

Blockchain-Enabled Reinforcement Learning for Privacy-Preserving and Trust Management in Vehicular Networks

Elham Mohammadzadeh Mianji

Supervisors: Dr. Irina Tal and Prof. Gabriel-Miro Muntean

External supervisor: Dr. Anca Delia Jurcut (Univ. College Dublin)



A thesis presented for the degree of Doctor of Philosophy

SCHOOL OF COMPUTING
DUBLIN CITY UNIVERSITY

June 2026

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work, and that I have exercised reasonable care to ensure that the work is original and have conformed to the regulations on the use and declaration of Generative AI, and does not to the best of my knowledge breach any law of copyright, and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed:

Elham Mohammadzadeh Mianji

Student ID: 22267911

Date: 2026/August/13

Appendix A - Reporting of Gen AI Tool Use in Research Thesis

Tools	Claude, Opus 4.6 and 4.7, used between 05/02/2026 and 30/04/2026
Outputs	I used Claude to suggest re-phrasings and simplify complex academic passages. I also used Claude to summarize articles in order to assess their relevance for inclusion in the reading list and literature review.
Prompts	Typical prompts included requests such as "Rephrase this paragraph based on academic style" or "Suggest a more concise and formal alternative" or "Revise and rewrite this sentence in passive voice."
Iterations	Several iterations were conducted to refine sentence flow and maintain authorial tone. All AI outputs were critically reviewed and verified by the author and rewritten where necessary.
Critical Reflection	Gen AI support in this project remained limited. I found the use of Generative AI tools helpful in improving grammar and text readability, which is important as I am not a native English speaker. Their use was limited to language refinement and did not contribute to research design, analysis, or results.

*To my beloved husband, and
my incredibly supportive parents and siblings.*

Acknowledgment

I would like to extend my heartfelt appreciation to my beloved husband and respected colleague, whose unwavering encouragement, faith, and companionship have been invaluable throughout my life and PhD journey. His support and collaborative spirit have given me the strength and motivation to persevere through every challenge.

I am deeply grateful to my parents for their enduring love, wisdom, and guidance, which have shaped my values of integrity, determination, and hard work. I also thank my siblings for their kindness, understanding, and continuous encouragement, which have inspired me to pursue my goals with confidence.

I am especially grateful to my supervisors, Dr. Irina Tal, Prof. Gabriel-Miro Muntean, and my external supervisor, Dr. Anca Delia Jurcut (UCD), for their dedicated supervision, insightful advice, and continued confidence in my research. Their expertise, mentorship, and encouragement have been fundamental to the development of this work and to my growth as a researcher during my PhD. I greatly value their commitment to guiding me and challenging me to achieve the best of my abilities.

I am thankful to my friends and colleagues, whose encouragement, friendship, and support have made this experience far more meaningful. In particular, I would like to acknowledge the members of the PEL LAB group for their cooperation and shared support throughout this journey. I also warmly thank the colleagues and organizers at the ML-Labs Research Center for creating a welcoming, stimulating, and collaborative research community.

Finally, I would like to acknowledge the esteemed members of the examination board, Prof. Eva Ibarrola, Dr. Sahraoui Dhelim, and Dr. Marija Bezbradica, for their time, expertise, and valued contribution to this significant milestone in my academic journey.

Publications

Journals:

1. E. Mohammadzadeh Mianji, G.-M. Muntean, and I. Tal, "Enhancing vehicular network security, privacy, and trust through reinforcement learning: A comprehensive survey," in *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–34, 2025, doi:10.1109/TITS.2025.3612202.
2. E. Mohammadzadeh Mianji, G. -M. Muntean and I. Tal, "Cost-Aware Pseudonym Management in Blockchain-Based VEC-IoV: A Multi-Agent Deep Reinforcement Learning Approach," in *IEEE Transactions on Vehicular Networks*, accepted.
3. M. Fardad, E. Mohammadzadeh Mianji, G. -M. Muntean and I. Tal, "Permissioned Distributed Ledgers-based Cooperative Decentralized Vehicular Edge Computing for Secure and Efficient Task Computation," in *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–16, 2026, doi:10.1109/TITS.2026.3703872.
4. E. Mohammadzadeh Mianji, G. -M. Muntean, A. Delia Jurcut and I. Tal, "Blockchain-Based Reputation Management for Privacy Preservation in the Internet of Vehicles: A Deep Reinforcement Learning Approach," in *IEEE Transactions on Dependable and Secure Computing*, under review (revision submitted).

Conferences:

1. E. M. Mianji, G. -M. Muntean and I. Tal, "Trustworthy Routing in VANET: A Q-learning Approach to Protect Against Black Hole and Gray Hole Attacks," in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, Florence, Italy, 2023, pp. 1-6, doi:10.1109/VTC2023-Spring57618.2023.10201086.

-
2. E. M. Mianji, M. Fardad, G. -M. Muntean and I. Tal, "A Survey on Multi-Agent Reinforcement Learning Applications in the Internet of Vehicles," in 2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring), Singapore, Singapore, 2024, pp. 1-7, doi:10.1109/VTC2024-Spring62846.2024.10683524.
 3. E. M. Mianji, G. -M. Muntean and I. Tal, "Trust and Reputation Management for Data Trading in Vehicular Edge Computing: A DRL-Based Approach," in 2024 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Toronto, ON, Canada, 2024, pp. 1-7, doi:10.1109/BMSB62888.2024.10608229.
 4. M. Fardad, E. M. Mianji, G. -M. Muntean and I. Tal, "Hybrid Consensus Networks for Scalable and Secure Internet of Vehicles," in 2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC), Osaka, Japan, 2024, pp. 1671-1676, doi:10.1109/COMPSAC61105.2024.00263.
 5. E. M. Mianji, G. -M. Muntean and I. Tal, "Dynamic Pseudonym Management With Privacy-Energy Trade-Offs in Iov Networks," in 2025 IEEE 101st Vehicular Technology Conference (VTC2025-Spring), Oslo, Norway, 2025, pp. 1-7, doi:10.1109/VTC2025-Spring65109.2025.11174685.

Other Conferences:

6. M. Fardad, E. M. Mianji, M. Basereh, F. Delaney, S. Wurster, K. Blind, and I. Tal, "Standardization and Regulation of Blockchain and Distributed Ledger Technology: Landscape Overview and Gap Analysis," in 2025 Cyber Research Conference - Ireland (Cyber-RCI), Galway, Ireland, 2025, pp. 1-7, doi:10.1109/Cyber-RCI68134.2025.11385239.
7. J. Palomares, M. Zaccaro, M. Basereh, E. M. Mianji, M. Fardad and I. Tal, "Balancing Innovation and Regulation: Integrating Blockchain into the European Digital Identity Framework," in 2025 Cyber Research Conference - Ireland (Cyber-RCI), Galway, Ireland, 2025, pp. 1-7, doi:10.1109/Cyber-RCI68134.2025.11384865.
8. M. Fardad, E. M. Mianji, G. -M. Muntean and I. Tal, "Rolling-Horizon VCG Pricing for Fair Task Offloading in Internet of Vehicles," in 2026 IEEE 29th International Conference on Intelligent Transportation Systems (ITSC), Naples, Italy, accepted.

Contents

1	Introduction	1
1.1	Research Motivation	1
1.2	Research Context and Problem Statement	3
1.3	Thesis Contributions	7
1.4	Thesis Structure	9
2	Technical Background	11
2.1	Overview of Vehicular Networking and Related Concepts	11
2.1.1	VANET and IoV Architecture	11
2.1.2	MEC in Vehicular Networks	15
2.1.3	SDN in Vehicular Networks	16
2.1.4	UAV in Vehicular Networks	17
2.1.5	Challenges in Vehicular Networks	18
2.2	SPT in Vehicular Networks	19
2.2.1	Security	19
2.2.2	Privacy	25
2.2.3	Trust Management	26
2.2.4	SPT Standardization in Vehicular Networks	27
2.3	Overview of Reinforcement Learning	28
2.3.1	Classification of RL Techniques in Vehicular Networks	38
2.4	Blockchain Technology	44
2.4.1	Fundamentals of Blockchain	44
2.4.2	Blockchain in Vehicular Networks	47
3	Related Works	51
3.1	RL-Based Solutions for Enhancing SPT in Vehicular Networks: A Taxonomy Based on Focus Area	52
3.1.1	RL-based security Approaches	53
3.1.2	RL-based Trust Management Approaches	65
3.1.3	RL-based Privacy Protection Approaches	75
3.2	RL-based Approaches for Enhancing SPT: a Taxonomy Based on RL Methods and SPT Protection	87
3.3	Non-RL Approaches for SPT Provisioning in Vehicular Networks	95
3.3.1	Non-RL Pseudonym Management Approaches	95
3.3.2	Non-RL Reputation Management Approaches	97

3.3.3	Joint Privacy-Preserving and Trust Management	98
3.4	Summary and Research Gaps	99
4	Reinforcement Learning-Based Trust Management in Vehicular Networks	101
4.1	Link-Centric Trust Management: Trustworthy Routing Using Q-Learning	102
4.1.1	System Model	103
4.1.2	Problem Formulation	104
4.1.3	Performance Evaluation	109
4.1.4	Summary	115
4.2	Vehicle-Centric Trust Management: Dynamic Selection of Trusted Sellers Using DDPG	115
4.2.1	System Model	116
4.2.2	Dynamic Credibility Score Calculation	118
4.2.3	DSTS-DDPG Proposed Algorithm	120
4.2.4	Performance Analysis	124
4.3	Conclusion	128
5	Blockchain-Enabled Pseudonym Management in VEC-IoV	130
5.1	System Model	132
5.1.1	Architecture of BPMV Framework	132
5.1.2	Security Model	134
5.2	Pseudonym Management and Revocation Framework	135
5.2.1	System Initialization	135
5.2.2	Vehicle and VEC Server Registration	137
5.2.3	Vehicle Authentication	140
5.2.4	Message Signing and Batch Verification	143
5.2.5	Revocation Report Generation	145
5.3	Problem Formulation	147
5.3.1	Privacy Entropy in VEC-IoV Networks	147
5.3.2	Utility Functions for Cost-Aware Pseudonym Management	148
5.4	Cost-Aware Pseudonym Management MADRL Approach	150
5.4.1	State Space	151
5.4.2	Action Space	152
5.4.3	Reward Function	153
5.4.4	MAPPO Training with CTDE	154
5.5	Simulation Results and Discussions	157
5.5.1	Simulation Setup	157
5.5.2	Results and Analysis	160
5.6	Conclusions	166

6	Blockchain-Based Reputation Management for Privacy Preservation	168
6.1	System Model	169
6.1.1	System Model for VEC-enabled IoV	170
6.1.2	Security Model	171
6.2	BRMPP Framework	172
6.2.1	Pseudonym Identity Issuance and Authentication	172
6.2.2	Reputation Management Under Different Pseudonyms	181
6.3	Problem Formulation	183
6.3.1	Privacy Entropy Formulation	184
6.3.2	Reputation Discrepancy Formulation	184
6.3.3	Optimization Problem Formulation	185
6.4	Deep Reinforcement Learning Approach	186
6.4.1	MDP Formulation	187
6.5	Simulation Results and Discussions	193
6.5.1	Simulation Setup	194
6.5.2	Results and Analysis	197
6.6	Conclusions	204
7	Conclusion	205
7.1	Contributions: Overview & Limitations	205
7.2	Future Work	207
7.3	Final Remarks	209
7.4	Acknowledgment	210

List of Figures

1.1	Future growth of AV	2
2.1	IoV Network	12
2.2	SPT in vehicular networks	20
2.3	The anatomy of a RL algorithm	29
2.4	Deep Q-learning [96]	32
2.5	Classification of RL techniques	39
2.6	Comparison of multi-agent learning topologies: (a) Fully Decentralized Learning and (b) Centralized Training with Decentralized Execution.	41
2.7	Model-based vs. model-free RL methods [127]	42
2.8	Structure of a blockchain	45
2.9	Different types of blockchain [146]	47
2.10	Layered view of blockchain in the vehicular networking stack [155].	49
3.1	A taxonomy based on focus area	53
3.2	A taxonomy of RL-based SPT protection in vehicular networks, classified by SPT protection type and RL method type	88
4.1	Black/gray hole attack on VANET	103
4.2	PDR with a variable number of malicious vehicles	112
4.3	End-to-end delay with variable number of malicious nodes	113
4.4	PDR as a function of vehicle speed	113
4.5	End-to-end delay as a function of vehicle speed	114
4.6	System Model	117
4.7	Mean reward comparison during the training process (an episode contains of about 30 steps)	126
4.8	Comparison of percentage of selecting trustworthy transactions to number of transactions under different scheme.	127
4.9	Comparison of percentage of selecting trustworthy transactions to number of transactions under different scheme.	128
5.1	Architecture of BPMV	132
5.2	Flow Diagram	136
5.3	Overview of CAPM-MAPPO algorithm details	151

5.4	Average reward under different methods	161
5.5	Average reward under different learning rates	162
5.6	Average privacy entropy vs number of vehicles	163
5.7	Social welfare vs number of vehicles	164
5.8	Social welfare vs VEC-side cost(ms)	165
5.9	Social welfare vs decay rate	166
6.1	Blockchain-based system model	170
6.2	The Workflow of Pseudonym Identity Issuance and Authentication . .	177
6.3	The block and blockchain structure of ChainPID	181
6.4	Average reward under different methods	198
6.5	Average episode reward under different learning rates	199
6.6	Average privacy entropy versus number of vehicles	200
6.7	Cost Analysis across Malicious Percentages	201
6.8	Sensitivity to Privacy-Cost Weighting	203

List of Tables

3.1	Summary of Literature on RL-enabled Security in Vehicular Networks	61
3.2	Summary of Literature on RL-enabled Trust Management in Vehicular Networks	72
3.3	Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks	81
3.4	Summary of RL Approaches for Enhancing SPT Based on RL Methods	90
4.1	Simulation Parameters for the QL-TRT Algorithm	110
4.2	Simulation Parameters for the DSTS-DDPG Algorithm	125
5.1	Vehicle-Side Cost Components	159
5.2	VEC-Side Cost Components	159
5.3	MAPPO Algorithm Training Parameters	159
5.4	Simulation Parameters	160
6.1	Action Masking Outcomes for Vehicle V_i at Time t	189
6.2	Simulation Settings	196

List of Acronyms

AI	Artificial Intelligence
A2C	Advanced Actor-Critic
A3C	Asynchronous Advanced Actor-Critic
AV	Autonomous Vehicle
BFT	Byzantine Fault Tolerant
CA	Certificate Authority
CAV	Connected Autonomous Vehicle
CNN	Convolutional Neural Network
CTDE	Centralized Training with Decentralized Execution
DDPG/MADDPG	Deep Deterministic Policy Gradient/Multi-Agent DDPG
DQN/DDQN	Deep Q-Network/Double Deep Q-Network
DRL	Deep Reinforcement Learning
DSRC	Dedicated Short-Range Communication
ECC	Elliptic Curve Cryptography
ETSI	European Telecommunications Standards Institute
FL/FRL	Federated Learning/Federated Reinforcement Learning
HRL	Hierarchical Reinforcement Learning
IoT	Internet of Things
IoV	Internet of Vehicles
ITS	Intelligent Transportation System
MAB	Multi-Armed Bandit
MADRL	Multi-Agent Deep Reinforcement Learning
MAFRL	Multi-Agent Federated Reinforcement Learning
MANET	Mobile Ad-Hoc Network
MARL	Multi-Agent Reinforcement Learning
MBRL	Model-Based Reinforcement Learning
MDP	Markov Decision Process
MEC	Multi-Access Edge Computing

MFRL Model-Free Reinforcement Learning

ML Machine Learning

OBU On-Board Unit

PBFT Practical Byzantine Fault Tolerance

PHC Policy Hill Climbing

PPO/MAPPO Proximal Policy Optimization/Multi-Agent PPO

QoS Quality of Service

RL Reinforcement Learning

RSU Roadside Unit

SAE Society of Automotive Engineers

SDN Software-Defined Networking

SPT Security, Privacy and Trust

TA Trusted Authority

TRL Transfer Reinforcement Learning

UAV Unmanned Aerial Vehicle

VANET Vehicular Ad-Hoc Network

VEC Vehicular Edge Computing

V2I Vehicle-to-Infrastructure

V2N Vehicle-to-Network

V2P Vehicle-to-Pedestrian

V2S Vehicle-to-Sensor

V2V Vehicle-to-Vehicle

V2X Vehicle-to-Everything

Abstract

Elham Mohammadzadeh Mianji

Title: Blockchain-Enabled Reinforcement Learning for Privacy Preservation and Trust Management in Vehicular Networks

Vehicular networks have become a fundamental component of next-generation intelligent transportation systems, with the Internet of Vehicles (IoV) enabling communication among vehicles and infrastructure. While this connectivity offers significant benefits, the large scale of participating vehicles, the high frequency of message exchanges, and inherent vehicle mobility introduce substantial challenges to ensuring strong security, privacy preservation, and trust management during data sharing. This thesis proposes novel Reinforcement Learning (RL)-based approaches supported by immutable blockchain infrastructure to enhance security, privacy, and trust (SPT) provisioning in vehicular networks. To structure the analysis of existing approaches, two distinct taxonomies are proposed in this thesis: a focus-centric taxonomy and a taxonomy based on two dimensions: the type of SPT protection and the type of the RL methodology employed. To enhance trust management, two complementary RL-based approaches are proposed: a link-centric Q-learning-based Trustworthy Routing Technique (QL-TRT) that selects trusted routes while strengthening availability, and a vehicle-centric Dynamic Selection of Trusted Seller using Deep Deterministic Policy Gradient algorithm (DSTS-DDPG) that dynamically optimizes credibility score thresholds for trusted data provider selection in vehicular edge computing (VEC). To enhance privacy preservation, a Blockchain-enabled Pseudonym Management for VEC (BPMV) framework is proposed, integrating a permissioned blockchain with a hierarchical pseudonym structure, together with a Cost-Aware Pseudonym Management algorithm using Multi-Agent Proximal Policy Optimization (CAPM-MAPPO) that balances vehicle-side and VEC-side costs against privacy entropy. To address privacy and trust jointly, a Blockchain-based Reputation Management for Privacy Preservation (BRMPP) framework is proposed, together with a Trust-Aware Pseudonym management algorithm based on Proximal Policy Optimization (TAP-PPO) that enforces reputation constraints on the policy space through a trust-aware action-masking mechanism, ensuring reputation continuity across pseudonym rotations in different Certificate Authority regions. Extensive simulations were conducted across diverse scenarios to evaluate the proposed solutions. Average reward was assessed for all RL-based approaches, while additional metrics including privacy entropy, packet delivery ratio, energy consumption, and cost were applied selectively based on each contribution's specific objectives. The results collectively validate the effectiveness of the proposed algorithms in real-time vehicular environments and highlight their potential for large-scale deployment.

Chapter 1

Introduction

1.1 Research Motivation

The evolution of modern transportation systems has been instrumental in the prosperity of human civilization, with successive technological innovations progressively improving the performance and efficiency with which people and goods are moved [1]. Over recent decades, advances in sensing, computing, analytics, and communication technologies have profoundly transformed the way transportation services are consumed, giving rise to the modern Intelligent Transportation System (ITS) [1]. Vehicular networks are positioned at the forefront of modern ITS [2]. Originally conceived as Vehicular Ad-Hoc Networks (VANETs) and subsequently evolved into the Internet of Vehicles (IoV), these networks have attracted sustained attention from both the research community and the automotive industry, particularly in light of the increasing deployment of Autonomous Vehicles (AVs). Beyond inter-vehicle communication, vehicular networks enable interaction with roadside infrastructure, vulnerable road users, and cloud-based services. This interconnected ecosystem is expected to enhance road safety [3], alleviate traffic congestion, and improve overall transportation efficiency. As the automotive industry advances toward a more connected and autonomous future, vehicular networks are assuming a pivotal role in shaping the transportation landscape and in enabling safer, more efficient driving experiences.

The human cost of road transportation underscores the urgency of this transition. According to the World Health Organization ¹, road traffic injuries are the leading cause of death among individuals aged 5 to 29, with approximately 1.19 million fatalities and between 20 and 50 million non-fatal injuries recorded annually, many

¹<https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries>

of which result in long-term disability. A key enabler in reducing road accidents, alleviating congestion, and addressing environmental concerns is vehicle automation. The Society of Automotive Engineers (SAE) defines six distinct levels of vehicle automation [4]: Level 0 (no automation), Level 1 (driver assistance), Level 2 (partial automation), Level 3 (conditional automation), Level 4 (high automation), and Level 5 (full automation), where Level 5 represents complete self-driving capability without human intervention [5], [6]. AVs at Level 3 and above are experiencing rapid commercial growth. Global sales of AVs are projected to rise from approximately 1.4 million units in 2019 to around 58 million units in 2030 [7], with the corresponding market value forecast to grow from roughly 33 billion US dollars in 2023 to approximately 114 billion US dollars by 2029 [8], as illustrated in Figure 1.1. Connectivity, a complementary capability that enables vehicles to exchange information with other vehicles, infrastructure, and network services, has achieved even broader market penetration: the global connected car fleet is estimated to have reached around 237 million units in 2021 and is projected to exceed 400 million units by 2025, while approximately 96% of all new vehicles shipped worldwide are expected to be equipped with built-in connectivity by 2030 [9]. The convergence of these two trajectories, automation and connectivity, forms the foundation for the emerging class of Connected and Autonomous Vehicles (CAVs). The rapid adoption is expected to play a significant role in improving road safety and in reducing crashes, injuries, and fatalities attributable to human error.

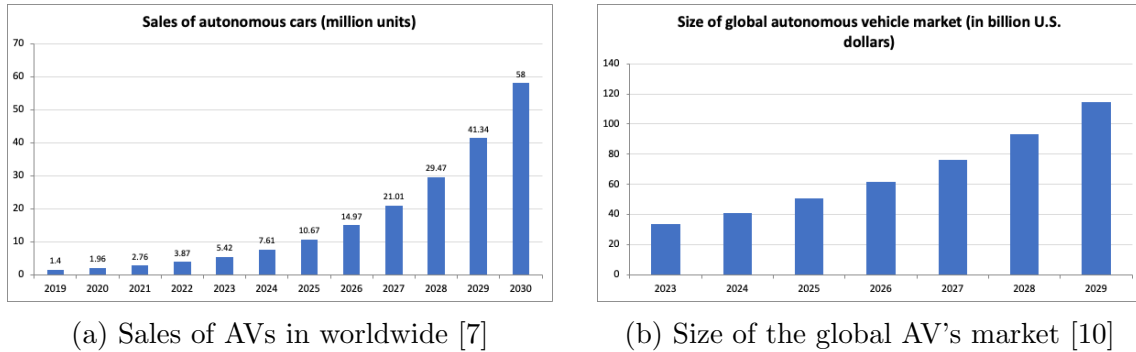


Figure 1.1: Future growth of AV

While AVs have been estimated to have the potential to reduce road accidents by up to 90%, this potential is contingent on the resilience of the underlying communication and computation infrastructure [11]. The widespread deployment of AVs or CAVs within these highly interconnected and dynamic VANETs and IoVs landscapes introduces complex security, privacy, and trust management challenges. The

high density of users, the vast array of connected vehicles, and the frequent, if not constant, communication requirements, all taking place in ever-changing and unpredictable environments, necessitate a reevaluation and strengthening of the existing frameworks to safeguard against potential threats. A critical concern therefore remains to enhance Security, Privacy and Trust (SPT) provisioning within vehicular networks. The SPT provisioning challenge is further complicated by the integration of vehicular networks with emerging technologies such as Multi-Access Edge Computing (MEC), its vehicular counterpart Vehicular Edge Computing (VEC), and blockchain, each of which introduces new security, privacy, and trust concerns.

Traditional approaches to SPT provisioning in vehicular networks have faced several fundamental challenges. They often lack the flexibility required to operate in the dynamic and non-stationary conditions of vehicular networks [2], and struggle with the scalability demands of large, heterogeneous IoV deployments. These limitations motivate a shift toward adaptive, data-driven methods, such as Machine Learning (ML) as a branch of Artificial Intelligence (AI), and particularly Reinforcement Learning (RL), which can adapt to evolving network conditions and adversarial behavior without relying on labeled datasets or fixed attack models.

1.2 Research Context and Problem Statement

Addressing the challenges of vehicular networks requires the secure, privacy-preserving, and trustworthy operation of advanced wireless technologies, distributed edge computing, and decentralized coordination mechanisms. Through real-time data exchange between vehicles, roadside units (RSUs), and cloud platforms, vehicular networks enable informed cooperative decision-making to improve road safety, optimize traffic flows, reduce emissions, and support infotainment services. As automotive technology advances toward higher levels of automation and connectivity, the integration of vehicle automation with Vehicle-to-Everything (V2X) communication promises to transform transportation into a safer, greener, and more efficient system, with broad societal, economic, and environmental benefits. However, the realization of this vision depends critically on the protection of the data exchanges that underpin it: cooperative decision-making cannot be trusted unless the messages on which it relies are authenticated, the identities and locations of the participating vehicles are kept private, and the behavioral reliability of cooperating entities can be assessed. SPT provisioning is therefore not an auxiliary concern but a defining prerequisite for the safe and widespread deployment of vehicular networks.

