group 2.2 outperforms Group 2.1 and 2.5. According to the result in Group 2.5, if the learning rate is too small, the learning process would be unstable. From the comparison of Groups 2.3 and 2.4, improving the learning rate of the critic network is shown more effective in developing the MIMO control performances. By investigating the



(a) The average rewards



(b) Computation time



(c) Reward change curve under different episodes with 5 groups

Fig. 5.8. The results of different learning rates

Table 5.4
Policy noise combination

Group	3.1	3.2	3.3
β	1×10^{-4}	1×10^{-4}	1×10^{-3}
σ	0.2	0.5	0.2

Table 5.5
Comparison of SOC and Fuel consumption

Group	3.1	3.2	3.3
Initial SoC	0.28	0.28	0.28
End SoC	0.294	0.313	0.309
Fuel Consumption(L/100km)	4.547	4.689	4.625

learning performance at each episode from 0 to 100 with an interval of 10 episodes in Fig. 5.8 (c), the MIMO controller with the settings in Group 2.3 is the best. It requires the least computation time and is the fastest to reach the coverage point, although its fuel consumption is 0.9% higher than Group 2.2.

Policy noise

As defined in Eq. 5.17, the OU process, dominating the ratio of exploration during reinforcement learning, is determined by the weighting value of the Wiener process, β, and the variation of the noise, σ. By implementing the network setting in Group 1.2 and Group 2.3, three groups of test with different values of β and σ, as illustrated in Table 5.4 are conducted.

The three groups of policy noise combinations are implemented respectively in the studied vehicle driving under the learning cycle, and the average reward, battery SOC, and fuel economy (L/100 km) are compared in Fig. 5.9 and Table 5.5. The results indicate that the controller with the setting defined in Group 3.1 is the best since it requires less time to converge to the best system performance which leads to the best fuel economy and battery charge sustaining. The results also show that the weighting value of the Wiener process, β, impacts the learning speed more significantly compared to the variation of the OU process, σ.

Importance analysis

By introducing a sensitivity level, R_s, which is defined as in Eq. 5.27:

$$R_s = \frac{\left|y_{best} - y_{worst}\right| \cdot \| x_{best} \|_2}{y_{best} \cdot \| x_{best} - x_{worst} \|_2} \times 100\% \tag{5.27}$$

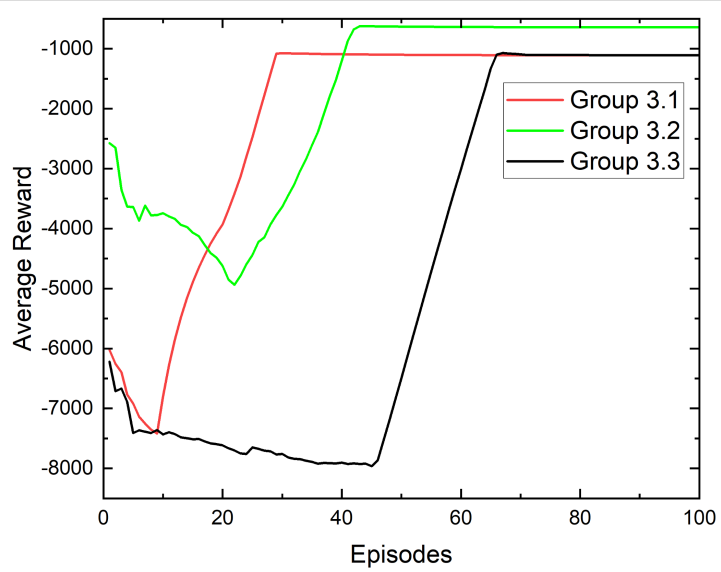


Fig. 5.9. The average rewards of different policy noise

where y_{best} and y_{worst} are the best-case values and the worse-case values of the indicators including computation time (CT), the number of episodes for convergence (CE), and fuel economy (FE); x_{best} and x_{worst} are projected values of the learning agent settings on a one-dimension vector for the best case and the worst case, respectively. The principal component analysis (PCA) method is utilized to project multi-dimension network setting matrices into the one-dimensional vector. The best-case values, the worst-case values, and the sensitivity levels of each group of tests are compared in Table 5.6. The results highlight that the learning rate is the most important influencing factor on learning performance in terms of CT, CE, and FE. Policy noise is the second contributor to the number of episodes for convergence (CE) while the number of network layers is the second contributor to fuel economy (FE). The policy noise has no contribution to computation time (CT).

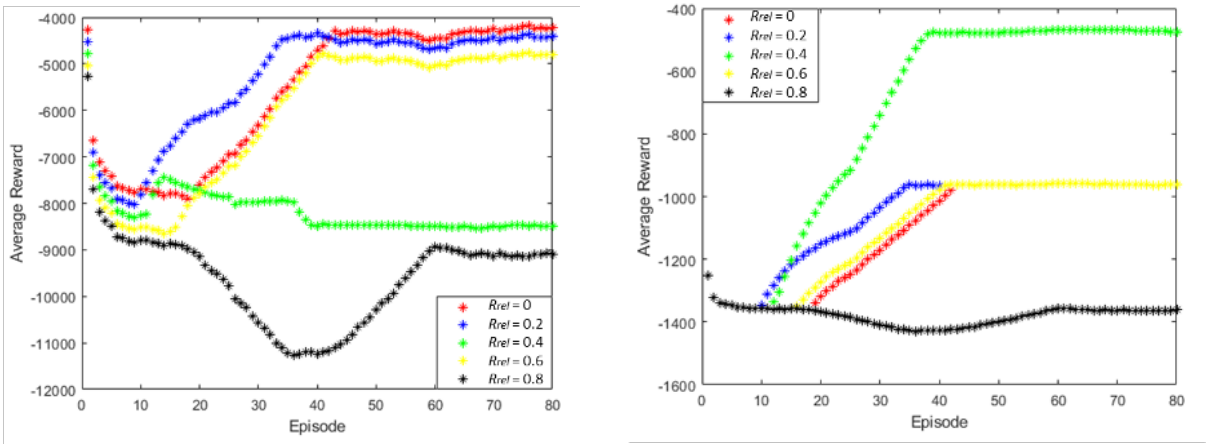
5.3.2 Hand-shaking of the multi-agent systems with different R_{rel} value

Following the study in Section 5.3.1, the two DDPG agents in the hand-shaking learning system share the same network parameters. To determine the unified setting for the relevance ratio in the multi-agent system, this paper investigated the performances of a PHEV controlled by the MADRL systems with the R_{rel} value changing from 0 to 0.8 with 0.2 intervals. The simulation experiments were conducted under the learning cycles to investigate how the relevance ratio

Table 5.6
Comparison of importance levels

		Computation time (CT)	Convergence episodes (CE)	Fuel economy (FE)
Networks layers	The worst	3069.40	33	4.876
	The best	2833.40	20	4.665
	R_s (%)	**7.90**	**59**	**1.430**
Learning rate	The worst	3456.10	50	4.726
	The best	2434.72	20	4.547
	R_s (%)	**33.12**	**118**	**6.186**
Policy noise	The worst	-	68	4.689
	The best	-	30	4.547
	R_s (%)	-	**90**	**1.397**

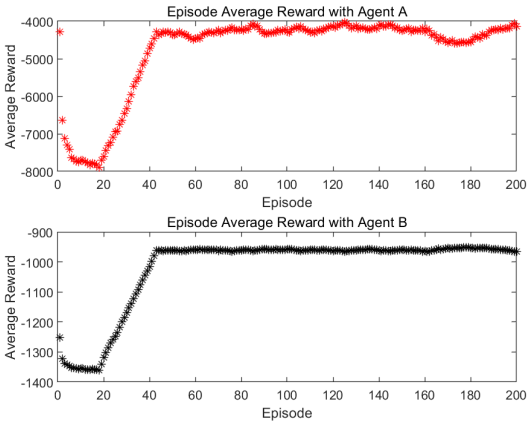
affects the multi-agent collaboration and the contributions of the learning agents to the global goal. And the learning curves of the two DDPG agents illustrating the evolution of agent rewards in 80 episodes are compared in Fig. 5.10.



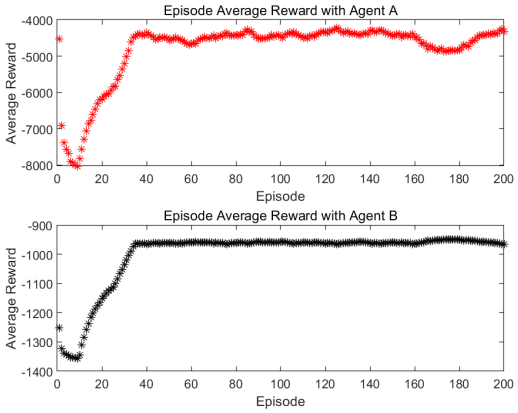
(a) Agent A **(b)** Agent B

Fig. 5.10. Learning performance of different ratios (SoC initial value is 0.28)

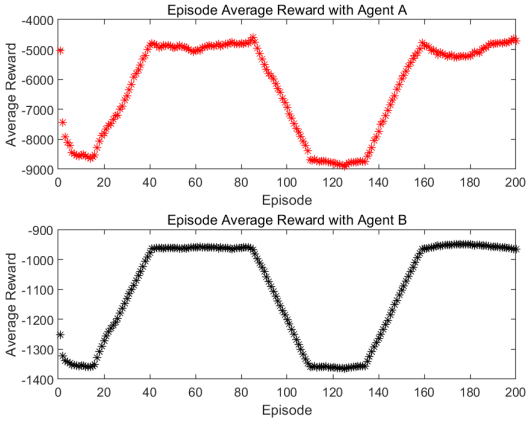
The two agents are growing in opposite directions in the system with a R_{rel} value of 0.4. In the system with a R_{rel} value of 0.8, although both agents have the same tendency, the reward values of the multi-agent system decrease over time. Most of the studied cases (R_{rel} = 0, 0.2, and 0.6) demonstrate the way of handshaking in the multi-agent system, in which both agents have the same tendency to grow with their reward values increase over time. Then, when R_{rel} values of 0, 0.2, and 0.6, two agents perform well and achieve fast convergence at about 40 episodes and have the same tendency to reach the common goal. Therefore, the learning performance with more training time has been investigated in this paper.



(a) $R_{rel} = 0$



(b) $R_{rel} = 0.2$



(c) $R_{rel} = 0.6$

Fig. 5.11. Learning performance of $a) R_{rel} = 0; b) R_{rel} = 0.2; c) R_{rel} = 0.6$

Table 5.7
Average values and standard deviation.

R_{rel}	Agent	Average reward value (All 200 episodes)	Standard deviation (400-200 episodes)
0	Agent A	-4782.1	48.064
	Agent B	-1018.8	48.064
0.2	Agent A	-4803.3	46.621
	Agent B	-999.1	46.621

The learning performance of the two agents with relevance ratio of 0, 0.2, and 0.6 are compared in Fig. 5.11. The result suggested that when the two agents have a relevant high level of relevance, i.e., R_{rel} = 0.6, although they follow the same trend during the training, both tend to be unstable and thus cannot converge at the steady point. By comparing the average value and standard deviation of the rewards obtained during the training with R_{rel} values of 0 and 0.2 in Table 5.7, the study suggested that the multi-agent system with a R_{rel} value of 0.2 is the best since it can achieve the highest average reward with less level of variation.

5.3.3 Performance comparison with different methods

To demonstrate the advantage of the proposed multi-agent system, a comparison study was conducted by using the rule-based method, the equivalent consumption minimization strategy (ECMS), and a single-agent system as baselines. By testing the PHEV under three driving cycles, the vehicle performances including the battery SoC at the end of the driving cycle, SoC sustaining error, fuel economy, and fuel saving rate over the rule-based method were obtained and compared in Table 5.8. The configuration of the single-agent system is summarized in Section 5.2, and the DDPG algorithm with the same setting as the multi-agent system is used for the online optimization of the single-agent system. By defining the battery SoC sustaining error, SOC_{error}, and fuel saving rate, $Saving$, as:

$$\begin{aligned} SoC_{error} &= \frac{|SoC_{End} - SoC_{Initial}|}{SoC_{Initial}} * 100\% \\ \text{Saving} &= \frac{|Fuel_{Multi-agent} - Fuel_{Single-agent}|}{Fuel_{Single-agent}} * 100\% \end{aligned} \tag{5.28}$$

where $SoC_{Initial}$ and SoC_{End} are the battery SoC value at the beginning and the end of a driving cycle, respectively; $Fuel_{Single-agent}$ and $Fuel_{Multi-agent}$ are the fuel consumption in L/100 km for the single-agent system and the multi-agent system, respectively. Vehicle

performances including fuel economy and battery SoC sustaining error are monitored under the learning cycle and two worldwide driving cycles, including UDDS and WLTC in Table 5.8a, 5.8b, and 5.8c, respectively.

Among the two traditional methods, the rule-based approach exhibits poorer performance in terms of both SoC sustaining and fuel economy. On the other hand, the ECMS method lacks stability for the testing environments with different initial SoC settings, limiting its generalization when compared to the single-agent system and the proposed multi-agent system. Notably, the ECMS methods demonstrated inconsistent performance in fuel economy and SoC sustaining error. For instance, when the initial SoC is set at 0.28 for the learning driving cycle, ECMS achieved very good fuel economy, however, the SoC sustaining error exceeded 5%, which is a performance threshold defined in legislation.

The results indicate that the multi-agent system can help maintain the battery SoC error within 5% when the initial SoC is relatively low (below 0.30). This means the PHEV controlled by the multi-agent system aligns with the requirement of most vehicle testing regulations around the world. The multi-agent system outperforms the single-agent in all the study cases in terms of using less fuel and battery SoC. This is because the multi-agent system generates two control variables to allow MG1 and MG2 to be controlled independently for their best performance. The PHEV controlled by the proposed multi-agent system achieved the best fuel economy in the learning driving cycle with all pre-set initial SoC values. Notably, significant fuel savings of up to 23.54% were achieved when the initial battery SoC is set at 0.25. Comparatively, in all driving cycles, the fuel economy performance is the most notable when the initial SoC is set at 0.25. Specifically, when the vehicle is driven under the WLTC cycle with an initial SoC of 0.25, the multi-agent system achieves fuel savings of 16.73% with a battery SoC sustaining error of only 1.60%, which far exceeded the industry standard (5%).

To address concerns about the fairness of energy efficiency comparisons under varying final SoC levels, Fig. 5.12 is presented to visualize the relationship between fuel consumption and final battery SoC across three driving cycles. It highlights the trade-off between energy usage and battery depletion for four energy management strategies. Multi-agent system (green) consistently achieves lower fuel consumption with moderate SoC depletion, indicating effective energy coordination. Single-agent system (orange) performs slightly less effectively, suggesting

Table 5.8
Learning performance of different methods. (– represents the baseline performance.)

(a) Learning driving cycle

Initial SoC	Method	End SoC	SOC Error	Fuel (L/100km)	Saving
0.25	Rule-based	0.292	16.72%	5.779	–
	ECMS	0.280	12.04%	5.625	2.67%
	Single-agent	0.272	8.80%	4.534	21.55%
	Multi-agent	**0.241**	**3.60%**	**4.419**	**23.54%**
0.28	Rule-based	0.306	9.11%	5.357	–
	ECMS	0.280	0.04%	4.598	14.17%
	Single-agent	0.305	8.93%	4.547	15.12%
	Multi-agent	**0.271**	**3.21%**	**4.450**	**16.93%**
0.3	Rule-based	0.312	4.10%	5.027	–
	ECMS	0.280	6.63%	3.926	21.89%
	Single-agent	0.328	9.33%	4.534	9.81%
	Multi-agent	**0.286**	**4.67%**	**4.418**	**12.11%**

(b) UDDS driving cycle

Initial SoC	Method	End SoC	SOC Error	Fuel (L/100km)	Saving
0.25	Rule-based	0.307	22.88%	5.368	–
	ECMS	0.298	19.36%	5.077	5.42%
	Single-agent	0.262	4.80%	4.430	17.47%
	Multi-agent	**0.261**	**4.40%**	**4.351**	**18.95%**
0.28	Rule-based	0.322	14.93%	4.916	–
	ECMS	0.298	6.57%	3.899	20.69%
	Single-agent	0.292	4.29%	4.430	9.89%
	Multi-agent	**0.290**	**3.57%**	**4.306**	**12.41%**
0.3	Rule-based	0.330	10.00%	4.573	–
	ECMS	0.298	0.53%	4.423	3.27%
	Single-agent	0.312	4.00%	4.431	3.10%
	Multi-agent	**0.310**	**3.33%**	**4.327**	**5.37%**

(c) WLTP driving cycle

Initial SoC	Method	End SoC	SOC Error	Fuel (L/100km)	Saving
0.25	Rule-based	0.315	26.08%	4.990	–
	ECMS	0.281	12.44%	4.518	9.45%
	Single-agent	0.261	4.40%	4.324	13.34%
	Multi-agent	**0.246**	**1.60%**	**4.155**	**16.73%**
0.28	Rule-based	0.321	14.50%	4.640	–
	ECMS	0.281	0.40%	4.205	9.38%
	Single-agent	0.293	4.64%	4.324	6.82%
	Multi-agent	**0.292**	**4.29%**	**4.148**	**10.61%**
0.3	Rule-based	0.324	7.90%	4.413	–
	ECMS	0.281	6.30%	3.501	20.67%
	Single-agent	0.314	4.67%	4.325	1.99%
	Multi-agent	**0.304**	**1.33%**	**4.162**	**5.68%**

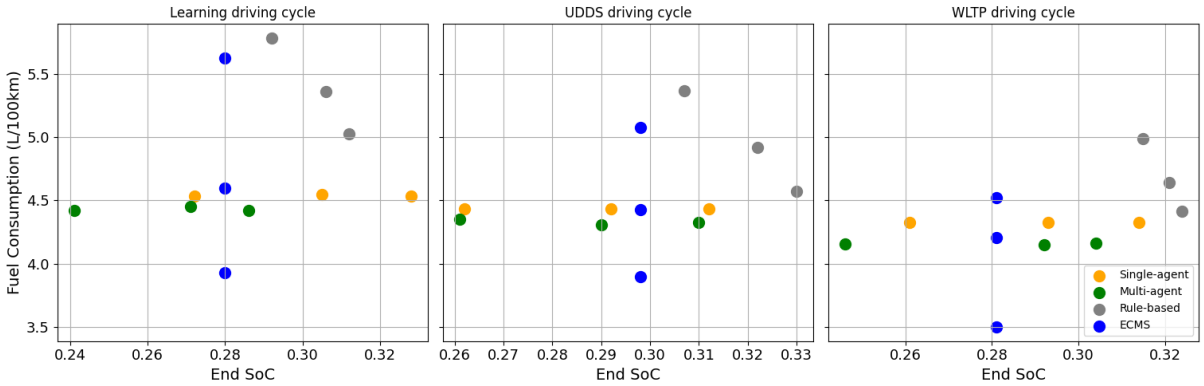


Fig. 5.12. Scatter plots of fuel consumption versus final SoC for four control strategies.

weaker mode coordination. ECMS (blue) shows strong performance in some cases (e.g., WLTP) but suffers from inconsistency, with some instances of high fuel usage or aggressive SoC depletion. Rule-based control (gray), while retaining higher SoC, results in the highest fuel consumption, indicating inefficient energy utilization. Overall, multi-agent RL demonstrates the most balanced performance, optimizing both fuel efficiency and battery use.

5.4 | Chapter summary

This chapter studied a new MIMO control method for the multi-mode PHEV based on MADRL. By introducing a relevance ratio, a hand-shaking strategy has been proposed to enable two learning agents to work collaboratively under the MADRL framework using the DDPG algorithm. Through sensitivity analysis, parametric study, and software-in-the-loop testing, the conclusions drawn from the investigation are as follows:

(1) The learning rate of the DDPG agents in the proposed MADRL-based EMS is the most significant influencing factor in determining the learning performance including computing time, the number of episodes for convergence, and fuel economy.

(2) Hand-shaking among the DDPG agents is achievable by tuning the relevance ratio. The optimal setting for the relevance ratio is 0.2 for control of the studied multi-mode PHEV. A smaller relevance ratio would lead to a low learning speed while a higher value will make the system unstable.

(3) The proposed MADRL method outperforms the baseline methods (single agent learning method, rule-based method, and ECMS) under all studied driving cycles in terms of mitigating

energy consumption and battery SoC sustaining error. Up to 23.54% of fuel can be saved with battery SoC error well-controlled within 5% compared to the baseline methods.

CHAPTER 6

Efficient Energy Management of Plug-in Hybrid Electric Vehicles through Ensemble with In-target Minimization Q-learning

This chapter builds upon previous work on complex multi-mode PHEVs and shifts the focus to a specified operating mode, addressing key challenges related to real-time implementation and control stability. While multi-mode PHEVs offer greater flexibility in energy management, their computational burden poses challenges for real-time optimization. This work builds on our group's established research, which was demonstrated in a prior study published in *IEEE Transactions on Intelligent Vehicles* [292], where I contributed as co-first author alongside a senior colleague two years ahead of me. Building on the limitations identified in that research, I further developed an efficient learning-based control strategy for the PHEV, extending the findings from the previous chapter by focusing on a specific operating mode to better demonstrate the effectiveness of RL and its practical applicability.

To further address the persistent issue of overestimation bias in Q-learning methods, we introduce a novel ensemble approach that integrates in-target minimization and an Update-to-Data (UTD) mechanism for end-to-end control optimization. This extended work, done by myself, is currently under review at *IEEE Transactions on Transportation Electrification*. The main contributions of this chapter are:

1) We propose an efficient ensemble Q-learning scheme with in-target minimization to mitigate Q-value overestimation in PHEV energy management. By applying a minimization operation across Q-functions, this method introduces a stable pessimistic bias that improves learning

consistency and significantly reduces overestimation by bias visualization and theoretical analysis.