Within this setting, security, privacy, and trust constitute the three defining concerns. The central objective of security in vehicular networks is to safeguard V2X communication channels against unauthorized access, modification, and disruption [12]. Effective security mechanisms are expected to deliver high-quality services across six requirements: availability, confidentiality, authentication, authorization, integrity, and non-repudiation. Privacy is concerned with preventing the disclosure of sensitive location and identity information to unauthorized observers, while trust management deals with how a vehicle can trust other vehicles and the received messages.

Although the three pillars address different concerns, they are tightly coupled in practice. Various authentication mechanisms are used to verify that messages originate from legitimate entities. However, security mechanisms alone cannot stop an authorized vehicle from intentionally transmitting false or manipulated information. Such misleading messages can reduce transportation efficiency and, in severe cases, lead to accidents that endanger human lives [2], [12]. Addressing this limitation requires trust and reputation management, which evaluates trust level of entities beyond the guarantees provided by authentication.

The interdependence becomes further apparent when privacy is considered. Authentication in its standard form verifies the identity of the sender, which is in direct conflict with the privacy requirement of concealing vehicle identity. This tension has motivated the development of anonymous authentication schemes, in which a vehicle proves its identity of an authorized set without disclosing its real identity, commonly through pseudonym-based mechanisms. However, anonymity alone remains insufficient in vehicular networks, because the predictable dynamics of vehicles allow an attacker to link consecutive pseudonymous messages and reconstruct the trajectory of a target vehicle, thereby compromising location privacy even when identity privacy is preserved. Location-privacy mechanisms, such as pseudonym-change strategies, are therefore layered on top of anonymous authentication to prevent long-term likability of pseudonyms.

The relationship between trust and privacy introduces a further tension. In vehicular networks, pseudonym-based privacy schemes protect vehicle identity and location, but at the same time enable misbehaving vehicles to bypass reputation tracking by discarding and regenerating pseudonyms, disrupting the continuous reputation that pure privacy mechanisms cannot maintain. Reputation-based trust management mechanisms, conversely, require the continuous observation of vehicle behavior over time, which is fundamentally in tension with the privacy goal of

hiding vehicle identity and location. Resolving this tension requires the co-design of privacy and trust: pseudonym management and reputation management should be integrated, so that trustworthiness can be accumulated and transferred across pseudonym changes without compromising identity or location privacy.

Although cloud computing has traditionally handled computation-intensive tasks in mobile networks, its centralized architecture introduces a prohibitive latency and single point of failure in vehicular environments [13]. VEC brings computational capacity closer to the vehicle, supporting latency-sensitive SPT operations such as real-time trust evaluation, reputation management, and pseudonym rotation that cannot be offloaded to distant cloud servers without compromising safety-critical response times [14], [15], [16], [17]. This proximity is a critical enabler of SPT mechanisms.

Blockchain technology has emerged as a promising foundation for strengthening security, privacy, and trust in vehicular network data exchange [6]. Through decentralized validation and tamper-evident transaction records, blockchain can protect VEC networks against unauthorized access, data manipulation, and single points of failure, while supporting auditable pseudonym issuance and reputation management across the network [18], [19], [20]. Specifically, blockchain can replace the centralized authorities of traditional public key infrastructure (PKI)-based and identity-based security schemes, support identity and location privacy through blockchain-enabled pseudonym management, and serve as a distributed reputation system for storing vehicle trust scores [21]. These properties make blockchain particularly well-suited to the integrated SPT provisioning.

Traditional methods, such as graph theory, game-theoretic approaches and heuristic algorithms, have been widely used for control and decision-making in vehicular networks, but they are increasingly challenged by the scale, heterogeneity, and non-stationarity of modern IoV deployments [22]. ML has emerged as a powerful alternative, and within ML, RL and its deep variant (DRL) are particularly well-suited to the class of problems that arises in SPT provisioning [2]. RL formalizes the problem as sequential decision-making under uncertainty, in which an agent learns a policy through continuous interaction with the environment, without relying on labeled datasets or precise system models [23]. This learning paradigm is naturally aligned with the dynamic, adversarial, and partially observable nature of vehicular networks, and supports a proactive mode of operation in which policies continuously adapt to evolving network conditions and attacker behavior. These properties mo-

tivate the use of RL as the primary decision-making framework underpinning the SPT mechanisms considered in this thesis.

Despite the promise of each of these technologies individually, their joint application to integrated SPT provisioning in vehicular networks remains at an early stage. Existing RL-based approaches exhibit structural limitations across several dimensions [2]. First, many approaches rely on centralized aggregators or single-agent formulations that introduce single points of failure and do not scale to edge-computing deployments [24], [25]. Second, the use of static rule-based thresholds and fixed heuristics for decisions such as pseudonym rotation, route selection, and trust evaluation limits the capacity of these approaches to adapt to time-varying adversarial conditions [26]. Third, the majority of existing work is validated on simplified topologies, simple traffic patterns, and static attacker models that do not reflect the non-stationary mobility and adaptive adversarial behavior observed in operational vehicular networks [27], [28]. Finally, the trade-offs among these SPT requirements are rarely addressed jointly: security typically requires the sharing of information that can compromise privacy. Moreover, privacy-preserving mechanisms such as pseudonym rotation can disrupt reputation continuity and undermine trust.

In the context presented above we formulated the following **hypothesis**:

Through the use of ML, and specifically RL, SPT provisioning in vehicular networks can be enhanced more effectively than through approaches that rely on static rule-based mechanisms or that address only classical security challenges. RL enables adaptive decision-making under the dynamic and adversarial conditions of vehicular networks. In addition, Blockchain provides decentralized, immutable, and auditable infrastructure. Together, RL and blockchain have the potential of preserving privacy, enhancing trust, and their joint challenges in vehicular networks and related environments (i.e. IoV). They address threats that classical cryptographic approaches cannot capture, such as misbehaving but authenticated vehicles (untrustworthy participants), pseudonym linkability challenges compromising privacy, and reputation discontinuity across pseudonym rotating in the IoV environment.

In order to effectively investigate this hypothesis, my research aims to address the following research questions:

- **Research Question 1 (RQ1).** What are the key challenges in ensuring security, privacy and trust within vehicular networks (RQ1a)? How can RL be tailored to address trust management challenges in vehicular networks (RQ1b)?

- **Research Question 2 (RQ2).** What design model would effectively integrate blockchain technology with RL strategies to improve privacy in vehicular networks (RQ2a)? What design model would effectively integrate blockchain technology with RL strategies to improve privacy and trust at the same time in vehicular networks (RQ2b)?

1.3 Thesis Contributions

This thesis introduces a suite of novel algorithms and frameworks that address the stringent SPT provisioning in vehicular networks. By integrating advanced RL and blockchain technologies, the following key contributions are presented:

- To identify and structure the key challenges in ensuring security, privacy, and trust within vehicular networks, **two distinct taxonomies** are proposed and analyzed. The first is a focus-centric taxonomy that classifies existing RL-based approaches according to their primary focus area within the SPT provisioning landscape. The second classifies the same body of work along two different dimensions: the type of SPT protection provided, and the type of RL methodology employed. The first dimension examines RL-based approaches within three primary domains: security-enhancing RL, privacy-preserving RL, and trust-enhancing RL, with each domain organized according to the specific characteristics of the approaches proposed in the literature. The second dimension systematically categorizes the studies based on whether they employ MAB or MDP formulations, with the MDP category further classified according to learning agent, learning model, and learning topology.
- To enhance trust management in vehicular networks, two complementary RL-based approaches are proposed. The first approach focuses on link-centric trust, assessing the trustworthiness of communication links between vehicles, while the second focuses on vehicle-centric trust, assessing the credibility of individual vehicles as data providers. The link-centric Q-learning-based Trustworthy Routing Technique (**QL-TRT**), defends vehicular networks against black-hole and gray-hole attacks by learning trust values of directed links from packet-forwarding ratio, residual energy ratio, and expected transmission time, thereby strengthening availability. The vehicle-centric Dynamic Selection of Trusted Sellers using Deep Deterministic Policy Gradient (**DSTS-DDPG**) algorithm for trusted data provider selection in VEC, dynamically optimizes

a credibility score threshold using DDPG over a reputation model based on belief, distrust, uncertainty, familiarity, and freshness.

- To enhance privacy preservation and overcome challenges associated with centralized pseudonym management approaches, the Blockchain-based Pseudonym Management for VEC (**BPMV**) framework is proposed. The framework integrates a permissioned blockchain into the VEC environment and introduces a hierarchical pseudonym structure. This structure is based on two types of pseudonym identities: systemic pseudonym identities (SPIDs) and communication pseudonym identities (CPIDs). Within this framework, decentralized protocols are developed for SPID authentication and revocation, eliminating the single point of failure. Furthermore, a social welfare function is formulated that incorporates vehicle-side and VEC-side costs together with privacy entropy to enable dynamic, cost-aware communication pseudonym updates within **BPMV**. The social welfare optimization is formulated as a Multi-Agent Deep Reinforcement Learning (MADRL) problem. Hence, a Cost-Aware Pseudonym Management algorithm using Multi-Agent Proximal Policy Optimization (**CAPM-MAPPO**) is proposed to govern dynamic CPID updates under this formulation. Together, the **BPMV** framework and the **CAPM-MAPPO** algorithm directly strengthen the vehicle identity and location privacy-preserving. Beyond its primary focus on privacy preservation, the BPMV framework supports the authentication requirements in security services through its cryptographic protocols components.
- To enhance privacy preservation and trust management jointly, the Blockchain-based Reputation Management for Privacy Preservation (**BRMPP**) framework is proposed. The framework introduces a hierarchical pseudonym structure comprising SPIDs for long-term reputation management and CPIDs for short-term privacy-preserving interactions. A permissioned blockchain is integrated within the framework to provide immutable storage of pseudonym identities and reputation records. In addition, a two-level reputation mechanism is proposed, in which local reputation is assessed at the VEC servers and globally aggregated at the Certificate Authority (CA) level. By enabling reputation continuity across SPIDs and across different CA regions, the framework ensures reliable and trustworthy data sharing in IoV environments. Complementing the framework, a Trust-Aware Pseudonym management algorithm based on Proximal Policy Optimization (**TAP-PPO**) is proposed to govern

CPID rotation decisions. The algorithm formulates CPID rotation as a constrained utility maximization problem that balances privacy entropy against cost. A trust-aware action-masking mechanism is then employed to enforce reputation constraints on the policy space, so that only vehicles with sufficient reputation are permitted to execute pseudonym rotations. Together, the **BRMPP** framework and the **TAP-PPO** algorithm strengthen privacy and trust, while supporting the authentication security service through the decentralized authentication protocols and the integrity security service through the immutable blockchain storage of pseudonym identity and reputation records.

1.4 Thesis Structure

This thesis is organized as follows:

- **Chapter 1** presents the research motivation, outlines the challenges in current vehicular networks, and summarizes the main contributions of this work.
- **Chapter 2** provides the technical background, including vehicular networks, edge computing, blockchain, and various types of RL, that supports the proposed solutions. The chapter also introduces SPT concepts in vehicular networks and related environments.
- **Chapter 3** provides a comprehensive survey of the state of the art in SPT provisioning for vehicular networks. The reviewed literature is organized through two complementary taxonomies: a focus-centric taxonomy, and a taxonomy based on the SPT protection type and RL method type. Together, these two taxonomies answers the **RQ1a** and motivate the contributions presented in subsequent chapters.
- **Chapter 4** details the QL-TRT algorithm, for link-centric trust management and DSTS-DDPG algorithm for vehicle-centric trust management, further addressing **RQ1b**.
- **Chapter 5** outlines the BPMV framework, which integrates a permissioned blockchain with decentralized pseudonym management. This chapter primarily tackles **RQ2a**.
- **Chapter 6** describes the BRMPP framework, which integrates blockchain-enabled verification with deep reinforcement learning for joint privacy preservation and trust management. This chapter addresses **RQ2b**.

- **Chapter 7** concludes the thesis by summarizing the key findings and outlining potential directions for future research.

Chapter 2

Technical Background

2.1 Overview of Vehicular Networking and Related Concepts

2.1.1 VANET and IoV Architecture

VANET is a specialized form of a Mobile Ad-Hoc Network (MANET). Unlike traditional networks, VANETs are established dynamically over wireless communication channels, operating without any fixed infrastructure. The intention of this approach is to adapt the principles of MANETs to the automotive environment, addressing the unique challenges of this domain, including constant changes in the composition of the network as a vehicle enters or leaves the range of communication [29].

VANETs are comprised of three main components: Onboard Units (OBUs), which enable vehicles to communicate with other OBUs or nearby Roadside Units (RSUs) [12]; RSUs, which are fixed devices positioned along roads to provide services to vehicles; and a Trusted Authority (TA), responsible for registering the various components within the VANET [30].

On the other hand, when the IoV is considered, a landscape emerges in which vehicles are transformed into intelligent entities equipped with sophisticated interior systems and advanced sensors. IoV merged VANET and the Internet of Things (IoT), building upon VANET by incorporating communication standards that enable vehicles to connect and share data seamlessly. These smart vehicles are typically linked with cloud-based data platforms, allowing them to access and exchange information effortlessly. As a specialized subset of IoT, IoV specifically focuses on enhancing connectivity and communication within the automotive field. V2V (vehicle-to-

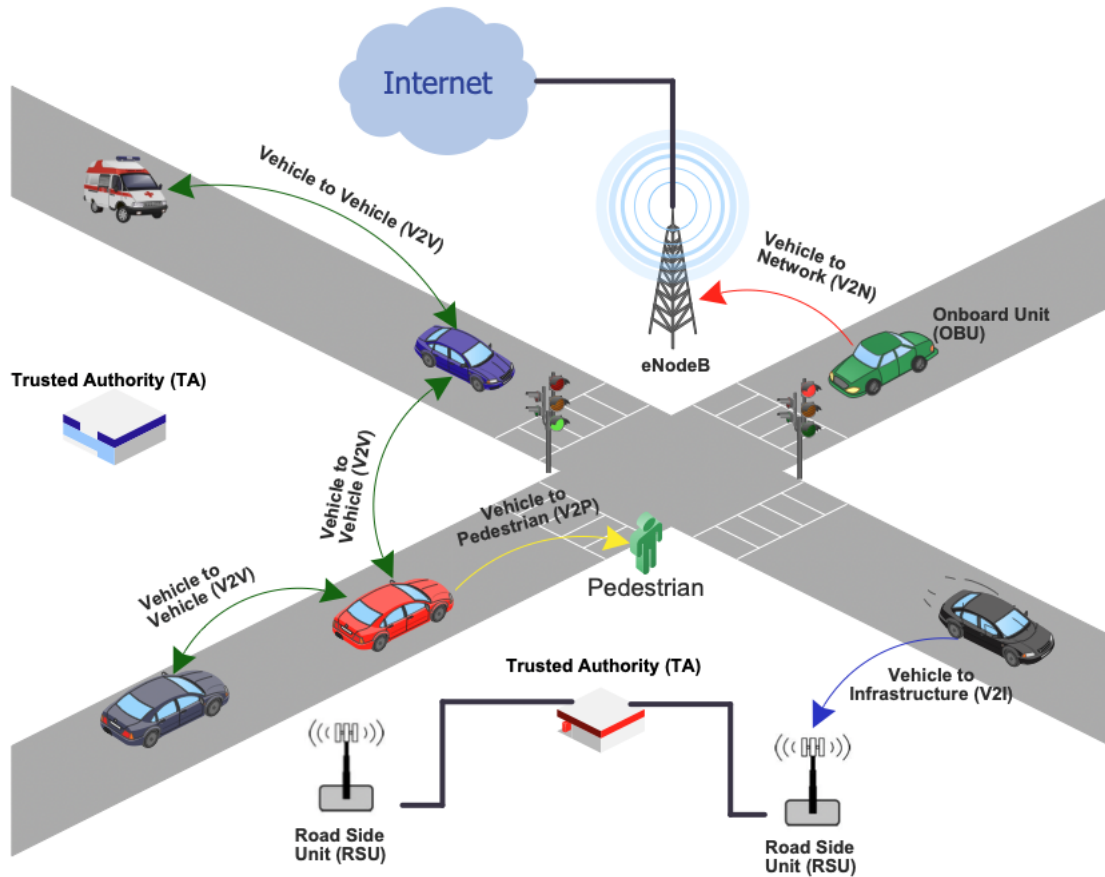


Figure 2.1: IoV Network

vehicle), V2P (vehicle-to-pedestrian), V2I (vehicle-to-infrastructure), V2N (vehicle-to-network), and V2S (vehicle-to-sensor), which collectively support communications within the IoV environment [31], [32], [33], [34], are briefly introduced. Figure 2.1 depicts these various modes of V2X (vehicle-to-everything) communication in order to provide a better understanding.

- **V2V Communications:** enable vehicles to broadcast, unicast and group-cast valuable information, including traffic conditions and collision detection. Typically, if two vehicles are within communication range of each other, they communicate directly; otherwise, they communicate via multi-hop communication.
- **V2P Communications:** allow devices carried by pedestrians, passengers, cyclists or people using wheelchairs or other mobility devices to communicate with vehicles.
- **V2I Communications:** refer to the direct data exchange between vehicles and RSUs that are fixed along the roads. V2I communications require larger bandwidth than those associated with V2V. They are less vulnerable to attacks.
- **V2N Communications:** describes communications between a vehicle and a network provider that provides access and connectivity to vehicles at different frequencies. Moreover, vehicles are able to communicate information with their management systems through a high-bandwidth, low-latency and high-reliability network infrastructure.
- **V2S Communications:** During their movement, vehicles can monitor their behavior using this type of communications by detecting their speed, position, tire pressure, oil pressure, etc. [32].

As compared to VANETs, the architecture of IoV includes a wide range of devices, making it more complex and market-oriented. The motivation for the design and development of IoV is threefold: addressing the commercialization issues of VANETs, examining the volume of growing traffic casualties, and assessing the massive market opportunity ahead for IoV. IoV can be viewed in two ways: as a collection of novel network technologies that provide an intelligent, layered architecture somewhat independent of VANETs, or as a network that utilizes an established

VANET network for some layers. IoV brings intelligence to the network, distributing, sharing, and valuing information in a way that is meaningful to users, vehicular systems, authorities, and service providers. One key feature of IoV is its ability to overcome the heterogeneity of systems, services, applications, and devices, making it the universal and unlimited network of the future. The capability of this network to manage global information is greater than that of VANETs. It integrates devices, vehicles, and users. IoV can analyze, process, and evaluate information collected from vehicles globally, which is not typically within the capacity of VANETs [35].

A fundamental enabler of IoV is the ability of vehicles and infrastructure nodes to share and exchange data in real time. Data sharing in vehicular networks encompasses a wide range of activities, including the dissemination of safety messages such as accident alerts and hazard warnings, the exchange of sensor readings and traffic conditions among neighboring vehicles and infrastructures.

V2X connectivity is supported by two dominant radio technologies: IEEE 802.11p-based communications [36] and 3rd Generation Partnership Project (3GPP) cellular-based communications [37].

- **IEEE 802.11p-based communications:** generally, the Wireless Access in Vehicular Environment (WAVE) is based on the IEEE 802.11p standard and provides the basic radio standard for Dedicated Short-Range Communications (DSRC). The IEEE 802.11p standard [36] has established a framework for V2V and V2I data exchange. Both Europe and the United States have designated the 5.9 GHz frequency band as exclusive for ITS services, with an emphasis on safety-related applications. It is noteworthy that IEEE 802.11p is the basis of two well-established standards for radio communications: DSRC in the United States [38] and the original radio access technology adopted by European Telecommunications Standards Institute (ETSI) Cooperative ITS (C-ITS) standards in Europe [39]. In addition, both DSRC and ETSI C-ITS standards incorporate cryptographic protocols to ensure communication security.
- **Cellular-based communications:** On another hand, there are already 3GPP standards available for communications in vehicular networks, such as the Long-Term Evolution (LTE) technology. Additionally, the 5G New Radio V2X (5G NR-V2X) was also standardized in Release 16.

The cellular V2X (C-V2X) protocol defines its lower layers, including the Physical (PHY) and Media Access Control (MAC) layers, through 3GPP radio ac-

cess specifications, while its upper layers, encompassing applications, facilities, networking, transport, and security, are built on the DSRC and ETSI C-ITS standards [40].

While there is some overlap in the functions and goals of VANETs and IoV, each of these networks has its unique characteristics and focuses. The choice between these technologies depends on the specific application, infrastructure availability, and the desired level of autonomy and connectivity. Nevertheless, reliable and efficient communication between vehicles is a challenge in all vehicular networks, including VANETs and IoV. In order to overcome these challenges, researchers have explored the integration of various technologies to enhance the performance and capabilities of vehicular networks [41]. However, the adoption of these technologies exposes vehicular networks to various attacks that threaten their security, privacy, and trust. Hence, it is imperative that vehicular networks are comprehensively analyzed when combined with other technologies, paying special attention to the security aspect of the network.

Next, this work provides an overview of three prominent technologies in vehicular networks: MEC [42], Software-Defined Networking (SDN), and Unmanned Aerial Vehicles (UAVs), as well as discussing the implications of these technologies for vehicular networks and the proposed solutions.

2.1.2 MEC in Vehicular Networks

MEC is a transformative technology that brings computing and storage capabilities closer to mobile networks, enabling data to be processed more rapidly and efficiently [43]. When MEC principles are applied to vehicular networks, the resulting paradigm is referred to as VEC [44]. Owing to the proximity of edge resources to end users, latency is reduced and the performance of vehicular applications is improved, which is particularly advantageous for safety-critical services such as autonomous driving that demand low latency and high data rates. MEC architectures applied to vehicular networks are commonly categorized into three main variants [45]:

- **MEC Server:** MEC servers are typically integrated with base stations (BSs) or RSUs, providing a common platform to which vehicles can offload their computational tasks via the V2I communication [46]. With robust computing and storage resources, these servers process vehicle requests quickly and efficiently without requiring assistance from a remote cloud service provider. However,

due to the mobility of vehicles, the coverage area served by a single MEC server is inherently limited and malicious MEC servers are capable of posing a significant threat to the integrity and security of the network [47].

- **Nearby Vehicles:** In addition to offloading computing tasks to the roadside MEC server at the RSU/BS, the vehicle can also offload them to surrounding vehicles through the V2V mode which is known as Vehicular Edge Computing (VEC) networks. It is possible for some vehicles to be equipped with computational and storage resources. Through the utilization of nearby vehicles idle computing resources, this resource sharing mechanism can maximize the use of available processing power. Alternatively, the cost of operating a large number of RSUs will be high. Thus, it is a good idea to utilize surrounding vehicles as edge servers. However, this also opens up potential vulnerabilities as it requires vehicles to share their resources, potentially exposing them to security threats. Malicious vehicles can exploit this to disrupt the network or compromise other vehicles data.
- **Hybrid MEC:** Hybrid MEC combines both the advantages of MEC server offloading and vehicle offloading in vehicular networks. Parallel computing is enabled by allocating tasks between the MEC server and nearby vehicles [48]. The system can determine the optimal offloading approach according to network conditions, resource availability, and real-time computation requirements. The task can be offloaded to nearby vehicles when a vehicle moves out of the RSU coverage area. This ensures continuous processing and minimizes disruptions.

2.1.3 SDN in Vehicular Networks

Vehicular networks can benefit from the implementation of SDN, known as Vehicular Software-Defined Networking (VSDN). SDN allows for the development of network switches and routers capable of running network management programs, enabling support for applications in distributed systems and heterogeneous network architectures [49]. The fundamental architecture of SDN separates the data plane, which handles data forwarding, from the control plane, which determines packet forwarding rules. This separation is facilitated by software controllers that manage programmable network components via standardized communication protocols like OpenFlow [50]. The SDN controller manages traffic flow by enforcing forwarding policies, enhancing network adaptability by separating policy execution from

physical hardware. In vehicular networks, OBUs in vehicles can perform roles such as power control, interface management, and data packet forwarding, making them ideal for SDN deployment. However, adopting SDN in vehicular networks introduces security and scalability challenges. Securing the centralized controller is essential to prevent single points of failure, unauthorized access, and breaches of sensitive V2X data [51]. Protecting the communication channels and securing the software running on SDN controllers is essential to maintain the integrity and privacy of network operations within vehicular networks. By addressing these security challenges, SDN can contribute to the efficient and flexible management of vehicular networks, enabling improved performance and enhanced services.

The centralized control mode and distributed control mode are two approaches within VSDN. In the centralized control mode, the SDN controller holds absolute regulation authority over all network nodes, defining traffic actions through flow rules. On the other hand, the distributed control mode allows SDN nodes to operate independently during data packet delivery, with local agents acting as internal SDN wireless controllers within clusters [52]. These control modes further shape the functionality and management of VSDN, enabling tailored approaches based on specific network requirements and characteristics.

2.1.4 UAV in Vehicular Networks

UAVs play a pivotal role in enhancing vehicular networks by automating various components of the transportation system beyond just vehicles. UAVs can replace or support human personnel in tasks such as road support, traffic monitoring, and communication infrastructure [53]. For example, UAVs can fly around incident locations, providing basic support or survey reports, and they can monitor vehicles on highways, reporting traffic violations. Additionally, UAVs can act as flying base stations, extending network coverage in areas lacking traditional infrastructure. However, integrating UAVs into vehicular networks introduces security and privacy challenges. Unauthorized access to UAVs can lead to operational disruptions, data breaches, or misuse of their capabilities. The wireless communication used by UAVs is susceptible to eavesdropping, unauthorized access, or data interception [54]. Safeguarding data privacy is a key concern, as UAVs collect and transmit sensitive information. Moreover, ensuring the integrity of traffic monitoring data is essential to prevent manipulation that could result in incorrect decisions or compromised safety. Protecting UAVs from physical tampering and unauthorized modifications is vital to maintain their secure operation within vehicular networks. Addressing these secu-

rity and privacy challenges is paramount to harnessing the full potential of UAVs in improving the efficiency and reliability of vehicular networks.

2.1.5 Challenges in Vehicular Networks

Despite advances in V2X communication and edge computing, several fundamental challenges persist in vehicular networks. These challenges complicate the delivery of latency-sensitive applications and affect the reliability and enhancing security, privacy and trust. The main ones are discussed next.

- ***Mobility and Dynamic Topology:*** The high mobility of vehicles results in a continuously evolving network topology, with vehicles frequently joining and leaving the coverage area of edge servers and other network entities [55]. The rapid pace of topology change increases the network's exposure to attacks and complicates the timely identification of malicious vehicles [12].
- ***Scalability:*** Scalability is one of the principal challenges in vehicular communication, requiring the network to adapt and maintain reliable performance as the number of participating nodes grows or as conditions change [56]. In dense urban environments and high-mobility traffic scenarios, the volume of communication and computational requests can grow rapidly, placing significant demands on the underlying infrastructure. Designing vehicular network architectures that scale gracefully under such conditions therefore remains challenging.
- ***Low Latency and High Throughput:*** Vehicular networks support a wide range of safety-critical and time-sensitive applications, such as cooperative collision avoidance, autonomous driving coordination, and emergency notification, all of which impose strict latency and throughput requirements [57]. Cryptographic operations, authentication procedures, reputation evaluations, and blockchain-based consensus all introduce computational and communication overhead, which must be carefully balanced against the timing constraints of safety-critical applications.
- ***Security, Privacy and Trust Concern:*** The open and decentralized nature of vehicular networks introduces many attacks in the network. Ensuring authentication, data integrity, and availability is particularly challenging in such a distributed environment, where traditional centralized security approaches may not be suitable. At the same time, vehicles continuously broad-

cast detailed contextual information, including location, speed, and heading, which exposes drivers to location-tracking threats and raises serious privacy concerns [12]. Establishing trust in data originating from anonymous or frequently changing identifiers further compounds these difficulties, particularly because short-lived interactions between vehicles leave limited time to assess peer trustworthiness, and because trust mechanisms must remain effective without compromising privacy guarantees.

Each of these challenges is interrelated, and their combined effect significantly shapes the design space of vehicular network protocols and applications. Among them, the robust provisioning of security, privacy, and trust, SPT, stands out as the most critical concern. The open, decentralized, and highly dynamic nature of vehicular networks introduces severe vulnerabilities that traditional protection mechanisms cannot adequately address, while mobility, scalability, and stringent latency requirements further compound the difficulty of designing effective SPT solutions. SPT provisioning is therefore a paramount concern in vehicular networks and constitutes the central focus of this thesis. The following subsection, introduces the core concepts of SPT, outlining the critical role each plays to safeguard the vehicular networks.

2.2 SPT in Vehicular Networks

As vehicular networks continue to evolve to make intelligent transportation safer and more efficient, safeguarding these environments has become a critical priority. Enhancing SPT in vehicular networks is essential to ensure that communication remains reliable, sensitive user data is protected, and participating entities behave honestly. While often grouped together, each of these three dimensions addresses a distinct aspect of network protection. To provide a comprehensive understanding of these foundational concepts, the remainder of this subsection details the specific requirements, services, and common attacks associated with each of these three areas separately. Furthermore, Figure 2.2 provides a visual representation summarizing the core SPT requirements in vehicular networks. The specific attacks associated with each security requirement are discussed in detail in the following subsections.

2.2.1 Security

Security in vehicular networks focuses on defending the network infrastructure against malicious actions, such as the unauthorized insertion, alteration, or deletion

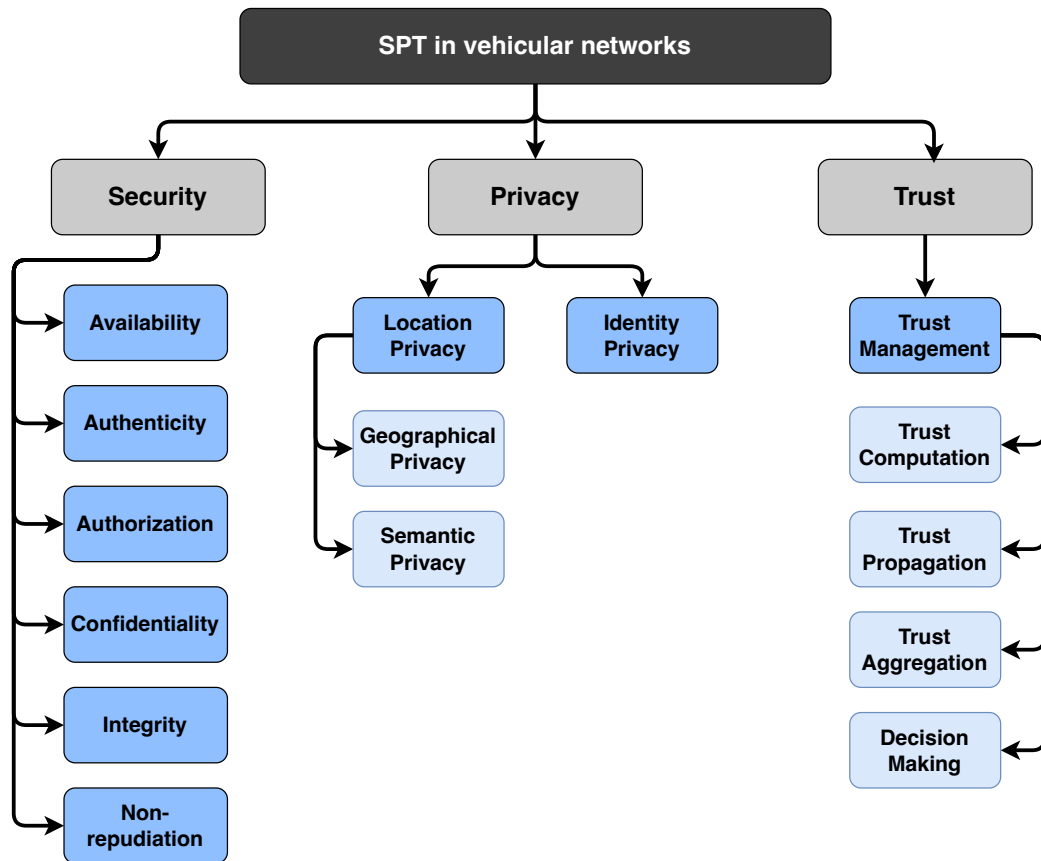


Figure 2.2: SPT in vehicular networks

of messages, by enforcing requirements such as availability, confidentiality, authenticity, data integrity, and non-repudiation [58], [59].

Availability

Availability ensures that vehicular network services and communication channels remain available to legitimate users at all times, even under adverse conditions [60]. The primary attacks that compromise availability in vehicular networks include the following:

- ***Denial-of-Service (DOS) Attack:*** A DOS attack is one of the most common attacks on vehicular networks [61]. An attacker in this context sends a massive number of requests that the system cannot handle. Due to this, the server becomes overloaded and denies legitimate users access, stopping communication between vehicles and/or RSUs. Distributed Denial-of-Service (DDoS) attacks can take place from multiple locations, making it more difficult to detect an attack. A DDoS attack could target not only vehicles in the network, but also RSUs [62].
- ***Jamming Attack:*** In the context of a Jamming attack, the attacker interferes with radio waves within the vehicular environments by transmitting an overpowered signal in the equivalent frequency range [63]. Therefore, signal quality is decreased until the user is no longer able to use the frequency band. Intelligent anti-jamming techniques could be adopted to secure the vehicular environments [64].
- ***Malware Attack:*** Malware like worms or viruses could be installed into OBUs and RSUs during software exchange or update. By using this method, attackers can penetrate vehicular networks such as VANETs and cause other components of the VANET system to malfunction [65].
- ***Spamming Attack:*** The attacker injects large volumes of spam messages into the vehicular network, consuming a significant amount of bandwidth and causing collisions. As a result, legitimate nodes are unable to receive genuine messages on time due to increased transmission latency, which can severely affect the performance of the network [60].
- ***Blackhole and Grayhole Attack:*** In this attack, a malicious node advertises itself to neighboring nodes with a route that has a low hop count and the fastest direction to the destination node. When a vehicle forwards a packet

through this node, it silently drops the packet instead of relaying it to the recipient. By redirecting and dropping all or a selective subset of packets, the attacker prevents real-time safety messages from reaching their intended recipients, which can disrupt routing tables and compromise the safety of drivers and passengers [60], [66].

Authenticity

Authenticity guarantees that the identities of entities participating in vehicular communication are genuine and that received messages originate from legitimate sources. To prevent potential data manipulation, network nodes must be authenticated prior to accessing any shared information [60], [67].

- **Sybil Attack:** This type of attack allows a malicious node to create multiple virtual identities and broadcast multiple messages using fictitious identities. The attacker can manipulate the behaviors of other vehicles so that the receiving vehicle believes the messages are coming from different vehicles. Thus, the receiving vehicles recognize a traffic jam and adjust their travel paths to keep the roadway clear. Sybil attacks make it difficult to detect the virtual nodes involved [68].
- **Tunneling Attack:** Attackers utilize the same network to establish a private connection, which is called a tunnel. Tunneling provides an additional communication channel between two far-away parts of the vehicular network. Consequently, distant victims nodes can communicate with each other as neighbors [61].
- **GPS Spoofing/ Position Faking Attack:** In most VANETs, position information plays a crucial role. The attacker tries to create signals that are stronger than those generated by real satellites around the vehicle where the GPS sensor is located. GPS spoofing attacks can facilitate other types of attacks, such as attacks against applications that rely on the location of the node [65].
- **Replay Attack:** A replay attack is a situation in which an attacker captures previously transmitted packets and replays them over the network at a later time. In fact, valid data is fraudulently retransmitted, causing the receiving vehicle or RSU to accept outdated information as authentic. Many of the existing protocols lack a mechanism for detecting duplicated packets, which

as a result allows an adversary to compromise the security of the network [64]. Vehicular network messages must have timestamps to indicate their freshness or sequence numbers ensuring that messages are processed in order. As a result, vehicular networks can reduce the risk of replay attacks and enhance the overall security.

Authorization

Authorization refers to the process of verifying that an authenticated entity possesses the appropriate permissions to access specific resources, services, or network functionalities [69]. User access control is also an essential security service for providing access rights only to authorized registered users for various services, information, and resources available in the IoV environment [60]. While authentication confirms the identity of a communicating party, authorization determines what that entity is permitted to do within the network. In vehicular environments, authorization mechanisms are essential to enforce access control policies over safety-critical services such as computation offloading at MEC servers, privileged V2X communication channels, and emergency message dissemination [70], [71]. The absence of robust authorization controls can expose vehicular networks to the different attacks.

Confidentiality

Confidentiality ensures that sensitive information exchanged within vehicular networks is available only to authenticated and authorized entities [72]. Although certain information in vehicular networks, such as basic safety messages, is intended to be publicly broadcast, the protection of private and sensitive data remains of utmost importance [60]. Encryption mechanisms are typically employed to prevent unauthorized disclosure of sensitive data, including vehicle identifiers, location information, and driving patterns. Attacks targeting confidentiality aim to intercept or access private communications between vehicles and infrastructure without the knowledge of the communicating parties.

- ***Man-in-the-Middle Attack:*** In this attack, a malicious entity positions itself between two communicating nodes and intercepts and changes the messages exchanged between them [73]. The attacker can change the messages while both the sender and receiver believe that communication takes place directly between them in private [74].

- ***Eavesdropping Attack:*** An attacker is able to snoop on information transmitted between two parties by listening to the media in vehicular networks. Several types of valuable information can be collected via this attack type by non-registered users, such as location data that can be used for vehicle tracking. In this attack, the attacker does not alter or reply to any of the entities, but merely listens and gathers personal information [72].

Integrity

- ***Masquerading Attack:*** An attacker fakes its identity in the network by using a registered user ID, which is called a mask. Moreover, an attacker creates fake messages and broadcasts them over the network. For instance, a malicious vehicle acts as an emergency vehicle to defraud other vehicles to slow down [65].
- ***Message Tampering Attack:*** A message tampering attack occurs when an attacker modifies a particular part of a transmitted message. Furthermore, the attacker can delete a portion of the message, alter its content, or create new messages in order to falsify requests or responses [67].
- ***Illusion Attack:*** In an illusion attack, an adversary vehicle deliberately generates false sensor data or fabricates misleading readings from its onboard sensors. The attacker, for instance, broadcasts false warning messages such as "accident ahead" to neighboring vehicles in order to misguide them into making incorrect driving decisions. Because the attacker is a legitimately authenticated member of the network, conventional authentication mechanisms are unable to detect this attack [75].

Non-repudiation

Non-repudiation ensures that vehicles, and other network entities cannot deny having sent or received a message [55]. In vehicular networks, this property is particularly critical for accident investigation, liability attribution, and traffic-rule enforcement, since malicious or negligent entities must not be able to evade responsibility for their transmitted messages and actions [76].

- ***Repudiation Attack:*** A repudiation attack occurs when a vehicle or node denies having transmitted a message or performed an action within the network. For instance, a vehicle that has disseminated false traffic information may subsequently deny its involvement [55].

2.2.2 Privacy

Privacy preservation in various vehicular networks such as IoV and VANETs refers to the measures taken to protect the privacy of vehicles and passengers when they communicate within the network. VANETs, a specialized subset of ad-hoc networks, are characterized by vehicles equipped with wireless communication devices, allowing them to share information not only with each other but also with RSUs like traffic lights and road signs. While this communication is essential for enhancing road safety, traffic efficiency, and various other applications, it also introduces vulnerabilities that could lead to the leakage of sensitive information about individuals and vehicles.

One of the primary objectives of privacy protection in vehicular networks is to prevent unauthorized parties from gaining access to sensitive data, which may include real-time information about a vehicle's location, speed, and travel direction. The potential consequences of privacy breaches can be severe, ranging from tracking the movements of specific vehicles to uncovering behavioral patterns of drivers and passengers [77]. To address these concerns, privacy protection in vehicular networks can be broadly categorized into two main categories: vehicle identity privacy and vehicle location privacy.

- **Vehicle Identity Privacy:** Vehicle identity privacy refers to the protection of the identities of drivers and passengers within the network. Ensuring that personal information remains confidential is crucial, as the compromise of this data can lead to identity theft or other malicious activities. Therefore, mechanisms such as pseudonym schemes and anonymous credentials are employed to preserve the real identities of vehicles while still enabling legitimate communication [78].
- **Vehicle Location Privacy:** Vehicle location privacy focuses on protecting the whereabouts of vehicles within the network. This aspect of privacy protection is multi-dimensional and can be dissected into two distinct dimensions: geographical privacy [79] and semantic privacy [80].
 - Geographical privacy involves hiding the exact geographic coordinates of a vehicle. Various techniques are employed to allow vehicles to mask their exact locations while still ensuring the effective functioning of the vehicular network. By doing so, the risk of malicious entities tracking a vehicle's movements is significantly reduced.

- Semantic privacy extends the concept of location preservation by not only hiding geographic coordinates but also masking contextual information associated with a location. This encompasses details about destinations, routes, and user preferences. By introducing context-aware modifications to location data, semantic attributes can be altered such that even if an adversary manages to access location information, the insights they can derive are limited, preserving the privacy of users' behavioral preferences and intentions.

2.2.3 Trust Management

Due to the open and vulnerable nature of vehicular networks, establishing trust among communicating entities is one of the main challenges when enhancing the protection level of the network against malicious nodes. Trust management refers to the process of evaluating the trustworthiness of vehicles and the information they transmit, in order to ensure the security and reliability of communications [81]. While trust management is closely related to data integrity, they serve distinct purposes: integrity mechanisms ensure that messages are not altered during transmission, whereas trust management evaluates whether the content of a message from an authenticated sender should be believed. Trust computation is a key component of trust management and involves the calculation of a trust value for each vehicle in the network. The trust value represents the degree of confidence that can be placed in a particular vehicle or the information it transmits.

Trust management can be implemented via either a centralized or decentralized approach. In a centralized approach, a central entity such as a cloud server, RSU, or trusted authority is responsible for collecting behavioral data, computing trust scores, and disseminating trust information to vehicles within the network. In a decentralized approach, each vehicle independently maintains trust values of its neighbors and shares this information with other vehicles through the exchange of trust management messages. Additionally, blockchain technology has been adopted to support decentralized trust management by providing an immutable and transparent ledger for recording trust-related transactions. In all approaches, vehicles make decisions about which other entities to trust or avoid based on the trust values they receive. In a decentralized approach, for instance, during the first exchange of information between two vehicles, the receiving vehicle evaluates the sender's trust level by collecting recommendations from its neighbors. The receiver then aggregates these recommendations to assess whether the sender should be trusted. The

process of establishing trust in a vehicular network involves four steps: computation of trust, propagation of trust, aggregation of trust, and decision making [81]. Due to the dynamic nature of vehicular networks, the lack of a centralized trust authority, and the highly mobile nature of the network, trust management and trust computation remain very challenging.

2.2.4 SPT Standardization in Vehicular Networks

The security, privacy, and trust requirements discussed in the previous subsections are addressed by several international standards that define the security framework for vehicular communications. This subsection provides an overview of the key standards relevant to the scope of this thesis.

The IEEE 1609 family of standards defines the WAVE architecture for V2X communication. Within this family, IEEE 1609.2 [82] specifies secure message formats, digital signature mechanisms, and encryption methods for vehicular applications. It defines a certificate-based PKI framework in which vehicles and infrastructure nodes are issued digital certificates to ensure message authenticity, integrity, and non-repudiation. To protect vehicle privacy, IEEE 1609.2 introduces pseudonym certificates that contain no identity information, allowing vehicles to sign V2X safety messages without revealing their real identities. The associated standard IEEE 1609.2.1 [83] further specifies the certificate management interfaces, including the protocols for provisioning, managing, and revoking both enrollment and pseudonym certificates on end entities. In the United States, the Security Credential Management System (SCMS) [84] manages these pseudonym certificates, enabling each vehicle to possess multiple short-lived certificates that are rotated periodically to prevent long-term tracking.

In Europe, the ETSI ITS security framework provides an equivalent set of standards. ETSI TS 102 940 [85] specifies the ITS communications security architecture, defining the roles of enrollment authorities and authorization authorities within the ITS PKI. ETSI TS 103 097 [86], which serves as a profile of IEEE 1609.2 adapted for the European C-ITS context, specifies the security header and certificate formats. In the ETSI framework, pseudonym certificates are referred to as authorization tickets, and all identifiers associated with a vehicle across different layers of the communication stack must be changed synchronously when the authorization ticket is rotated, ensuring unlikable pseudonymity [87]. These standardized pseudonym schemes provide the foundational framework upon which the pseudonym management contributions of this thesis are built.

2.3 Overview of Reinforcement Learning

The potential of vehicular networks to improve transportation security and safety has garnered significant research interest. To enhance vehicular network performance and security, powerful techniques such as ML are employed [88], [89]. ML techniques, including supervised learning, unsupervised learning, and RL, train computer systems to perform tasks without explicit programming and can be used to bolster routing, trust management, privacy-preserving, and security threat detection in vehicular networks. In the dynamic environment of vehicular networks, supervised and unsupervised learning methods designed to enhance security can face constraints. In contrast, RL-based security solutions do not depend on labeled datasets [90] or prior knowledge of the attack and network model.

RL is a type of ML that allows agents to learn from their experiences and improve their performance over time by receiving rewards from the environment. RL focuses on training agents to make optimal decisions based on the feedback received. An agent receives a reward or penalty for each action based on its effect on the environment. The agent learns to take actions that maximize rewards and avoid penalties [91]. To formulate the problem, RL is represented as a Markov Decision Process (MDP) or Multi-Armed Bandit (MAB) problem. MDP is a mathematical framework for modeling decision-making problems in which agents interact with their environment over time. MDP models the decision-making process as a sequence of steps. In each step, the agent is in a particular state and must decide what action to take. Depending on some probabilistic rules, the chosen action transitions the agent from its current state to a new state. The transition probabilities define the dynamics of the environment. MAB learning is a type of RL that can be classified as an MDP with a single state. A MAB agent learns about its environment by taking actions and receiving rewards. Instead of tracking the environment, the agent receives feedback based on its actions and typically chooses actions in a greedy manner, by updating its estimates of each action's benefits independent of other actions [92].

A MDP can be characterized by a set $\{\mathcal{S}, \mathcal{A}, P, r, \gamma\}$, where:

- $\mathcal{S}$ is the state space that represents the set of discrete or continuous states $s \in \mathcal{S}$.
- $\mathcal{A}$ is the action space that represents the set of discrete or continuous actions $a \in \mathcal{A}$ that the agent can take.

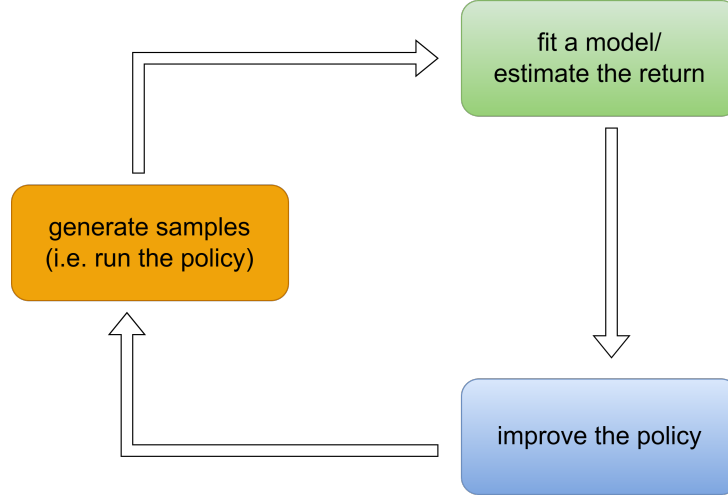


Figure 2.3: The anatomy of a RL algorithm

- P represents the transition probability matrix. Specifically, at a time step t , it determines the probability of going from state s_t to state s_{t+1} . Mathematically, the entry at the i^{th} row and j^{th} column of this matrix gives $P_{ij} = p(s_{t+1} = j | s_t = i)$, where i and j are state indices.
- $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ represents the reward function $r(s_t, a_t)$ received by the agent when transitioning from state s_t to state s_{t+1} by executing action a_t .
- γ is the discount factor that lies in the range 0 and 1, i.e., $\gamma \in [0, 1]$, and determines the importance the agent assigns to future rewards. A value of $\gamma = 0$ indicates that the agent prioritizes only immediate rewards, whereas $\gamma = 1$ implies that the agent values future rewards equally with immediate ones.

The discounted return, R_t , represents the sum of all future rewards, each weighted by the discount factor γ . The agent's primary goal is to identify the optimal policy for taking actions in specific states to maximize its cumulative rewards, i.e., $\max \mathbb{E}\{R_t\}$.

The RL process involves a series of interconnected steps, each playing an important role in the learning and improvement of an agent's actions within an environment. Figure 2.3 illustrates the three fundamental components of RL algorithms, and each stage is described as follows:

- **Sample Generation:** By deploying the current policy in the designated environment, the algorithm begins the process of generating samples or trajectories. These trajectories encompass a sequence of states, actions, and rewards

that the agent obtains through its interactions with the environment. During this phase, the agent explores its surroundings and evaluates the outcome of its actions as it learns through trial and error.

- ***Fitting a Model***: During the model fitting or return estimation phase, the collected samples are analyzed. An important component of this step is the analysis of the collected data in order to estimate relevant aspects of the policy's performance. The estimation process can incorporate a variety of techniques. For example, the rewards attained along trajectories may be aggregated to assess the overall effectiveness of certain actions. Furthermore, advanced approaches such as constructing value functions and modeling dynamics can be employed to achieve a more detailed understanding of the environment's behavior.
- ***Policy Improvement***: Using the previous step's return estimation, the algorithm refines the policy to improve its effectiveness. Policy improvement involves modifying the parameters of the policy to increase the probability of selecting actions that produce favorable results while decreasing the probability of selecting actions that do not. This iterative process is equivalent to learning from past experiences and adapting them to optimize future decisions.