2) We incorporate the update-to-data (UTD) mechanism into the learning process and identify an optimal update ratio that balances sample efficiency and computational cost. This enables faster convergence by extracting more value from each environmental interaction while avoiding overfitting.

3) We validate the proposed method through extensive SiL and HiL testings under standard and real-world driving cycles. By analyzing the impact of main driving pattern characteristics, such as speed variability and deceleration, we demonstrate the robustness, adaptability, and energy-saving effectiveness of the proposed strategy across diverse scenarios.

6.1 | The Hybrid Electric Vehicle System

The RL-based EMS for the PHEV system can be described in this section.

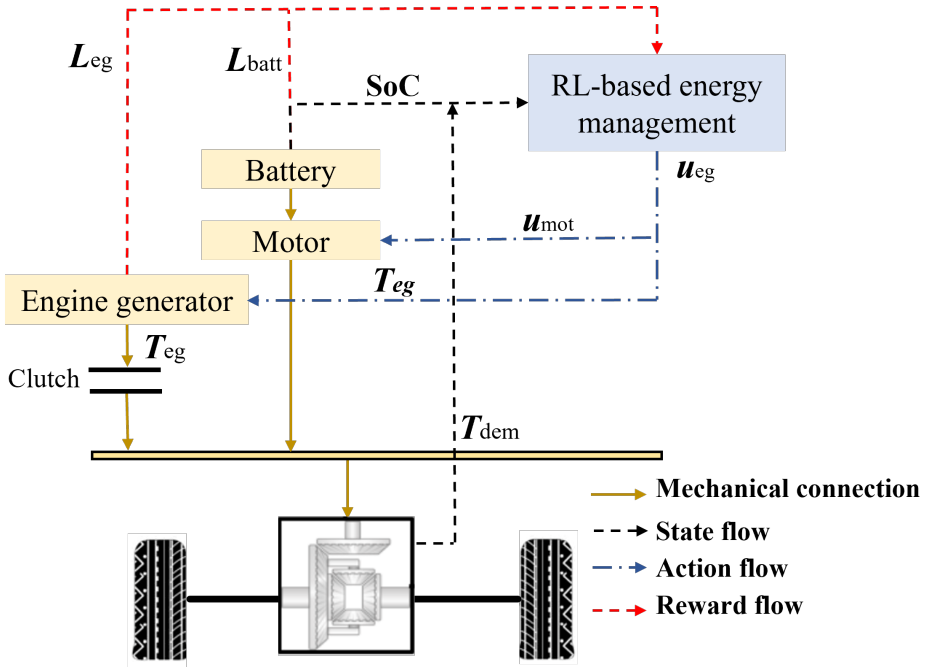


Fig. 6.1. The schematic of the studied PHEV

Table 6.1
Vehicle specifications of the studied PHEV.

Specifications	Value
Vehicle mass (m_{veh})	1700 kg
Tire rolling radius (r_{whl})	0.337 m
Rolling resistant coefficient (f_{roll})	0.011
Front Area (A)	2.5 m^2
Aerodynamic drag coefficient (C_d)	0.32
Engine displacement (D)	1 L
Heat value of fuel (Q_{lhv})	4.4×10^7 J/kg
Battery capacity (Q_{batt})	54.3 Ah
Battery nominal open-circuit-voltage (U_{OCV})	350 V
Differential ratio (i_{RD})	4.14
Transmission ratio of gearbox-engine (i_{eg})	0.95
Transmission ratio of gearbox-motor (i_{mot})	3.91

6.1.1 Vehicle modelling

Vehicle power flow

The vehicle has a plug-in hybrid electric powertrain, as illustrated in Fig. 6.1. The main power units of this powertrain include a 70 kW ICE, the motor with a nominal power of 135 kW, and a 20 kWh battery. The vehicle specifications are detailed in TABLE 6.1. The power flow of the studied vehicle is illustrated in Fig. 6.1, which obeys:

$$T_{\text{dem}} = T_{\text{eg}}(u_{\text{eg}}, n_{\text{eg}}) + T_{\text{mot}}(u_{\text{mot}}, n_{\text{mot}}) \tag{6.1}$$

where T_{eg} is the ICE torque, controlled by $u_{\text{eg}} \in [0, 1]$ at speed n_{eg}; T_{mot} is the motor torque, determined by battery SOC and motor speed n_{mot}. The demanded torque and speed are calculated based on vehicle dynamics [293]:

$$\begin{cases} T_{\text{dem}}(t) = \left(\dfrac{C_d A v_{\text{veh}}^2(t)}{21.15} + m_{\text{veh}} \dot{v}_{\text{veh}}(t) + m_{\text{veh}} g f_{\text{roll}} \right) \cdot r_{\text{whl}} \\ n_{\text{dem}}(t) = \dfrac{60 \cdot v_{\text{veh}}(t)}{2\pi r_{\text{whl}}} \cdot i_{\text{RD}} \cdot i_{\text{mot}} \end{cases} \tag{6.2}$$

where m_{veh} is the mass of the PHEV; v_{veh} and $\dot{v}_{\text{veh}}$ are the vehicle speed and acceleration, respectively; C_d is the drag coefficient; A_f is the PHEV's front area; g is the gravity; f_{roll} is the rolling resistance; r_{whl} is the radius of wheels; i_{RD} is the differential ratio; and i_{mot} is the

transmission ratio of the gearbox-motor. The mechanical power can be calculated as:

$$P_{\mathrm{dem}}(t) = T_{\mathrm{dem}}(t) \cdot n_{\mathrm{dem}}(t)/9550 \tag{6.3}$$

Engine-generator

The mass flow rate of the ICE $\dot{m}_f$, can be determined in each sampling time by

$$\dot{m}_{ef} = \mathbf{BSFC}(T_{\mathrm{eg}}, n_{\mathrm{eg}}) \tag{6.4}$$

where **BSFC** is the ICE fuel consumption map, which is calibrated ICE torque T_{eg} and speed n_{eg}.

Then, the engine power loss L_{eg} is calculated by

$$\begin{cases} L_{\mathrm{eg}}(t) = P_{\mathrm{ef}}(t) - P_{\mathrm{eg}}(t) \\ P_{\mathrm{ef}}(t) = \dot{m}_{\mathrm{ef}} \cdot H_f \\ P_{\mathrm{eg}}(t) = \dfrac{T_{\mathrm{eg}} \cdot n_{\mathrm{eg}}}{9550 \cdot \eta_{\mathrm{eg}}(T_{\mathrm{eg}}, n_{\mathrm{eg}})} \end{cases} \tag{6.5}$$

where H_f is the heat value of fuel ($H_f = 43.5KJ/g$).

Battery modelling

The battery model is established based on an equivalent circuit, as follows:

$$\begin{cases} \dfrac{P_{\mathrm{batt}}(u_{\mathrm{batt}}(t))}{I_{\mathrm{batt}}} = U_{\mathrm{OCV}} - I_{\mathrm{batt}}(t) \cdot R_{\mathrm{batt}}(\mathrm{SoC}) \\ \mathrm{SoC}(t) = \mathrm{SoC}_0 - \displaystyle\int_{t_0}^{t} \frac{I_{\mathrm{batt}}(t)}{Q_{\mathrm{batt}}}\, dt \\ L_{\mathrm{batt}}(t) = I_{\mathrm{batt}}^2(t) \cdot R_{\mathrm{batt}}(\mathrm{SoC}) \end{cases} \tag{6.6}$$

where $I_{\mathrm{batt}}(t)$ is the battery current at time step t; $R_{\mathrm{batt}}(\mathrm{SoC})$ and U_{OCV} are battery internal resistance and open circuitvoltage; $\mathrm{SoC}(t)$ is battery state-of-charge (SoC); Q_{batt} is the capacity of battery; L_{batt} is the battery power loss.

6.1.2 RL-based energy management

As illustrated in Fig. 6.1, the EMS for the hybrid powertrain is developed utilizing RL.

State and action space

The state space consists of two key variables:

$$s(t) = [T_{\text{dem}}(t), \text{SoC}(t)] \tag{6.7}$$

where $s(t) \in S$ is the state observed at time t; S is a two-dimensional finite set that is defined based on vehicle working conditions.

The control actions are determined using an ϵ-greedy strategy [174]:

$$a(t) = \begin{cases} \Pi(s(t)) & \text{if } \epsilon \geq \theta \\ \text{Rnd}(0,1) & \text{if } \epsilon < \theta \end{cases} \tag{6.8}$$

where ϵ is a random number between 0 and 1 generated in every sample time; θ is the output of a decay function; $\text{Rnd}(0,1)$ is a random number between 0 and 1; Π is a control policy, which is based on a knowledge base that maps the merit values Q, to different state-action pairs (s, a).

The motor control signal is then computed as:

$$u_{\text{mot}} = \frac{T_{\text{dem}} - u_{\text{eg}} \cdot T_{\text{eg}}^{\max} \cdot i_{\text{eg}}}{T_{\text{mot}}^{\max}} \tag{6.9}$$

where T_{eg}^{*} and T_{mot}^{*} are the maximum torque that can be generated by the engine generator and motor, respectively; and i_{eg} is the transmission ratio of the gearbox-engine.

Reward

$r(t)$ is the reward received at time t, which is defined by

$$r(t) = \begin{cases} L_{\text{loss}}(t) = L_{\text{eg}}(t) + L_{\text{batt}}(t) & \\ -L_{\text{loss}}(t) & \text{if } \mathrm{SoC}(t) \geq \mathrm{SoC}_{\text{ref}} \\ -L_{\text{loss}}(t) & \\ -\mu \left| \mathrm{SoC}_{\text{ref}} - \mathrm{SoC}(t) \right| & \text{if } \mathrm{SoC}(t) < \mathrm{SoC}_{\text{ref}} \end{cases} \tag{6.10}$$

where SoC_{ref} is the reference battery SoC to be maintained (for the best performance and health of the battery $\mathrm{SOC}_{\text{ref}} = 28\%$). μ is a scale factor to balance the consideration of battery SoC level and power loss; P_{loss} is the total power loss of the engine-generator, L_{eng}, and battery, L_{batt}.

6.2 | Ensemble Q-learning with in-target minimization and UTD ratio

6.2.1 Single Q-learning

Q-learning and its variants are known to suffer from overestimation bias in their Q-value estimates. The standard Q-learning update rule is given by:

$$\begin{aligned} Q(s(t), a(t)) \leftarrow Q(s(t), a(t)) + \alpha \Big(& r(t) \\ & + \gamma \cdot \max Q(s(t+1), \cdot) - Q(s(t), a(t)) \Big) \end{aligned} \tag{6.11}$$

where $Q(s, a)$ represents the Q-value at each step, α is the learning rate, γ is the discount factor, and $\max Q(s(t+1), \cdot)$ denotes the maximum estimated Q-value for the next state $s(t+1)$. The overestimation bias arises from two key factors:

1. The maximization operation in Q-learning inherently introduces positive bias by always selecting the highest Q-value estimate regardless of estimation noise.

2. Q-value estimates from noisy samples tend to accumulate positive bias through repeated maximum selection during updates.

6.2.2 Ensemble Q-learning design

Decay functions

Three decay functions were examined in our previous work [292]: the conventional exponential-based decay (EXD), step-based decay (SBD), and reciprocal function-based decay (RBD). Based on prior research findings, SBD and EXD have shown superior performance in regulating exploration rates and have therefore been chosen to manage the decay functions of the two agents. This chapter applies EXD to Agent A and SBD to Agent B, where the decay functions generate threshold values ($\theta_A(i)$ and $\theta_B(i)$) in each learning episode i. These values are compared with the random number ϵ to determine action selection under the ϵ-greedy strategy, as formulated below:

$$a_A(t) = \begin{cases} \Pi_A(s(t)) & \text{if } \epsilon \geq \theta_A(i) \\ \mathrm{Rnd}(0,1) & \text{if } \epsilon < \theta_A(i) \end{cases} \tag{6.12}$$

$$a_B(t) = \begin{cases} \Pi_B(s(t)) & \text{if } \epsilon \geq \theta_B(i) \\ \mathrm{Rnd}(0,1) & \text{if } \epsilon < \theta_B(i) \end{cases} \tag{6.13}$$

where Π_A and Π_B represent the control policies of Agent A and Agent B, respectively. The mathematical models for the decay functions are presented as follows:

1) Exponential-based decay function

The EXD function is a widely used exploration strategy where the decay follows an exponential pattern. The decay function is generated as:

$$\theta_A(i) = {\theta_0}^i \tag{6.14}$$

where θ_0 is the initial value of the decay function. This approach allows for a smooth reduction in exploration over time, balancing between exploration and exploitation.

2) Step-based decay function

Step-based decay is a popular learning rate scheduler where the exploration rate remains constant for a certain number of episodes before decreasing at predefined intervals. This structured reduction pattern improves convergence stability. The decay function is defined as:

$$\theta_B(i) = \alpha_1 \times F^{\text{round}\left(\frac{1+E}{D}\right)} \tag{6.15}$$

where $\alpha_1$is the initial learning rate (Usually initial learning rate = 0.5). F is a factor value controlling the rate at which the learning rate drops; the larger factor F is, the slower the learning rate or E2E will decay. Conversely, the smaller the factor F, the faster the learning rate or E2E will decay. D is the scale factor (typically the θ decreases after ten times of learning) that controls how many learning iterations the θ drops; E is the current learning iteration.

Decision making with different agents

Three ensemble learning methods are proposed to determine the final action, i.e., a maximum value-based method, a random-based method, and a weighted-based method. Previous research has shown that the weighted-based method outperforms the other two by improving vehicle energy efficiency [292]. In the weighted-based approach, the final action is determined by evaluating the contributions of Agent A and Agent B, assigning them proportional weights as:

$$a_{\text{final}} = \mu a_{A(t)} + \delta a_B(t) \tag{6.16}$$

where μ and δ are the coefficient factors determining the proportions in action $a_{A(t)}$ and action $a_{B(t)}$, respectively. This method ensures a balanced decision-making process, leveraging the strengths of both agents to optimize control performance.

6.2.3 Ensemble learning with in-target minimization and UTD

Building upon the ideas of ensemble learning, we propose ensemble learning with in-target minimization and update-to-data (UTD) mechanism. This method introduces two key improvements:

1. An ensemble of N independently initialized Q-functions that share a common update target, a minimization step over Q-functions within the target calculation, $N = 2$ is set in this paper and more details are in Appendix.

2. UTD mechanism significantly enhances sample efficiency and accelerates convergence by extracting more value from each collected data sample. And UTD ratio G is the number of updates taken by the agent compared to the number of actual interactions with the environment, which is defined as:

$$G = \frac{\text{Number of updates}}{\text{Number of actual interactions}} \tag{6.17}$$

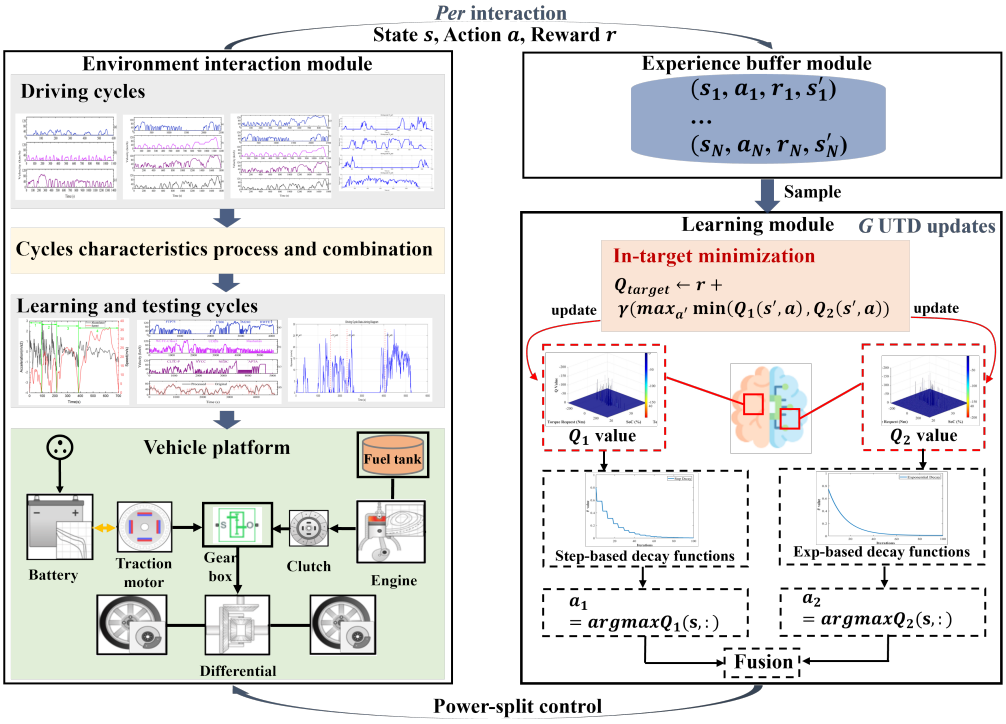


Fig. 6.2. Ensemble learning with in-target minimization and UTD updates in end-to-end EMS.

The proposed double Q learning with the minimization and UTD (DQM-UTD), as described in Fig. 6.2, consists of three key components: (1) **an environment interaction module**, which includes a complete hybrid powertrain system that generates states and rewards; (2) **an experience buffer**, which stores (s, a, r, s') tuples for batch sampling, where s' is next state $s(t+1)$, and enhances learning efficiency by performing G updates per environment interaction to reduce the sample complexity, and (3) **a learning module**, which features dual

Q tables with in-target minimization, where both Q-tables are updated using two different decay functions. The system operates by collecting experiences from environment interactions, storing them in the buffer, performing UTD updates using randomly sampled batches, and generating power-split control actions that are then executed in the environment to effectively address the overestimation bias while maintaining learning efficiency. The update rule for Q is given by:

$$\begin{aligned} &Q_i\left(s(t), a(t)\right) \leftarrow Q_i\left(s(t), a(t)\right) \\ &+\gamma\left[r + \max \min Q_i(s(t+1), :) - Q_i\left(s(t), a(t)\right)\right] \end{aligned} \tag{6.18}$$

Therefore, the $\min Q_i(s(t+1), :)$ operation selects the minimum Q-value between the two tables for each action, introducing a pessimistic estimate that mitigates the optimistic bias from the subsequent $\max$ operation. By using a shared target for both Q-tables, the approach ensures consistent learning objectives while promoting diverse exploration through different decay functions. Additionally, the integration of the UTD ratio G enhances learning efficiency by increasing the effective updates per environment interaction, reducing the overall data required for convergence. The complete algorithm is summarized in Algorithm 2, and more theoretical analysis is found in **Theoretical analysis**.

Algorithm 2: Ensemble Q-learning with the Minimization and UTD Algorithm

Input: Initialize $Q_i(s, a)$ for all $s \in S$, $a \in A$, $i = 1, 2$
Experience replay buffer
Output: Trained Q_1 and Q_2

for $k = 1$ *to* K **do**
  for *each time step* **do**
    Select action $a \in A$ based on $\{Q_i(s, a)\}_{i=1,2}$;
    Observe reward r and next state s';
    Store (s, a, r, s') in Buffer;
  end
  for $g = 1$ *to* G ; // UTD updates
  do
    batch $\leftarrow$ SampleBatch(Buffer);
    for *each* (s, a, r, s') *in batch* **do**
      $Q_{\text{target}} \leftarrow r + \gamma \max_a \min(Q_1(s', a), Q_2(s', a))$;
      $Q_1(s, a) \leftarrow Q_1(s, a) + \alpha(Q_{\text{target}} - Q_1(s, a))$;
      $Q_2(s, a) \leftarrow Q_2(s, a) + \alpha(Q_{\text{target}} - Q_2(s, a))$;
    end
  end
end

6.2.4 Normalized Bias Estimation

To quantify the effectiveness of our approach in reducing overestimation bias, we develop a comprehensive bias analysis framework. The analysis consists of several key components:

Monte Carlo Return Calculation

The true Q-value is computed using the discounted Monte Carlo (MC) method, which sums the discounted rewards from state s until the end of the episode:

$$Q_\pi(s,a) = r_0 + \gamma r_1 + \gamma^2 r_2 + \cdots = \sum_{t=0}^{T} \gamma^t r_t, \tag{6.19}$$

where r_t is the reward at time step t, γ is the discount factor, and T is the episode length.

Normalized Bias Calculation

To make bias measurements comparable across different scales, we introduce normalized bias:

$$\textit{Normalized Bias} = \frac{Q(s(t),a(t)) - Q_\pi(s(t),a(t))}{\left| E_{s(t),a(t)\sim\pi}\left[Q_\pi((t)s,a(t))\right]\right|}, \tag{6.20}$$

where the expected value of true Q-values is:

$$E_{s(t),a(t)\sim\pi}\left[Q_\pi(s(t),a(t))\right] = \sum_{i=1}^{N} P(s_i(t),a_i(t)) Q_\pi(s_i(t),a_i(t)), \tag{6.21}$$

where the state-action probability $P(s_i(t),a_i(t))$ is computed using the visit frequency:

$$P(s_i(t),a_i(t)) = \frac{N_{visit}(s_i(t),a_i(t))}{\sum_{j=1}^{N} N_{visit}(s_j(t),a_j(t))}, \tag{6.22}$$

$N_{visit}(s_i(t),a_i(t))$ represents the number of times state-action pair $(s_i(t),a_i(t))$ has been visited during training.

Statistical Analysis

We analyze both the mean and standard deviation (Std) of the normalized bias across state-action pairs to understand, where mean bias indicates systematic overestimation or underestimation and standard deviation captures the consistency of estimates. This analysis

allows us to quantitatively evaluate the effectiveness of DQM-UTD in reducing overestimation bias compared to traditional Q-learning approaches.

6.3 | Theoretical analysis

6.3.1 Preliminaries

Consider the DQM-UTD algorithm with two Q-functions Q_1 and Q_2. Let Q^* denote the optimal Q-function. The update rule combines in-target minimization with the UTD ratio G:

$$\begin{aligned} Q_i(s,a) \leftarrow Q_i(s,a) + \\ \alpha\left(r + \gamma \max_{a'} \min\left(Q_1(s',a'), Q_2(s',a')\right) - Q_i(s,a)\right), \end{aligned} \tag{6.23}$$

where $i = 1, 2$.

6.3.2 Convergence analysis

Target value analysis

Let T be the Bellman operator defined by minimization:

$$T(Q_1, Q_2)(s,a) = \mathbb{E}[r + \gamma \max_{a'} \min(Q_1(s',a'), Q_2(s',a'))|s,a], \tag{6.24}$$

where r is the immediate reward, $\gamma \in [0,1]$ is the discount factor, and s' represents the next state given the current state s and action a. The operator T updates the Q-functions using a conservative strategy that minimizes over Q_1 and Q_2.

Proposition 1 (Contraction mapping): The operator T is a contraction mapping under the $\sup$-norm, with contraction factor γ:

$$\| T(Q_1, Q_2) - T(Q_1', Q_2') \|_\infty \leq \gamma \max\left(\| Q_1 - Q_1' \|_\infty, \| Q_2 - Q_2' \|_\infty\right). \tag{6.25}$$

This contraction guarantees the uniqueness of the fixed point Q^* and ensures that iterative applications of the Bellman operator T converge to Q^*.

Step 1: Express the difference between two operators. From the definition of T, the difference between $T(Q_1, Q_2)$ and $T(Q_1', Q_2')$ is:

$$|T(Q_1, Q_2)(s,a) - T(Q_1', Q_2')(s,a)| = \gamma \Big| \mathbb{E}\left[\max_{a'} \min(Q_1(s',a'), Q_2(s',a'))\right] - \mathbb{E}\left[\max_{a'} \min(Q_1'(s',a'), Q_2'(s',a'))\right] \Big|. \quad (6.26)$$

Step 2: Use properties of $\min$ and $\max$. For any action a', the difference between $\min(Q_1(s',a'), Q_2(s',a'))$ and $\min(Q_1'(s',a'), Q_2'(s',a'))$ satisfies:

$$\left|\min(Q_1(s',a'), Q_2(s',a')) - \min(Q_1'(s',a'), Q_2'(s',a'))\right| \leq \max\left(|Q_1(s',a') - Q_1'(s',a')|, |Q_2(s',a') - Q_2'(s',a')|\right). \quad (6.27)$$

Taking the $\max$ over a' does not increase the magnitude of the difference, so:

$$\left|\max_{a'} \min(Q_1(s',a'), Q_2(s',a')) - \max_{a'} \min(Q_1'(s',a'), Q_2'(s',a'))\right| \leq \max\left(\| Q_1 - Q_1' \|_\infty, \| Q_2 - Q_2' \|_\infty\right). \quad (6.28)$$

Step 3: Combine with the discount factor γ. Using the linearity of expectation and the contraction property of γ, we get:

$$\left|T(Q_1, Q_2)(s,a) - T(Q_1', Q_2')(s,a)\right| \leq \gamma \max\left(\| Q_1 - Q_1' \|_\infty, \| Q_2 - Q_2' \|_\infty\right). \quad (6.29)$$

Step 4: Take the supremum. Taking the supremum over all state-action pairs (s,a), we arrive at:

$$\| T(Q_1, Q_2) - T(Q_1', Q_2') \|_\infty \leq \gamma \max\left(\| Q_1 - Q_1' \|_\infty, \| Q_2 - Q_2' \|_\infty\right). \quad (6.30)$$

The above derivation proves that T is a contraction mapping with factor γ, ensuring: 1) A unique fixed point Q^*, where $T(Q^*) = Q^*$; 2) Convergence of the Q-functions to Q^* under iterative applications of T, regardless of the initialization of Q_1 and Q_2.

This property guarantees the stability and reliability of using the Bellman operator for reinforcement learning tasks.

6.3.3 UTD impact analysis

Theorem 1 (Sample efficiency). With a UTD ratio G, each environmental sample generates G updates. For a fixed number of environment interactions N, the effective sample size becomes GN, leading to accelerated convergence:

$$\mathbb{E}[\| Q_t - Q^* \|_\infty] \leq O\left(\frac{1}{\sqrt{GN}}\right). \tag{6.31}$$

Proof of Theorem 1

The error between the current Q-function Q_t and the optimal Q-function Q^* can be decomposed into two components:

$$\mathbb{E}[\| Q_t - Q^* \|_\infty] \leq \mathbb{E}[\text{Bias}] + \mathbb{E}[\text{Variance}], \tag{6.32}$$

where **Bias** accounts for approximation errors due to the Bellman operator and limited updates, and **Variance** captures stochasticity introduced by sampling from the environment.
The UTD ratio G increases the number of updates per environment sample, effectively scaling the total number of updates to GN, where N is the number of environment interactions. According to standard statistical results, the variance decreases with the square root of the total updates:

$$\mathbb{E}[\text{Variance}] \propto \frac{1}{\sqrt{GN}}. \tag{6.33}$$

The convergence of Q-learning is governed by the contraction property of the Bellman operator, which ensures that the error after t updates is inversely proportional to the square root of the total number of updates:

$$\mathbb{E}[\| Q_t - Q^* \|_\infty] \leq O\left(\frac{1}{\sqrt{GN}}\right). \tag{6.34}$$

Thus, increasing G accelerates convergence by reducing variance and improving the sample efficiency.

6.3.4 Bias control

Theorem 2 (Bias reduction): The in-target minimization operation provides an upper bound on overestimation:

$$\mathbb{E}[\max_{a'} \min(Q_1(s', a'), Q_2(s', a')) - \max_{a'} Q^*(s', a')] \leq 0. \tag{6.35}$$

Proof: For any action a', the minimization operator acts as a conservative estimator:

$$\min(Q_1(s', a'), Q_2(s', a')) \leq \frac{Q_1(s', a') + Q_2(s', a')}{2},$$
$$\mathbb{E}[\min(Q_1(s', a'), Q_2(s', a'))] \leq Q^*(s', a').$$

Thus, the maximization over a' maintains the inequality, ensuring that in-target minimization avoids overestimation and remains pessimistically biased.

Explanation: Unlike traditional Q-learning, which maximizes over a single Q-function, the minimization step in DQM ensures a more conservative update by avoiding over-reliance on overestimated Q-values. This bias reduction improves stability and policy robustness.

6.3.5 Sample complexity analysis

Theorem 3 (Sample complexity): To achieve an ϵ-optimal policy with probability $1 - \delta$, DQM-UTD requires:

$$O\left(\frac{1}{G(1-\gamma)^4\epsilon^2} \log \frac{1}{\delta}\right) \tag{6.36}$$

environment interactions, where G is the UTD ratio.

Proof

To analyze the sample complexity, we decompose the error for any state-action pair (s, a) as:

$$|Q_t(s,a) - Q^*(s,a)| \leq \underbrace{|T^t Q_0(s,a) - Q^*(s,a)|}_{\text{Bias}} + \underbrace{\left|\sum_{k=0}^{N-1} \alpha_k (T^k Q_0 - T^{k+1} Q_0)\right|}_{\text{Variance}}, \tag{6.37}$$

where the first term (**Bias**) reflects the initialization error, which decreases as more Bellman updates are applied; the second term (**Variance**) represents the stochastic error caused by sampling and finite updates.

To ensure $|Q_t(s,a) - Q^*(s,a)| \leq \epsilon$, we control both terms to be at most $\epsilon/2$.

Initialization Bias The Bellman operator T is a contraction mapping with factor γ, so after t iterations, the initialization bias satisfies:

$$|T^t Q_0(s,a) - Q^*(s,a)| \leq \gamma^t \cdot \| Q_0 - Q^* \|_\infty. \tag{6.38}$$

To ensure the bias is within $\epsilon/2$, we require:

$$\gamma^t \cdot \| Q_0 - Q^* \|_\infty \leq \frac{\epsilon}{2}. \tag{6.39}$$

Taking the logarithm on both sides and solving for t:

$$t \geq \frac{\log(2\| Q_0 - Q^* \|_\infty/\epsilon)}{\log(1/\gamma)}. \tag{6.40}$$

Using $\log(1/\gamma) \approx 1 - \gamma$ for small $(1-\gamma)$, this simplifies to:

$$t \geq O\left(\frac{1}{1-\gamma}\log\frac{1}{\epsilon}\right). \tag{6.41}$$

Variance Term The variance term arises due to stochastic sampling and finite updates. Its expected magnitude is bounded by:

$$\mathbb{E}\left[\left|\sum_{k=0}^{N-1}\alpha_k(T^k Q_0 - T^{k+1}Q_0)\right|\right] \leq \frac{R_{\max}}{\sqrt{N}}, \tag{6.42}$$

where $R_{\max}$ is the maximum reward. With a UTD ratio G, the effective variance is reduced by a factor of $1/G$, giving:

$$\mathbb{E}\left[\left|\sum_{k=0}^{N-1}\alpha_k(T^kQ_0 - T^{k+1}Q_0)\right|\right] \leq \frac{R_{\max}}{G\sqrt{N}}. \tag{6.43}$$

To ensure the variance term is within $\epsilon/2$, we require:

$$\frac{R_{\max}}{G\sqrt{N}} \leq \frac{\epsilon}{2}. \tag{6.44}$$

Rearranging for N, we obtain:

$$N \geq \frac{4R_{\max}^2}{G\epsilon^2}. \tag{6.45}$$

Combining Bias and Variance To achieve an ϵ-optimal policy, both the bias and variance terms must be controlled:

$$t \geq O\left(\frac{1}{1-\gamma}\log\frac{1}{\epsilon}\right), \quad N \geq \frac{4R_{\max}^2}{G\epsilon^2}. \tag{6.46}$$

In Q-learning, the convergence of both terms is influenced by the discount factor γ, as it affects the propagation of rewards. Specifically, the bias depends on $(1-\gamma)^{-1}$, reflecting the slow contraction of the Bellman operator for large γ; the variance depends on $(1-\gamma)^{-3}$, due to its role in determining the effective sample size and reward propagation over time.

Combining these effects, the overall sample complexity becomes:

$$N \geq O\left(\frac{1}{G(1-\gamma)^4\epsilon^2}\log\frac{1}{\delta}\right), \tag{6.47}$$

where the logarithmic term accounts for confidence level $1-\delta$.

The sample complexity of DQM-UTD is influenced by: 1)The UTD ratio G, which improves sample efficiency by reducing variance; 2) The discount factor γ, which slows convergence as γ approaches 1; 3) The error tolerance ϵ and confidence level $1-\delta$.

Thus, to achieve an ϵ-optimal policy with high probability, DQM-UTD requires:

$$O\left(\frac{1}{G(1-\gamma)^4\epsilon^2}\log\frac{1}{\delta}\right) \tag{6.48}$$

environment interactions.

6.3.6 Stability Analysis

Under the following assumptions:

1. Bounded rewards: $|r| \leq R_{\max}$;

2. Learning rates satisfy Robbins-Monro conditions: $\sum_t \alpha_t = \infty, \sum_t \alpha_t^2 < \infty$;

3. Sufficient exploration: all state-action pairs visited infinitely often;

4. Discount factor: $\gamma < 1$.

Theorem 4 (Convergence): The DQM-UTD algorithm converges with probability 1:

$$P(\lim_{t\to\infty} Q_i(s,a) = Q^*(s,a)) = 1, \quad i = 1,2. \tag{6.49}$$

Define the combined Bellman operator $\hat{T}$ for DQM-UTD:

$$\hat{T}Q(s,a) = \mathbb{E}[r + \gamma \max_{a'} \min(Q_1(s',a'), Q_2(s',a'))|s,a], \tag{6.50}$$

where $\hat{T}$ is a contraction mapping in the sup-norm:

$$\| \hat{T}Q_1 - \hat{T}Q_2 \|_\infty \leq \gamma \| Q_1 - Q_2 \|_\infty. \tag{6.51}$$

With appropriate learning rates:

$$\sum_t \alpha_t = \infty, \quad \sum_t \alpha_t^2 < \infty, \tag{6.52}$$

the convergence to the fixed point Q^* is guaranteed:

$$\lim_{t\to\infty} Q_i(s,a) = Q^*(s,a) \quad \text{with probability 1.} \tag{6.53}$$

Remarks: The UTD ratio G enables a trade-off between computational efficiency and sample efficiency. Meanwhile, the minimization operator ensures robustness by preventing overestimation, leading to faster convergence and stable policy updates compared to standard Q-learning.

6.4 | Results and discussion

6.4.1 Validation setup

All simulations were conducted in a software-in-the-loop (SiL) with an initial battery SoC of 30% for 100 learning iterations on an Intel(R) Core(TM) i7-10700 CPU. Building upon our previous work [292], we analyze the double Q-learning method by implementing both step-based and conventional exponential-based decay functions for double agents. During each learning episode, these decay functions generate specific values θ_1 and θ_2, which are compared against a random number ϵ to execute the ϵ-greedy strategy for action selection in two agents. The final action is determined through a weighted combination of the two agents with a 9:1 proportional ratio. The algorithm parameters include a discount factor $\gamma = 0.9$, learning rate $\alpha = 0.1$, and SoC reference value of 28%.

We first compare the learning performance between single Q and double Q-learning systems to visualize the reduction in overestimation bias achieved by the double structure. Then we evaluate the performance improvements obtained through ensemble learning with in-target minimization and update ratio optimization. Subsequently, we conduct comparative analyses across three standard driving cycles and one real-world driving cycle. Finally, we assess real-time performance through HiL testing.

Driving cycles

The proposed strategy is trained on the learning cycle from [293] and validated across four diverse driving conditions, as shown in Fig. 6.3 [294]. These include three synthesized Standard

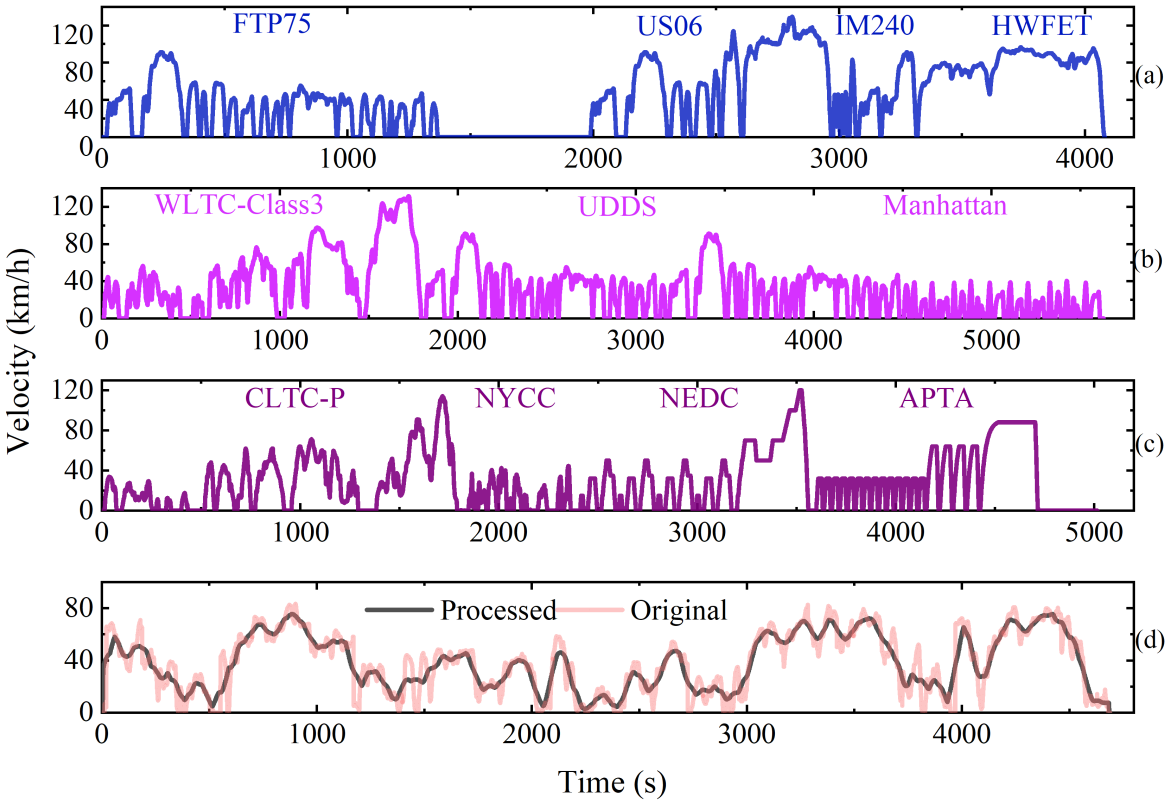


Fig. 6.3. The driving cycles for proposed strategy validation: (a) Driving Cycle I, (b) Driving Cycle II, (c) Driving Cycle III, (d) Driving Cycle IV

Driving Cycles (I, II, III) and a Real-world Driving Cycle (IV). The first three cycles are generated from eleven standard driving cycles to enhance complexity and cover a broad range of road characteristics. The fourth is derived from real-world driving data collected from passenger vehicles in Wuhan via the China Automotive Traffic Condition (CATC) information system.

Hardware-in-the-loop testing platform

To validate the practical feasibility of the proposed end-to-end control methods, we implemented a comprehensive hardware-in-the-loop (HiL) testing platform, as illustrated in Fig. 6.4. The HiL setup consists of a host PC and a Speedgoat real-time target machine, connected through a physical interface. The host PC is responsible for designing and compiling the real-time model of the vehicle model and the optimized control prototypes. Once compiled, the models are downloaded to the Speedgoat machine, which executes them in real time.

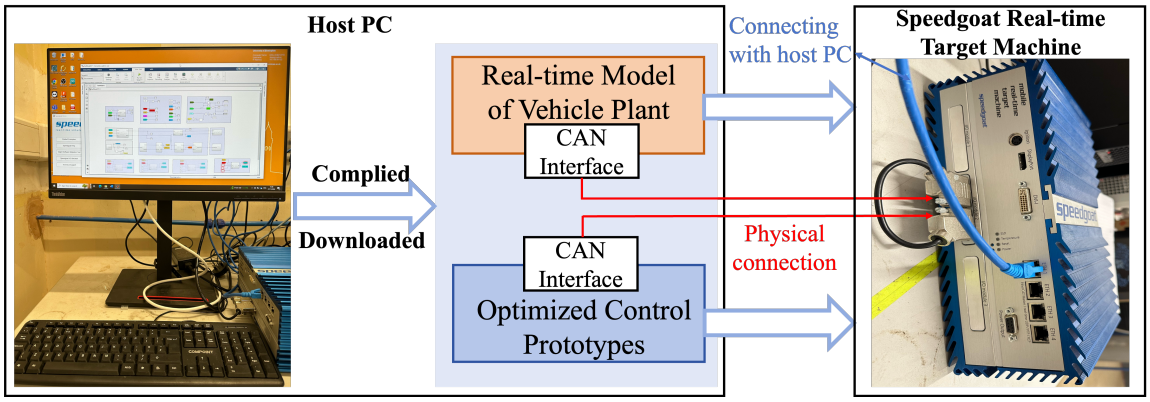


Fig. 6.4. Hardware-in-the-loop testing platform for real-time validation and testing.

6.4.2 Comparative analysis with the single and double Q learning systems

Fig. 6.5 illustrates the change of normalized bias in single and double Q-learning approaches. The single Q learning system exhibits significant overestimation, with normalized bias peaking at 3.0 during early training phases and substantial variance. In contrast, the double Q-learning system demonstrates remarkable stability, maintaining near-zero bias throughout the training process with minimal variance, thus validating its effectiveness in mitigating overestimation issues.

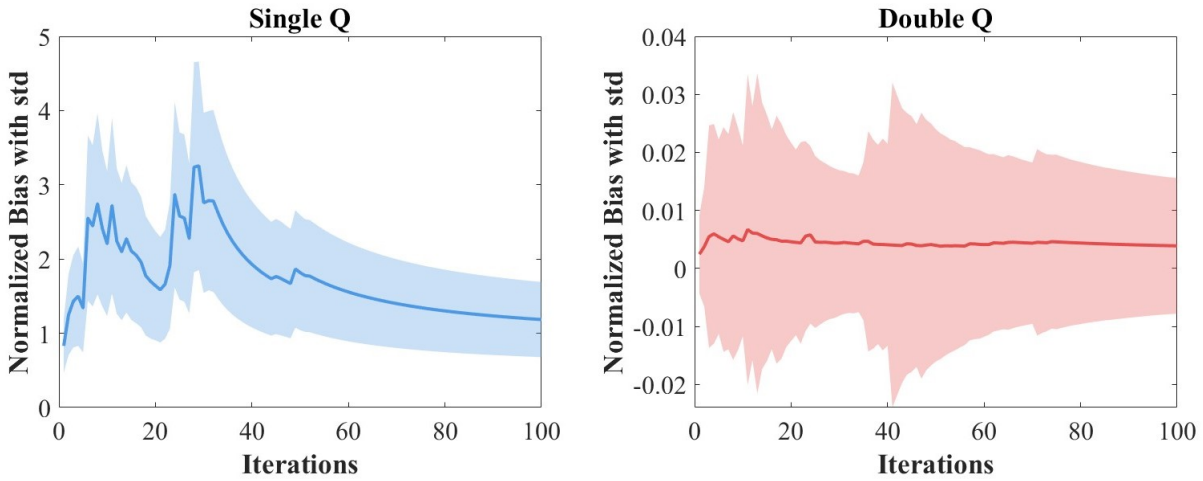


Fig. 6.5. Comparative analysis of overestimation bias between single and double Q-learning systems

The Q-value distributions across different control ratios (0.1, 0.5, 0.8, and 1.0) in the single Q-learning are presented in Fig. 6.6 to demonstrate the learned control policy at varying power levels. At a low control ratio (a=0.1), the Q-values maintain predominantly negative values,

with distinct variations in the mid-torque region near the reference SoC. As the control ratio increases to a=0.5, the distribution evolves to show more balanced Q-values, particularly in moderate torque demand regions. The higher control ratios (a=0.8 and 1.0) reveal increasingly pronounced negative spikes, especially at extreme torque demands, indicating the learned penalties for excessive engine utilization.