The implementation of RL algorithms entails costs at each phase. Generating samples can require substantial resources, particularly in environments that require real-time interactions to simulate real-life complexity. Learning an accurate model is a resource-intensive process since adequate data is essential for comprehensive training. In addition, updating highly complex policies results in substantial computing expenses. Therefore, the cost of each phase varies depending on the algorithm's design, the complexity of the problem, and the computational requirements.

A widely recognized traditional RL algorithm is Q-learning [93], which maintains a table of Q-values for each state-action pair and iteratively updates them based on the agent's interactions with the environment. The Q-value $Q(s, a)$ for a specific state-action pair (s, a) is updated according to the following rule:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right] \quad (2.1)$$

where $\alpha \in [0, 1]$ denotes the learning rate, r is the immediate reward, γ is the discount factor, s' is the next state, and a' represents the possible actions in the next state. Q-learning is an off-policy temporal difference learning method, enabling the

agent to learn from diverse policy experiences by updating the current Q-value with the maximum Q-value across all possible actions.

In addition to Q-learning, several other traditional RL algorithms have been widely adopted. Policy Hill Climbing (PHC) [94] iteratively adjusts action probabilities to improve the policy. Dyna-Q [23] combines direct interaction with the environment and planning through a learned model, enabling more sample-efficient learning. Monte Carlo RL methods [23] estimate value functions by averaging returns obtained from complete episodes of experience.

RL Methods and Extensions

However, traditional RL methods typically rely on tabular or linear function approximation, limiting their scalability when applied to complex, high-dimensional environments like vehicular networks. To address this limitation, advanced RL extensions are required. For instance, Deep Reinforcement Learning (DRL) is frequently employed to manage large state spaces using deep neural networks as function approximators. Furthermore, the dynamic nature of vehicular environments necessitates additional advanced techniques, including Safe RL, Transfer RL, Hierarchical RL, and Federated RL, to manage decentralized training, knowledge transfer, and strict safety constraints, respectively. These RL techniques are explored further below.

- **Deep Reinforcement Learning:** DRL integrates RL with deep neural networks, enabling agents to process high-dimensional input spaces. While traditional RL algorithms rely on tabular representations, these methods become inefficient as the state-action space expands exponentially with the addition of features [95]. The integration of deep neural networks in DRL provides several key benefits. First, deep neural networks automatically learn meaningful feature representations from raw sensory input, removing the need for manual feature engineering. This capability enables agents to effectively process complex, high-dimensional observations. Second, deep neural networks can model intricate non-linear functions, allowing the development of more advanced and nuanced policies. Third, they support end-to-end learning, enabling the agent to map raw input directly to actions without relying on predefined intermediate representations. As a result, DRL algorithms have achieved, and in some cases exceeded, human-level performance on complex tasks. Furthermore, DRL frameworks make use of both offline and online data to optimize the association between state-action pairs and their corresponding values [42].

By defining the policy as $\pi_\theta(a|s)$, which represents the probability of taking action a in state s under the deep neural network parameter θ , the probability of an MDP trajectory τ can be defined as:

$$p_\theta(\underbrace{s_1, a_1, s_2, a_2, \dots}_\tau) = p(s_1) \prod_{t=1}^{\infty} \pi_\theta(a_t|s_t) p(s_{t+1}|s_t, a_t) \quad (2.2)$$

and the RL objective can be expressed by:

$$J(\theta^*) = \max_{\theta} \mathbb{E}_{\tau \sim p_\theta(\tau)} [R_t] \quad (2.3)$$

where θ^* indicates the optimal parameter.

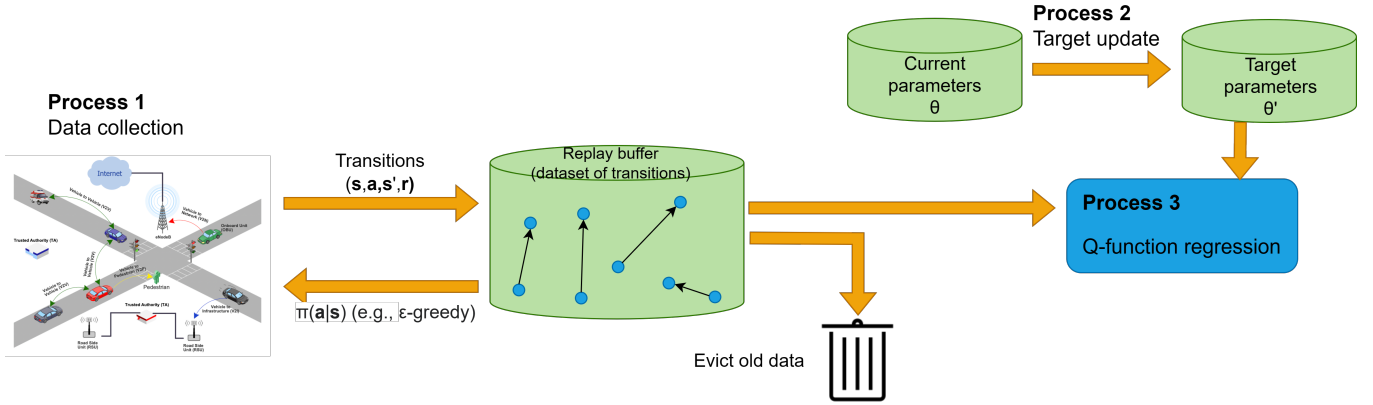


Figure 2.4: Deep Q-learning [96]

Different DRL algorithms adopt different strategies to optimize this objective function. The following presents three prominent DRL algorithms that are directly employed in the contributions of this thesis:

- **Deep Q-Network**: Deep Q-Network (DQN) is a DRL method that uses neural networks to replace traditional Q-tables. These networks map input states to actions and their corresponding Q-values, making learning more adaptable and efficient, especially in complex or high-dimensional state environments. To compress the state space, a Convolutional Neural Network (CNN) can be employed in the deep Q-learning framework [97]. The architecture of deep Q-learning is shown in Figure 2.4. Deep Q-learning utilizes two neural networks that have the same structure but different weights: a main network and a target network. The epsilon-greedy exploration strategy is used to select actions, where the action

with the highest predicted Q-value for a given state is considered the optimal choice. Once the agent has selected an action, it performs it and updates the main and target networks using the Bellman equation [98]. To improve the learning process of DQN agents, techniques such as experience replay [99] can be employed. A replay buffer is used to store past experiences, including the state, action, reward, and next state, during training. A batch of experiences is randomly selected from this buffer, enabling the agent to stabilize its learning process and mitigate the effects of temporal correlations within sequential data [100]. Another variation of deep Q-learning, dueling Deep Q-Learning, separates the estimation of state values from the estimation of action advantages. By utilizing a dueling network architecture, this approach provides an understanding of the importance of states and the benefits of different actions. As a result, this architecture allows better policy evaluation when there are many actions with similar values [101].

- Proximal Policy Optimization (PPO): PPO [102] is a DRL algorithm that aims to improve training stability by constraining the magnitude of policy updates at each iteration. In standard policy gradient methods, large policy updates can lead to performance collapse or instability. PPO addresses this by introducing a clipped surrogate objective function that limits how much the policy can change in a single update step. The clipped objective is defined as:

$$L^{CLIP}(\theta) = \mathbb{E}_t \left[\min \left(r_t(\theta) \hat{A}_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) \hat{A}_t \right) \right] \quad (2.4)$$

where $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$ is the probability ratio between the new and old policies, $\hat{A}_t$ is the estimated advantage function, and ϵ is a small hyperparameter that controls the clipping range. By preventing excessively large updates, PPO achieves a balance between sample efficiency and training stability, making it one of the most widely adopted DRL algorithms for both single-agent and multi-agent applications.

- Deep Deterministic Policy Gradient (DDPG): DDPG [103] is an actor-critic DRL algorithm designed for environments with continuous action spaces. DDPG maintains four neural networks: an actor network that outputs a deterministic action given the current state, a critic network

that estimates the Q-value for a given state-action pair, and their corresponding target networks that provide stable training targets. The actor is updated by applying the chain rule to the expected return with respect to the actor parameters, while the critic is updated by minimizing the temporal difference error. Similar to DQN, DDPG employs experience replay to store and sample past transitions, and uses soft target updates to gradually update the target networks, thereby improving training stability. DDPG is particularly well-suited for vehicular network applications that require continuous control decisions, such as trust score optimization and power allocation.

- **Multi-Agent Reinforcement Learning:** Single-agent RL alone is insufficient for modeling scalable and self-organizing systems that consist of interconnected agents. Multi-Agent RL (MARL) addresses this limitation by enabling multiple agents to learn effective policies while considering the actions and observations of other agents [104]. MARL is formally modeled as a Markov game. In a Markov game, the state of the environment is represented by s and the joint action of all agents is denoted by $\mathbf{a} = (a_1, a_2, \dots, a_N)$, where N is the total number of agents. Each agent i selects an action from its own action space $\mathcal{A}_i$ based on its individual policy π_i , resulting in a joint action $\mathbf{a}$. The environment transitions to a new state s' according to the transition dynamics $p(s' | s, \mathbf{a})$, and each agent receives a reward $r_i(s, \mathbf{a})$ based on the chosen action and resulting state.

The goal of each agent in MARL is to learn its optimal policy π_i^* that maximizes its expected cumulative reward over time. The joint policy π of all agents is defined as $\pi(\mathbf{a} | s) = \prod_{i=1}^N \pi_i(a_i | s)$. The Q-value for agent i is defined as the expected cumulative reward when taking joint action $\mathbf{a}$ in state s and following the joint policy π :

$$Q_i^\pi(s, \mathbf{a}) = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t r_i(s_t, \mathbf{a}_t) \mid s_0 = s, \mathbf{a}_0 = \mathbf{a} \right] \quad (2.5)$$

In the fully cooperative case, the optimal Q-value for agent i can be defined as:

$$Q_i^*(s, \mathbf{a}) = \max_{\pi} Q_i^\pi(s, \mathbf{a}) \quad (2.6)$$

Consequently, the value function of agent i is expressed as:

$$V_i^\pi(s) = \mathbb{E}_{\mathbf{a} \sim \pi} [Q_i^\pi(s, \mathbf{a})] \quad (2.7)$$

Hence, in general-sum or competitive settings, the optimal policy for agent i depends on the policies of its opponents, denoted as π_{-i} [105]. In MARL, the relationship between agents' reward functions determines the nature of the game. In cooperative settings, all agents share the same reward function ($r_i = r$). In zero-sum Markov games ($\sum_i r_i = 0$) and general-sum Markov games (no constraint on $\sum_i r_i$), the optimal joint policy is typically addressed by searching for a Nash equilibrium [105], [106].

When MARL is combined with deep neural networks, it is referred to as Multi-Agent Deep Reinforcement Learning (MADRL). MADRL extends the DRL algorithms discussed previously to multi-agent settings. For instance, Multi-Agent DDPG (MADDPG) extends DDPG to mixed cooperative-competitive multi-agent settings using a centralized critic with decentralized actors, while Multi-Agent PPO (MAPPO) extends PPO to multi-agent settings, typically under the centralized training with decentralized execution paradigm. MAPPO is directly employed in the contributions of this thesis for cooperative decision-making in vehicular networks.

- **Safe Reinforcement Learning:** Although RL has made considerable progress in handling complex decision-making tasks, its application in real-world scenarios often raises safety concerns when certain types of mistakes pose a threat to people and/or objects [107]. This has resulted in an increasing need for safe RL algorithms, particularly in autonomous driving, where ensuring the safety of the systems is of paramount importance [108].

Safe RL is a specialized area within RL that concentrates on creating algorithms and methodologies to guarantee the secure and dependable conduct of RL agents in environments characterized by uncertainty or potential risks. The goal is to design agents that can learn from interaction with the environment while taking precautions to prevent harmful actions or catastrophic outcomes. To tackle safety concerns during the training of RL agents, two primary approaches can be employed: modifying the objective function and constraining the search space. Modifying the objective involves addressing rare catastrophic events that may not heavily impact the overall expected re-

turn. These modifications take into account both the risk-sensitive criterion, aiming to balance rewards and risks, and the worst-case criterion, focusing on policies resilient to unfavorable scenarios and ensuring robustness against potential risks [109], [110]. Additionally, safe RL employs a constrained criterion that maximizes return while adhering to specific constraints. This approach ensures that policies optimize performance while taking into account safety limitations.

In the context of vehicular network security, conventional RL methods explore all possible state-action pairs, including those that could result in network failure or privacy leakage, leading to slower and less efficient learning in safety-critical scenarios. Safe RL addresses this by incorporating security performance metrics, such as the probability of privacy leakage or network failure, to assess the risk associated with each state-action pair during the learning process [111], [112]. However, most existing safe RL algorithms depend on an accurate reward signal, which is not commonly available in wireless security applications, and can suffer from performance degradation when agents must select policies from large action spaces.

- **Transfer Reinforcement Learning:** Transfer Reinforcement Learning (TRL) extends traditional RL by allowing agents to utilize knowledge acquired from one or multiple source tasks to improve their performance on a target task [113]. In standard RL, an agent learns to maximize rewards by interacting with its environment. However, this procedure can be time-consuming and data-intensive, especially for complex tasks [114]. TRL overcomes this limitation by leveraging pre-existing knowledge to accelerate the learning process and facilitate adaptation to new tasks.

A successful implementation of TRL hinges on defining three key aspects: what knowledge should be transferred, when it should be transferred, and how it should be transferred [115]. In this regard, TRL can be broadly categorized into intra-agent and inter-agent transfer. In intra-agent transfer, an agent reuses its own previously acquired knowledge in new tasks or domains without requiring explicit communication with other agents. In inter-agent transfer, knowledge is shared between different agents, potentially with different observations or representations, to improve the learning efficiency of the receiving agent [116]. Overall, TRL enhances learning efficiency in multi-

task or multi-agent systems by reducing the training time required to achieve effective policies [117].

- **Hierarchical Reinforcement Learning:** Hierarchical Reinforcement Learning (HRL) involves decomposing a complex RL task into smaller sub-problems, forming a hierarchy of policies [118], [119]. The higher-level policy selects optimal sub-tasks as its actions, while each sub-task is treated as a separate RL problem with its own lower-level policy. This decomposition converts the original long-term objective into a series of shorter-term sub-tasks, offering several advantages including reduced computational complexity and increased reusability, which contribute to faster learning. HRL techniques employ a variety of abstractions to handle the exponentially growing number of parameters in large-scale problems, enabling the agent to find optimal solutions more efficiently. Notable HRL algorithms include Hierarchical-DQN (H-DQN) [120], which utilizes a high-level controller to select sub-goals and a low-level Q-network to execute primitive actions, and Feudal Networks [121], [122], which employ a Manager module to set abstract goals and a Worker module to achieve them through environment interactions.
- **Federated Reinforcement Learning:** Federated Learning (FL) is an approach to ML that enables training models across multiple decentralized devices without exposing raw data [123], [124]. Unlike traditional ML approaches that require centralizing data from various sources, FL allows training to occur locally on each device, while only aggregated model updates are shared with a central server. In vehicular networks, the current model is distributed to vehicles within the IoV for local training, and the locally updated models are subsequently sent back to a central server, where they are aggregated to create a unified global model which is then redistributed to the vehicles. This approach helps maintain data privacy and reduces the need to transmit sensitive raw data.

Federated Reinforcement Learning (FRL) extends FL to sequential decision-making problems, where multiple agents, each with its own local dataset and potentially unique environment, collaborate to learn a shared or coordinated policy using RL techniques [125]. FRL enables autonomous policy training on individual devices, followed by central server aggregation to produce a cohesive global policy. This allows agents to learn effective decisions while maintaining data privacy by avoiding the need to share privately collected

data during exploration. FRL is particularly relevant to vehicular networks, where ensuring the security and privacy of vehicles during the learning process is of utmost importance [126].

2.3.1 Classification of RL Techniques in Vehicular Networks

As discussed, RL problems can be formulated as either MDP-based or MAB-based methods. Within MDP-based methodologies, a more detailed classification can be established along three fundamental dimensions: learning model, learning topology, and learning agent. It is important to note that these three dimensions are independent of each other, meaning that any given RL algorithm is simultaneously characterized by all three dimensions. The learning model dimension distinguishes between model-based and model-free approaches, where model-free methods are further divided into value-based, policy-based, and actor-critic categories. The learning topology dimension categorizes approaches based on whether training and decision-making are performed in a centralized or decentralized manner. The learning agent dimension differentiates between single-agent and multi-agent approaches. Figure 2.5 illustrates this classification framework. Each of these dimensions is discussed in more detail in the following.

Learning Topology

The learning topology defines how training and decision-making responsibilities are distributed across the network.

- **Centralized:** In a centralized learning topology, a single central entity serves as the focal point for collecting data and making decisions that affect the entire vehicular network. This approach offers the advantage of a unified and coordinated view of the network, which can facilitate prompt responses to emerging threats and changes in network conditions. However, it presents certain trade-offs. One significant concern is scalability, as a central entity may become overwhelmed when dealing with large-scale vehicular networks. Additionally, the centralized model introduces a single point of failure, making it susceptible to disruptions aimed at the central decision-making process.

A notable variant of centralized learning is Centralized Training with Decentralized Execution (CTDE), where a central entity is responsible for training the RL models, but the execution and decision-making processes are decentralized among individual vehicles or clusters. This hybrid approach combines the advantages of centralized training for model accuracy with the benefits of

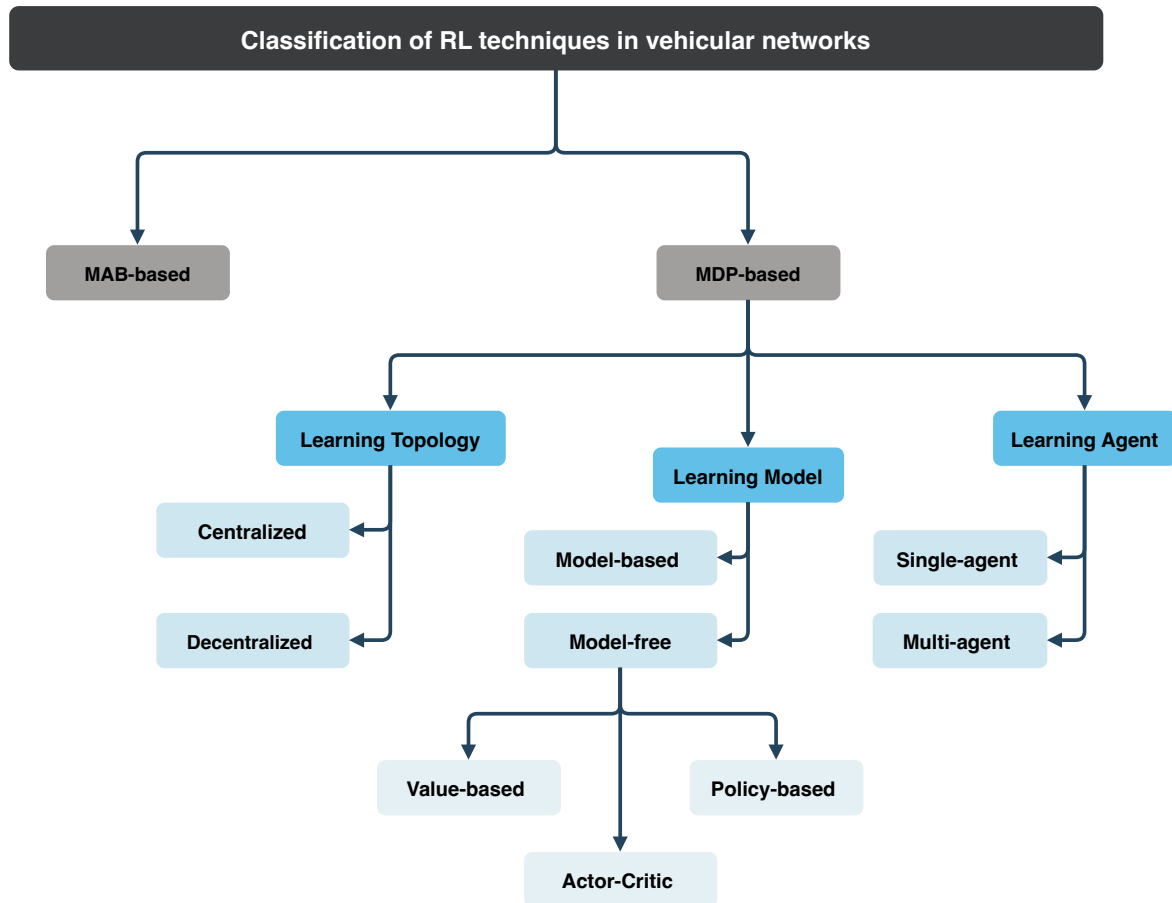


Figure 2.5: Classification of RL techniques

decentralized execution for real-time adaptability and scalability. However, it requires careful synchronization between the central trainer and decentralized agents to ensure that the trained models align with the dynamic conditions of the vehicular network. It is worth noting that, CTDE is a variant of centralized learning and a multi-agent paradigm under the learning agent classification.

- **Decentralized:** In a decentralized learning topology, individual vehicles or clusters of vehicles make decisions autonomously based on local data and observations. This approach fosters scalability and resilience within the network, as each vehicle acts as an independent node capable of evaluating and responding to changes in its environment. Such autonomy is particularly valuable in scenarios where communication with a central entity may be intermittent or unreliable. However, decentralized learning necessitates effective communication mechanisms among network participants to ensure that information is shared efficiently and that the network as a whole benefits from collective intelligence. Similar to CTDE, decentralized learning is inherently a multi-agent paradigm. Figure 2.6 illustrates the distinction between fully decentralized learning and CTDE in a vehicular network setting.

Learning Model

RL techniques can be broadly categorized into two main groups based on their learning model: model-based and model-free, as illustrated in Figure 2.7.

- **Model-Based:** In Model-Based Reinforcement Learning (MBRL), a predictive model of the environment is created and utilized by the agent. The agent learns not only from direct interactions with the environment, but also by developing an internal model that captures the dynamics and relationships between states, actions, and rewards. Using this model, the agent can simulate possible scenarios and plan ahead, enabling more informed decisions. MBRL is a powerful approach that approximates the environment dynamics, allowing learning to occur with guaranteed stability [128] or with minimal interactions [129], [130]. A key advantage of MBRL is its superior sample efficiency compared to model-free methods, as the learned model can be reused to solve a variety of tasks within the same environment. Additionally, MBRL methods such as Dyna-Q enable agents to plan with the learned model and evaluate potential outcomes before taking actions in the real environment. However, MBRL also presents challenges, as inaccuracies in the learned model can lead

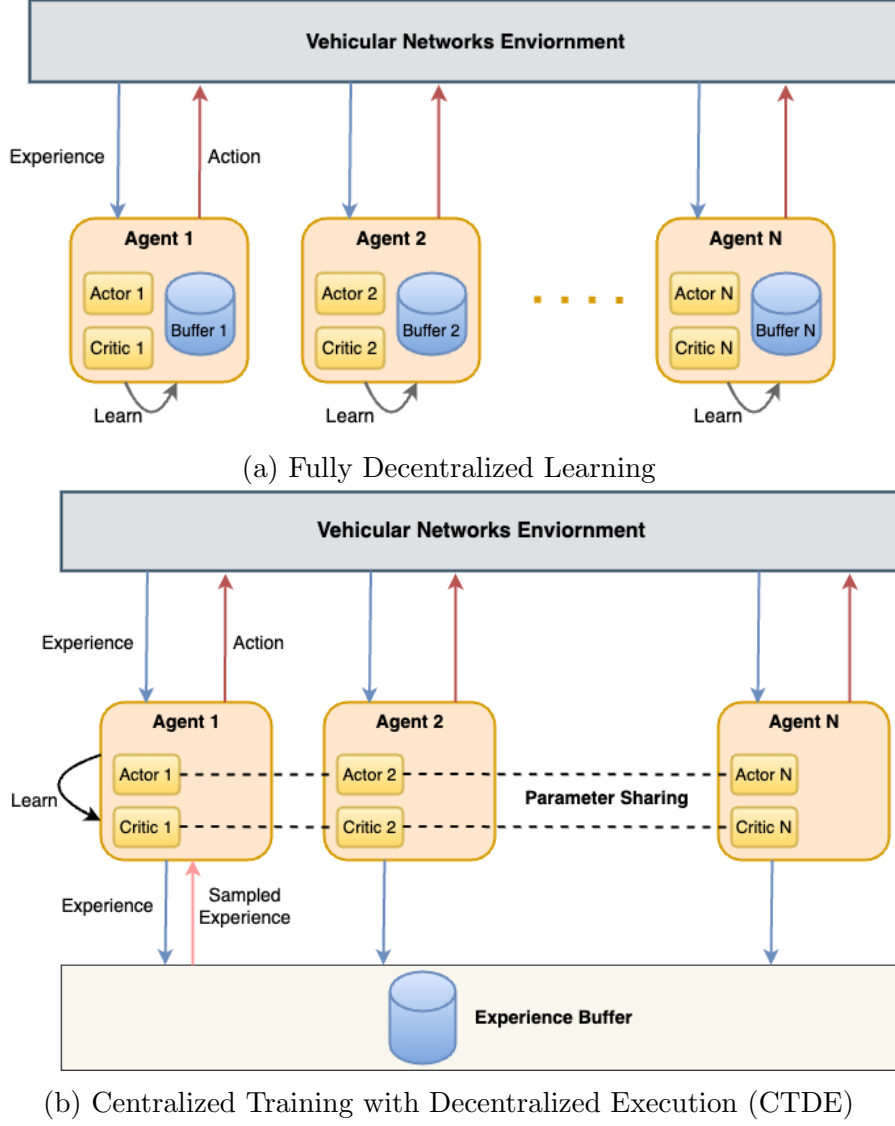


Figure 2.6: Comparison of multi-agent learning topologies: (a) Fully Decentralized Learning and (b) Centralized Training with Decentralized Execution.

to suboptimal decisions, and model learning itself can be computationally costly [131], [132].