For comparison, Fig. 6.7 presents the Q-value distributions of Q1 and Q2 in the double Q learning at a=0.5, representing the critical operating region where power balancing decisions are most frequent. Double Q exhibits similar patterns, with Q-values distributed between -200 and 0, primarily concentrated in the -50 to 0 range. While Q1 displays more pronounced spikes in mid-range torque values, Q2 demonstrates smoother transitions, highlighting their complementary learning characteristics. The state space exhibits stable control behavior at its extremes (high torque demands and SoC), with higher variations in central regions reflecting the adaptation to typical vehicle operating conditions.

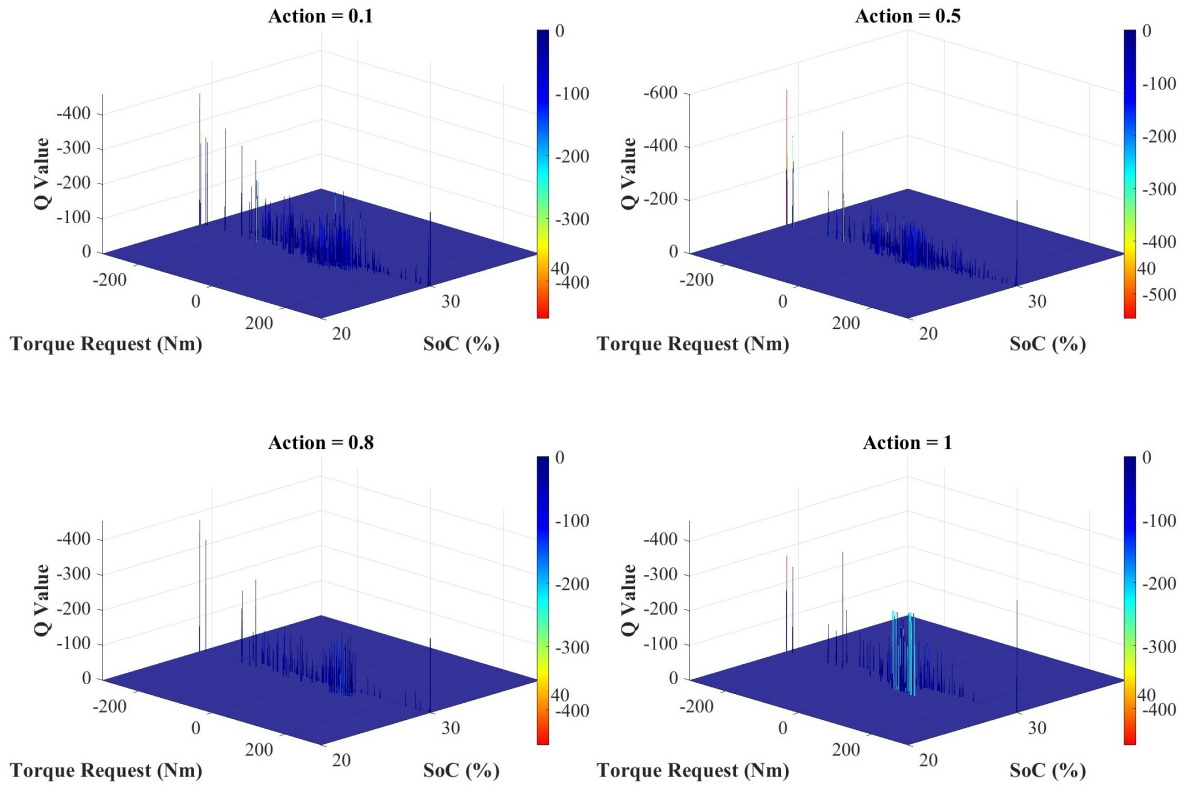


Fig. 6.6. Q-value distributions across different control ratios (a=0.1, 0.5, 0.8, 1.0) in single Q-learning system

Fig. 6.8 compares the learning performance between single and double Q-learning systems through average reward and system efficiency. Double Q demonstrates superior performance,

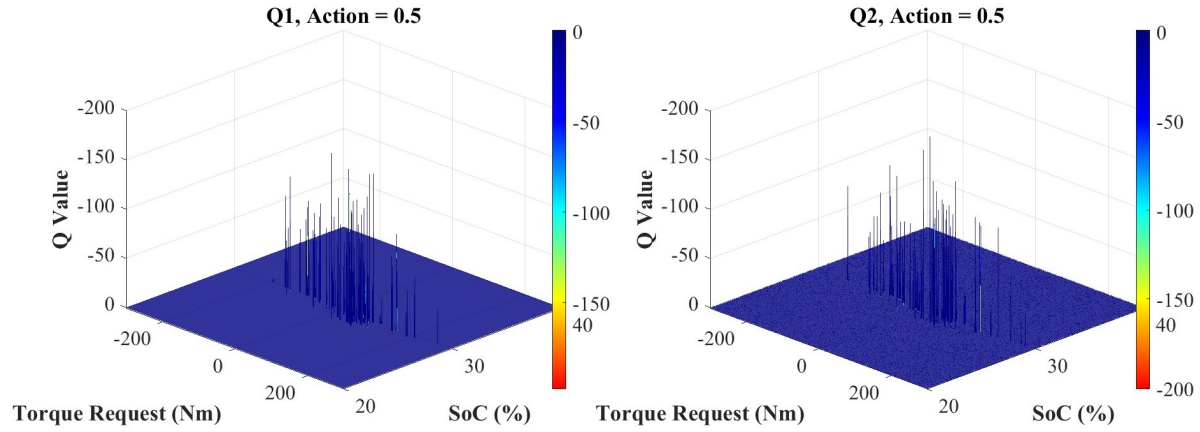


Fig. 6.7. Q-value distributions between Q1 and Q2 at control ratio a=0.5

achieving higher convergent reward and higher system efficiency. The double Q system also exhibits higher learning stability with fewer fluctuations after 40 iterations. These improvements indicate that double Q learning provides more robust and reliable control policies for PHEV.

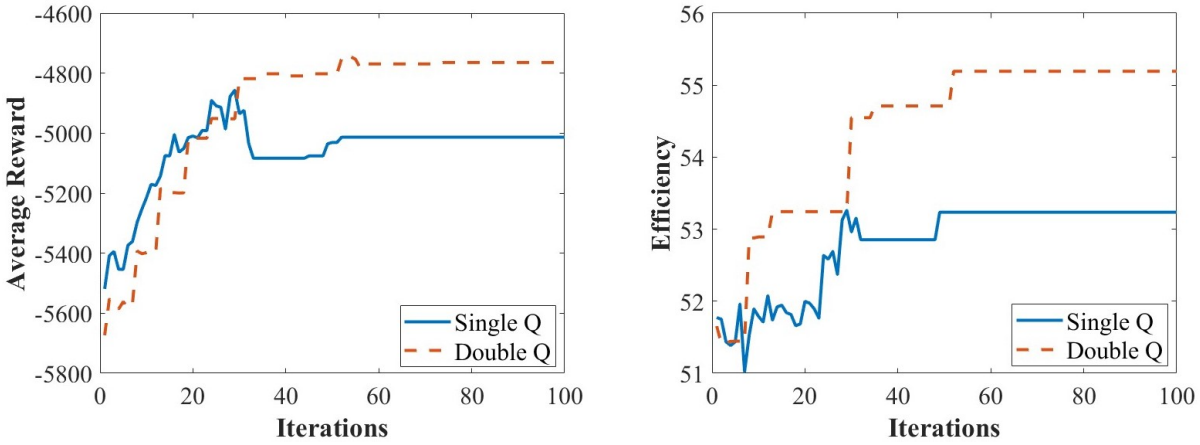


Fig. 6.8. Performance comparison of average reward and system efficiency between single and double Q learning systems.

6.4.3 Improvement in learning performance with the ensemble with in-target minimization and UTD ratio

Although higher UTD ratios generally lead to more stable learning and can significantly reduce the total number of environment interactions, the optimal ratio requires careful tuning to balance computational cost with learning efficiency, ensuring effective control policy without overfitting on limited data. Therefore, the relationship between UTD ratios (G) and learning performance was conducted particularly in addressing overestimation bias across various G, as shown in Fig. 6.9.

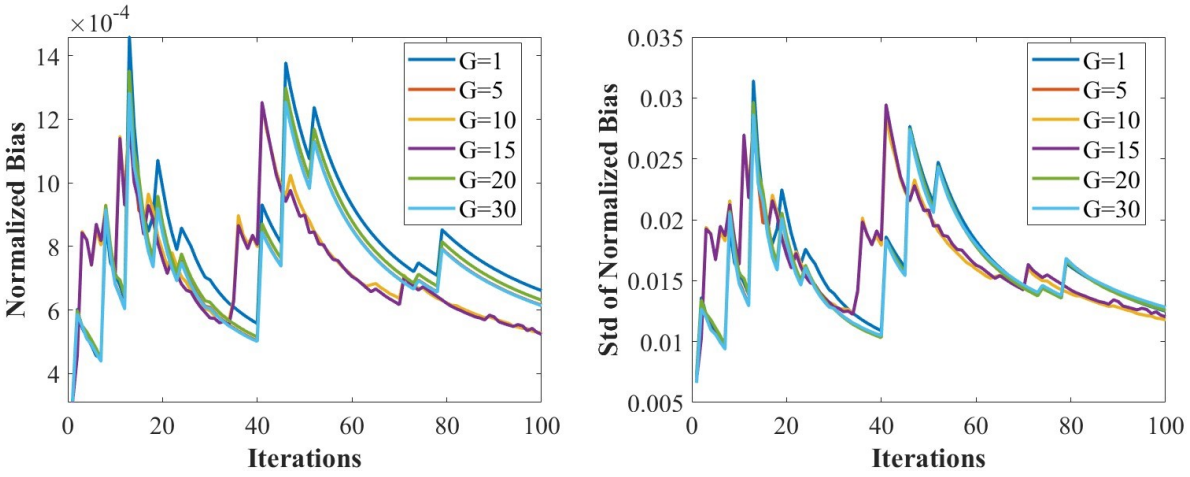


Fig. 6.9. Average and std of normalized bias under different UTD ratios

Lower UTD ratios (G=1,5) demonstrate suboptimal performance with higher normalized bias reaching 14×10^{-4} and larger Std up to 0.035, indicating unstable learning behavior. As G increases, both metrics show marked improvement with G=10-15 as an optimal range. In this range, the normalized bias reduces to approximately 8×10^{-4} with Std around 0.015, while maintaining efficient learning progress. However, increasing G beyond 20 yields diminishing returns and potential instability, accompanied by increased computational cost and a higher risk of overfitting on limited data.

The average reward and system efficiency across different UTD ratios are presented in Fig. 6.10. G=1 achieves the best final performance, with average rewards stabilizing at -4643.9 and system efficiency reaching 61.4% due to tighter coupling between experience and learning. However, medium UTD ratios $G = 10 - 15$ show faster convergence with about 70 iterations, demonstrating better sample efficiency through more frequent value updates.

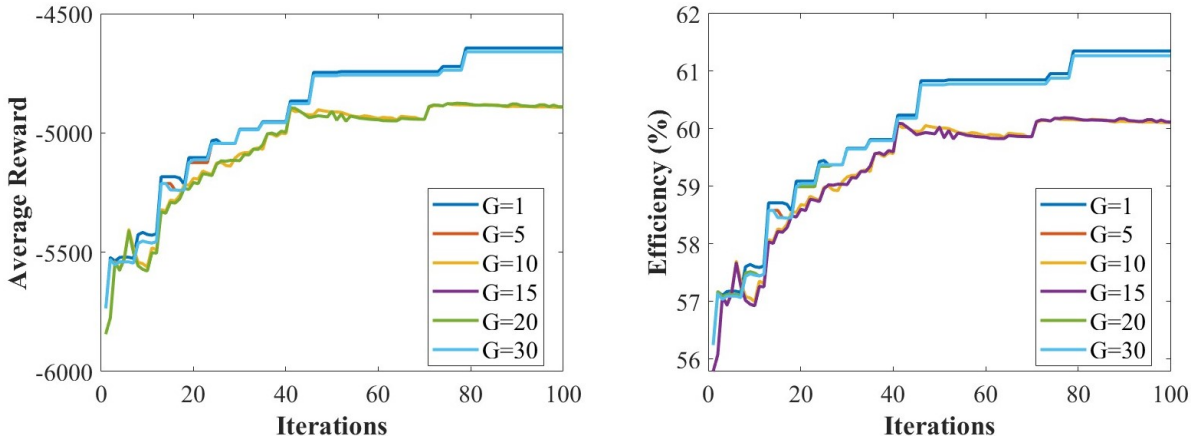


Fig. 6.10. Average reward and system efficiency under different UTD ratios

Table 6.2 compares different UTD ratios against baseline methods across multiple metrics. Single Q converges fastest (52 iterations) but suffers from high bias (1.18) and low efficiency (53.24%). Double Q reduces bias (3.810^{-3}) and improves efficiency (55.19%) but increases computation time. Analyzing UTD variations reveals distinct trade-offs. Low UTD (G=1-5) achieves the highest reward and efficiency but requires more iterations due to stronger correlation with real experiences. Medium UTD (G=10-15) strikes a balance, offering faster convergence with stable performance. However, high UTD ($G \geq 20$) yields diminishing returns, increasing computational cost without significant performance gains, and sometimes degrading due to over-updating and misalignment with real-world interactions.

From a computational perspective, DQM-UTD with G=15 requires only approximately 4.6% more computation time (4301.3s) than standard Q-learning (4112.1s), while double Q requires 38.2% more time (5681.5s). Importantly, DQM-UTD's runtime execution adds minimal overhead (only requiring two table lookups and one minimization operation per control step), making it highly suitable for real-time vehicular applications. The memory requirements scale as $O(2\times|S|\times|A|)$, identical to double Q-learning, while the improved sample efficiency reduces the overall training resource requirements compared to conventional approaches.

These results suggest that DQM-UTD with the appropriate UTD ratio, particularly G=10-15, offers a robust approach requiring both learning efficiency and stability while maintaining computational feasibility, more analysis is described in Section 6.3.

Table 6.2
Performance comparison under different UTDs

UTD ratio	Average reward	System efficiency (%)	Convergence iterations	Computation time (s)	Normalized bias
$G = 1$	**-4643.9**	**61.4**	80	5300.5	6.610^{-4}
$G = 5$	-4659.8	61.3	80	5122.3	6.1×10^{-4}
$G = 10$	-4892.5	60.1	71	4713.6	5.3×10^{-4}
$G = 15$	-4890.5	60.1	71	4301.3	$\mathbf{5.2} \times 10^{-4}$
$G = 20$	-4890.5	61.3	71	4837.7	6.3×10^{-4}
$G = 30$	-4659.9	61.3	80	5335.9	6.1×10^{-4}
Single Q	-5013.4	53.24	**52**	**4112.1**	1.18
Double Q	-4764.3	55.19	57	5681.5	3.8×10^{-3}

The Q-value distributions at $G = 15$ in Fig. 6.11 demonstrate balanced learning characteristics between Q1 and Q2, with similar pattern structures but sufficient diversity to maintain effective

ensemble properties, contributing to robust learning and stable policy. Compared with Fig. 6.7, the traditional double Q learning shows significant value spikes and irregularities near SoC = 28%, due to independent updates causing discordant value estimates between Q1 and Q2. In contrast, DQM-UTD achieves smoother value surfaces through minimization-based shared targets and increased data utilization via UTD, leading to more consistent and stable Q-value estimates across the state space.

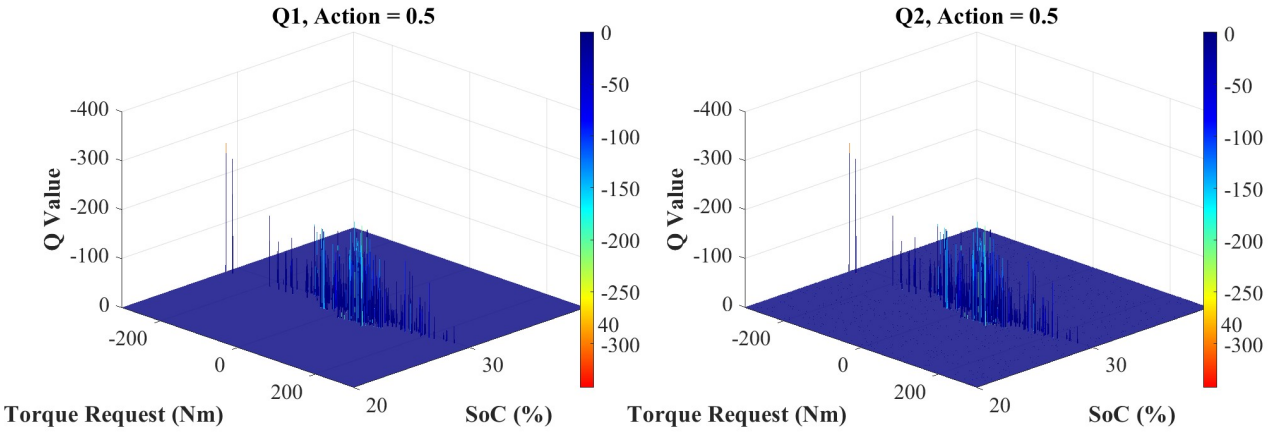


Fig. 6.11. Average and Std of normalized bias when $G = 15$

6.4.4 Real-time performance of DQM-UTD algorithm

The real-time performance of the DQM-UTD method was evaluated on the HiL testing platform under four driving cycles with an initial battery SoC of 30% and compared to rule-based, single Q, and double Q approaches. The results, demonstrated in Fig. 6.12, reveal distinct performance patterns across different driving conditions. While all learning-based methods perform comparably in standardized driving cycles I-III, DQM-UTD demonstrates superior performance in the real-world driving cycle IV, achieving notably lower energy consumption. Throughout all test scenarios, DQM-UTD exhibits better SoC maintenance capabilities, particularly evident in cycle IV where it maintains stable SoC levels while simultaneously delivering more efficient energy management compared to conventional approaches.

The exceptional performance in cycle IV can be attributed to its distinctive driving characteristics. This real-world driving cycle exhibits smoother driving patterns with moderate maximum speed, milder deceleration, and lower speed variability compared to standardized test cycles. In such driving conditions, DQM-UTD's in-target minimization mechanism can more effectively

optimize power distribution during consistent operational states. These results suggest that the stability and consistency of driving patterns can influence algorithm performance, then a relationship that will be further examined across additional driving conditions in the robustness testing section.

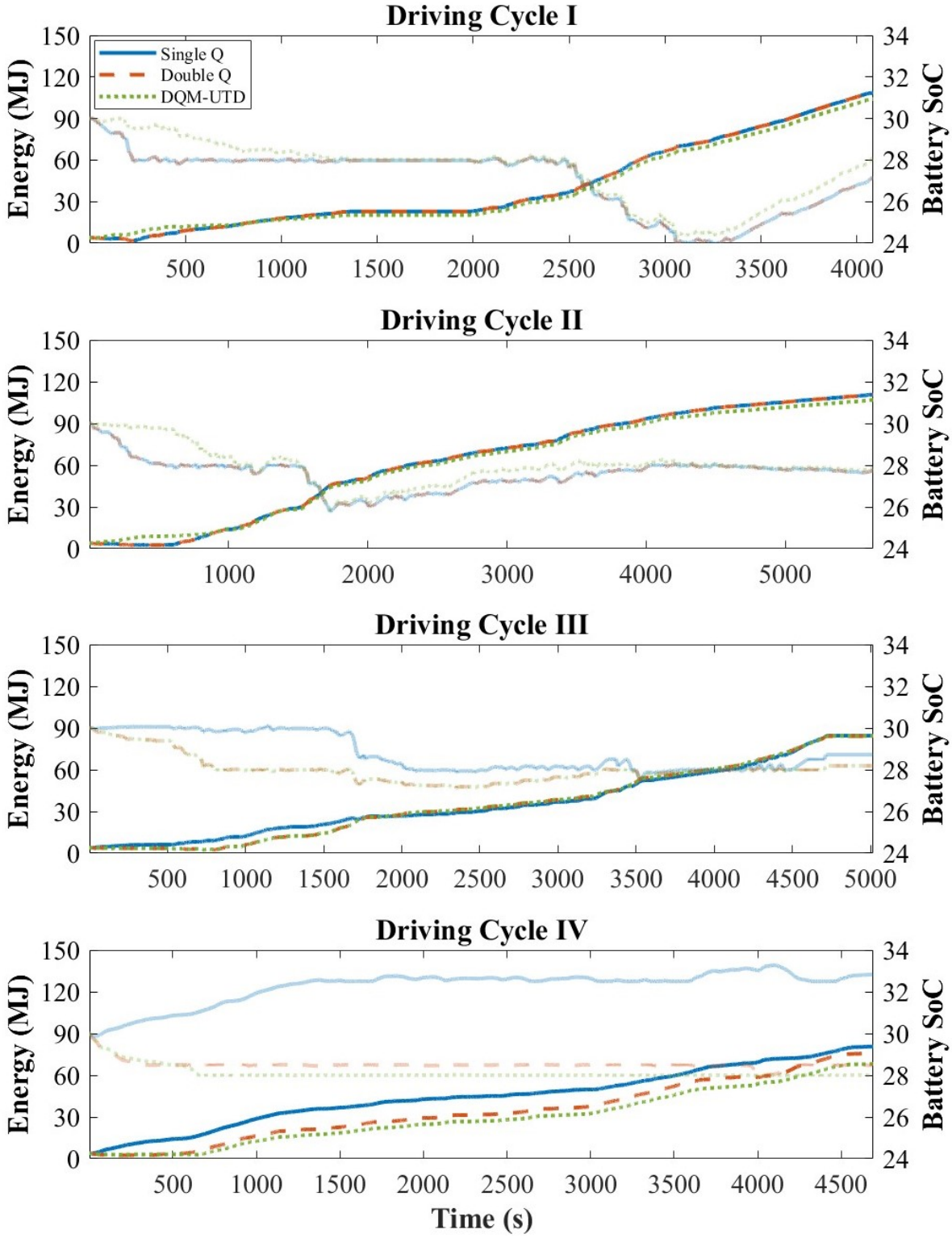


Fig. 6.12. Real-time performance comparison in HiL testing platform

6.4.5 Robustness testing of DQM-UTD algorithm

In real driving scenarios, the PHEV encounters a wide range of driving conditions. Therefore, it is essential to conduct robustness tests on the proposed DQM-UTD algorithm across diverse driving patterns. To further validate the findings from real-time performance tests and explore algorithm performance across a broader range of driving conditions, we extended our analysis to include an additional open dataset cycle, collected by the Joint Research Centre of the European Commission [**Anesiadou2020**], with data processing scripts and visualization available at JRC-DATA-PROCESSING. Five different driving cycles, including standard driving cycles, real-world driving conditions, and an open access dataset are analyzed in Table 6.3.

Table 6.3
Statistics analysis of driving cycles

Cycles	Max Speed (m/s)	Abs Dec (m/s^2)	Std Speed (m/s)	Speed Var
I	35.90	**3.09**	**10.11**	0.82
II	**36.47**	2.49	8.18	0.91
III	33.35	2.64	7.99	1.00
IV	**21.00**	**1.04**	**5.65**	**0.53**
V	27.78	5.90	6.98	**1.18**

The four characteristics, maximum speed (Max Speed), speed standard deviation (Std Speed), absolute deceleration (Abs Dec), and speed variability (Speed Var), are chosen based on their significant impact on energy management through extensive analysis. Specifically, Max Speed influences peak power demands and state space limits, while Std Speed reflects driving pattern stability, affecting algorithm learning. Abs Dec directly impacts regenerative braking and energy recovery, whereas Speed Var (the ratio of speed standard deviation to mean speed) indicates driving consistency, influencing the effectiveness of the UTD mechanism in utilizing collected samples. Finally, Table 6.4 compares DQM-UTD with rule-based, ECMS, single Q, and double Q methods in terms of SoC sustaining, equivalent energy consumption, and relative energy savings.

Table 6.4 presents the comparison results with 'End SoC' (the state of charge after completing a driving cycle), 'Energy (MJ)' (the equivalent energy consumption) [295], and Δ % (energy saving percentage compared to the rule-based baseline). The experimental results demonstrate

Table 6.4
Performance comparison of DQM-UTD and other methods under various driving cycles

Cycle	Init SoC%	Method	End SoC%	Energy (MJ)	Δ %
(I)	30	Rule-based	28.65	107.86	-
	30	ECMS	28.08	107.11	0.70
	30	Single Q	27.14	108.46	-0.55
	30	Double Q	27.15	108.44	-0.54
	30	DQM-UTD	27.72	106.12	**1.61**
	25	Rule-based	26.56	121.91	-
	25	ECMS	28.02	119.15	2.26
	25	Single Q	25.66	118.76	2.58
	25	Double Q	25.66	118.76	2.58
	25	DQM-UTD	27.13	115.88	**4.95**
(II)	30	Rule-based	27.90	109.07	-
	30	ECMS	27.82	108.72	0.32
	30	Single Q	27.72	110.39	-1.2
	30	Double Q	27.72	110.52	-1.33
	30	DQM-UTD	27.85	108.34	**0.67**
	25	Rule-based	27.29	129.32	-
	25	ECMS	28.03	123.72	4.33
	25	Single Q	26.42	121.38	6.14
	25	Double Q	26.42	121.38	6.14
	25	DQM-UTD	27.85	118.22	**8.58**
(III)	30	Rule-based	30.46	87.93	-
	30	ECMS	27.98	85.86	2.35
	30	Single Q	28.72	84.43	3.98
	30	Double Q	28.18	84.26	4.17
	30	DQM-UTD	28.19	84.14	**4.31**
	25	Rule-based	29.17	101.21	-
	25	ECMS	27.77	98.40	2.78
	25	Single Q	28.71	96.63	4.53
	25	Double Q	28.18	96.53	4.62
	25	DQM-UTD	28.19	96.37	**4.78**
(IV)	30	Rule-based	42.50	137.81	-
	30	ECMS	27.76	95.37	30.80
	30	Single Q	32.50	86.50	37.23
	30	Double Q	28.00	80.64	41.48
	30	DQM-UTD	28.00	68.07	**50.60**
	25	Rule-based	38.48	134.71	-
	25	ECMS	28.20	103.45	23.20
	25	Single Q	28.49	87.33	35.17
	25	Double Q	28.00	89.06	33.89
	25	DQM-UTD	28.00	84.44	**37.32**
(V)	30	Rule-based	28.35	81.18	-
	30	ECMS	27.97	78.58	3.20
	30	Single Q	27.89	77.47	4.57
	30	Double Q	27.78	77.47	4.57
	30	DQM-UTD	27.97	74.85	**7.80**
	25	Rule-based	25.60	91.34	-
	25	ECMS	28.04	83.52	8.56
	25	Single Q	25.04	80.17	12.23
	25	Double Q	25.04	80.17	12.23
	25	DQM-UTD	26.15	79.11	**13.39**

the superior performance of our proposed DQM-UTD approach under various driving conditions and initial SOC.

Compared to the traditional methods in Table 6.4, the most significant improvements are observed in the real-world driving cycle (IV), where DQM-UTD achieves remarkable energy savings of 50.60% and 37.32% under different initial SOC conditions compared to the rule-based method. It also maintains solid performance in the highly dynamic Cycle V, achieving up to 13.39% savings despite high deceleration and variability. Although the rule-based method demonstrates competitive performance in some regular, standardized driving cycles (I-III), it lacks robustness when facing complex real-world driving conditions, highlighting the adaptive and robust advantages of the learning-based approaches.

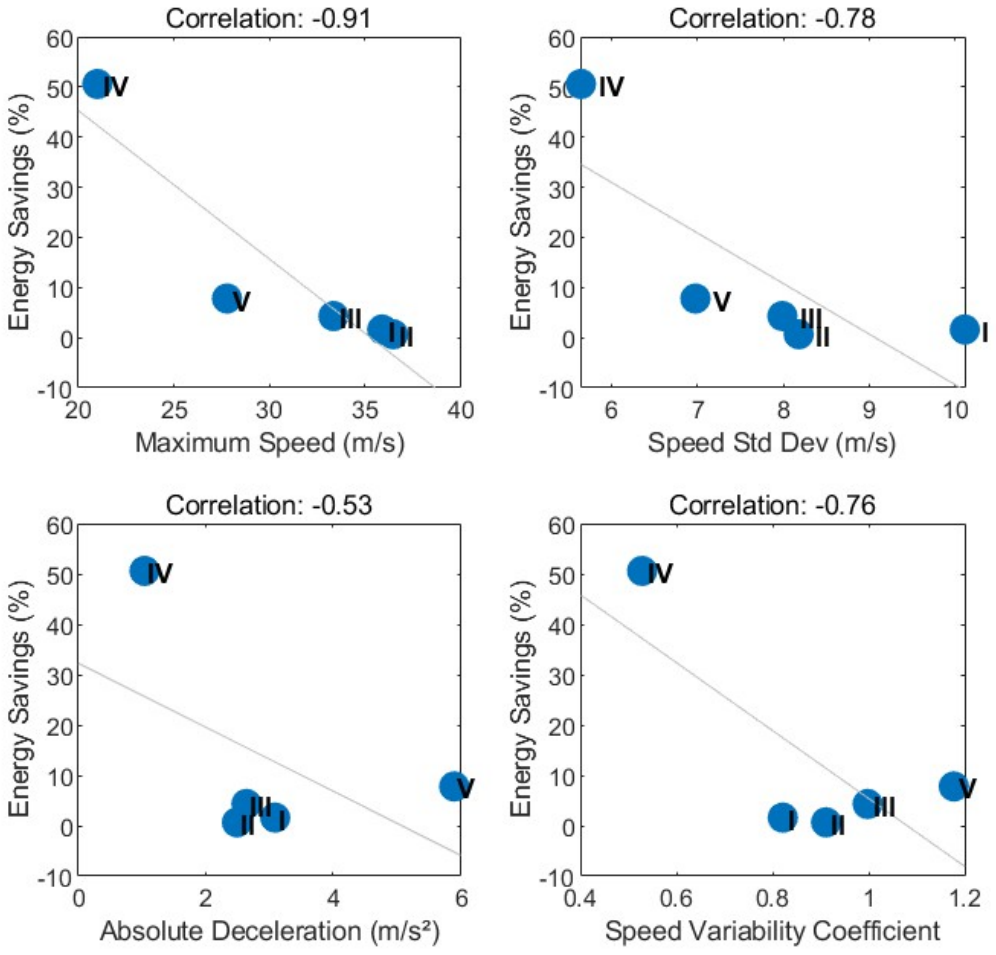


Fig. 6.13. Correlation between driving cycle characteristics and DQM-UTD energy savings

To further investigate how specific driving characteristics influence the performance of DQM-UTD and to better explain the observed performance, a correlation analysis was conducted

between key cycle features and the corresponding energy savings, as illustrated in Fig. 6.13. A strong negative correlation (–0.91) with maximum speed suggests that DQM-UTD performs exceptionally well in environments with moderate speed limits, which indicates that the algorithm effectively optimizes energy distribution when peak power demands are less aggressive. Similarly, a high negative correlation with speed variability (–0.76) indicates that consistent driving patterns enhance the algorithm's ability to learn and adapt effectively. This is exemplified in Cycle IV, which features the lowest variability (0.53) and yields the highest energy savings. The standard deviation of speed also shows a significant negative correlation (–0.78) with performance, reinforcing the importance of stable driving patterns. Lower fluctuations allow for more accurate energy allocation, making the learning process more effective. Absolute deceleration exhibits a moderate negative correlation (–0.53), implying that moderate braking profiles, as found in urban conditions, offer ideal opportunities for regenerative energy recovery without introducing destabilizing dynamics. These results collectively confirm that DQM-UTD thrives under structured driving conditions where traditional rule-based strategies fail to adapt effectively.

6.5 | Chapter summary

This chapter proposed the DQM-UTD algorithm, a novel ensemble-based RL approach for end-to-end EMS in PHEVs. By addressing key challenges such as Q-function overestimation bias and sample inefficiency, the proposed method demonstrates significant improvements in learning performance and real-world applicability. The main conclusions are as follows:

(1) The DQM-UTD algorithm reduces Q-function overestimation bias from 1.18 to 5.2×10^{-4} using an in-target minimization update mechanism. This results in more reliable policy learning and contributes to overall stability during training. Additionally, the algorithm achieves optimal sample efficiency with a UTD ratio of 15, balancing computational cost and learning effectiveness.

(2) In SiL testing, the proposed method achieves a system efficiency of 60.1% with stable convergence within 71 iterations, outperforming conventional Q-learning and double Q-learning approaches in terms of convergence speed and stability.

(3) HiL validation highlights the practical effectiveness of DQM-UTD in real-world driving conditions. The algorithm delivers up to 50.6% energy savings while ensuring superior SoC stability, demonstrating its robustness and adaptability under dynamic operational scenarios.

CHAPTER 7

Conclusion and future work

This thesis has investigated advanced methodologies for energy management in CAVs and multi-mode PHEVs. By leveraging cutting-edge techniques, such as MARL and RL, the research has addressed critical challenges in scalability, energy efficiency, and real-world applicability. This chapter summarizes the key contributions and findings of the thesis, highlighting the impact of the proposed methods. Additionally, it outlines the broader implications of this research and discusses potential directions for future work to further advance energy optimization and intelligent transportation systems.

7.1 | Conclusion

The main conclusions of the PhD research are summarized below:

1) Communication-Efficient MARL for Platoon Stability and Energy-efficiency Co-optimization in Cooperative Adaptive Cruise Control of CAVs

This part presents the **BDC-MARL** approach, designed to enhance system robustness and efficiency by reducing reliance on potentially delayed or unreliable vehicle-to-vehicle communication. By optimizing individual vehicle responses to real-time traffic conditions, BDC-MARL significantly improves energy savings, achieving up to **5.8%** efficiency gains with an average velocity of **15.26 m/s** and an IVS of **20.76 m**. Unlike IA2C and CommNet, which struggle with delayed message transmission in CACC scenarios, BDC-MARL leverages an efficient communication mechanism to mitigate this issue, ensuring stable coordination among vehicles. Integrating **QSGD** further enhances communication efficiency, enabling rapid convergence to a stable IVS while maintaining smooth velocity and acceleration profiles. This optimized bandwidth usage prevents data overflow and minimizes transmission delays,

contributing to more reliable system performance. Additionally, a **6-vehicle** platoon is identified as the optimal configuration for balancing energy efficiency and stability, as it benefits from aerodynamic drag reduction without a proportional increase in power consumption. While BDC-MARL and the **OpenACC-based** scenarios exhibit comparable average speeds, BDC-MARL demonstrates superior IVS stability due to more effective vehicle coordination. Despite similar short-term energy consumption, its robust communication protocols and optimized response strategies significantly enhance vehicle dynamics, reducing fluctuations in platoon behavior and ensuring smoother and more efficient operation.

2) Energy Management of Multi-mode Plug-in Hybrid Electric Vehicle Using Multi-agent Deep Reinforcement Learning

Expanding on the previous MARL-based approach for macro-level CACC in CAVs, this part investigates a **MADRL-based MIMO** control method for the micro-level energy management of a complex **multi-mode PHEV**. Given the multiple operating modes, treating each power source as an independent agent enables more adaptive and cooperative decision-making in power distribution. To facilitate effective collaboration between the two learning agents, a **hand-shaking strategy** is introduced using a **relevance ratio** within the DDPG framework. Through **sensitivity analysis**, **parametric study**, and **SiL testing**, several key findings emerge. The **learning rate** of DDPG agents is identified as the most influential factor, affecting **computational time**, **convergence speed**, and **fuel economy**. The **hand-shaking mechanism** is fine-tuned via the **relevance ratio**, with an optimal value of **0.2** for the studied PHEV. A lower relevance ratio results in slower learning, while a higher value induces instability. Furthermore, the proposed MADRL method surpasses **baseline approaches** (single-agent learning, rule-based control, and ECMS) across all tested driving cycles, achieving fuel savings of up to **23.54%** while maintaining battery SoC errors within **5%**. These results highlight the effectiveness of MADRL in managing the complex energy control challenges of multi-mode PHEVs.

3) Efficient Energy Management of Plug-in Hybrid Electric Vehicles through Random Ensemble with In-target Minimization Q-learning

Building on the previous findings in MADRL-based micro-level EMSs, this study shifts the focus from complex multi-mode PHEVs to a specified operating mode, addressing key challenges in

real-time implementation and control stability. This part presented the **DQM-UTD algorithm** for energy management in PHEVs, addressing critical challenges in **overestimation bias**, **sample efficiency**, and **real-world applicability**. The proposed algorithm significantly reduces overestimation bias from $\mathbf{1.18}$ to $\mathbf{5.2 \times 10^{-4}}$ through **in-target minimization**, ensuring more accurate Q-value updates and enabling robust decision-making. In SiL testing, the algorithm demonstrates remarkable system efficiency, achieving **60.1% efficiency** with stable convergence within **71 iterations**, highlighting its ability to learn quickly and effectively. By optimizing the **UTD ratio** to **15**, DQM-UTD balances computational cost and learning performance, leveraging data efficiently to minimize redundancy while maximizing performance. The HiL validation further confirms the **practical feasibility** of the algorithm, achieving up to **50.6%** energy savings in real-world driving conditions while ensuring improved SOC stability. These results emphasize the potential of DQM-UTD for deployment in PHEVs, offering a scalable and efficient solution for EMSs.

7.2 | Research Impact

The outcomes of this research have significantly contributed to the fields of energy management. The findings have been disseminated and cited through various publications and achievements:

1. Parts of the literature review in Chapter 2 were published in the *International Journal of Automotive Manufacturing and Materials, 2023* [121] and presented on the open-source platform https://github.com/huahuaedi/MARL_in_CAV_control_review, receiving several stars [86]. Additionally, the part of work has been accepted in the top journal *IEEE Transactions on Automation Science and Engineering* (IF = 5.9, JCR Q1).

2. Research from Chapter 4 was published in the top journal *IEEE Transactions on Vehicular Technology* (IF = 6.1, JCR Q1) [281], focusing on MARL-based CACC systems. This work also involved collaboration with a visiting researcher from Southeast University, and the results were published in [296].

3. Research from Chapter 5 was published in the top-tier journal *Applied Energy* (IF = 10.1, JCR Q1) [283]. Findings were also presented at two major conferences: *The*

8th Applied Energy Symposium: Low Carbon Cities and Urban Energy Systems, 2022 [289] and the *International Conference on Applied Energy, 2021* [279]. Furthermore, this research was conducted in collaboration with visiting researchers from Tsinghua University, with results published in [297, 298], and Wuhan University of Technology, with results published in [295, 299, 300].

4. Research from Chapter 6 was published as co-first author in the top journal *IEEE Transactions on Intelligent Vehicles* (IF = 14, JCR Q1) [292], showcasing RL-based ensemble methods in EMSs for PHEVs. Additionally, a part of this work has been submitted to the top journal *IEEE Transactions on Transportation Electrification* (IF = 7.2, JCR Q1, round 1).

5. An undergraduate student under my guidance published their work in the conference *2023 7th CAA International Conference on Vehicular Control and Intelligence (CVCI)* [301].

6. A national silver award in an entrepreneurship competition was won in 2022.

7.3 | Future work

While this research has achieved significant progress, several areas remain for future exploration:

1) BDC-MARL for Long-Term Energy Savings and Adverse Conditions

Future work will extend the analysis of BDC-MARL to longer periods to better evaluate its potential for sustained energy savings. Refining communication protocols to further reduce variability in IVS will also be a focus, along with addressing challenges posed by adverse conditions, such as poor weather and challenging road surfaces.

2) Real-Time Decision-Making and Hardware Integration

The development and validation of low-latency decision-making algorithms and real-time coordination among multiple agents will be pursued. These efforts will be supported by advanced HiL facilities, bridging the gap between simulation-based results and real-world applications. This work will aim to enable commercial deployment of the proposed energy management systems in PHEVs.

3) Integration of Traffic Prediction and Road Condition Information

Future research will integrate traffic prediction and road condition information into the proposed frameworks to enhance adaptability in real-world environments. This integration aims to improve the robustness and efficiency of EMSs, particularly in dynamic and unpredictable driving scenarios.

By addressing these areas, future work will further strengthen the practical utility and scalability of the proposed methods, contributing to the advancement of sustainable and intelligent EMSs.

Reference

1. Sabet, S. & Farooq, B. in *Handbook of Smart Energy Systems* 2003–2023 Springer, 2023.
2. Almassalkhi, M. R. & Kundu, S. "Intelligent Electrification as an enabler of Clean Energy and Decarbonization". *Current Sustainable/Renewable Energy Reports* **10,** 183–196 2023.
3. Zhao, J., Chen, Y. & Gong, Y. *Study of connectivity probability of vehicle-to-vehicle and vehicle-to-infrastructure communication systems.* in *2016 IEEE 83rd vehicular technology conference (VTC Spring)* 2016, 1–4.
4. Connor, W. D., Wang, Y., Malikopoulos, A. A., Advani, S. G. & Prasad, A. K. "Impact of connectivity on energy consumption and battery life for electric vehicles". *IEEE Transactions on Intelligent Vehicles* **6,** 14–23 2020.
5. Market.us, *Connected Vehicles Market Size By Communication*. `https://market.us/report/connected-vehicles-market/` 2024.
6. Michel, P., Karbowski, D. & Rousseau, A. *Impact of connectivity and automation on vehicle energy use.* Tech. rep. SAE Technical Paper, 2016.
7. GAMBOA-CRUZADO, J., PAYI-QUISPE, A., ALZAMORA, C., RIVERO, A. D.-C., KONG, M. R. & VALENZUELA, J. N. "5G TECHNOLOGY AND ITS IMPACT ON THE USE OF ONLINE VIDEOGAMES: A COMPREHENSIVE SYSTEMATIC". *Journal of Theoretical and Applied Information Technology* **101** 2023.
8. Shichun, Y., Zheng, Z., Bin, M., Yifan, Z., Sida, Z., Mingyan, L., Yu, L., Qiangwei, L., Xinan, Z., Mengyue, Z., Yang, H., Fei, C. & Yaoguang, C., "Essential Technics of Cybersecurity for Intelligent Connected Vehicles: Comprehensive Review and Perspective". *IEEE Internet of Things Journal* **10**, 21787–21810 2023.

9. Zhao, L. & Malikopoulos, A. A. "Enhanced mobility with connectivity and automation: A review of shared autonomous vehicle systems". *IEEE Intelligent Transportation Systems Magazine* **14,** 87–102 2020.

10. GHD, *Connected and Autonomous Vehicles: Crossroads in Transport Evolution*. Accessed: 2024-01-20. GHD. https://www.ghd.com/en-gb/crossroads.

11. Garikapati, D. & Shetiya, S. S. "Autonomous Vehicles: Evolution of Artificial Intelligence and the Current Industry Landscape". *Big Data and Cognitive Computing* **8,** 42 2024.