- **Model-Free:** In contrast to MBRL, Model-Free Reinforcement Learning or MFRL does not require the agent to learn or maintain a model of the environment’s dynamics. Instead, the agent learns directly by interacting with the environment, collecting rewards, and updating its value function or policy estimates [105], [133]. MFRL techniques have become increasingly popular due to their applicability in complex and unfamiliar environments such as vehicular

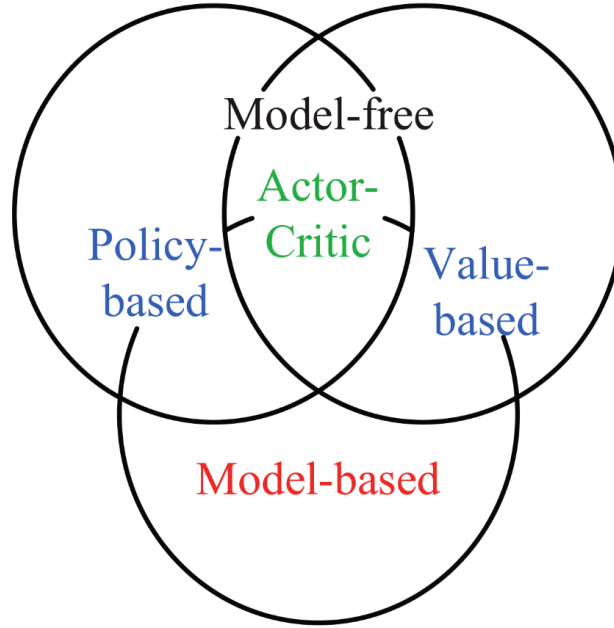


Figure 2.7: Model-based vs. model-free RL methods [127]

networks, where accurate modeling of state transitions and reward generation is not always feasible. Within MFRL, various subcategories emerge: value-based, policy-based, and actor-critic methods.

- **Value-Based:** Value-based RL methods enable agents to learn optimal strategies by estimating the optimal state-action value function, referred to as the Q-function (Q^*). These methods indirectly derive an optimal policy by identifying the most rewarding actions within a given state space. Q-learning is a foundational value-based algorithm, while its deep extensions, DQN, Double DQN, and Dueling DQN, as defined and explained earlier in this section, address the scalability limitations of tabular Q-learning.
- **Policy-Based:** In contrast to value-based methods, policy-based RL methods learn the optimal policy directly without estimating a value function. The policy is expressed as a parameterized function $\pi(a|s; \theta)$, where π denotes the probability of taking action a in state s with parameters θ . Policy-based methods typically employ gradient ascent to iteratively update the policy parameters in the direction that maximizes the expected cumulative reward. Based on the type of policy, these methods can be further classified into stochastic policy gradient methods, which

output a probability distribution over actions, and deterministic policy gradient methods, which output a single fixed action and are particularly suited to continuous action domains [134]. PPO, as defined earlier in this section, is a notable policy-based algorithm that has been widely adopted for its training stability.

- **Actor-Critic:** Actor-critic methods combine value-based and policy-based approaches by maintaining two components: an actor that learns the policy, and a critic that estimates the value function to evaluate the actor’s performance [135]. The actor adjusts its policy parameters based on feedback from the critic, while the critic updates its value estimates using temporal difference learning. By concurrently optimizing the policy and estimating the value function, actor-critic methods enhance learning stability and effectiveness in challenging environments. Notable actor-critic algorithms include Actor-Critic (A2C), Asynchronous Advantage Actor-Critic (A3C), which utilizes multiple agents running in parallel to accelerate the learning process [136], [137], and Soft Actor-Critic (SAC), which incorporates entropy regularization to encourage exploration. DDPG, as defined earlier in this section, is also an actor-critic method that extends policy gradient approaches to continuous action spaces.

Learning Agent

The learning agent dimension classifies RL techniques based on the number of agents involved in the learning and decision-making process. The nature of these learning agents can vary significantly depending on the network architecture and the problem being addressed, and they can be generally classified into two primary categories:

- **Single-Agent:** Single-agent RL involves a single entity responsible for making decisions within the environment. In vehicular networks, this entity is typically a centralized component such as an SDN controller, cloud or a MEC server, which serves as the decision-maker for the entire network or a specific segment of it. The agent makes decisions based on the collective information and state of the network rather than relying on individual vehicles. This approach can lead to a more coordinated system, where a central entity can efficiently manage policies, resource allocation, and response strategies for the network.

- **Multi-Agent:** In multi-agent RL, multiple agents are deployed across the network, each learning its own policy while considering the actions and observations of other agents. In vehicular networks, multiple vehicles, RSUs, or edge servers often need to make coordinated or independent decisions simultaneously. To address this, MARL is modeled as a Markov game, where agents simultaneously interact with each other and the environment, as formally defined earlier in this section.

2.4 Blockchain Technology

2.4.1 Fundamentals of Blockchain

Blockchain is a Distributed Ledger Technology (DLT) that integrates cryptographic techniques, such as hashing and digital signatures, with consensus mechanisms to enable a decentralized network of participants to cooperatively maintain an immutable and ordered record of transactions [55], [138]. A blockchain consists of a sequence of blocks, where each block comprises a block header and a block body. The block body contains a set of validated transactions, while the block header stores metadata including a timestamp, the Merkle tree root, and the cryptographic hash of the previous block's header. As shown in Figure 2.8, each block is linked to its predecessor through this hash value, forming an interlinked chain. This chaining structure ensures that if a transaction in any block is modified, the corresponding block header changes, producing a hash value that differs from the one stored in the subsequent block, making any tampering immediately evident [139], [140]. Newly generated blocks are continuously appended to the chain through a consensus process, in which participating nodes collectively validate and agree on the legitimacy of new transactions before they are recorded on the ledger. This process is designed to be computationally expensive to perform but straightforward to verify, ensuring that malicious entities cannot feasibly alter historical records [141].

A key innovation enabled by blockchain is the concept of smart contracts. Smart contracts are self-executing and immutable programs with predefined logic that are deployed on the blockchain and automatically enforce agreed-upon rules when specific conditions are met, without the need for intermediaries or third-party organizations to establish a trustworthy environment [142]. A smart contract interacts with users through exposed public functions to offer predefined business logic. It can execute read-write operations that alter data on the blockchain and store the results on the ledger itself, or operate in a read-only mode for computational purposes

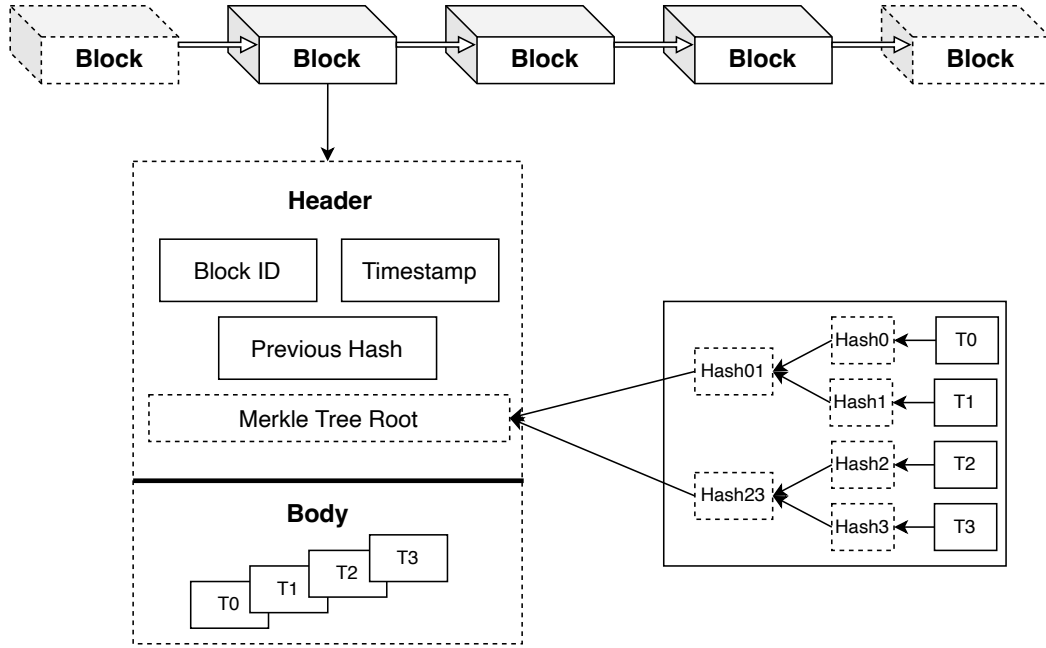


Figure 2.8: Structure of a blockchain

[140]. In the context of vehicular networks, smart contracts provide a mechanism to automate complex tasks and enable a large number of vehicles and infrastructure nodes to interact intelligently [142].

Consensus Mechanisms

Consensus mechanisms are fundamental in blockchain, as they enable distributed participants to agree on the current state of the ledger without relying on a central authority [141]. The choice of consensus mechanism directly impacts the performance, scalability, energy efficiency, and security of the blockchain system. The most prominent consensus mechanisms are described below.

- **Proof-of-Work (PoW):** PoW, first introduced in Bitcoin [143], requires validators (referred to as miners) to compete in solving complex cryptographic puzzles in order to append a new block to the chain. The first miner to find a valid solution broadcasts the block to the network for verification. This process ensures that attacking the network would require an immense amount of computational power, making it economically infeasible. However, PoW suffers from high energy consumption and limited transaction throughput, which restrict its suitability for latency-sensitive applications such as vehicular networks [138].

- ***Proof-of-Stake (PoS)***: In contrast to PoW, PoS selects validators based on the amount of cryptocurrency they stake in the system rather than computational effort. Participants who stake more funds receive a higher probability of being chosen to validate transactions, while they risk losing their stake if they engage in malicious activity. PoS significantly reduces energy consumption compared to PoW and offers improved transaction throughput. However, PoS can introduce centralization risks, as participants with larger stakes have a disproportionate influence on the validation process [142]. A notable variant is Delegated Proof-of-Stake (DPoS), in which stakeholders vote to elect a fixed number of delegates who are responsible for validating transactions and maintaining the blockchain on behalf of the network [141].
- ***Practical Byzantine Fault Tolerance (PBFT)***: Byzantine Fault Tolerance (BFT) refers to the ability of a distributed system to reach consensus despite the presence of malicious or faulty nodes, known as Byzantine faults. PBFT [144] is the most widely adopted BFT algorithm, designed for permissioned networks with a known set of validators. In PBFT, consensus is reached through a multi-round voting process among validators, tolerating up to one-third of malicious or faulty nodes (i.e., the system requires at least $3f + 1$ nodes to tolerate f Byzantine faults). PBFT offers low latency and high throughput compared to PoW and PoS, making it well-suited for vehicular network applications that demand fast transaction finality. However, PBFT faces scalability limitations as the communication overhead grows quadratically with the number of validators, restricting its applicability to networks with a relatively small number of nodes [55], [145].

Blockchain Types

Based on their participation and governance models, blockchains can be broadly categorized into four types, as illustrated in Figure 2.9 [145], [146]:

- **Public blockchains** are fully open systems where any participant can join the network as a validator or user without requiring approval from a central authority. Public blockchains are decentralized and non-permissioned [147]. Notable examples of public blockchains include Bitcoin, Ethereum, and Litecoin [148].

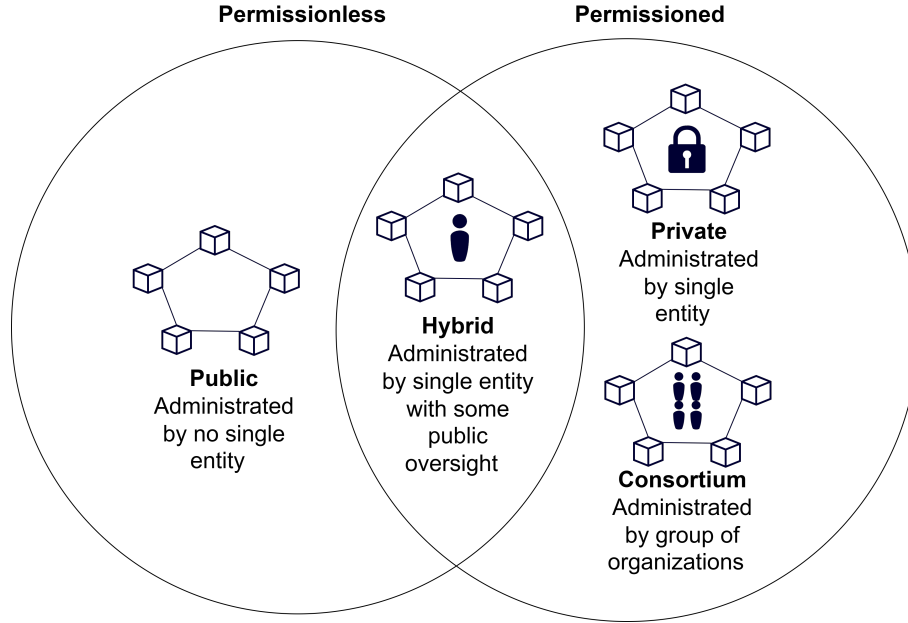


Figure 2.9: Different types of blockchain [146]

- **Private blockchains** are permissioned systems where participation is restricted to authorized entities verified by a central authority. Private blockchains include Hyperledger [149], Corda [150], and Ripple.
- **Consortium blockchains** represent a middle ground between public and private blockchains. They are permissioned networks governed by a group of organizations rather than a single central authority. Validator nodes are selected from a predefined set of trusted entities [139].
- **Hybrid blockchains** combine elements of both public and private blockchains, allowing selective transparency where certain data and transactions are made publicly accessible while sensitive information remains restricted to authorized participants [145].

2.4.2 Blockchain in Vehicular Networks

Motivation and Benefits

The adoption of blockchain technology in vehicular networks is primarily driven by its potential to address the inherent security, privacy, and trust challenges of decentralized vehicular communication systems [138], [151]. The open and dynamic nature of vehicular networks, combined with the absence of a persistent centralized authority, makes them susceptible to various attacks including data tampering,

identity spoofing, and false information dissemination, as discussed in Section 2.2. Blockchain offers key features designed to eliminate single points of failure, the reliance on trusted third parties, and other vulnerabilities [151].

First, the immutability of blockchain ensures that once data is recorded on the ledger, it cannot be altered or deleted, providing tamper-resistant storage for critical vehicular data such as trust scores, reputation records, and safety messages [145]. Second, the decentralized architecture of blockchain eliminates single points of failure, distributing control across multiple nodes and thereby enhancing network resilience [152]. Third, the transparency of blockchain enables all authorized participants to verify recorded transactions, supporting real-time auditing and accountability in scenarios such as accident forensics and misbehavior detection [142]. Fourth, smart contracts deployed on the blockchain can automate enforcement of security policies, trust evaluation rules, and pseudonym management procedures without requiring manual intervention [140], [153]. These properties collectively establish blockchain as a promising foundation for building secure, transparent, and trustworthy vehicular network services.

Network Architecture

Blockchain-enabled vehicular networks typically adopt a multi-layer architecture that integrates V2X communication with blockchain infrastructure [154]. Due to the limited resources and intermittent connectivity of CAVs, many designs rely on supporting infrastructure, such as RSUs or edge servers, which act as blockchain nodes within a hierarchical, multi-layer framework [142]. Generally, the blockchain layer operates between the networking layer (utilizing technologies such as 5G, DSRC, or C-V2X) and the application layer, as depicted in Figure 2.10 [155].

Key Applications in Vehicular Networks

Blockchain technology enables a range of applications in vehicular networks. The following highlights some of the applications, with a particular focus on the ones most relevant to the scope of this thesis:

- **Secure Data Exchange:** Vehicles can exchange essential information, such as accident notifications, hazard alerts, and sensor readings, using a blockchain-based framework that helps preserve data integrity. In this setting, multiple nearby vehicles can independently verify event reports, and the agreed-upon result is stored on the blockchain as an immutable record. This process helps ensure that misleading or malicious messages are detected and discarded. For

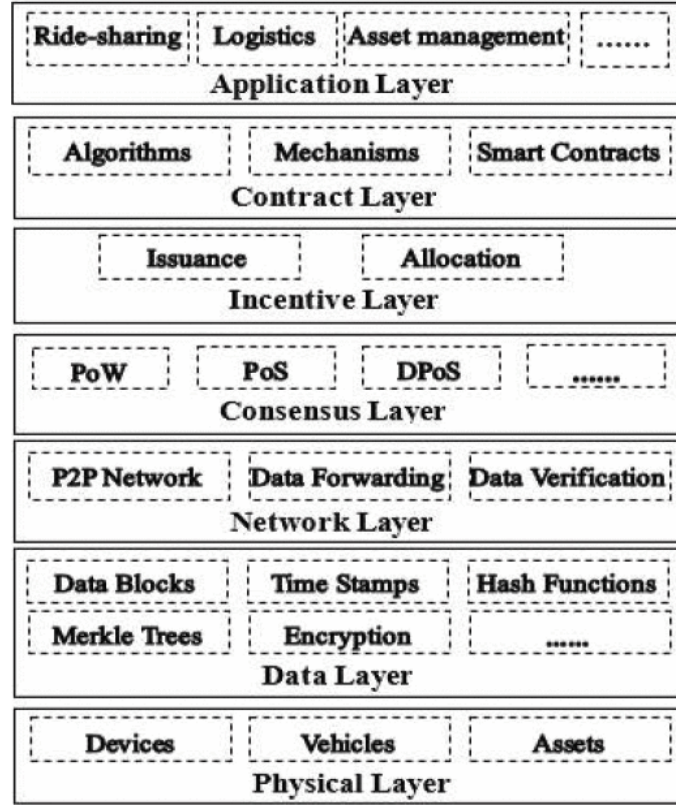


Figure 2.10: Layered view of blockchain in the vehicular networking stack [155].

instance, when an accident is reported, surrounding vehicles may jointly confirm its validity, and the RSU can then record the verified consensus as a transaction in a new block. Vehicles can share critical information, such as accident alerts, hazard warnings, or sensor data, in a tamper-proof manner through blockchain. Multiple vehicles can cross-verify event reports, and the consensus result is recorded on the immutable ledger, ensuring that false or malicious messages are identified and rejected. For instance, when an accident is reported, surrounding vehicles may jointly confirm its validity, and the RSU can then record the verified consensus as a transaction in a new block [156].

- **Trust Management:** Beyond secure data sharing, the trustworthiness of exchanged information is equally critical. In this context, a blockchain-based distributed trust and reputation management system provides a decentralized and tamper-resistant framework for recording, sharing, and updating trust scores based on vehicle behavior and data credibility. By storing reputation records on the blockchain, the system ensures that trust information remains transparent, auditable, and resistant to manipulation [140]. This application

is directly relevant to the reputation management framework presented in Chapters 6 of this thesis.

- **Privacy-Preserving Identity Management:** Protecting user privacy in vehicular communications is also vital. In this context, blockchain-based privacy-preserving frameworks utilize advanced cryptographic mechanisms, such as pseudonymity, zero-knowledge proofs, and homomorphic encryption, to safeguard vehicle identities and location information. Blockchain further supports the management of pseudonymous identities, allowing vehicles to communicate without disclosing their real identities while still ensuring accountability through its inherent traceability. This approach aligns closely with the pseudonym management frameworks presented in Chapters 5 and 6 of this thesis.
- **Resource Marketplace** In addition, blockchain enables a resource marketplace where vehicles trade computational power, storage, and energy. In this ecosystem, resource-constrained vehicles offload intensive tasks like data pre-processing to neighboring nodes or edge infrastructure [157], while electric vehicles (EVs) engage in vehicle-to-grid (V2G) transactions to buy or sell electricity, ensuring stable energy supply through smart contracts and immutable settlements.
- **Toll Payments** Blockchain supports seamless toll payments and microtransactions for vehicular services [158]. Vehicles equipped with digital wallets can automatically settle tolls, parking fees, or charging costs via smart contracts [159], such as triggering payments at toll gates and logging immutable transactions on the ledger to prevent fraud, enable pay-per-use models like congestion charges, and eliminate intermediaries.

Chapter 3

Related Works

This chapter presents a comprehensive review of existing approaches to SPT provisioning in vehicular networks, with the dual goal of surveying the state of the art and addressing research question **RQ1a**, which concerns the identification of key challenges in ensuring security, privacy, and trust within vehicular networks. In this regard, two complementary taxonomies are developed and used to systematically structure the review: a focus-centric taxonomy and a taxonomy based on RL method type and SPT protection type. The focus-centric taxonomy classifies the literature based on primary focus areas within vehicular networks, prioritizing the specific aspects each approach aims to enhance and it is discussed in Section 3.1. The second taxonomy classifies approaches based on the specific type of RL method employed, highlighting how each technique contributes to enhancing SPT in vehicular networks, while also categorizing studies according to the type of SPT protection they address and it is presented in Section 3.2. Through these two complementary classifications, this chapter addresses the first research question: *”What are the key challenges in ensuring security, privacy, and trust within vehicular networks? (RQ1a)”*. Both taxonomies were originally proposed in [2]. In addition, Section 3.3 reviews non-RL approaches closely related to the contributions presented in this thesis, with particular emphasis on trust-aware routing protocols, pseudonym management schemes, and reputation management schemes.

3.1 RL-Based Solutions for Enhancing SPT in Vehicular Networks: A Taxonomy Based on Focus Area

Following the overview of RL methodologies presented in Chapter 2, such as DRL, FRL, and Safe RL, it is now appropriate to examine existing approaches in vehicular networks that utilize RL techniques to improve security, protect privacy, and enhance trust management. The purpose of this section is to systematically review the existing literature in this field in order to categorize the approaches according to their ability to mitigate different vulnerabilities. The proposed focus-centric taxonomy in SPT provisioning is depicted in Figure 3.1. For this purpose, the discussion is divided into three subsections. RL-based security approaches, RL-based trust management approaches, and RL-based privacy protection approaches, with three corresponding tables (Tables 3.1, 3.2, and 3.3). In all three tables, the literature is categorized based on its focus centric. Six primary limitation categories are also included. The six limitation categories are defined as follows. Memory inefficiency (M) denotes approaches that require bulky in-RAM structures, such as large replay buffers, multiple DNNs, Q-tables, or consensus histories, that exceed typical onboard or edge RAM. High computational cost (C) captures per-step operations (e.g., DNN gradient updates, consensus or blockchain protocols) that can overload constrained processors. High training overhead (T) flags methods needing thousands of offline episodes or very long online learning before their policies stabilize. Simplified scenario (S) applies to evaluations conducted on simple topologies or simplified models rather than realistic, dynamic environments. Scalability limitations (L) identify algorithms whose state/action spaces, hierarchical levels, or sequential update loops grow super linearly with network size, preventing deployment at scale. Finally, Global state aggregation (G) marks design that depends on a single logical aggregator to collect all agents' observations or model updates, thereby introducing latency, signaling overhead. This challenge can occur even in decentralized schemes that use decentralized execution and centralized training (or a common coordinator in federated setups) to aggregate updates. All these defined challenges can critically undermine the practical deployment of RL-based approaches in vehicular networks.