12. Department, S. R., *Connected Cars - Statistics & Facts*. Statista. https://www.statista.com/topics/1918/connected-cars/ 2024.

13. Schiegg, F. A. "Vehicular networks: a value of information driven distributed resource allocation protocol (VALINDRA)" 2022.

14. Hernández, R. H. "Economic liberalization and trade relations between Mexico and China". *Journal of Current Chinese Affairs* **41,** 49–96 2012.

15. Heineke, K., Laverty, N., Möller, T. & Ziegler, F. *The Future of Mobility*. https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/the-future-of-mobility-mobility-evolves. McKinsey Quarterly, April 19, 2023. Apr. 2023.

16. Elassy, M., Al-Hattab, M., Takruri, M. & Badawi, S. "Intelligent transportation systems for sustainable smart cities". *Transportation Engineering,* 100252 2024.

17. Cao, D., Wang, X., Li, L., Lv, C., Na, X., Xing, Y., Li, X., Li, Y., Chen, Y. & Wang, F.-Y. "Future directions of intelligent vehicles: Potentials, possibilities, and perspectives". *IEEE Transactions on Intelligent Vehicles* **7,** 7–10 2022.

18. Vermesan, O., John, R., Pype, P., Daalderop, G., Kriegel, K., Mitic, G., Lorentz, V., Bahr, R., Sand, H. E., Bockrath, S., *et al.* "Automotive intelligence embedded in electric connected autonomous and shared vehicles technology for sustainable green mobility". *Frontiers in Future Transportation* **2,** 688482 2021.

19. World Health Organization, *Global Status Report on Road Safety 2018*. Accessed: 2024-11-30. Geneva, Switzerland, World Health Organization, 2018.

20. Intergovernmental Panel on Climate Change, *Climate Change 2021: The Physical Science Basis*. Part of the Sixth Assessment Report of the IPCC. Geneva, Switzerland, IPCC, 2021.

21. Liu, B., Han, C., Liu, X. & Li, W. "Vehicle artificial intelligence system based on intelligent image analysis and 5G network". *International Journal of Wireless Information Networks* **30,** 86–102 2023.

22. Feng, S., Yan, X., Sun, H., Feng, Y. & Liu, H. X. "Intelligent driving intelligence test for autonomous vehicles with naturalistic and adversarial environment". *Nature communications* **12,** 748 2021.

23. Li, J., Cheng, H., Guo, H. & Qiu, S. "Survey on artificial intelligence for vehicles". *Automotive Innovation* **1,** 2–14 2018.

24. Fu, Y., Li, C., Yu, F. R., Luan, T. H. & Zhang, Y. "A survey of driving safety with sensing, vehicular communications, and artificial intelligence-based collision avoidance". *IEEE transactions on intelligent transportation systems* **23,** 6142–6163 2021.

25. Future, M. R., *Vehicle Intelligence System Market Research Report*. https://www.marketresearchfuture.com/reports/vehicle-intelligence-system-market-2447 2024.

26. Yang, Z. & Bandivadekar, A. "2017 Global update: Light-duty vehicle greenhouse gas and fuel economy standards" 2017.

27. Kouridis, C. & Vlachokostas, C. "Towards decarbonizing road transport: Environmental and social benefit of vehicle fleet electrification in urban areas of Greece". *Renewable and Sustainable Energy Reviews* **153,** 111775 2022.

28. Sacchi, R., Bauer, C., Cox, B. & Mutel, C. "When, where and how can the electrification of passenger cars reduce greenhouse gas emissions?" *Renewable and Sustainable Energy Reviews* **162,** 112475 2022.

29. IEA, *Global EV Outlook 2024*. Tech. rep.. International Energy Agency, Paris, 2024.

30. Corradi, C., Sica, E. & Morone, P. "What drives electric vehicle adoption? Insights from a systematic review on European transport actors and behaviours". *Energy Research & Social Science* **95,** 102908 2023.

31. Rottoli, M., Dirnaichner, A., Pietzcker, R., Schreyer, F. & Luderer, G. "Alternative electrification pathways for light-duty vehicles in the European transport sector". *Transportation Research Part D: Transport and Environment* **99,** 103005 2021.

32. Mohan, A., Sripad, S., Vaishnav, P. & Viswanathan, V. "Trade-offs between automation and light vehicle electrification". *Nature Energy* **5,** 543–549 2020.

33. Yuan, M., Thellufsen, J. Z., Lund, H. & Liang, Y. "The electrification of transportation in energy transition". *Energy* **236,** 121564 2021.

34. Rezaei, H., Abdollahi, S. E., Abdollahi, S. & Filizadeh, S. "Energy management strategies of battery-ultracapacitor hybrid storage systems for electric vehicles: Review, challenges, and future trends". *Journal of Energy Storage* **53,** 105045 2022.

35. Girade, P., Shah, H., Kaushik, K., Patheria, A. & Xu, B. "Comparative analysis of state of charge based adaptive supervisory control strategies of plug-in Hybrid Electric Vehicles". *Energy* **230,** 120856 2021.

36. Dogdu, M. F. & Reyhancan, I. A. "The comparison of gasoline powered vehicle and serial hybrid vehicle on emissions". *Heliyon* **10** 2024.

37. Talele, V., Thorat, P., Gokhale, Y. P. & Desai, H. "Technical review on battery thermal management system for electric vehicle application". *Energy Storage Systems,* 177–225 2023.

38. Chu, W., Wuniri, Q., Du, X., Xiong, Q., Huang, T. & Li, K. "Cloud control system architectures, technologies and applications on intelligent and connected vehicles: a review". *Chinese Journal of Mechanical Engineering* **34,** 139 2021.

39. Li, Y., Tang, X., Lin, X., Grzesiak, L. & Hu, X. "The role and application of convex modeling and optimization in electrified vehicles". *Renewable and Sustainable Energy Reviews* **153,** 111796 2022.

40. Yi, C., Hofmann, H. & Epureanu, B. I. "Energy efficient platooning of connected electrified vehicles enabled by a mixed hybrid electric powertrain architecture". *IEEE Transactions on Intelligent Transportation Systems* **23,** 20383–20397 2022.

41. Zhuang, W., Li, S., Zhang, X., Kum, D., Song, Z., Yin, G. & Ju, F. "A survey of powertrain configuration studies on hybrid electric vehicles". *Applied Energy* **262,** 114553 2020.

42. Zhao, Z., Tang, P. & Li, H. "Generation, screening, and optimization of powertrain configurations for power-split hybrid electric vehicle: A comprehensive overview". *IEEE Transactions on Transportation Electrification* **8,** 325–344 2021.

43. Vu, T. M., Moezzi, R., Cyrus, J., Hlava, J. & Petru, M. "Parallel hybrid electric vehicle modelling and model predictive control". *Applied Sciences* **11,** 10668 2021.

44. Hofman, T., Ebbesen, S. & Guzzella, L. "Topology optimization for hybrid electric vehicles with automated transmissions". *IEEE Transactions on Vehicular Technology* **61,** 2442–2451 2012.

45. Anselma, P. G., Biswas, A., Belingardi, G. & Emadi, A. "Rapid assessment of the fuel economy capability of parallel and series-parallel hybrid electric vehicles". *Applied Energy* **275,** 115319 2020.

46. Dong, P., Zhao, J., Xu, X., Liu, Y., Wang, S., Huang, H., Wang, R., Zheng, L. & Zhou, Z. "Performance comparison of series–parallel hybrid transmissions with multiple gears and modes based on efficiency model". *Energy Conversion and Management* **274,** 116442 2022.

47. Xu, X., Zhao, J., Zhao, J., Shi, K., Dong, P., Wang, S., Liu, Y., Guo, W. & Liu, X. "Comparative study on fuel saving potential of series-parallel hybrid transmission and series hybrid transmission". *Energy Conversion and Management* **252,** 114970 2022.

48. Seyfi, M., Alinaghian, M., Ghorbani, E., Çatay, B. & Sabbagh, M. S. "Multi-mode hybrid electric vehicle routing problem". *Transportation Research Part E: Logistics and Transportation Review* **166,** 102882 2022.

49. Zhang, L., Liu, W. & Qi, B. "Energy optimization of multi-mode coupling drive plug-in hybrid electric vehicles based on speed prediction". *Energy* **206,** 118126 2020.

50. Zhang, Y., Zhao, H., Huang, K., Qiu, M. & Geng, L. "Hybrid optimization and its applications for multi-mode plug-in hybrid electric vehicle". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* **234,** 228–244 2020.

51. Milovanoff, A., Posen, I. D. & MacLean, H. L. "Electrification of light-duty vehicle fleet alone will not meet mitigation targets". *Nature Climate Change* **10,** 1102–1107 2020.

52. Han, J., Sciarretta, A., Ojeda, L. L., De Nunzio, G. & Thibault, L. "Safe-and eco-driving control for connected and automated electric vehicles using analytical state-constrained optimal solution". *IEEE Transactions on Intelligent Vehicles* **3,** 163–172 2018.

53. Rios-Torres, J. & Malikopoulos, A. A. "Impact of partial penetrations of connected and automated vehicles on fuel consumption and traffic flow". *IEEE Transactions on Intelligent Vehicles* **3,** 453–462 2018.

54. Wang, H., Hao, W., So, J., Xiao, X., Chen, Z. & Hu, J. "A Faster Cooperative Lane Change Controller Enabled by Formulating in Spatial Domain". *IEEE Transactions on Intelligent Vehicles* 2023.

55. Liu, W., Hua, M., Deng, Z., Meng, Z., Huang, Y., Hu, C., Song, S., Gao, L., Liu, C., Shuai, B., *et al.* "A systematic survey of control techniques and applications in connected and automated vehicles". *IEEE Internet of Things Journal* 2023.

56. Boddupalli, S., Rao, A. S. & Ray, S. "Resilient cooperative adaptive cruise control for autonomous vehicles using machine learning". *IEEE Transactions on Intelligent Transportation Systems* **23,** 15655–15672 2022.

57. Dai, C., Zong, C., Zhang, D., Hua, M., Zheng, H. & Chuyo, K. "A Bargaining Game-Based Human–Machine Shared Driving Control Authority Allocation Strategy". *IEEE Transactions on Intelligent Transportation Systems* 2023.

58. Tsugawa, S., Jeschke, S. & Shladover, S. E. "A review of truck platooning projects for energy savings". *IEEE Transactions on Intelligent Vehicles* **1,** 68–77 2016.

59. Liu, B., Ding, Z. & Lv, C. "Platoon control of connected autonomous vehicles: A distributed reinforcement learning method by consensus". *IFAC-PapersOnLine* **53,** 15241–15246 2020.

60. Ding, J., Li, L., Peng, H. & Zhang, Y. "A rule-based cooperative merging strategy for connected and automated vehicles". *IEEE Transactions on Intelligent Transportation Systems* **21,** 3436–3446 2019.

61. Makridis, M. A., Schaniel, J. & Kouvelas, A. "Rule-based on-off traffic control strategy for cavs on motorway networks: Assessing cooperation level and driving homogeneity". *IEEE Access* **11,** 35111–35121 2023.

62. Gao, M. & Zhou, M. *Control strategy selection for autonomous vehicles in a dynamic environment.* in *2005 IEEE International Conference on Systems, Man and Cybernetics* **2** 2005, 1651–1656.

63. Wang, J., Jiang, Z. & Pant, Y. V. "Improving safety in mixed traffic: A learning-based model predictive control for autonomous and human-driven vehicle platooning". *Knowledge-Based Systems,* 111673 2024.

64. Yang, F., Wang, H., Pi, D., Sun, X. & Wang, X. "Research on collaborative adaptive cruise control based on MPC and improved spacing policy". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering,* 09544070241240166 2024.

65. Mosharafian, S. & Velni, J. M. "A hybrid stochastic model predictive design approach for cooperative adaptive cruise control in connected vehicle applications". *Control Engineering Practice* **130,** 105383 2023.

66. Zhong, Z., Lee, J. & Zhao, L. "Multiobjective optimization framework for cooperative adaptive cruise control vehicles in the automated vehicle platooning environment". *Transportation Research Record* **2625,** 32–42 2017.

67. Zhao, S. & Zhang, K. "A distributionally robust stochastic optimization-based model predictive control with distributionally robust chance constraints for cooperative adaptive cruise control under uncertain traffic conditions". *Transportation Research Part B: Methodological* **138,** 144–178 2020.

68. Park, S., Rakha, H. A., Ahn, K., Moran, K., Saerens, B. & Van den Bulck, E. "Predictive ecocruise control system: model logic and preliminary testing". *Transportation research record* **2270,** 113–123 2012.

69. Ma, F., Yang, Y., Wang, J., Liu, Z., Li, J., Nie, J., Shen, Y. & Wu, L. "Predictive energy-saving optimization based on nonlinear model predictive control for cooperative connected vehicles platoon with V2V communication". *Energy* **189,** 116120 2019.

70. Wang, Z., Chen, X., Ouyang, Y. & Li, M. "Emission mitigation via longitudinal control of intelligent vehicles in a congested platoon". *Computer-Aided Civil and Infrastructure Engineering* **30,** 490–506 2015.

71. Kazemi, H., Mahjoub, H. N., Tahmasbi-Sarvestani, A. & Fallah, Y. P. "A Learning-Based Stochastic MPC Design for Cooperative Adaptive Cruise Control to Handle Interfering Vehicles". *IEEE Transactions on Intelligent Vehicles* **3,** 266–275 2018.

72. Chu, T. & Kalabić, U. *Model-based deep reinforcement learning for CACC in mixed-autonomy vehicle platoon.* in *2019 IEEE 58th Conference on Decision and Control (CDC)* 2019, 4079–4084.

73. Tian, B., Wang, G., Xu, Z., Zhang, Y. & Zhao, X. "Communication delay compensation for string stability of CACC system using LSTM prediction". *Vehicular Communications* **29,** 100333 2021.

74. Hao, W., Liu, L., Yang, X., Li, Y. & Byon, Y.-J. "Reducing CACC platoon disturbances caused by state jitters by combining two stages driving state recognition with multiple platoons' strategies and risk prediction". *IEEE Transactions on Intelligent Transportation Systems* **23,** 2144–2154 2020.

75. Lin, Y., McPhee, J. & Azad, N. L. "Comparison of deep reinforcement learning and model predictive control for adaptive cruise control". *IEEE Transactions on Intelligent Vehicles* **6,** 221–231 2020.

76. Neto, A. A. & Mozelli, L. A. "Robust Longitudinal Control for Vehicular Autonomous Platoons Using Deep Reinforcement Learning". *arXiv preprint arXiv:2206.01175* 2022.

77. Kanipakam, L., Sakr, A. H., Avedisov, S. S. & Moradipari, A., *Cooperative Adaptive Cruise Control Based on Intent Sharing Messages and Reinforcement Learning*. *2024 IEEE Vehicular Networking Conference (VNC)*. 2024, 188–194.

78. Ke, H., Mozaffari, S., Alirezaee, S. & Saif, M. *Cooperative adaptive cruise control using vehicle-to-vehicle communication and deep learning.* in *2022 IEEE Intelligent Vehicles Symposium (IV)* 2022, 435–440.

79. Liu, T., Lei, L., Zheng, K. & Shen, X. "Multi-Timescale Control and Communications With Deep Reinforcement Learning—Part I: Communication-Aware Vehicle Control". *IEEE Internet of Things Journal* 2023.

80. Fu, Y., Li, C., Yu, F. R., Luan, T. H. & Zhang, Y. "Hybrid Autonomous Driving Guidance Strategy Combining Deep Reinforcement Learning and Expert System". *IEEE Transactions on Intelligent Transportation Systems* **23,** 11273–11286 2022.

81. Albeaik, S., Wu, T., Vurimi, G., Chou, F.-C., Lu, X.-Y. & Bayen, A. M. "Longitudinal deep truck: Deep longitudinal model with application to sim2real deep reinforcement learning for heavy-duty truck control in the field". *Journal of Field Robotics* **40,** 306–329 2023.

82. Wang, J. & Sun, L. "Robust Dynamic Bus Control: a Distributional Multi-Agent Reinforcement Learning Approach". *IEEE Transactions on Intelligent Transportation Systems* **24,** 4075–4088 2023.

83. Chellapandi, V. P., Yuan, L., Brinton, C. G., Żak, S. H. & Wang, Z. "Federated learning for connected and automated vehicles: A survey of existing approaches and challenges". *IEEE Transactions on Intelligent Vehicles* 2023.

84. Parvini, M., González, A., Villamil, A., Schulz, P. & Fettweis, G. *Joint resource allocation and string-stable CACC design with multi-agent reinforcement learning.* in *ICC 2023-IEEE International Conference on Communications* 2023, 1232–1237.

85. Chen, G., Gao, Z., Hua, M., Shuai, B. & Gao, Z. "Lane change trajectory prediction considering driving style uncertainty for autonomous vehicles". *Mechanical Systems and Signal Processing* **206,** 110854 2024.

86. Hua, M., Chen, D., Qi, X., Jiang, K., Liu, Z. E., Zhou, Q. & Xu, H. "Multi-agent reinforcement learning for connected and automated vehicles control: Recent advancements and future prospects". *arXiv preprint arXiv:2312.11084* 2023.

87. Han, S., Zhou, S., Wang, J., Pepin, L., Ding, C., Fu, J. & Miao, F. "A Multi-Agent Reinforcement Learning Approach for Safe and Efficient Behavior Planning of Connected Autonomous Vehicles". *IEEE Transactions on Intelligent Transportation Systems* 2023.

88. Zhang, H., Peng, J., Dong, H., Ding, F. & Tan, H. "Integrated velocity optimization and energy management strategy for hybrid electric vehicle platoon: A multi-agent reinforcement learning approach". *IEEE Transactions on Transportation Electrification* 2023.

89. Zhang, W., Bastani, O. & Kumar, V. "Mamps: Safe multi-agent reinforcement learning via model predictive shielding". *arXiv preprint arXiv:1910.12639* 2019.

90. Du, Y., Ma, C., Liu, Y., Lin, R., Dong, H., Wang, J. & Yang, Y. *Scalable model-based policy optimization for decentralized networked systems.* in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* 2022, 9019–9026.

91. Chen, J., Wu, X., Lv, Z., Xu, Z. & Wang, W. "Collaborative Control of Vehicle Platoon Based on Deep Reinforcement Learning". *IEEE Transactions on Vehicular Technology* **73,** 14399–14414 2024.

92. Gangopadhyay, B., Dasgupta, P. & Dey, S. *Counterexample-Guided Policy Refinement in Multi-Agent Reinforcement Learning.* in *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems* 2023, 1606–1614.

93. Tung, T.-Y., Kobus, S., Roig, J. P. & Gündüz, D. "Effective communications: A joint learning and communication framework for multi-agent reinforcement learning over noisy channels". *IEEE Journal on Selected Areas in Communications* **39,** 2590–2603 2021.

94. Peake, A., McCalmon, J., Raiford, B., Liu, T. & Alqahtani, S. *Multi-agent reinforcement learning for cooperative adaptive cruise control.* in *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)* 2020, 15–22.

95. Song, L. & Fan, W. ". "Performance of State-Shared Multiagent Deep Reinforcement Learning Controlled Signal Corridor with Platooning-Based CAVs". *Journal of Transportation Engineering, Part A: Systems* **149,** 04023072 2023.

96. Chen, D., Zhang, K., Wang, Y., Yin, X., Li, Z. & Filev, D. "Communication-Efficient Decentralized Multi-Agent Reinforcement Learning for Cooperative Adaptive Cruise Control". *IEEE Transactions on Intelligent Vehicles* 2024.

97. Li, M., Cao, Z. & Li, Z. "A reinforcement learning-based vehicle platoon control strategy for reducing energy consumption in traffic oscillations". *IEEE Transactions on Neural Networks and Learning Systems* **32,** 5309–5322 2021.

98. Chu, T., Chinchali, S. & Katti, S., *Multi-agent Reinforcement Learning for Networked System Control*. *International Conference on Learning Representations*. 2020.

99. Liu, B., Han, W., Wang, E., Ma, X., Xiong, S., Qiao, C. & Wang, J. *An efficient message dissemination scheme for cooperative drivings via multi-agent hierarchical attention reinforcement learning.* in *2021 IEEE 41st International Conference on Distributed Computing Systems (ICDCS)* 2021, 326–336.

100. Li, B., Xie, K., Huang, X., Wu, Y. & Xie, S. "Deep reinforcement learning based incentive mechanism design for platoon autonomous driving with social effect". *IEEE Transactions on Vehicular Technology* **71,** 7719–7729 2022.

101. Zohdy, I. H. & Rakha, H. *Game theory algorithm for intersection-based cooperative adaptive cruise control (CACC) systems.* in *2012 15th International IEEE Conference on Intelligent Transportation Systems* 2012, 1097–1102.

102. Fu, M., Li, S., Guo, M., Yang, Z., Sun, Y., Qiu, C., Wang, X. & Li, X. "Cooperative decision-making of multiple autonomous vehicles in a connected mixed traffic environment: A coalition game-based model". *Transportation research part C: emerging technologies* **157,** 104415 2023.

103. Xu, Y., Zhu, K., Xu, H. & Ji, J. "Deep Reinforcement Learning for Multi-Objective Resource Allocation in Multi-Platoon Cooperative Vehicular Networks". *IEEE Transactions on Wireless Communications* 2023.

104. Parvini, M., Javan, M. R., Mokari, N., Abbasi, B. & Jorswieck, E. A. "AoI-aware resource allocation for platoon-based C-V2X networks via multi-agent multi-task reinforcement learning". *IEEE Transactions on Vehicular Technology* 2023.

105. Wang, Z., Xue, Y., Liu, L., Zhang, H., Qu, C. & Fang, C. "Multi-Agent DRL-Controlled Connected and Automated Vehicles in Mixed Traffic With Time Delays". *IEEE Transactions on Intelligent Transportation Systems* 2024.

106. Wang, Z., Wu, G. & Barth, M. J. "Cooperative eco-driving at signalized intersections in a partially connected and automated vehicle environment". *IEEE Transactions on Intelligent Transportation Systems* **21,** 2029–2038 2019.

107. Lu, S., Cai, Y., Chen, L., Wang, H., Sun, X. & Gao, H. "Altruistic cooperative adaptive cruise control of mixed traffic platoon based on deep reinforcement learning". *IET Intelligent Transport Systems* 2023.

108. Shi, H., Chen, D., Zheng, N., Wang, X., Zhou, Y. & Ran, B. "A deep reinforcement learning based distributed control strategy for connected automated vehicles in mixed traffic platoon". *Transportation Research Part C: Emerging Technologies* **148,** 104019 2023.

109. Zhang, K., Yang, Z. & Başar, T. "Decentralized multi-agent reinforcement learning with networked agents: Recent advances". *Frontiers of Information Technology & Electronic Engineering* **22,** 802–814 2021.

110. Petrillo, A., Pescape, A. & Santini, S. "A secure adaptive control for cooperative driving of autonomous connected vehicles in the presence of heterogeneous communication delays and cyberattacks". *IEEE transactions on cybernetics* **51,** 1134–1149 2020.

111. Xie, S., Hu, J., Ding, Z. & Arvin, F. "Cooperative Adaptive Cruise Control for Connected Autonomous Vehicles Using Spring Damping Energy Model". *IEEE Transactions on Vehicular Technology* **72,** 2974–2987 2023.

112. Long, H., Khalatbarisoltani, A., Yang, Y. & Hu, X. "Hierarchical Control Strategies for Connected Heavy-Duty Modular Fuel Cell Vehicles via Decentralized Convex Optimization". *IEEE Transactions on Vehicular Technology* 2023.

113. Foerster, J., Assael, I. A., De Freitas, N. & Whiteson, S. "Learning to communicate with deep multi-agent reinforcement learning". *Advances in neural information processing systems* **29** 2016.

114. Kuutti, S., Bowden, R., Jin, Y., Barber, P. & Fallah, S. "A survey of deep learning applications to autonomous vehicle control". *IEEE Transactions on Intelligent Transportation Systems* **22,** 712–733 2020.

115. Zhang, Y., Xu, Z., Wang, Z., Yao, X. & Xu, Z. "Impacts of communication delay on vehicle platoon string stability and its compensation strategy: A review". *Journal of traffic and transportation engineering (English edition)* 2023.

116. Gronauer, S. & Diepold, K. "Multi-agent deep reinforcement learning: a survey". *Artificial Intelligence Review,* 1–49 2022.

117. Hofman, T., Steinbuch, M., Van Druten, R. & Serrarens, A. "Rule-based energy management strategies for hybrid vehicles". *International Journal of Electric and Hybrid Vehicles* **1,** 71–94 2007.

118. Yuan, H.-B., Zou, W.-J., Jung, S. & Kim, Y.-B. "Optimized rule-based energy management for a polymer electrolyte membrane fuel cell/battery hybrid power system using a genetic algorithm". *International Journal of Hydrogen Energy* **47,** 7932–7948 2022.

119. Serpi, A. & Porru, M. *An MPC-based energy management system for a hybrid electric vehicle.* in *2020 IEEE Vehicle Power and Propulsion Conference (VPPC)* 2020, 1–6.

120. Zhang, F., Hu, X., Langari, R. & Cao, D. "Energy management strategies of connected HEVs and PHEVs: Recent progress and outlook". *Progress in Energy and Combustion Science* **73,** 235–256 2019.

121. Hua, M., Shuai, B., Zhou, Q., Wang, J., He, Y. & Xu, H. "Recent Progress in Energy Management of Connected Hybrid Electric Vehicles Using Reinforcement Learning". *International Journal of Automotive Manufacturing and Materials* **2,** 6–22 2023.

122. Wu, J., He, H., Peng, J., Li, Y. & Li, Z. "Continuous reinforcement learning of energy management with deep Q network for a power split hybrid electric bus". *Applied energy* **222,** 799–811 2018.

123. Qi, X., Luo, Y., Wu, G., Boriboonsomsin, K. & Barth, M. J. *Deep reinforcement learning-based vehicle energy efficiency autonomous learning system.* in *2017 IEEE Intelligent Vehicles Symposium (IV)* 2017, 1228–1233.

124. Qi, X., Luo, Y., Wu, G., Boriboonsomsin, K. & Barth, M. "Deep reinforcement learning enabled self-learning control for energy efficient driving". *Transportation Research Part C: Emerging Technologies* **99,** 67–81 2019.

125. Hu, Y., Li, W., Xu, K., Zahid, T., Qin, F. & Li, C. "Energy management strategy for a hybrid electric vehicle based on deep reinforcement learning". *Applied Sciences* **8,** 187 2018.

126. Han, X., He, H., Wu, J., Peng, J. & Li, Y. "Energy management based on reinforcement learning with double deep Q-learning for a hybrid electric tracked vehicle". *Applied Energy* **254,** 113708 2019.

127. He, D., Zou, Y., Wu, J., Zhang, X., Zhang, Z. & Wang, R. *Deep Q-learning based energy management strategy for a series hybrid electric tracked vehicle and its adaptability validation.* in *2019 IEEE Transportation Electrification Conference and Expo (ITEC)* 2019, 1–6.

128. Du, G., Zou, Y., Zhang, X., Kong, Z., Wu, J. & He, D. "Intelligent energy management for hybrid electric tracked vehicles using online reinforcement learning". *Applied Energy* **251,** 113388 2019.

129. Du, G., Zou, Y., Zhang, X., Liu, T., Wu, J. & He, D. "Deep reinforcement learning based energy management for a hybrid electric vehicle". *Energy* **201,** 117591 2020.

130. Du, G., Zou, Y., Zhang, X., Guo, L. & Guo, N. "Heuristic energy management strategy of hybrid electric vehicle based on deep reinforcement learning with accelerated gradient optimization". *IEEE Transactions on Transportation Electrification* **7,** 2194–2208 2021.

131. Du, G., Zou, Y., Zhang, X., Guo, L. & Guo, N. "Energy management for a hybrid electric vehicle based on prioritized deep reinforcement learning framework". *Energy* **241,** 122523 2022.

132. Zhang, H., Fan, Q., Liu, S., Li, S. E., Huang, J. & Wang, Z. "Hierarchical energy management strategy for plug-in hybrid electric powertrain integrated with dual-mode combustion engine". *Applied Energy* **304,** 117869 2021.

133. Li, Y., He, H., Peng, J. & Zhang, H. "Power management for a plug-in hybrid electric vehicle based on reinforcement learning with continuous state and action spaces". *Energy Procedia* **142,** 2270–2275 2017.

134. Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. "Proximal policy optimization algorithms". *arXiv preprint arXiv:1707.06347* 2017.

135. Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D. & Wierstra, D. "Continuous control with deep reinforcement learning". *arXiv preprint arXiv:1509.02971* 2015.

136. Zhang, J., Tao, J., Hu, Y. & Ma, L. "An energy management strategy based on DDPG with improved exploration for battery/supercapacitor hybrid electric vehicle". *IEEE Transactions on Intelligent Transportation Systems* 2023.

137. Wu, J., Wei, Z., Liu, K., Quan, Z. & Li, Y. "Battery-involved energy management for hybrid electric bus based on expert-assistance deep deterministic policy gradient algorithm". *IEEE Transactions on Vehicular Technology* **69,** 12786–12796 2020.

138. Zhang, B., Zou, Y., Zhang, X., Du, G., Jiao, F. & Guo, N. "Online updating energy management strategy based on deep reinforcement learning with accelerated training for hybrid electric tracked vehicles". *IEEE Transactions on Transportation Electrification* **8,** 3289–3306 2022.

139. Li, Y., He, H., Peng, J. & Wang, H. "Deep reinforcement learning-based energy management for a series hybrid electric vehicle enabled by history cumulative trip information". *IEEE Transactions on Vehicular Technology* **68,** 7416–7430 2019.

140. Hu, B. & Li, J. "An adaptive hierarchical energy management strategy for hybrid electric vehicles combining heuristic domain knowledge and data-driven deep reinforcement learning". *IEEE Transactions on Transportation Electrification* **8,** 3275–3288 2021.

141. Chen, H., Tang, X., Hu, G. & Liu, T. "Human-like energy management based on deep reinforcement learning and historical driving experiences". *arXiv preprint arXiv:2007.10126* 2020.

142. Guo, X., Liu, T., Tang, B., Tang, X., Zhang, J., Tan, W. & Jin, S. "Transfer deep reinforcement learning-enabled energy management strategy for hybrid tracked vehicle". *IEEE Access* **8,** 165837–165848 2020.

143. Lian, R., Tan, H., Peng, J., Li, Q. & Wu, Y. "Cross-type transfer for deep reinforcement learning based hybrid electric vehicle energy management". *IEEE Transactions on Vehicular Technology* **69,** 8367–8380 2020.

144. Hu, B. & Li, J. "An Adaptive Hierarchical Energy Management Strategy for Hybrid Electric Vehicles Combining Heuristic Domain Knowledge and Data-Driven Deep Reinforcement Learning". *IEEE Transactions on Transportation Electrification* **8,** 3275–3288 2022.

145. Fujimoto, S., Hoof, H. & Meger, D. *Addressing function approximation error in actor-critic methods.* in *International conference on machine learning* 2018, 1587–1596.

146. Zhou, J., Xue, S., Xue, Y., Liao, Y., Liu, J. & Zhao, W. "A novel energy management strategy of hybrid electric vehicle via an improved TD3 deep reinforcement learning". *Energy* **224,** 120118 2021.

147. Huang, R., He, H., Zhao, X., Wang, Y. & Li, M. "Battery health-aware and naturalistic data-driven energy management for hybrid electric bus based on TD3 deep reinforcement learning algorithm". *Applied Energy* **321,** 119353 2022.

148. Liu, Z. E., 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* **54,** 546–551 2021.

149. Niu, Z., Huang, R., He, H., Zhou, Z. & Su, Q. *Energy Management Strategy Based on an Improved TD3 Reinforcement Algorithm with Novel Experience Replay.* in *2023 IEEE Vehicle Power and Propulsion Conference (VPPC)* 2023, 1–5.

150. Tan, H., Zhang, H., Peng, J., Jiang, Z. & Wu, Y. "Energy management of hybrid electric bus based on deep reinforcement learning in continuous state and action space". *Energy Conversion and Management* **195,** 548–560 2019.

151. Li, Y., He, H., Khajepour, A., Wang, H. & Peng, J. "Energy management for a power-split hybrid electric bus via deep reinforcement learning with terrain information". *Applied Energy* **255,** 113762 2019.

152. 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* **8,** 1376–1388 2021.

153. Sewak, M. & Sewak, M. "Actor-critic models and the A3C: The asynchronous advantage actor-critic model". *Deep reinforcement learning: frontiers of artificial intelligence,* 141–152 2019.

154. Zhang, Z., Luo, X., Liu, T., Xie, S., Wang, J., Wang, W., Li, Y. & Peng, Y. *Proximal policy optimization with mixed distributed training.* in *2019 IEEE 31st international conference on tools with artificial intelligence (ICTAI)* 2019, 1452–1456.

155. Zhou, J., Xue, Y., Xu, D., Li, C. & Zhao, W. "Self-learning energy management strategy for hybrid electric vehicle via curiosity-inspired asynchronous deep reinforcement learning". *Energy* **242,** 122548 2022.

156. Biswas, A., Anselma, P. G. & Emadi, A. "Real-time optimal energy management of multimode hybrid electric powertrain with online trainable asynchronous advantage actor–critic algorithm". *IEEE Transactions on Transportation Electrification* **8,** 2676–2694 2021.

157. Huang, R., He, H. & Gao, M. "Training-efficient and cost-optimal energy management for fuel cell hybrid electric bus based on a novel distributed deep reinforcement learning framework". *Applied Energy* **346,** 121358 2023.

158. Tang, X., Chen, J., Liu, T., Qin, Y. & Cao, D. "Distributed deep reinforcement learning-based energy and emission management strategy for hybrid electric vehicles". *IEEE Transactions on Vehicular Technology* **70,** 9922–9934 2021.

159. Li, Y., He, H., Khajepour, A., Chen, Y., Huo, W. & Wang, H. "Deep reinforcement learning for intelligent energy management systems of hybrid-electric powertrains: Recent advances, open issues, and prospects". *IEEE Transactions on Transportation Electrification,* 1–1 2024.

160. Hua, M., Chen, G., Zhang, B. & Huang, Y. "A hierarchical energy efficiency optimization control strategy for distributed drive electric vehicles". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* **233,** 605–621 2019.

161. Kong, W., Cai, T., Luo, Y., Lian, X. & Jiang, F. "Energy Management Optimization Based on Aging Adaptive Functional State Model of Battery for Internal Combustion Engine Vehicles". *Automotive Innovation* **6,** 62–75 2023.

162. Maamria, D., Gillet, K., Colin, G., Chamaillard, Y. & Nouillant, C. "Computation of eco-driving cycles for Hybrid Electric Vehicles: Comparative analysis". *Control Engineering Practice* **71,** 44–52 2018.

163. Chen, G., Hua, M., Zong, C., Zhang, B. & Huang, Y. "Comprehensive chassis control strategy of FWIC-EV based on sliding mode control". *IET Intelligent Transport Systems* **13,** 703–713 2019.

164. Zhou, Q., Li, Y., Zhao, D., Li, J., Williams, H., Xu, H. & Yan, F. "Transferable representation modelling for real-time energy management of the plug-in hybrid vehicle

based on k-fold fuzzy learning and Gaussian process regression". *Applied energy* **305,** 117853 2022.

165. Jendoubi, I. & Bouffard, F. "Multi-agent hierarchical reinforcement learning for energy management". *Applied Energy* **332,** 120500 2023.

166. 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* **332,** 120563 2023.

167. Deng, P., Wu, X., Yang, J., Yang, G., Jiang, P., Yang, J. & Bian, X. "Optimal online energy management strategy of a fuel cell hybrid bus via reinforcement learning". *Energy Conversion and Management* **300,** 117921 2024.

168. Xu, B., Rathod, D., Zhang, D., Yebi, A., Zhang, X., Li, X. & Filipi, Z. "Parametric study on reinforcement learning optimized energy management strategy for a hybrid electric vehicle". *Applied Energy* **259,** 114200 2020.

169. Xiong, R., Cao, J. & Yu, Q. "Reinforcement learning-based real-time power management for hybrid energy storage system in the plug-in hybrid electric vehicle". *Applied energy* **211,** 538–548 2018.

170. Liu, C. & Murphey, Y. L. *Power management for plug-in hybrid electric vehicles using reinforcement learning with trip information.* in *2014 IEEE transportation electrification conference and expo (ITEC)* 2014, 1–6.

171. Zou, Y., Liu, T., Liu, D. & Sun, F. "Reinforcement learning-based real-time energy management for a hybrid tracked vehicle". *Applied energy* **171,** 372–382 2016.

172. Fang, Y., Song, C., Xia, B. & Song, Q. *An energy management strategy for hybrid electric bus based on reinforcement learning.* in *The 27th Chinese control and decision conference (2015 CCDC)* 2015, 4973–4977.

173. Qi, X., Wu, G., Boriboonsomsin, K. & Barth, M. J. *A novel blended real-time energy management strategy for plug-in hybrid electric vehicle commute trips.* in *2015 IEEE 18th international conference on intelligent transportation systems* 2015, 1002–1007.

174. Sutton, R. S. & Barto, A. G. *Reinforcement learning: An introduction* MIT press, 2018.

175. Yuan, J., Yang, L. & Chen, Q. "Intelligent energy management strategy based on hierarchical approximate global optimization for plug-in fuel cell hybrid electric vehicles". *International Journal of Hydrogen Energy* **43,** 8063–8078 2018.

176. Cao, J. & Xiong, R. "Reinforcement learning-based real-time energy management for plug-in hybrid electric vehicle with hybrid energy storage system". *Energy Procedia* **142,** 1896–1901 2017.

177. Xu, B., Malmir, F., Rathod, D., Filipi, Z., *et al. Real-time reinforcement learning optimized energy management for a 48V mild hybrid electric vehicle.* Tech. rep. 2019.

178. Qi, X., Wu, G., Boriboonsomsin, K., Barth, M. J. & Gonder, J. "Data-driven reinforcement learning–based real-time energy management system for plug-in hybrid electric vehicles". *Transportation Research Record* **2572,** 1–8 2016.

179. Fayyazi, M., Abdoos, M., Phan, D., Golafrouz, M., Jalili, M., Jazar, R. N., Langari, R. & Khayyam, H. "Real-time self-adaptive Q-learning controller for energy management of conventional autonomous vehicles". *Expert Systems with Applications* **222,** 119770 2023.

180. Thrun, S. & Schwartz, A. *Issues in using function approximation for reinforcement learning.* in *Proceedings of the 1993 connectionist models summer school* 2014, 255–263.

181. Hasselt, H. "Double Q-learning". *Advances in neural information processing systems* **23** 2010.

182. Meng, X., Li, Q., Zhang, G., Wang, X. & Chen, W. *Double Q-learning-based energy management strategy for overall energy consumption optimization of fuel cell/battery vehicle.* in *2021 IEEE Transportation Electrification Conference & Expo (ITEC)* 2021, 1–6.

183. Han, L., Yang, K., Zhang, X., Yang, N., Liu, H. & Liu, J. *Energy management strategy for hybrid electric vehicles based on double Q-learning.* in *International Conference on Mechanical Design and Simulation (MDS 2022)* **12261** 2022, 639–648.

184. Zhang, Z., Pan, Z. & Kochenderfer, M. J. *Weighted double Q-learning.* in *IJCAI* 2017, 3455–3461.

185. Lee, D., Defourny, B. & Powell, W. B. *Bias-corrected q-learning to control max-operator bias in q-learning.* in *2013 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)* 2013, 93–99.

186. Meng, L., Gorbet, R. & Kulić, D. *The effect of multi-step methods on overestimation in deep reinforcement learning.* in *2020 25th International Conference on Pattern Recognition (ICPR)* 2021, 347–353.

187. Lee, K., Laskin, M., Srinivas, A. & Abbeel, P. *Sunrise: A simple unified framework for ensemble learning in deep reinforcement learning.* in *International Conference on Machine Learning* 2021, 6131–6141.

188. Zhou, Q., Li, J., Shuai, B., Williams, H., He, Y., Li, Z., Xu, H. & Yan, F. "Multi-step reinforcement learning for model-free predictive energy management of an electrified off-highway vehicle". *Applied Energy* **255,** 113755 2019.

189. Schaul, T. "Prioritized Experience Replay". *arXiv preprint arXiv:1511.05952* 2015.

190. Faußer, S. & Schwenker, F. "Neural network ensembles in reinforcement learning". *Neural Processing Letters* **41,** 55–69 2015.

191. Osband, I., Blundell, C., Pritzel, A. & Van Roy, B. "Deep exploration via bootstrapped DQN". *Advances in neural information processing systems* **29** 2016.

192. Anschel, O., Baram, N. & Shimkin, N. *Averaged-dqn: Variance reduction and stabilization for deep reinforcement learning.* in *International conference on machine learning* 2017, 176–185.

193. Agarwal, R., Schuurmans, D. & Norouzi, M. *An optimistic perspective on offline reinforcement learning.* in *International conference on machine learning* 2020, 104–114.

194. Xu, B., Hu, X., Tang, X., Lin, X., Li, H., Rathod, D. & Filipi, Z. "Ensemble reinforcement learning-based supervisory control of hybrid electric vehicle for fuel economy improvement". *IEEE Transactions on Transportation Electrification* **6,** 717–727 2020.

195. Yang, N., Han, L., Liu, R., Wei, Z., Liu, H. & Xiang, C. "Multiobjective intelligent energy management for hybrid electric vehicles based on multiagent reinforcement learning". *IEEE Transactions on Transportation Electrification* **9,** 4294–4305 2023.

196. Lan, Q., Pan, Y., Fyshe, A. & White, M. "Maxmin q-learning: Controlling the estimation bias of q-learning". *arXiv preprint arXiv:2002.06487* 2020.

197. Kielak, K. P. "Do recent advancements in model-based deep reinforcement learning really improve data efficiency?" 2019.

198. 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* **543,** 231841 2022.

199. Li, Y., Tao, J., Xie, L., Zhang, R., Ma, L. & Qiao, Z. "Enhanced Q-learning for real-time hybrid electric vehicle energy management with deterministic rule". *Measurement and Control* **53,** 1493–1503 2020.

200. Ghaderi, R., Kandidayeni, M., Boulon, L. & Trovão, J. P. "Q-learning based energy management strategy for a hybrid multi-stack fuel cell system considering degradation". *Energy Conversion and Management* **293,** 117524 2023.

201. Tao, J., Zhang, R., Qiao, Z. & Ma, L. "Q-Learning-based fuzzy energy management for fuel cell/supercapacitor HEV". *Transactions of the Institute of Measurement and Control* **44,** 1939–1949 2022.

202. 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* **245,** 123182 2022.

203. Yin, Y., Huang, X., Zhan, S., Zhang, X. & Wang, F. "Hierarchical model predictive control strategy based on Q-Learning algorithm for hybrid electric vehicle platoon". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* **238,** 385–402 2024.

204. Shen, S., Gao, S., Liu, Y., Zhang, Y., Shen, J., Chen, Z. & Lei, Z. "Real-time energy management for plug-in hybrid electric vehicles via incorporating double-delay Q-learning and model prediction control". *IEEE Access* **10,** 131076–131089 2022.

205. Zhang, B., Chang, L., Teng, T., Chen, Q., Li, Q., Cao, Y., Yang, S. & Zhang, X. "Multi-objective optimization with Q-learning for cruise and power allocation control parameters of connected fuel cell hybrid vehicles". *Applied Energy* **373,** 123910 2024.