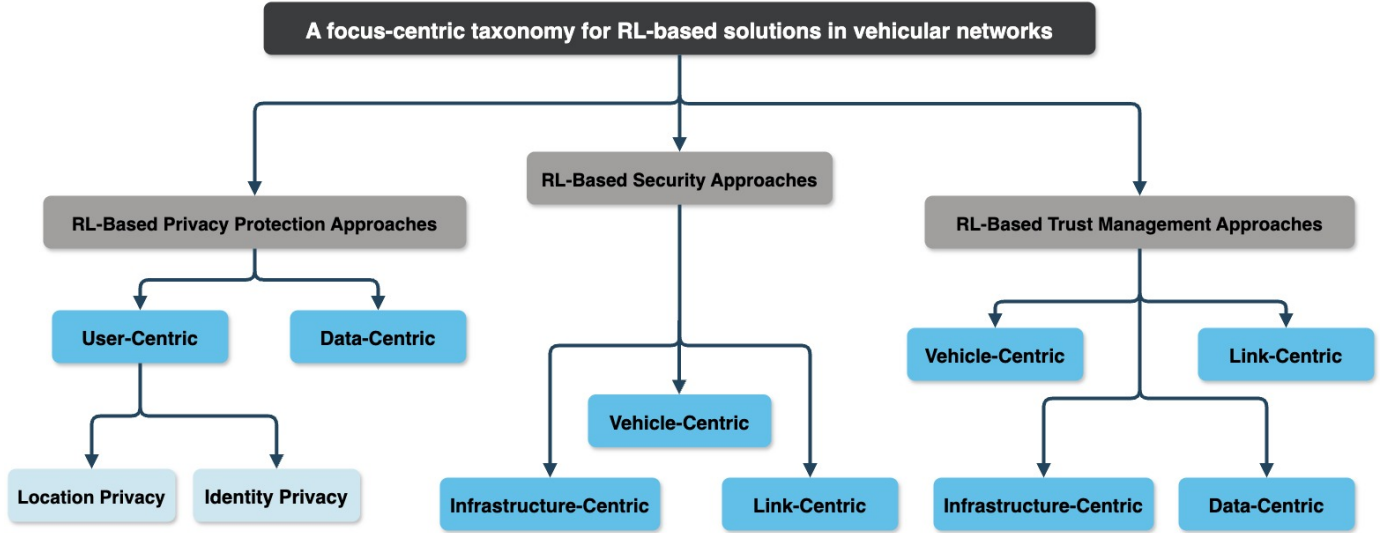


Figure 3.1: A taxonomy based on focus area

3.1.1 RL-based security Approaches

The objective of this subsection is to categorize various vehicular networks attack targets over V2X communication. Several RL-based techniques have been applied in order to improve security in vehicular networks. These techniques are discussed in detail below, while Table 3.1 provides a summary. Security attacks over V2X communication can be categorized based on their target in: vehicle-centric, infrastructure-centric and link-centric. Each of these categories can be affected by various attack types, including Distributed Denial of Service (DDoS), eavesdropping, jamming, and spoofing. In Table 3.1, both the focus-centric features of each approach and the specific attacks addressed are listed.

Vehicle-centric RL-based Approaches for security

Vehicle-centric RL-based approaches focus on enhancing the security of individual vehicles by monitoring their behaviors and interactions within the network. Moreover, blockchain technology holds considerable promise and is experiencing a growing adoption in fortifying vehicular application security [160]. In [161], an innovative approach was introduced that combined blockchain technology with MADRL, effectively increasing the efficiency of Cooperative Adaptive Cruise Control (CACC) within AVs. This setup secured vehicles, specifically targeting malicious activities like message forgery and acceleration falsification that could directly affect the CACC function within each vehicle. Notably, the algorithm developed in that work served a dual purpose: enhancing CACC's performance by optimizing fuel con-

sumption and throughput, while concurrently collaborating in the identification of disruptive jamming attacker vehicles within the network. A distinctive aspect of the proposed algorithm was its utilization of blockchain to securely transmit attacker data to RSUs. This methodology further integrated MADRL techniques to determine configurations that optimized fuel usage and throughput for CACC. The proposed algorithm utilized a neighborhood-based approach to detect acceleration forgery attacks by integrating a cooperative verification mechanism. This verification process involved vehicles taking turns to assess safety messages within their proximity, cross-referencing reported values with sensor-based estimates. The computed differences were transmitted to the nearest RSU, where a low-complexity Extended Isolation Forest assessed the vector, allowing it to gauge the likelihood of an attack.

In order to detect spoofing attacks, [162] proposed an RL-based authentication framework for the Controller Area Network (CAN) bus, specifically within the vehicle's in-vehicle network. This vehicle-centric scheme dynamically selected the authentication mode without prior knowledge of authentication or spoofing models. Electronic Control Units in CAN buses typically send and receive messages without authentication or encryption, making them vulnerable to spoofing attacks. To address this, the proposed framework used a monitor node to extract voltage signals from recent messages and verified these against historical message identifiers to confirm their legitimacy. Optimizing authentication parameters such as protection level and test threshold was challenging in CAN networks with unknown spoofing models. To overcome this, the authors employed a Q-learning-based approach, treating the dynamic authentication process as a MDP. Experimental results showed that this framework significantly improved security performance by reducing the false alarm rate by 22.2%, decreasing the miss detection rate by 34.3%, and increasing utility by 8.8% by the 2000th time slot. Furthermore, in [163], RL was applied to optimize the authentication for controller area network bus communication, which managed in-vehicle communication among Electronic Control Units. The proposed approach utilized RL to select a mode of authentication that captured the information from the physical layer (i.e., arrival intervals and signal voltages of the messages received) and the test threshold required to detect spoofing messages. On the basis of a hierarchical structure, a deep RL version was also designed to improve the accuracy of authentication and to reduce the optimization latency.

Infrastructure-centric RL-based Approaches for Security

Infrastructure-centric RL-based approaches prioritize the security and reliability of network infrastructure components, such as RSUs, edge servers, and centralized controllers, within vehicular networks. For instance, [164] aimed to protect 5G wireless networks against attack techniques, such as jamming and DDOS. The authors proposed an HRL method for identifying attacks targeting various network entities (access points, base stations, etc.), making it an infrastructure-centric approach. Networks were monitored for anomalies by Intrusion Detection Systems (IDS). The identification of attackers involved two steps: the First Level-IDS and the Second Level-IDS. The first Level-IDS monitored the user equipment to detect anomalies using a one-class support vector machine classifier; however, the verification of detected anomalies was performed by the SL-IDS to determine whether or not the target had been compromised by an attacker. In [28], DDOS attacks were investigated using fuzzy and Q-learning methods in the context of IoV networks. Different clusters were considered where the sensor nodes collected data from end users and transmitted it to the cluster head. Packets in real-time network traffic were held by the packet capturing module and analyzed by fuzzy logic. In the next step, the system utilized the Q-learning algorithm to make a decision about DDOS attack detection, and log files were saved accordingly. Additionally, the interaction between attackers and inspection modules was evaluated based on buffer size usage, communication channel efficiency, response time, and throughput. In [165], a new method utilizing a transfer DDQN approach for detecting DDOS attacks on the IoV was proposed. The first algorithm in that work utilized an RL approach with two deep neural networks, in which the base stations already in place acted as agents to estimate the action-value function. Additionally, a transfer DDQN method based on traffic flow similarity was designed for new base stations, using knowledge from adjacent similar base stations to expedite training. The method was evaluated for detecting DDOS attacks in IoV by simulating traffic attack flows using the public dataset “CIC-DDoS2019”¹.

The authors of [166] aimed to tackle the susceptibility of MEC-enabled SDN-based VANET to DDOS attacks. They developed a collaborative DRL approach that allocated computing resources and mitigated edge DDOS attacks. This was accomplished through the evaluation of vehicle trust values and making informed offloading decisions for vehicular computing tasks. Considering the paper’s emphasis

¹<https://www.unb.ca/cic/datasets/ddos-2019.html>

on protecting MEC servers and SDVN infrastructure, this approach can be classified as an infrastructure-centric approach. In the context of SDN and IoV, a novel approach was introduced by [167] to combat DDOS attacks through an intelligent moving target defense scheme by using policy-based PPO. In this strategy, network configurations within RSUs were periodically altered using DRL. Additionally, the trust level of vehicles was assessed by dynamically rearranging the association between the RSUs and the vehicles. By incorporating multiple constraints into the evaluation process, the trust assessment mechanism enhanced the capability of identifying suspicious vehicles.

According to [168], RL-based anomaly detection, specifically the threat of DoS attacks, was implemented at the edge RSUs, which were recognized as agents that interacted with the IoV environment to determine the best detection policy. The approach addressed attacks on multiple IoV infrastructure components. A Q-learning algorithm was adopted, and an open-source dataset [169], was used for evaluation. The reward offered to an agent for an action performed in a particular state was determined by the ground truth values of basic safety messages. A positive reward was given if the agent correctly detected an attack or a normal state (true positive or true negative), and a negative reward was given if it incorrectly identified a normal state as an attack or an attack as a normal state (false positive or false negative). To evaluate the RL algorithm's detection performance, accuracy, precision, recall, and F1 score were used as the most common metrics.

Regarding eavesdropping, [170] proposed a single-agent and multi-agent RL approach in UAV-assisted MEC networks. This scheme incorporated a spiral placement algorithm for optimal UAV deployment, ensuring efficient user coverage with a minimal number of UAVs. The primary goal was to mitigate information eavesdropping from airborne threats while optimizing secure offloading for maximum system utility. This approach accounted for varying user task preferences, the residual energy of computing equipment, and processing delays, effectively securing the infrastructure within a UAV-assisted MEC environment. Similarly, [171] introduced an innovative online edge learning offloading approach aimed at enhancing secure communication performance in UAV-assisted MEC settings. The scheme employed DRL to improve secure computation efficiency by optimizing binary offloading decisions and resource allocation for terminal users. Additionally, it addressed security concerns related to UAV line-of-sight transmissions, which could expose offloading information of terminal users to eavesdropping. Both approaches focused on securing infrastructure-centric aspects within UAV-assisted MEC systems to counter

potential eavesdropping risks. In [14], eavesdropping in VEC technology was also addressed through a joint secure offloading and resource allocation approach based on DRL to reduce system processing latency while ensuring the wireless offloading process, using physical layer security technology and spectrum-sharing architecture. This approach focused on securing the MEC infrastructure within a VEC network.

Link-centric RL-based Approaches for Security

Link-centric RL-based approaches focus on securing communication links within vehicular networks, ensuring reliable data transmission between vehicles and infrastructure. By applying RL, these approaches adaptively assess and select secure routing paths, mitigating attacks such as eavesdropping and jamming.

On the other hand, as discussed in Section 2.1, the usage of MEC in vehicular networks presents security challenges, and malicious vehicles can pose a threat to the network. Consequently, several studies have proposed various methods to counter different attacks in vehicular networks utilizing MEC. For example, Li *et al.* [172] introduced a method called FAST and utilized RL to protect communication links between vehicles and base stations from DDoS attacks within the IoV under a MEC framework. The authors developed a specialized action space and reward function that integrated the Kalman filter method with historical traffic data, enabling FAST to rapidly and accurately detect DDoS attacks by leveraging space-time flow regularities in the IoV. To handle the dynamic nature of DDoS attacks without relying on labeled data, Q-learning was implemented to facilitate efficient attack mitigation. Additionally, Double DQN enhanced Q-learning by allowing FAST to learn from prior experiences, thereby optimizing its performance over time.

[24] aimed to secure the communication channels in MANETs against jamming attacks. The DQN model was designed to help nodes identify optimal time slots for transmitting data to avoid interference, directly targeting the protection of the communication link between nodes. In this approach, the transmitter needed to determine when to send packets in order to avoid jamming. Q-learning was employed to learn the history of actions based on past experience, and then the data were entered into the DQN to predict Q-values for the current state. As a result of this strategy, the transmitter was able to make more efficient decisions. The authors in [173] presented a Dyna-Q RL approach for preventing jamming in VANET platoons by focusing on securing communication links. In this method, vehicles utilized V2V communication through periodic beacons, allowing the transmitter to make more efficient decisions under jamming conditions. In this approach, jamming consumed

channel energy and prevented data from being transmitted efficiently between vehicles. Without knowing the jammer's strategy, each vehicle used Dyna-Q RL to independently learn the optimal power control policy that maximized its utility. The proposed RL method calculated channel utility and used it to control VANET power through a historical study. Compared with Q-learning, the algorithm not only achieved optimal utility but also increased the Signal-to-Interference-plus-Noise Ratio (SINR) of the channel. Similarly, the authors in [174] presented an anti-jamming V2V and V2I communication links scheme for VANET based on environment-aware RL. They aimed to optimize the power allocation and channel selection to combat jamming and interference, without relying on a jamming attack model. The post-decision state algorithm was employed to incorporate environment information such as traffic density, vehicle location, and building structure and height into the optimization process. The authors used estimated future communication distances of moving vehicles to accelerate the learning process. The simulation results showed that the proposed scheme significantly reduced both energy consumption and bit error rate.

Another study explored the use of RL for anti-jamming measures involving power control and channel selection. The work in [175] focused on maintaining secure and reliable V2V communication links in vehicular networks despite jamming threats. It achieved this by implementing a cognitive dynamic system for adaptive power control and channel selection, utilizing a brain-inspired cognitive risk control framework to address jamming challenges effectively.

Xiao *et al.* [176] proposed using a UAV as a relay to enhance VANET communication links, particularly under smart jamming conditions. UAVs were employed to relay messages from OBUs to RSUs without knowing the VANET model and jamming model, thereby avoiding jamming. The proposed algorithm utilized a Policy Hill-Climbing (PHC) strategy, which was less predictable, resulting in more uncertainty for the jammer to exploit. Due to this, the jammer could not determine the optimal relay policy and therefore could not attack the UAV accordingly. The OBU sent a message to RSU1, and the UAV connected with RSU2. Depending on the quality of the channels and the BER, the UAV decided whether to relay the message to RSU2. In this case, the UAV transmitted the message to another RSU in better condition and not exposed to jamming. As a result, the UAV and the smart jammer engaged in a game of anti-jamming. By using RL, they developed a hot-booting policy for hill climbing-based UAV relay strategies to strengthen VANETs against

jamming. According to the proposed scheme, the BER of the OBU message was reduced by 65.7%, and the utility of the UAV increased by 42.0%.

Lv *et al.* [177] presented a MARL approach for optimizing relay selection and power allocation in UAV swarms to counteract jamming. The approach specifically aimed to improve the reliability and stability of communication links between UAVs in a dynamic swarm, focusing on relay efficiency and link robustness against jamming interference. To improve exploration and stability of the learning process, the proposed scheme utilized network topology, channel states, historical performance, and observations shared by neighboring UAVs. To facilitate initial learning, transfer learning was employed by utilizing shared swarm experiences. A deep RL-based approach was proposed to reduce state quantization errors resulting from rapid changes in channel state during high-speed swarm movement, thereby improving the effectiveness of anti-jamming methods. The scheme utilized network states from neighboring UAVs in both request-to-send and clear-to-send messages for learning state formulation. The learning process was guided by expected long-term utility (Q-value) and average policy distribution. The learning rate was adaptively adjusted based on previous and current Q-values, facilitating swift learning under dynamic jamming policies.

In another study aligned with the communication link-centric category, [178] proposed a MARL framework based on DDPG for spectrum allocation and UAV trajectory optimization in a dynamic spectrum-sharing network with anti-jamming capabilities. In this approach, secondary user agents independently learned spectrum allocation policies using A2C, while the UAV agent optimized its trajectory policy via DDPG. Training was conducted in a decentralized manner with decentralized execution, where agents collaborated through a shared global reward signal. In addition, [111] proposed a method to select the frequency channel and transmit power of UAVs. A safe RL algorithm was introduced, aimed for enhancing the efficiency of learning and the security of exploration in wireless security applications. This approach utilized a hierarchical structure based on policy priority, dividing policies into sub-policies based on their priority of selection. In [179], a novel defense approach against jamming attacks in flying ad-hoc networks (FANETs) was proposed to protect communication links between UAVs. The study presented an adaptive FRL-based strategy specifically designed for jamming defense within UAV nodes. Additionally, the authors introduced a Q-learning-based spatial retreat defense strategy driven by an epsilon-greedy policy, which built upon an FL jamming detection mechanism to enhance resilience against jamming threats.

Several recent studies emphasized enhancing V2V communication security against eavesdropping attacks. For instance, Yao *et al.* [180], [181], [182] developed an approach that combined Distributed Kalman Filtering (DKF) with DRL techniques. The DKF algorithm improved the accuracy of attacker tracking by allowing adjacent nodes to share state estimates, enhancing the reliability of V2V communication. In [180], the authors introduced a method to boost anti-eavesdropping communication capacity and mitigate jamming interference by implementing a hierarchical DQN for dynamic power control and channel selection policies. Additionally, [181], [182] presented a secure vehicular joint radar-communication system for V2V communication, utilizing a MAB approach to optimize secure communication strategies under threat conditions.

In UAV-enabled secure communication systems, ground eavesdroppers [183], [184] were investigated in MEC systems. In [183], the author analyzed the ground-to-UAV (G2U) communication system with a malicious eavesdropper and implemented the DDQN algorithm to maximize the accumulated secrecy rate, while [184] investigated a Q-learning scheme to maximize the average secrecy rate against intentional or unintentional eavesdropping. The approach aimed to enhance the security of the UAV communication link by optimizing transmit power and UAV positioning.

In addition, [185] proposed an RL-based physical-layer authentication scheme to secure V2V communication in VANETs against rogue edge attackers sending spoofing signals. The scheme exploits both the physical properties of received signals and ambient radio signals to detect spoofing, using Q-learning combined with the Dyna-Q architecture to accelerate convergence through simulated experiences. Transfer learning is further employed to initialize authentication policies from prior experiences in similar VANET scenarios. A DRL extension is also proposed for mobile devices with greater computational resources. Compared to earlier RL-based baselines, the scheme reduces the miss detection rate by 76.08% and the false alarm rate by 70.44% after 1000 time slots.

Table 3.1: Summary of Literature on RL-enabled Security in Vehicular Networks

Ref.	Scope	Addressed At- tack(s)	Comm. type	Category	Algorithm	Contextual Usage	Limitations & Challenges					
							M	C	T	S	L	G
[165]	Infra- Centric	DDOS	V2I	Availability	Transfer DDQN	SDN-based IoV	✓	✓			✓	✓
[167]	Infra- Centric	DDOS	V2I	Availability	PPO	SD-IoV		✓	✓			✓
[172]	Link- Centric	DDOS	V2I	Availability	Q-learning and DDQN	IoV under MEC	✓	✓			✓	
[166]	Infra- Centric	DDOS	V2I	Availability	DDPG	SDN-based vehicular networks	✓	✓			✓	✓
[168]	Infra- Centric	DOS, Sybil at- tack	V2I	Availability, Authenticity	Q-learning	5G-enabled IoV	✓				✓	
[164]	Infra- Centric	DDOS, Jam- ming	V2V	Availability	Hierarchical RL	Intrusion de- tection in 5G network	✓			✓	✓	✓
[28]	Infra- Centric	DDOS	V2V, V2I	Availability	Fuzzy and Q-learning	Routing in IoV	✓			✓		✓
[177]	Link- Centric	Jamming	V2V	Availability	DRL, MARL	UAV swarm network	✓	✓	✓	✓	✓	

Continued on next page

Table 3.1: Summary of Literature on RL-enabled Security in Vehicular Networks (*continued*)

Ref.	Scope	Addressed At- tack(s)	Comm. type	Category	Algorithm	Contextual Usage	Limitations & Challenges					
							M	C	T	S	L	G
[178]	Link- Centric	Jamming	UAV	Availability	A2C, DDPG	UAV- assisted network	✓	✓		✓	✓	✓
[111]	Link- Centric	Jamming	V2I	Availability	Hierarchical safe DRL	UAV- assisted network	✓	✓		✓	✓	
[161]	Vehicle- Centric	Jamming	V2I	Availability	DDPG	Urban inter- sections with CACC	✓	✓	✓		✓	✓
[174]	Link- Centric	Jamming	V2V, V2I	Availability	Q-learning	VANETs	✓	✓		✓	✓	
[179]	Link- Centric	Jamming	UAV	Availability	FRL, Q- learning	FANETs	✓	✓		✓	✓	✓
[175]	Link- Centric	Jamming	V2V	Availability	MAB	VANETs	✓	✓		✓	✓	
[24]	Link- Centric	Jamming	V2V	Availability	DQN	MANETs	✓	✓	✓	✓	✓	
[173]	Link- Centric	Jamming	V2V	Availability	Dyna-Q RL	VANET- platoon	✓	✓		✓		

Continued on next page

Table 3.1: Summary of Literature on RL-enabled Security in Vehicular Networks (*continued*)

Ref.	Scope	Addressed At- tack(s)	Comm. type	Category	Algorithm	Contextual Usage	Limitations & Challenges					
							M	C	T	S	L	G
[176]	Link- Centric	Jamming	V2I	Availability	PHC	UAV relay in VANETs	✓	✓	✓	✓	✓	
[180]	Link- Centric	Jamming, Eavesdrop- ping	V2V	Availability, Confidential- ity	DQN	CAVs net- work	✓	✓	✓	✓	✓	
[181]	Link- Centric	Eavesdropping	V2V	Confidentiality	MAB	CAVs net- work	✓	✓		✓	✓	
[182]	Link- Centric	Eavesdropping	V2V	Confidentiality	MAB	CAVs net- work	✓	✓		✓	✓	
[183]	Link- Centric	Eavesdropping	V2I	Confidentiality	DDQN	UAV com- munication	✓	✓	✓	✓	✓	✓
[184]	Link- Centric	Eavesdropping	V2I	Confidentiality	Q-learning	UAV com- munication	✓			✓	✓	
[170]	Infra- Centric	Eavesdropping	V2I	Confidentiality	Q-learning, MARL	UAV- assisted network	✓	✓	✓	✓	✓	✓
[171]	Infra- Centric	Eavesdropping	V2I	Confidentiality	DRL	UAV- assisted MEC	✓	✓		✓	✓	✓

Continued on next page

Table 3.1: Summary of Literature on RL-enabled Security in Vehicular Networks (*continued*)

Ref.	Scope	Addressed At- tack(s)	Comm. type	Category	Algorithm	Contextual Usage	Limitations & Challenges					
							M	C	T	S	L	G
[14]	Infra- Centric	Eavesdropping	VEC	Confidentiality	Double DQL	VEC	✓	✓	✓		✓	✓
[162]	Vehicle- Centric	Spoofing	V2V	Authenticity	Q-learning	Controller area network bus				✓		
[163]	Vehicle- Centric	Spoofing	V2V	Authenticity	DQN	Controller area net- works	✓	✓	✓	✓	✓	
[185]	Link- Centric	Spoofing, Rogue node	V2I	Authenticity	Dyna-Q- RL, DRL	VANETs	✓	✓	✓	✓	✓	

3.1.2 RL-based Trust Management Approaches

The vulnerability of vehicular networks makes trust a major challenge. Ensuring the trustworthiness of both sender and receiver entities, as well as the transmitted data, is essential for effective trust management in these networks. This requires robust verification and validation processes [186]. A primary component of trust management is trust computation, which involves calculating a trust value for each vehicle in the network. This trust value represents the level of trustworthiness that can be attributed to a particular vehicle or the data it transmits.

This subsection discusses trust management in detail and reviews existing trust models using RL in vehicular networks. Table 3.2 provides a summary of these approaches. The classification is organized based on the focus of trust, categorized as vehicle-centric, infrastructure-centric, and data-centric approaches. In addition, three methods of trust management are distinguished: centralized, decentralized, and blockchain-based approaches [187]. This structure allows for a comprehensive analysis of existing trust management approaches, highlighting both their primary trust focus and management method.

Vehicle-Centric Trust Management

In vehicle-centric trust management, the primary focus is on evaluating and managing the trustworthiness of individual vehicles within the network. This approach assesses each vehicle based on its behavior and interactions, ensuring reliable communication and cooperation across the vehicular network. Vehicle-centric trust approaches emphasize decentralized structures, where each vehicle independently assesses the trust levels of its neighboring vehicles without relying on a central authority.

Decentralized: Using a decentralized trust management system, trust assessments are distributed across the network's nodes. In order to determine the trust level of others, each vehicle or entity independently evaluates the behavior and interactions of others. In order to accomplish this, mechanisms such as reputation systems may be used, in which vehicles accumulate reputation scores based on the actions they have taken in the past.

In [188], the authors proposed a decentralized trust management scheme aimed at enhancing trust-sharing among vehicles within a clustered IoV environment. This vehicle-centric approach applied a Deep Echo State Q-Network model, enabling each vehicle to assess and predict the trustworthiness of neighboring vehicles. By integrating Echo State Networks, which simplify the training of recurrent neural

networks, with DQNs, the scheme facilitated efficient trust information sharing and predicted vehicle routes at intersections based on observed traffic patterns. The method's state space included current trust assessments and traffic conditions, which together enhanced safety by supporting informed decision-making in real-time traffic scenarios.

While the above approach addresses the challenge of trust information sharing across clusters in a decentralized IoV environment, it assumes that the trust evaluation mechanism and the threshold for distinguishing trustworthy from untrustworthy vehicles are predetermined. The problem of dynamically selecting a trust-value threshold that adapts to changing network conditions in vehicular edge computing environments remains unaddressed. This gap is addressed by the DSTS-DDPG approach presented in Chapter 4, which employs a DDPG algorithm to dynamically select the trust-value threshold based on VEC conditions.

Infrastructure-Centric Trust Management

In infrastructure-centric trust management, the focus is on ensuring the trustworthiness of network infrastructure components, such as RSUs, edge servers, and other supporting nodes within the vehicular network. This approach evaluates the trustworthiness of these infrastructure elements, which play a central role in maintaining communication, processing and sharing data, and managing network resources.

Centralized: In a centralized trust management system, a central authority or entity is responsible for maintaining and assessing the trust level of entities within the vehicular network. In order to determine trust levels, the central authority collects and analyzes data from various sources, such as vehicle behavior, communication patterns, and historical data. A trust score or recommendation is then provided to other members of the network. Trust management in VANETs is particularly challenging due to factors such as the wireless communication, high mobility, decentralized nature, and autonomous operations of network participants. Therefore, a centralized controller like SDN along with RSUs can be used to compute and manage trust for vehicles. A centralized SDN control strategy, however, may be challenged by the bottlenecks and single points of failure associated with centralized controllers.