206. Mortabit, I., Rachid, A., Laaroussi, S., Tahiri, M. A. & Boudmane, B. *Integration of ECMS Control Strategies Using QLearning Algorithm In Hybrid Electric Vehicles.* in *2024 International Conference on Circuit, Systems and Communication (ICCSC)* 2024, 1–6.

207. Lin, X., Zhou, B. & Xia, Y. "Online recursive power management strategy based on the reinforcement learning algorithm with cosine similarity and a forgetting factor". *IEEE Transactions on Industrial Electronics* **68,** 5013–5023 2020.

208. Liu, T., Tan, W., Tang, X., Chen, J. & Cao, D. "Adaptive energy management for real driving conditions via transfer reinforcement learning". *arXiv preprint arXiv:2007.12560* 2020.

209. Nyong-Bassey, B. E., Giaouris, D., Patsios, C., Papadopoulou, S., Papadopoulos, A. I., Walker, S., Voutetakis, S., Seferlis, P. & Gadoue, S. "Reinforcement learning based adaptive power pinch analysis for energy management of stand-alone hybrid energy storage systems considering uncertainty". *Energy* **193,** 116622 2020.

210. Liu, T., Zou, Y., Liu, D. & Sun, F. "Reinforcement learning–based energy management strategy for a hybrid electric tracked vehicle". *Energies* **8,** 7243–7260 2015.

211. Lin, X., Zeng, S. & Li, X. "Online correction predictive energy management strategy using the Q-learning based swarm optimization with fuzzy neural network". *Energy* **223,** 120071 2021.

212. Zou, R., Fan, L., Dong, Y., Zheng, S. & Hu, C. "DQL energy management: An online-updated algorithm and its application in fix-line hybrid electric vehicle". *Energy* **225,** 120174 2021.

213. Kaelbling, L. P., Littman, M. L. & Moore, A. W. "Reinforcement learning: A survey". *Journal of artificial intelligence research* **4,** 237–285 1996.

214. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., *et al.* "Human-level control through deep reinforcement learning". *nature* **518,** 529–533 2015.

215. Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D. & Kavukcuoglu, K. *Asynchronous methods for deep reinforcement learning.* in *International conference on machine learning* 2016, 1928–1937.

216. Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P., *et al.* "Grandmaster level in StarCraft II using multi-agent reinforcement learning". *Nature* **575,** 350–354 2019.

217. Zhang, B., Lin, X., Zhu, Y., Tian, J. & Zhu, Z. "Enhancing Multi-UAV Reconnaissance and Search Through Double Critic DDPG With Belief Probability Maps". *IEEE Transactions on Intelligent Vehicles,* 1–16 2024.

218. Watkins, C. J. & Dayan, P. "Q-learning". *Machine learning* **8,** 279–292 1992.

219. Szepesvári, C. *Algorithms for reinforcement learning* Springer Nature, 2022.

220. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. & Riedmiller, M. "Playing atari with deep reinforcement learning". *arXiv preprint arXiv:1312.5602* 2013.

221. Van Hasselt, H., Guez, A. & Silver, D. *Deep reinforcement learning with double q-learning.* in *Proceedings of the AAAI conference on artificial intelligence* **30** 2016.

222. Andrychowicz, M., Wolski, F., Ray, A., Schneider, J., Fong, R., Welinder, P., McGrew, B., Tobin, J., Pieter Abbeel, O. & Zaremba, W. "Hindsight experience replay". *Advances in neural information processing systems* **30** 2017.

223. Chu, T., Wang, J., Codecà, L. & Li, Z. "Multi-agent deep reinforcement learning for large-scale traffic signal control". *IEEE Transactions on Intelligent Transportation Systems* **21,** 1086–1095 2019.

224. 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* 2018, 1861–1870.

225. Chen, D., Hajidavalloo, M. R., Li, Z., Chen, K., Wang, Y., Jiang, L. & Wang, Y. "Deep multi-agent reinforcement learning for highway on-ramp merging in mixed traffic". *IEEE Transactions on Intelligent Transportation Systems* 2023.

226. OpenAI, : Berner, C., Brockman, G., Chan, B., Cheung, V., Dębiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C., Józefowicz, R., Gray, S., Olsson, C., Pachocki, J., Petrov, M., d. O. Pinto, H. P., Raiman, J., Salimans, T., Schlatter, J., Schneider, J., Sidor, S., Sutskever, I., Tang, J., Wolski, F. & Zhang, S. *Dota 2 with Large Scale Deep Reinforcement Learning.* 2019. arXiv: 1912.06680 [cs.LG].

227. Naderializadeh, N., Sydir, J., Simsek, M. & Nikopour, H. "Resource management in wireless networks via multi-agent deep reinforcement learning". *IEEE Transactions on Wireless Communications* 2021.

228. Chen, D., Chen, K., Li, Z., Chu, T., Yao, R., Qiu, F. & Lin, K. "Powernet: Multi-agent deep reinforcement learning for scalable powergrid control". *IEEE Transactions on Power Systems* **37,** 1007–1017 2021.

229. Zhang, K., Yang, Z. & Başar, T. "Multi-agent reinforcement learning: A selective overview of theories and algorithms". *Handbook of reinforcement learning and control,* 321–384 2021.

230. Wang, J., Yuan, M., Li, Y. & Zhao, Z. "Hierarchical Attention Master–Slave for heterogeneous multi-agent reinforcement learning". *Neural Networks,* 359–368 2023.

231. Mahajan, A., Rashid, T., Samvelyan, M. & Whiteson, S. *MAVEN: Multi-Agent Variational Exploration.* in *Advances in Neural Information Processing Systems* 2019, 7611–7622.

232. Li, W., He, S., Mao, X., Li, B., Qiu, C., Yu, J., Peng, F. & Tan, X. "Multi-agent evolution reinforcement learning method for machining parameters optimization based on bootstrap aggregating graph attention network simulated environment". *Journal of Manufacturing Systems,* 424–438 2023.

233. Qiu, D., Wang, J., Dong, Z., Wang, Y. & Strbac, G. "Mean-Field Multi-Agent Reinforcement Learning for Peer-to-Peer Multi-Energy Trading". *IEEE Transactions on Power Systems,* 1–13 2022.

234. Tan, M. *Multi-agent reinforcement learning: Independent vs. cooperative agents.* in *Proceedings of the tenth international conference on machine learning* 1993, 330–337.

235. Nguyen, T. T., Nguyen, N. D. & Nahavandi, S. "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications". *IEEE transactions on cybernetics* **50,** 3826–3839 2020.

236. Zhang, K., Yang, Z., Liu, H., Zhang, T. & Basar, T. *Fully decentralized multi-agent reinforcement learning with networked agents.* in *International Conference on Machine Learning* 2018, 5872–5881.

237. Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V. F., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K. & Graepel, T. "Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward". *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems,* 2085–2087 2018.

238. Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J. & Whiteson, S. "QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning". *Proceedings of the 35th International Conference on Machine Learning,* 4295–4304 2018.

239. Son, K., Kim, D., Kang, W. J., Hostallero, D. E. & Yi, Y. "Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning". *International Conference on Machine Learning,* 5887–5896 2019.

240. Rashid, T., Farquhar, G., Peng, B. & Whiteson, S. "Weighted qmix: Expanding monotonic value function factorisation for deep multi-agent reinforcement learning". *Advances in neural information processing systems,* 10199–10210 2020.

241. Zhang, Y., Ma, H. & Wang, Y. "AVD-Net: Attention Value Decomposition Network For Deep Multi-Agent Reinforcement Learning". *25th International Conference on Pattern Recognition,* 7810–7816 2021.

242. Liu, S., Liu, W., Chen, W., Tian, G., Chen, J., Tong, Y., Cao, J. & Liu, Y. "Learning Multi-Agent Cooperation via Considering Actions of Teammates". *IEEE Transactions on Neural Networks and Learning Systems,* 1–12 2023.

243. Foerster, J., Nardelli, N., Farquhar, G., Afouras, T., Torr, P. H., Kohli, P. & Whiteson, S. *Stabilising experience replay for deep multi-agent reinforcement learning.* in *International conference on machine learning* 2017, 1146–1155.

244. Wang, R., He, X., Yu, R., Qiu, W., An, B. & Rabinovich, Z. *Learning efficient multi-agent communication: An information bottleneck approach.* in *International Conference on Machine Learning* 2020, 9908–9918.

245. Ahilan, S. & Dayan, P. "Feudal multi-agent hierarchies for cooperative reinforcement learning". *arXiv preprint arXiv:1901.08492* 2019.

246. Tang, H., Hao, J., Lv, T., Chen, Y., Zhang, Z., Jia, H., Ren, C., Zheng, Y., Meng, Z., Fan, C., *et al.* "Hierarchical deep multiagent reinforcement learning with temporal abstraction". *arXiv preprint arXiv:1809.09332* 2018.

247. Yang, J., Borovikov, I. & Zha, H. "Hierarchical cooperative multi-agent reinforcement learning with skill discovery". *arXiv preprint arXiv:1912.03558* 2019.

248. Wang, T., Gupta, T., Mahajan, A., Peng, B., Whiteson, S. & Zhang, C. "Rode: Learning roles to decompose multi-agent tasks". *arXiv preprint arXiv:2010.01523* 2020.

249. Xu, Z., Bai, Y., Zhang, B., Li, D. & Fan, G. *HAVEN: hierarchical cooperative multi-agent reinforcement learning with dual coordination mechanism.* in *Proceedings of the AAAI Conference on Artificial Intelligence* **37** 2023, 11735–11743.

250. Jaques, N., Lazaridou, A., Hughes, E., Gulcehre, C., Ortega, P. A., Strouse, D., Leibo, J. Z. & de Freitas, N., "Intrinsic social motivation via causal influence in multi-agent RL" 2018.

251. Pina, R., De Silva, V. & Artaud, C. "Discovering Causality for Efficient Cooperation in Multi-Agent Environments". *arXiv preprint arXiv:2306.11846* 2023.

252. Liu, B., Pu, Z., Pan, Y., Yi, J., Liang, Y. & Zhang, D. *Lazy agents: a new perspective on solving sparse reward problem in multi-agent reinforcement learning.* in *International Conference on Machine Learning* 2023, 21937–21950.

253. Wang, H., Yu, Y. & Jiang, Y. "Fully Decentralized Multiagent Communication via Causal Inference". *IEEE Transactions on Neural Networks and Learning Systems* 2022.

254. Li, J., Kuang, K., Wang, B., Liu, F., Chen, L., Fan, C., Wu, F. & Xiao, J. *Deconfounded value decomposition for multi-agent reinforcement learning.* in *International Conference on Machine Learning* 2022, 12843–12856.

255. Jiang, K., Liu, W., Wang, Y., Dong, L. & Sun, C. "Credit assignment in heterogeneous multi-agent reinforcement learning for fully cooperative tasks". *Applied Intelligence,* 1–18 2023.

256. Foerster, J., Farquhar, G., Afouras, T., Nardelli, N. & Whiteson, S. "Counterfactual multi-agent policy gradients". *Proceedings of the AAAI Conference on Artificial Intelligence,* 2974–2982 2018.

257. Guo, D., Tang, L., Zhang, X. & Liang, Y.-C. "Joint Optimization of Handover Control and Power Allocation Based on Multi-Agent Deep Reinforcement Learning". *IEEE Transactions on Vehicular Technology,* 13124–13138 2020.

258. Hou, Y., Sun, M., Zeng, Y., Ong, Y.-S., Jin, Y., Ge, H. & Zhang, Q. "A Multi-agent Cooperative Learning System with Evolution of Social Roles". *IEEE Transactions on Evolutionary Computation* 2023.

259. Lowe, R., Wu, Y. I., Tamar, A., Harb, J., Pieter Abbeel, O. & Mordatch, I. "Multi-agent actor-critic for mixed cooperative-competitive environments". *Advances in neural information processing systems* **30** 2017.

260. Foerster, J., Farquhar, G., Afouras, T., Nardelli, N. & Whiteson, S. *Counterfactual multi-agent policy gradients.* in *Proceedings of the AAAI conference on artificial intelligence* **32** 2018.

261. Bacon, P.-L., Harb, J. & Precup, D. *The option-critic architecture.* in *Proceedings of the AAAI conference on artificial intelligence* **31** 2017.

262. Harb, J., Bacon, P.-L., Klissarov, M. & Precup, D. *When waiting is not an option: Learning options with a deliberation cost.* in *Proceedings of the AAAI Conference on Artificial Intelligence* **32** 2018.

263. Vezhnevets, A. S., Osindero, S., Schaul, T., Heess, N., Jaderberg, M., Silver, D. & Kavukcuoglu, K. *Feudal networks for hierarchical reinforcement learning.* in *International Conference on Machine Learning* 2017, 3540–3549.

264. Nachum, O., Gu, S. S., Lee, H. & Levine, S. "Data-efficient hierarchical reinforcement learning". *Advances in neural information processing systems* **31** 2018.

265. Grimbly, S. J., Shock, J. & Pretorius, A. "Causal multi-agent reinforcement learning: Review and open problems". *arXiv preprint arXiv:2111.06721* 2021.

266. Pearl, J. "Theoretical impediments to machine learning with seven sparks from the causal revolution". *arXiv preprint arXiv:1801.04016* 2018.

267. Yang, S., Yang, B., Zeng, Z. & Kang, Z. "Causal inference multi-agent reinforcement learning for traffic signal control". *Information Fusion* **94,** 243–256 2023.

268. Ho, J. & Wang, C.-M. *Human-Centered AI using Ethical Causality and Learning Representation for Multi-Agent Deep Reinforcement Learning.* in *2021 IEEE 2nd International Conference on Human-Machine Systems (ICHMS)* 2021, 1–6.

269. Barlow, T. J., Latham, S., McCrae, I. & Boulter, P. "A reference book of driving cycles for use in the measurement of road vehicle emissions". *TRL Published Project Report* 2009.

270. Anesiadou, A., Makridis, M., Ciuffo, B. & Mattas, K. "Open ACC Database". *European Commission, Joint Research Centre (JRC) [Dataset] PID: http://data.europa.eu/89h/9702c950-c80f-4d2f-982f-44d06ea0009f* 2020.

271. Makridis, M., Mattas, K., Anesiadou, A. & Ciuffo, B. "OpenACC. An open database of car-following experiments to study the properties of commercial ACC systems". *Transportation research part C: emerging technologies* **125,** 103047 2021.

272. Xiao, Y., Zhang, Y., Kaku, I., Kang, R. & Pan, X. "Electric vehicle routing problem: A systematic review and a new comprehensive model with nonlinear energy recharging and consumption". *Renewable and Sustainable Energy Reviews* **151,** 111567 2021.

273. Mohan, G., Assadian, F. & Longo, S., *Comparative analysis of forward-facing models vs backwardfacing models in powertrain component sizing*. *IET Hybrid and Electric Vehicles Conference 2013 (HEVC 2013)*. 2013, 1–6.

274. Mohan, G., Assadian, F. & Longo, S. *Comparative analysis of forward-facing models vs backwardfacing models in powertrain component sizing.* in *IET hybrid and electric vehicles conference 2013 (HEVC 2013)* 2013, 1–6.

275. Maybury, L., Corcoran, P. & Cipcigan, L. "Mathematical modelling of electric vehicle adoption: A systematic literature review". *Transportation Research Part D: Transport and Environment* **107,** 103278 2022.

276. Gillespie, T. *Fundamentals of vehicle dynamics* SAE international, 2021.

277. Leach, F., Kalghatgi, G., Stone, R. & Miles, P. "The scope for improving the efficiency and environmental impact of internal combustion engines". *Transportation Engineering* **1,** 100005 2020.

278. Miri, I., Fotouhi, A. & Ewin, N. "Electric vehicle energy consumption modelling and estimation—A case study". *International Journal of Energy Research* **45,** 501–520 2021.

279. Hua, M., Zhou, Q., Wang, C., Zhang, C., Shuai, B. & Xu, H. "Surrogate Modelling for Battery State-of-Charge Estimation in Electric Vehicles Based on Pseudo-2-Dimensional Model and Gradient Boosting Machines" 2021.

280. Seaman, A., Dao, T.-S. & McPhee, J. "A survey of mathematics-based equivalent-circuit and electrochemical battery models for hybrid and electric vehicle simulation". *Journal of Power Sources* **256,** 410–423 2014.

281. Hua, M., Chen, D., Jiang, K., Zhang, F., Wang, J., Wang, B., Zhou, Q. & Xu, H., "Communication-Efficient MARL for Platoon Stability and Energy-Efficiency Co-Optimization in Cooperative Adaptive Cruise Control of CAVs". *IEEE Transactions on Vehicular Technology*, 1–12 2024.

282. Zhang, C., Zhou, Q., Hua, M., Xu, H., Bassett, M. & Zhang, F. "Cuboid equivalent consumption minimization strategy for energy management of multi-mode plug-in hybrid vehicles considering diverse time scale objectives". *Applied Energy* **351,** 121901 2023.

283. Hua, M., Zhang, C., Zhang, F., Li, Z., Yu, X., Xu, H. & Zhou, Q. "Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning". *Applied Energy* **348,** 121526 2023.

284. Bando, M., Hasebe, K., Nakayama, A., Shibata, A. & Sugiyama, Y. "Dynamical model of traffic congestion and numerical simulation". *Physical review E* **51,** 1035 1995.

285. Wang, Q., Dong, H., Ju, F., Zhuang, W., Lv, C., Wang, L. & Song, Z. "Adaptive Leading Cruise Control in Mixed Traffic Considering Human Behavioral Diversity". *IEEE Transactions on Intelligent Transportation Systems,* 1–12 2023.

286. Guzzella, L., Sciarretta, A., *et al. Vehicle propulsion systems* Springer, 2007.

287. Abdi, A. & Fekri, F. *Quantized compressive sampling of stochastic gradients for efficient communication in distributed deep learning.* in *Proceedings of the AAAI Conference on Artificial Intelligence* **34** 2020, 3105–3112.

288. Sukhbaatar, S., Fergus, R., *et al.* "Learning multiagent communication with backpropagation". *Advances in neural information processing systems* **29** 2016.

289. Hua, M., Zhang, C., Li, Z., Yu, X., Xu, H. & Zhou, Q. *Energy Management of Multi-mode Hybrid Electric Vehicles based on Hand-shaking Multi-agent Learning.* in *The 8th Applied Energy Symposium: Low Carbon Cities and Urban Energy Systems* 2022.

290. Shi, W., Huangfu, Y., Xu, L. & Pang, S. "Online energy management strategy considering fuel cell fault for multi-stack fuel cell hybrid vehicle based on multi-agent reinforcement learning". *Applied Energy* **328,** 120234 2022.

291. Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D. & Riedmiller, M. *Deterministic policy gradient algorithms.* in *International conference on machine learning* 2014, 387–395.

292. Shuai, B., Hua, M., Li, Y., Shuai, S., Xu, H. & Zhou, Q. "Optimal Energy Management of Plug-in Hybrid Electric Vehicles Through Ensemble Reinforcement Learning With Exploration-to-Exploitation Ratio Control". *IEEE Transactions on Intelligent Vehicles* 2024.

293. Hua, M., Zhang, C., Zhang, F., Li, Z., Yu, X., Xu, H. & Zhou, Q. "Energy management of multi-mode plug-in hybrid electric vehicle using multi-agent deep reinforcement learning". *Applied Energy* **348,** 121526 2023.

294. Wang, J., Du, C., Yan, F., Hua, M., Gongye, X., Yuan, Q., Xu, H. & Zhou, Q. "Bayesian Optimization for Hyper-Parameter Tuning of an Improved Twin Delayed Deep Deterministic Policy Gradients Based Energy Management Strategy for Plug-In Hybrid Electric Vehicles". *Available at SSRN 4832956* 2024.

295. Wang, J., Du, C., Yan, F., Duan, X., Hua, M., Xu, H. & Zhou, Q. "Energy Management of a Plug-in Hybrid Electric Vehicle Using Bayesian Optimization and Soft Actor-Critic Algorithm". *IEEE Transactions on Transportation Electrification* 2024.

296. Jiang, K., Hua, M., He, X., Dong, L., Zhou, Q., Xu, H. & Sun, C. "Improving String Stability in Cooperative Adaptive Cruise Control Through Multi-Agent Reinforcement Learning With Potential-Driven Motivation". *IEEE Transactions on Artificial Intelligence* 2024.

297. Liu, Z. E., Li, Y., Zhou, Q., Shuai, B., Hua, M., Xu, H., Xu, L., Tan, G. & Li, Y. "Real-time energy management for HEV combining naturalistic driving data and deep reinforcement learning with high generalization". *Applied Energy* **377,** 124350 2025.

298. Liu, Z. E., Li, Y., Zhou, Q., Li, Y., Shuai, B., Xu, H., Hua, M., Tan, G. & Xu, L. "Deep Reinforcement Learning based Energy Management for Heavy Duty HEV considering Discrete-Continuous Hybrid Action Space". *IEEE Transactions on Transportation Electrification* 2024.

299. Wang, J., Du, C., Yan, F., Hua, M., Gongye, X., Yuan, Q., Xu, H. & Zhou, Q. "Bayesian optimization for hyper-parameter tuning of an improved twin delayed deep deterministic policy gradients based energy management strategy for plug-in hybrid electric vehicles". *Applied Energy* **381,** 125171 2025.

300. Yan, F., Wang, J., Du, C. & Hua, M. "Multi-objective energy management strategy for hybrid electric vehicles based on TD3 with non-parametric reward function". *Energies* **16,** 74 2022.

301. Mo, L., Hua, M., Sun, H., Xu, H., Shuai, B. & Zhou, Q. "Study on the Impacts of Hazardous Behaviors on Autonomous Vehicle Collision Rates Based on Humanoid Scenario Generation in CARLA". *arXiv preprint arXiv:2307.10229* 2023.