In [189], the RL-based framework was developed to integrate trust quantification and trust-aware control algorithms into multi-agent transportation systems. This infrastructure-centric, centralized solution employed a trust authority unit to assess the trust levels of individual agents within the intelligent transportation infrastruc-

ture. Trust evaluations were based on observed behaviors and trust scores, which were securely stored on a centralized cloud or edge server. Agents that consistently followed protocols and provided accurate information received higher trust scores, while those with unreliable behavior were assigned lower scores. These trust scores directly influenced the control algorithm’s decision-making; for example, vehicles with high trust levels were granted smaller buffer zones when navigating intersections. To determine optimal buffer sizes, the study introduced a deep Q-learning method, allowing the system to adaptively manage intersection control based on the trustworthiness of participating agents.

Blockchain-based: Blockchain technology provides a distributed and immutable ledger that secures transactions and data [156]. Blockchain-based trust management systems for vehicular networks can record trust-related information. A smart contract can define rules for trust assessment and audit historical data.

In [190], the authors suggested leveraging blockchain technology to achieve synchronized messaging and enhanced security across numerous distributed Network Function Virtualization (NFV) management and orchestration systems. Through the utilization of MEC servers, the computational aspects of the blockchain framework were facilitated. This solution considered both the trust degrees and the computational capability of the MEC servers to enhance the performance of the blockchain-enabled distributed NFV-MANO framework. The approach involved formulating a comprehensive optimization strategy encompassing blockchain efficiency, processing latency, and associated costs. To solve the problem, they proposed the trust model and the resource allocation model as a MDP and utilized dueling deep Q-learning.

To address the vulnerabilities associated with single points of failure in SDN, [136] developed a new approach by integrating distributed SDN, blockchain technology, and DRL. This method facilitated the development of a dynamic service offloading and migration algorithm within VEC while ensuring trusted sharing of network topology information across domains. By leveraging a node trust evaluation model, the algorithm made optimal offloading decisions dynamically based on network conditions. An MDP was systematically formulated to address the inherent challenge of optimizing service offloading to minimize both delay and energy consumption. Hence, an A3C algorithm was applied to dynamically learn optimal offloading decisions. However, this centralized single-agent DRL architecture limits scalability in large-scale deployment environments.

Data-Centric Trust Management

Data-centric trust management emphasizes the assurance of the trustworthiness of data exchanged within vehicular networks. In this approach, trust evaluations are centered on verifying the accuracy and credibility of transmitted messages and shared data.

Decentralized: In [15], the authors introduced a data-centric reputation management scheme for VEC environments, specifically addressing the reliability of data trading transactions between vehicles and edge servers. Using a MAB model, the approach dynamically assigned and updated reputation scores based on each vehicle's historical trading behavior, promoting trustworthy data exchanges. By implementing this MAB-based framework at the edge server level, the system ensured that reputation updates and trust evaluations occurred locally within each trading session. This decentralized setup enabled edge servers to optimize data quality and trading reliability without requiring constant synchronization with a central server, effectively supporting autonomous reputation management across the VEC environment.

In order to overcome the lack of centralized infrastructure in VANETs, [191] proposed a novel vehicle-centric decentralized trust management scheme to evaluate the trustworthiness of individual vehicles. Fuzzy logic was used to evaluate direct trust of one-hop neighbors based on cooperativeness, honesty, and responsibility factors. Hence, it was possible to handle the uncertainty and complexity of the highly dynamic VANET topology and decentralized communications. A Q-learning method was proposed for indirectly estimating trust between non-neighbors by synthesizing reports from multiple nodes. By avoiding malicious nodes in routing decisions, the proposed trust management scheme improved network performance, increasing throughput and packet delivery ratio.

Trust management in vehicular communication is also challenged by feedback messages. In the context of vehicular networks, there have been recent research activities related to RL and trust management based on misbehavior detection systems (MDS). MDS is implemented in each vehicle and infrastructure node therefore each node can detect and prevent internal attacks. Collaboration between all detectors is required in order to reach a final decision regarding malicious activities. False feedback messages can occur in collaborative MDS. In this regard, [192] addressed a decentralized method based on collaborative ML and reputation-based MDS, which evaluated the trust level of feedback messages in VANETs by assigning reputation scores to each vehicle. To force vehicles to provide true feedback, a DRL

approach was applied to determine the optimal reputation updating policy. The Dempster-Shafer theory was used to combine the feedback from the vehicles and accurately predict the average number of true messages. Using the DQN, a local authority functioned as an agent and interacted with the vehicular environment to learn and broadcast the optimum reputation update policy accordingly. On the other hand, Guo *et al.* [193] presented a trust model to prevent bogus information from reaching decision-making entities. In this data-centric and decentralized model, a vehicle requested trust values from the trust evaluation model to assist with a driving decision-making event. An RL algorithm was used to determine the optimal strategy for trust schemes based on feedback from historical evaluation results.

Blockchain-based: In IoV environments, a consortium blockchain system was proposed in [194] to enable trust management during resource sharing. Using a novel Proof-of-Reputation consensus protocol, interactions between task owner vehicles and resource providers were recorded in the blockchain. Vehicles with higher reputations were given priority in transactions, encouraging good behavior. An RSU acted as a blockchain node, generating and validating blocks. Dynamic pricing and task assignment were implemented by a smart contract based on DRL.

Link-Centric Trust Management

This approach assesses the trustworthiness of connections such as V2V or V2I links to ensure that data is transmitted securely across trusted paths. Link-centric trust management is essential in dynamic vehicular networks where maintaining trusted communication links supports seamless and safe information flow across nodes.

Centralized: Zhang *et al.* [195] deployed a software-defined DRL algorithm to find an optimal routing path. The focus of this approach was on establishing secure routing paths within VANET. SDN acted as an intelligent agent, analyzing routing paths to identify the one with the highest trust value and assessing the behavior of neighboring nodes for secure packet forwarding. By interacting with the VANET environment, SDN determined a reliable and secure route from the source vehicle to the destination vehicle. To summarize, the agent selected the next hop vehicle node and received the path trust value as a reward. The SDN controller centrally collected trust information, applied a deep Q-learning algorithm, and then determined the optimal routing paths based on accumulated trust values. Thus, trust management was coordinated in a centralized manner through the SDN controller.

Similarly, [196] focused on optimizing the trustworthiness of routing paths between vehicles in a VANET environment. Like the previous study, it deployed a centralized SDN controller to collect trust information and make routing decisions. However, this work advanced traditional DRL by applying a dueling network architecture, known as dueling DRL, which significantly improved trust evaluation and routing efficiency by separating value and advantage functions within the model. This approach enhanced the reliability of V2V communication links by selecting trusted neighbors for data forwarding. Meanwhile, the SDN controller was introduced in [25] as a learning agent that used a deep Q-learning algorithm to obtain the optimal communication link quality policy.

Decentralized: In [197], a cluster-based routing protocol called CR-DRL was proposed for vehicular opportunistic networks, utilizing an Actor-Critic. Each node locally maintains its own encounter history, residual energy, and buffer size, and the AC algorithm runs based on these local observations. The approach demonstrated improvements in delivery ratio, end-to-end delay, throughput, and energy efficiency. However, trust evaluation of vehicles and the presence of malicious nodes were not considered. However, another approach presented trusted routing [198] in vehicular networks while avoiding malicious vehicles. The approach evaluated inter-vehicular trust based on direct interactions and indirect recommendations from neighboring vehicles, and integrated this trust value into a road segment scoring function alongside connectivity degree, traffic density distribution, and shortest distance to the destination. UAVs were employed to assist in the detection of malicious nodes through a clustering-based assessment gathering phase, where cluster heads reported suspected malicious vehicles to RSUs, which in turn forwarded the information to a Trusted Authority. In addition, [199] developed a decentralized secure routing approach to defend against packet drop attacks in VANETs. This study introduced a novel Grid-based extended Joint Action Learning approach (Grid-eJAL) that employed online and adaptive MARL to implement route mutation for attack mitigation in VANETs. Grid-eJAL divided the area of interest into grids, and during packet transmission, the optimal next hop was determined by the learning policy's minimum angle of mobility.

Although the above approaches address various aspects of trust-aware routing in vehicular networks, several limitations remain. The centralized approaches [25], [195], [196] rely on SDN controllers as single points of trust coordination, introducing scalability bottlenecks and single points of failure. Among the decentralized approaches, [197] achieves efficient cluster-based routing but does not incorporate

trust evaluation. In [198], trust is evaluated and integrated into a road segment scoring function; however, the approach depends on UAV-assisted infrastructure for trust aggregation and malicious node detection. In [199], route mutation is employed to proactively avoid compromised nodes; however, the threat model assumes that only RSUs, not vehicles, can be compromised, and no trust evaluation of individual vehicles is performed. None of these approaches address vehicle-level link trust evaluation and adaptive trust-aware route selection that can detect and mitigate black hole and gray hole attacks in fully decentralized manner and without infrastructure dependency. This gap motivates the QL-TRT approach presented in Chapter 4, which integrates Q-learning-based link trust computation directly into the routing process, enabling each vehicle to independently learn and select the most trustworthy routes without reliance on centralized controllers or external infrastructure.

Table 3.2: Summary of Literature on RL-enabled Trust Management in Vehicular Networks

Ref.	Scope	Method	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[195]	Link-Centric	Centralized	SDN	Find optimal routing path	Deep Q-learning				✓		✓
[196]	Link-Centric	Centralized	SDN	Find optimal routing path	Dueling deep RL	✓	✓	✓	✓		✓
[191]	Data-Centric	Decentralized	VANET	Propose a decentralized trust management scheme for vehicular networks	Fuzzy logic, Q-learning	✓			✓		
[194]	Data-Centric	Blockchain-based	IoV	Blockchain system is proposed to enable trust management during resource sharing	DDPG	✓	✓	✓	✓		
[25]	Link-Centric	Centralized	SDN	Obtain the optimal communication link quality policy	Deep Q-learning	✓		✓	✓		✓
[193]	Data-Centric	Decentralized	VANETs	Ensure bogus information will not influence the decision process	Q-learning				✓		

Continued on next page

Table 3.2: Summary of Literature on RL-enabled Trust Management in Vehicular Networks (*continued*)

Ref.	Scope	Method	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[190]	Infra-Centric	Blockchain-based	MEC	Autonomous management and orchestration of virtualized resources are facilitated, while blockchain ensures the synchronization of trust messages	Dueling deep Q-learning, Blockchain	✓	✓	✓		✓	
[189]	Infra-Centric	Centralized	ITS	Propose a framework for using the quantified trust level of agents	Deep Q-learning		✓	✓	✓	✓	✓
[192]	Data-Centric	Decentralized	VEC	Evaluate the trust level of both messages and nodes	Deep Q-learning	✓	✓	✓	✓		
[136]	Infra-Centric	Blockchain-based	MEC, SDN	Trusted communication and optimally offloading services to minimize delay and energy consumption	A3C		✓	✓	✓	✓	
[188]	Vehicle-Centric	Decentralized	IoV	Distributed trust sharing mechanism	DQN			✓	✓		

Continued on next page

Table 3.2: Summary of Literature on RL-enabled Trust Management in Vehicular Networks (*continued*)

Ref.	Scope	Method	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[199]	Link-Centric	Decentralized	VANET	Trust-based routing while avoiding packet drop attacks	MARL, Q-learning				✓	✓	
[15]	Data-Centric	Decentralized	VEC	Reputation management to select the vehicle for data trading	MAB				✓		

3.1.3 RL-based Privacy Protection Approaches

Privacy protection in various vehicular networks refers to the measures taken to protect the privacy of data, vehicles and passengers when they communicate within the network. IoV is characterized by vehicles equipped with wireless communication devices, allowing them to share information not only with each other but also with RSUs or other entities like traffic lights and road signs. While this communication is essential for enhancing road safety, traffic efficiency, and various other applications, it also introduces vulnerabilities that could compromise the privacy of individuals and vehicles. This subsection details privacy preservation approaches that utilize RL within vehicular networks. Table 3.3 summarizes these approaches, classified by their specific privacy focus.

User-Centric Privacy Protection

User-centric privacy protection focuses on giving individuals control over their personal information, particularly their location and identity. To address these privacy challenges, researchers have proposed various methods based on RL techniques.

Location Privacy Protection This aspect of user-centric privacy involves protecting individuals' location data from unauthorized access and tracking. Continuous monitoring of vehicle locations could lead to unauthorized surveillance, potentially exposing users' patterns and activities. As a result, a number of works have applied RL methodologies to safeguard vehicle geographical privacy [79] and semantic location privacy [80] as defined in the Chapter 2.

- ***Semantic Location Privacy Protection:*** In [200], [201], the released vehicle locations were randomized based on semantic trajectory. By employing RL in the privacy budget selection, algorithms optimized the obfuscation policy while lowering QoS loss and increasing privacy. While [200] used a Q-learning algorithm to solve the problem, which was only applicable to discrete perturbation action spaces, [201] utilized an actor-critic algorithm to explore the optimal policy in a continuous perturbation action space.
- ***Geographical Location Privacy Protection:*** The combination of the blockchain and RL has been explored to make the most of both technologies while addressing privacy concerns. For instance, [202] integrated DRL and a permissioned blockchain approach into vehicular networks. The aim was to suggest a smart and secure strategy for caching content. Vehicles had a

dual role: they acted as caching requesters and providers, making the caching process smoother and performing caching via V2V communication. Meanwhile, BSs maintained a blockchain ledger of all content transactions. To address the challenges of dynamic vehicular mobility and time-varying channels, a multi-agent actor-critic DRL algorithm was used to learn optimal caching policies that maximized system utility and found an optimal dynamic content caching policy between vehicles based on the changing topology and channel conditions of the network. A permissioned blockchain facilitated secure content transactions among untrusted entities. In such cases, FL emerged as a privacy-preserving paradigm for distributed vehicular networks [203]. For instance, [204] worked on an asynchronous FL scheme by adopting DRL for node selection to improve efficiency and for training local models while sharing parameters to improve privacy. A DDPG-based RL algorithm was proposed for optimized node selection to improve learning efficiency.

- ***Geographical and Semantic Location Privacy Protection:*** [205] proposed an on-policy RL model that addressed not only geographical location security but also semantic location security on vehicle trajectories. According to the proposed method, the privacy budget was dynamically allocated for each location on the vehicle trajectory to improve the balance between unclear geographical and semantic location protection. To guarantee the efficiency of the model, the scheme utilized two metrics to specify the reward function between different states.

Preserving the location privacy of vehicles during navigation was another challenge where Zhu *et al.* in [206] proposed a distributed, multi-agent navigation system using DQN combined with differential privacy. The agents were vehicles that learned to optimize routes by sharing perturbed traffic information under real-time traffic conditions with each other, balancing efficient navigation with privacy preservation.

In other works such as [207], [208], a method was outlined for jointly optimizing the throughput from RSUs to vehicles while maintaining vehicle privacy by providing obfuscated location information to the server. The authors employed the approach of providing the server with obscured location information. To tackle this challenge, they harnessed the power of Q-learning, training a state-action matrix. This matrix enabled the decision-making process for devising location obfuscation strategies. These strategies were contingent on the cur-

rent and previous reported locations of the vehicles. [207] considered a highway scenario and assigned a single privacy weight to all vehicles, whereas [208] investigated a more general scenario, including both ordinary and emergency vehicles with varying privacy importance.

Wei *et al.* in [209] proposed a privacy-aware MADRL approach for task offloading over VANET. They identified a vulnerability in the MADRL policy learning process, where an offloading preference inference attack could mislead vehicles into offloading tasks to malicious remote RSUs. To address this issue, the study formulated a joint optimization strategy for offloading decisions and transmission power, aiming to minimize system costs while safeguarding privacy preferences during the offloading process.

A related work in Edge Content Cache (ECC) [210] proposed a joint optimization problem involving real-time cache replacement strategies and cache prefetching strategies, aimed at improving the performance of collaborative ECC. It is important to note that this approach utilized an on-policy PPO methodology. In fact, this was a distributed, multi-agent RL approach in which each RSU ran an independent PPO agent that learned policies to minimize total request costs by making cache replacement or prefetch decisions on a request-by-request basis.

Other studies utilized FRL to allocate resources in vehicular networks while maintaining privacy. For instance, [211] developed a privacy-preserving, distributed MAFRL approach for cooperative content caching in vehicular edge networks. The DQN-based agents (RSUs) learned to make caching decisions that improved cache efficiency and met content delay requirements. However, [212] introduced a MAFRL framework for joint edge association and power allocation in vehicular networks, which achieved a good connectivity-cost trade-off while preserving privacy. Using this approach, the agents were trained centrally at a macro base station by sharing encrypted Q-values, but then executed policies in a distributed manner based on local observations. Consequently, it was a centralized training, decentralized execution approach.

A similar problem was addressed in [137], where an A3C-based algorithm was proposed to jointly optimize device selection, UAV placement, and resource management, aiming to improve the performance of federated convergence. Similarly, [213] Developed actor-critic RL algorithms integrated with FL to optimize trajectories and allocate resources in networks utilizing multiple UAVs

as aerial base stations. Specifically, the authors developed FL actor-critic and distillation FL actor-critic methods where each UAV agent learned decentralized policies but coordinated with other UAVs periodically to reach an overall optimal solution. The key innovation was to use FL for privacy-preserving collaboration between UAVs, while actor-critic methods handled the continuous action spaces.

Identity Privacy Protection User identity privacy pertains to the protection of the identities of drivers and passengers on the network. [16] proposed an algorithm for computation offloading in vehicular networks. It was based on a MARL approach with multiple decentralized agents. Each vehicle acted as an agent, making offloading decisions independently while ensuring the privacy of offloaded data, such as identity authentication and balancing load on the edge.

In [214], a cross-metaverse empowered dual pseudonym management framework was proposed for vehicular edge metaverses. The approach utilized cross-chain technology combining a main chain, relay chain, and multiple subchains to facilitate decentralized and secure pseudonym distribution and revocation during vehicle twin migrations. A privacy metric called Degree of Privacy Entropy (DoPE) was introduced to quantify the level of location privacy protection after pseudonym changes. The pseudonym generation problem was formulated using inventory theory, and a MADRL approach based on MAPPO was employed to derive the optimal pseudonym generating strategy across multiple local metaverses. In this user-centric approach, pseudonyms are used to hide the true identities of vehicles, while the DoPE metric quantifies the probability that attackers can successfully track a vehicle after a pseudonym change, addressing a location privacy concern. However, while this approach optimizes the number of pseudonyms generated by each local metaverse manager, the pseudonym change decision for vehicles is not adaptively controlled through RL. Furthermore, the pseudonym management operates independently of vehicle reputation, and no trust-based mechanism is integrated to link pseudonym lifecycle decisions with vehicle trustworthiness. These limitations motivate the novel approaches proposed and presented in this thesis in Chapter 5 and 6, respectively.

Data-Centric Privacy Protection

One of the primary objectives of privacy protection in vehicular networks was to prevent unauthorized parties from gaining access to sensitive data, which could include real-time information about a vehicle's location, speed, and travel direction. The potential consequences of privacy breaches could be severe, ranging from tracking the movements of specific vehicles to uncovering behavioral patterns of drivers and passengers [77]. In this regard, the authors of [215] presented a DRL algorithm to hide sensitive information while keeping the utility of a given vehicle connected over VANET. The algorithm used an MDP to sanitize private information. The methodology was able to adapt and receive feedback to correct wrong decisions. A sanitation process functioned adequately to hide and remove private information from OBU sensors. It was also important to consider offloading tasks to RSUs to maintain privacy. In addition, in [216], an incentive mechanism for privacy-preserving crowdsensing was proposed, formulating the interactions between platforms and participants as a multi-leader multi-follower Stackelberg game. Participants determined their sensing contributions based on sensing effort and differential privacy protection levels, while platforms optimized their payment strategies. Q-learning was employed by participants to learn optimal sensing contributions, and a Double Deep Q-Network with dueling architecture (DDQN) was utilized by platforms to accelerate the acquisition of optimal payment strategies.

In [17], a data-centric privacy-preserving task offloading framework is proposed for vehicular edge computing, aimed at protecting sensing data generated by vehicles and transmitted to RSUs. Federated learning, differential privacy, and TD3-based DRL are integrated across a three-tier vehicle-RSU-cloud architecture. An information-entropy-based privacy leakage metric is introduced, defined as the mutual information between a privacy source at the vehicle and a privacy sink at the RSU, and an entropy-based weighting method is employed to dynamically balance latency, energy, and privacy within the TD3 reward function.

RL and blockchain were combined to address privacy concerns in vehicular networks. In [217], an algorithm based on a DRL-enabled approach was proposed to optimize the blockchain by selecting suitable miners and transactions to maximize security and privacy while minimizing latency. Furthermore,

[218] proposed a computation offloading strategy for IoVs using DRL and blockchain. The system consisted of RSUs equipped with mobile edge servers, vehicles with computation tasks, and a high-altitude platform system (HAPS) as the central controller. HAPS determined optimal offloading decisions to minimize the total processing time of vehicles' tasks. It used a soft actor-critic RL algorithm to train a policy for assigning tasks and allocating transmission power. Blockchain technology was incorporated to coordinate task offloading between RSUs and provide security. The HAPS acted as a single agent with global knowledge to make centralized offloading decisions.

Li *et al.* suggested a new verification approach in [219] for verifying vehicles and infrastructures in the IoV, called Crowd-Learning. To prepare for the verification of an incoming vehicle in advance, an SDN controller notified some RSUs that the vehicle might arrive. The verification latency was effectively reduced in this manner. Through RL, each RSU identified the authenticity of passing vehicles and related participating RSUs in one incentive crowd-learning process based on Q-learning.

Similarly, [220] tackled data privacy and security challenges in SDN integrated with IoV by implementing an enhanced multi-blockchain framework and hierarchical task management system. This approach categorized and distributed spatial tasks based on privacy needs and geographic regions, effectively safeguarding data privacy. As a result, malicious workers could not obtain the complete privacy information of the task by launching Sybil attacks and Collusion attacks. Moreover, it employed a DRL agent trained with DQN to select reliable workers or task receivers for crowdsourcing tasks and dynamically configure blockchain parameters to optimize worker-task assignments, minimize worker-task distance, enhance worker credit ratings, and improve blockchain throughput. [221] also proposed a blockchain-empowered FL framework for secure data sharing in IoV.

In another work, [27] used MAFRL for cooperative content caching in vehicular edge networks. However, this work proposed an asynchronous FL approach, where the RSU aggregated local models as soon as they were received without waiting for all vehicles. This allowed more local model updates to be incorporated even if vehicles left the coverage area, improving global model accuracy.

Table 3.3: Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks

Ref.	Scope	Type of Privacy	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[200]	User-Centric	Semantic location privacy	VANETs	Randomize vehicle locations based on semantic trajectory	Q-learning			✓	✓		
[207]	User-Centric	Geographical location privacy	Vehicular networks	Jointly optimizing throughput and safeguarding vehicle privacy	Q-learning				✓		
[202]	User-Centric	Geographical location privacy	VEC	Intelligent content caching for vehicles while achieving security and privacy protection	Blockchain and DRL	✓	✓	✓	✓	✓	✓
[204]	User-Centric	Geographical location privacy	IoV	Guarantee the reliability of shared data, reduce the delay, and improve the efficiency	FL, DRL, Blockchain	✓	✓	✓			✓
[201]	User-Centric	Semantic location privacy	VANETs	Randomize the disclosed vehicle locations and dynamically adjust the perturbation strategy	Actor-Critic	✓	✓	✓	✓		

Continued on next page

Table 3.3: Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks (*continued*)

Ref.	Scope	Type of Privacy	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[217]	Data-Centric	Data privacy	IoV	Jointly maximize blockchain safety and minimize latency in crowd-sensing	Blockchain, DRL	✓	✓	✓	✓	✓	✓
[215]	Data-Centric	Data privacy	VANETs	Hide and remove private information from OBU sensors	DRL		✓	✓	✓		
[205]	User-Centric	Geographical and semantic location privacy	VANETs	Protect geographical and semantic locations	Monte Carlo RL		✓		✓		
[220]	Data-Centric	Data privacy	SDN-IoV	Protect the privacy of tasks during the task allocation process	DRL, Blockchain	✓	✓	✓	✓	✓	
[137]	User-Centric	Geographical and semantic location privacy	UAV	Joint device selection, UAV placement, and resource allocation while maintaining privacy	FRL, A3C		✓	✓	✓	✓	

Continued on next page

Table 3.3: Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks (*continued*)

Ref.	Scope	Type of Privacy	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[208]	User-Centric	Geographical and semantic location privacy	Vehicular networks	Jointly optimize the throughput and the privacy of vehicles' locations	Q-learning	✓					
[219]	Data-Centric	Data privacy	SDN, MEC	Behavior-based verification method that verifies vehicles and reduces verification latency by estimating vehicle behavior in advance	Q-learning	✓					
[209]	User-Centric	Geographical and semantic location privacy	VEC	Jointly optimize offloading decisions and transmission power to minimize system costs, while ensuring privacy	MADRL, DDPG	✓	✓	✓	✓	✓	✓
[211]	User-Centric	Geographical and semantic location privacy	IoV	Privacy-preserving and accurate caching decision	FRL, MAFRL, DQN	✓	✓	✓	✓		✓

Continued on next page

Table 3.3: Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks (*continued*)

Ref.	Scope	Type of Privacy	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[27]	Data-Centric	Data privacy	VEC	Vehicle privacy and cooperative caching scheme in VEC	FRL, DRL	✓	✓	✓	✓		
[212]	User-Centric	Geographical and semantic location privacy	IoV	Privacy-preserving joint edge association and power allocation	MAFRL, DQN	✓	✓	✓			✓
[206]	User-Centric	Geographical and semantic location privacy	IoV, V2V	Vehicles' privacy while enabling distributed navigation under real-time traffic conditions	DQN	✓	✓	✓			
[210]	User-Centric	Geographical and semantic location privacy	VEC	Train cache servers to perform cache prefetch in real time on each request while protecting user privacy	PPO		✓	✓	✓		
[16]	User-Centric	Identity privacy	VEC	Minimize delay and energy consumption in computation offloading while ensuring privacy	DDQN	✓	✓	✓	✓		✓

Continued on next page

Table 3.3: Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks (*continued*)

Ref.	Scope	Type of Privacy	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[221]	Data-Centric	Data privacy	Vehicular networks	Adaptive blockchain-enabled FL framework based on blockchain sharing and DRL to obtain vehicular data flow between a blockchain and multiple shards	FL and DRL, Blockchain	✓	✓	✓	✓		✓
[213]	User-Centric	Geographical and semantic location privacy	UAVs	Maximizing QoS and ensuring user information privacy	FL Actor-Critic and DFRL		✓	✓	✓		✓
[214]	User-Centric	Location and Identity privacy	VEC	Optimize pseudonym generation across local meta-verses during vehicle twin migrations	MADRL, MAPPO, Blockchain	✓	✓	✓	✓		✓
[216]	Data-Centric	Data privacy	IoV	Maximize utilities of platforms and participants while preserving sensing data privacy	Q-learning, DDDQN	✓	✓	✓	✓		

Continued on next page

Table 3.3: Summary of Literature on RL-enabled Privacy-Preserving in Vehicular Networks (*continued*)

Ref.	Scope	Type of Privacy	Network	Final Goal	Algorithm	Limitations & Challenges					
						M	C	T	S	L	G
[17]	Data-Centric	Data privacy	VEC	Entropy-based privacy leakage metric; protects sensing data via FL with Differential privacy noise on model updates; entropy-weighted reward balances latency, energy, and privacy	FL + DRL	✓	✓	✓	✓		✓
[218]	Data-Centric	Data privacy	IoV, VEC	Computation controlling and improving user privacy	Soft Actor-Critic, Blockchain	✓	✓	✓	✓		✓

3.2 RL-based Approaches for Enhancing SPT: a Taxonomy Based on RL Methods and SPT Protection

Section 3.1 classified the RL-based approaches based on their focus-centric on addressing SPT issues in vehicular networks. However, it is also essential to examine the distinctions between the various RL approaches outlined in the literature and compare them according to the methods used. Therefore, this section presents a complementary taxonomy that classifies these approaches along two different dimensions: the type of SPT protection they provide and the type of RL methodology they employ. It is important to note that the same body of literature reviewed in Section 3.1 is re-examined here from a different perspective, without introducing new references. Figure 3.2 illustrates various SPT protection types and various RL methods that have been employed across various approaches to address SPT provisioning problems.

The first classification examines RL-based approaches within three primary domains: security-enhancing RL, privacy-preserving RL, and trust-enhancing RL methods. Each subcategory is organized according to the specific characteristics of the approaches proposed in the literature. In the security-enhancing subcategory, various types of attacks commonly encountered in vehicular networks are categorized, such as Sybil attacks, eavesdropping, spoofing, jamming, and DoS/DDoS attacks, and the RL-based approaches developed to detect or mitigate them are investigated. The second subcategory focuses on privacy-preserving mechanisms. The RL-based approaches in this domain are classified into two subcategories: vehicle location privacy protection and vehicle identity protection. These approaches aim to safeguard sensitive vehicular information while maintaining system functionality. In the trust-enhancing subcategory, the existing literature on RL-based trust-enhancing mechanisms is classified into centralized and decentralized approaches. The decentralized category is further divided into blockchain-based and non-blockchain-based approaches.

The second classification systematically categorizes the available research papers in this field based on the specific RL methods they employ. This categorization provides a more comprehensive classification of the available approaches that address SPT provisioning problems using RL methods explained in Chapter 2, which are classified based on MAB/MDP. The MDP category is further classified according to

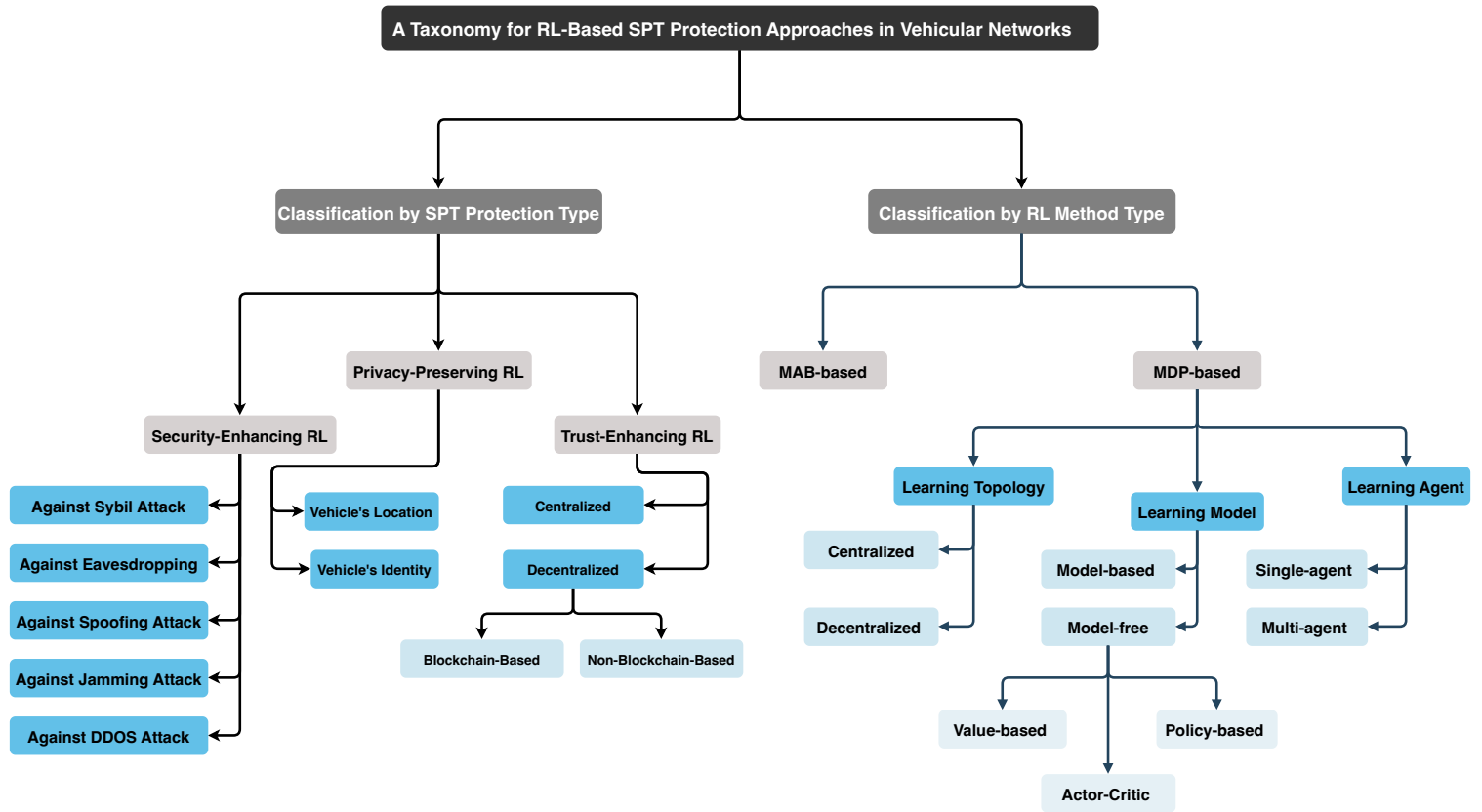


Figure 3.2: A taxonomy of RL-based SPT protection in vehicular networks, classified by SPT protection type and RL method type

learning agent, learning model, and learning topology. The details of each category were also explained in Chapter 2.

Table 3.4 presents a summary of the classification scheme and provides an overview of each approach. Since the same literature as in Section 3.1 is discussed here from a methodological perspective, details including the specific techniques employed, along with the challenges and limitations associated with each method, can be revisited in the aforementioned section - they are not repeated here.

Table 3.4: Summary of RL Approaches for Enhancing SPT Based on RL Methods

RL Type	Learning Model	Category	Algorithm	Agent	Topology	Citation	Sec.	Priv.	Trust
MDP	Model-Based	–	Dyna-Q	Multi-agent	Decentralized	[173]	✓(Jamming)		
MDP	Model-Based	–	Dyna-Q	Single-agent	Centralized	[163], [185]	✓(Spoofing)		
MDP	Model-Free	Value-Based	Q-learning	Single-agent	centralized	[168]	✓(DOS)		
MDP	Model-Free	Value-Based	Q-learning	Single-agent	Centralized	[170], [184]	✓(Eavesdropping)		
MDP	Model-Free	Value-Based	Q-learning	Single-agent	Centralized	[200]		✓	
MDP	Model-Free	Value-Based	Q-learning	Multi-agent	Decentralized	[170]	✓(Eavesdropping)		
MDP	Model-Free	Value-Based	Q-learning	Multi-agent	Decentralized	[174], [179]	✓(Jamming)		
MDP	Model-Free	Value-Based	Q-learning	Multi-agent	Decentralized	[191], [193], [199], [222]			✓
MDP	Model-Free	Value-Based	Q-learning	Multi-agent	Decentralized	[207], [208], [219]		✓	

Continued on next page

Table 3.4: Summary of RL Approaches for Enhancing SPT Based on RL Methods (*continued*)

RL Type	Learning Model	Category	Algorithm	Agent	Topology	Citation	Sec.	Priv.	Trust
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Single-agent	Centralized	[24]	✓(Jamming)		
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Single-agent	Centralized	[183]	✓(Eavesdropping)		
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Single-agent	Centralized	[25], [189], [192], [195]			✓
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Single-agent	Centralized	[215], [217]		✓	
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Multi-agent	Decentralized	[165], [172]	✓(DDOS)		
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Multi-agent	Decentralized	[14], [180]	✓(Eavesdropping)		

Continued on next page

Table 3.4: Summary of RL Approaches for Enhancing SPT Based on RL Methods (*continued*)

RL Type	Learning Model	Category	Algorithm	Agent	Topology	Citation	Sec.	Priv.	Trust
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Multi-agent	Decentralized	[188]			✓
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Multi-agent	Decentralized	[206], [211], [221]		✓	
MDP	Model-Free	Value-Based	DRL (DQN, DDQN)	Multi-agent	Centralized	[212], [220]		✓	
MDP	Model-Free	Value-Based	Dueling DRL	Single-agent	Centralized	[190], [196]			✓
MDP	Model-Free	Value-Based	Dueling DRL	Multi-agent	Decentralized	[16], [27]		✓	
MDP	Model-Free	Policy-Based	PPO	Single-agent	Centralized	[167]	✓(DDOS)		
MDP	Model-Free	Policy-Based	PPO	Multi-agent	Decentralized	[210]		✓	
MDP	Model-Free	Policy-Based	PHC	Single-agent	Centralized	[176]	✓(Jamming)		

Continued on next page

Table 3.4: Summary of RL Approaches for Enhancing SPT Based on RL Methods (*continued*)

RL Type	Learning Model	Category	Algorithm	Agent	Topology	Citation	Sec.	Priv.	Trust
MDP	Model-Free	Policy-Based	Monte Carlo	Multi-agent	Decentralized	[205]		✓	
MDP	Model-Free	Policy-Based	–	Single-agent	Centralized	[171]	✓(Eavesdropping)		
MDP	Model-Free	Policy-Based	–	Multi-agent	Decentralized	[177]	✓(Jamming)		
MDP	Model-Free	Actor-Critic	A3C	Single-agent	Centralized	[136]			✓
MDP	Model-Free	Actor-Critic	A3C	Multi-agent	Decentralized	[137]		✓	
MDP	Model-Free	Actor-Critic	FL-AC	Multi-agent	Decentralized	[213]		✓	
MDP	Model-Free	Actor-Critic	SAC	Single-agent	Centralized	[218]		✓	
MDP	Model-Free	Actor-Critic	A2C	Multi-agent	Decentralized	[178]	✓(Jamming)		
MDP	Model-Free	Actor-Critic	DDPG	Multi-agent	Centralized	[166]	✓(DDOS)		
MDP	Model-Free	Actor-Critic	DDPG	Multi-agent	Centralized	[194]			✓

Continued on next page

Table 3.4: Summary of RL Approaches for Enhancing SPT Based on RL Methods (*continued*)

RL Type	Learning Model	Category	Algorithm	Agent	Topology	Citation	Sec.	Priv.	Trust
MDP	Model-Free	Actor-Critic	DDPG	Multi-agent	Centralized	[201], [209]		✓	
MDP	Model-Free	Actor-Critic	DDPG	Multi-agent	Decentralized	[161]	✓(Jamming)		
MDP	Model-Free	Actor-Critic	DDPG	Multi-agent	Decentralized	[202], [204]		✓	
MAB	–	–	–	–	–	[175]	✓(Jamming)		
MAB	–	–	–	–	–	[181], [182]	✓(Eavesdropping)		
MAB	–	–	–	–	–	[15]			✓

3.3 Non-RL Approaches for SPT Provisioning in Vehicular Networks

The previous sections reviewed RL-based approaches for enhancing security, privacy, and trust in vehicular networks through two proposed complementary taxonomies. However, a substantial body of related work addresses identity privacy preservation through pseudonym management and enhancing trust through reputation management using non-RL techniques. To cover these aspects, three areas that are directly aligned with the contributions of this thesis are examined in this section: non-RL pseudonym management approaches, non-RL reputation management approaches, and joint non-RL privacy-preserving and trust management approaches.

3.3.1 Non-RL Pseudonym Management Approaches

This subsection discusses non-RL pseudonym management approaches classified into two main categories. The first category covers traditional pseudonym management, which includes mix-zone methods and context-aware strategies to enhance privacy in vehicular networks. The second category focuses on blockchain-based pseudonym management and authentication approaches.

Traditional Pseudonym Management

Recent advancements in vehicular privacy have led to two primary methodologies for pseudonym management: mix-zone-based approaches and mix-context-based approaches. Mix-zone-based approaches use high-traffic locations as predefined zones where vehicles collectively update their pseudonyms [223]. In [224], a pseudonym change scheme designated areas before red lights as silent mix zones, with zone lengths dynamically adjusted based on real-time traffic predictions. Similarly, in [225], mix-zone strategies were enhanced by combining them with broadcast and intermittent silence phases, and clustering techniques were used to adaptively select zones, reducing semantic and syntactic linking attacks. In [226], a scheme was proposed where each vehicle generates multiple pseudonyms valid for up to 10 days without relying on RSUs; however, this approach lacks adaptability to dynamic traffic conditions. Mix-context-based strategies coordinate pseudonym updates based on real-time factors such as traffic density, enabling adaptive, collaborative changes without relying on fixed infrastructure. In this regard, in [227], the pseudonym swap area was determined dynamically based on the communication time and vehicle density, ensuring a secure and efficient swapping process. Vehicles in proxim-

ity with sufficient communication time establish a pseudonym swap zone. In [228], mix-context zones were proposed for use in dense areas alongside differential privacy-based pseudonym perturbation in sparse areas. Vehicles in sparse regions transmit multiple perturbed messages, making it difficult for attackers to track real vehicle identities, effectively countering colluding attacks. In addition, in [229], a dynamic and context-aware pseudonym change mechanism was introduced. This approach enables vehicles to alter their pseudonyms based on real-time context information, such as the number of nearby vehicles that participate simultaneously in pseudonym changes. In [230], a hybrid model was introduced where vehicles swap pseudonyms cooperatively without RSU dependence within mix-contexts. The scheme incorporates two types of pseudonyms: swappable and non-swappable.

Blockchain-based Pseudonym Management and Authentication

Existing blockchain-based approaches offer tamper-proof mechanisms for decentralized pseudonym authentication, shuffling, and management to provide anonymity [231]. In [232], [233], a pseudonym authentication mechanism was introduced, designed to safeguard vehicle identities while ensuring efficient authentication in high-density IoV environments. In [234], an architecture was introduced that relies on RSUs to handle pseudonym issuance and authentication, reducing the burden on central authorities. In comparison, in [235], an approach was presented that, instead of relying on RSUs or a central TA, uses bilinear pairing on elliptic curves and blockchain technology to verify vehicle pseudonyms, while ensuring secure communication and supporting pseudonym revocation. In [236], pseudonym authentication was expanded by incorporating zero-knowledge proofs to enable privacy-preserving data verification. The decentralized nature of blockchain, combined with the computational efficiency of Zero-Knowledge Succinct Non-Interactive Arguments of Knowledge (zk-SNARK), illustrates how pseudonym management can be enhanced without compromising performance. In [20], a pseudonym distribution and shuffling mechanism was proposed, where the pseudonyms used are securely reallocated among vehicles instead of being discarded, significantly reducing the number of newly generated pseudonyms. In addition, in [237], blockchain technology was leveraged to enable secure multi-domain authentication while ensuring conditional privacy across domains. However, these approaches faced significant challenges, including high computational costs, increased latency, and potential bottlenecks during periods of high demand for pseudonym updates. Similarly, in [238], a blockchain-based data sharing scheme that uses dynamic pseudonyms was proposed. By leveraging

blockchain immutability and smart contracts, the scheme ensures secure data aggregation and pseudonym updates. However, it incurs communication overhead and increased response times in high-demand scenarios.

3.3.2 Non-RL Reputation Management Approaches

This subsection reviews non-RL reputation management approaches in IoV, including both traditional and blockchain-based approaches.

In [239], a multidimensional trust model based on the Dirichlet distribution was introduced to evaluate node reputation, which is then secured on a decentralized ledger using a Delegated Proof of Stake consensus mechanism. To ensure system integrity, the framework incorporates an incentive mechanism that rewards honest behavior and punishes malicious actors, dynamically adjusting reputation scores to filter out untrustworthy data sources. In [240], a scheme was proposed that utilizes edge computing to optimize trust management in VANETs. The model aggregates direct interactions, recommendations, and a group reputation factor, recognizing that vehicles in platoons often share trust characteristics. This hierarchical approach delegates assessment tasks to edge nodes. In [241], a blockchain-based and decentralized reputation management model for IoV networks was proposed, aiming to enhance data trustworthiness and malicious node detection. A reputation evaluation method was developed that assesses node reliability based on spatiotemporal feature perception fusion, considering both time continuity and spatial transitivity to accurately identify malicious actors. To facilitate efficient reputation data dissemination, a blockchain-based storage system utilizing a PBFT consensus scheme integrated with Bayesian inference was constructed. In another blockchain-based approach, in [242], a vehicle-centric reputation model was proposed where trust is intrinsically tied to a vehicle's historical behavior in data trading. To capture the realistic dynamics of trust, the Gompertz function combined with an ageing mechanism was employed; this ensures that a vehicle's reputation grows gradually through consistent honest behavior but declines rapidly upon the detection of malicious activity. Furthermore, the scheme introduces a Proof-of-Reputation consensus mechanism, supported by a parallel-chain structure, to dynamically select the RSU committee and optimize consensus efficiency. Similarly, in [243], a semi-distributed trust framework for Cyber-Physical Systems was proposed. It combines Bayesian inference for message verification with a slow reward, fast punishment strategy for vehicle reputation. The scheme also introduces a consensus algorithm that prioritizes high-trust nodes to significantly reduce consensus latency. In [244], a central-

ized dual reputation management scheme for vehicle platoons was proposed that separates each vehicle's reputation into leading-vehicle and following-vehicle values to counter the single reputation attack.

The approaches reviewed in this subsection contribute to reputation management in IoV through diverse mechanisms. However, several limitations persist across this body of work. Most notably, the majority of these schemes rely on static or rule-based reputation update mechanisms that lack adaptability to evolving network conditions and adversarial behaviors.

3.3.3 Joint Privacy-Preserving and Trust Management

Privacy-preserving and reputation management at the same time is critical in IoV, particularly during sensing-vehicle evaluation. Leakage of reputation values enables adversaries to analyze vehicle behavior, target high-reputation vehicles with attacks, or disrupt the sensing process by manipulating low-reputation ones [245]. In [246], reputation was extended to multi-dimensional vectors with dynamically updated threshold vectors for emergency message evaluation in VANETs. The scheme employed Paillier homomorphic encryption and ECC-based pseudonyms to preserve reputation value, feedback score, and identity privacy, and offload computation to honest-but-curious cloud servers. In addition, [247] proposed a lightweight centralized privacy-preserving scheme for mobile crowdsensing in vehicular networks that protected location, identity, sensing data, and reputation values. The design integrated k-anonymity-based task matching, Paillier homomorphic encryption, commitment-based reputation protection, and two zero-knowledge verification algorithms. Several studies adopted blockchain to decentralize privacy-preserving and reputation management [248], [249]. In [250], a blockchain-based efficient and traceable data sharing scheme was proposed that addresses the challenge of simultaneously ensuring message reliability and sender anonymity. The scheme combines verifiable ring signatures based on the zero-knowledge proof system to enable anonymous message signing while supporting malicious signer tracing. The reputation system comprises feedback collection by RSUs, reputation calculation incorporating weighted influence and objective scores, and reputation update using the beta distribution function. Privacy is preserved through one-time pseudonyms and the ring signature construction, which ensures unlinkability. In [251], InfoChain, a reputation-based blockchain framework for secure information sharing, was proposed. The scheme integrates a distributed reputation model directly into the blockchain through a novel Proof-of-Reputation consensus mechanism. Privacy

is also addressed through pseudonymous communication using blockchain wallet addresses and a Verifiable Random Function-based reputation verification mechanism that enables vehicles to prove their reputation level without exposing identity information. In [252], a blockchain-powered authenticated key agreement scheme for V2V communication in IoV was proposed, where privacy is preserved through pseudo-identities. A dual blockchain architecture was employed, comprising an authentication ledger for registration and revocation data, and a transaction ledger for V2V messages and trust scores. A reputation-incentive mechanism based on direct and indirect trust metrics rewards honest vehicles and penalizes malicious ones, with scores stored immutably on-chain.

3.4 Summary and Research Gaps

This chapter reviewed existing approaches for enhancing security, privacy, and trust in vehicular networks. To address **RQ1a**, two complementary taxonomies were proposed to systematically classify RL-based approaches: a focus-centric taxonomy and a taxonomy based on RL method type and SPT protection type. These taxonomies are proposed as part of the contributions of this thesis. In addition, non-RL approaches aligned with the contributions of this thesis were reviewed separately.

The review of the RL-based literature across both proposed taxonomies and the non-RL approaches reveals key research gaps that motivate the contributions of this thesis as outlined below.

Within link-centric trust management category, existing centralized RL-based routing approaches rely on coordinators such as SDN controllers that introduce single points of failure, while decentralized routing approaches either neglect trust evaluation or treat it as a secondary consideration. Therefore, existing approaches lack a fully decentralized, trust-aware routing mechanism that integrates trust evaluation directly into the routing process. This gap is addressed by the QL-TRT approach presented in Chapter 4, which integrates Q-learning-based trust evaluation into routing decisions to mitigate black hole and gray hole attacks. In addition, within vehicle-centric trust management category, existing approaches in VEC environments assume that the threshold for distinguishing trustworthy from untrustworthy vehicles is predetermined and static. The problem of dynamically adapting this threshold to changing network conditions and adversarial behaviors remains largely unaddressed. This gap is addressed by the DSTS-DDPG approach presented in Chapter 4, which dynamically learns the trust threshold.

Moreover, existing pseudonym management approaches are predominantly non-RL-based and rely on static rule-based mechanisms, such as mix-zone and mix-context techniques. In parallel, existing blockchain-based schemes focus primarily on providing immutable and decentralized pseudonym authentication and revocation, but they do not address the operational decision of pseudonyms rotation. Furthermore, the trade-off between the cost of pseudonym updates and the privacy they provide is rarely considered. This gap is addressed by the CAPM-MAPPO approach presented in Chapter 5, which combines a decentralized blockchain-based authentication and revocation protocol with a MAPPO algorithm for dynamic, cost-aware pseudonym updates.

Finally, regarding the joint provisioning of privacy and trust, existing approaches mostly employ pseudonyms to protect both identity and reputation values from disclosure. However, preserving reputation continuity across pseudonym rotations remains an open problem, since securely linking the trust information of the same vehicle under different pseudonyms is not adequately addressed. This gap is addressed by the BRMPP framework and the TAP-PPO algorithm presented in Chapter 6, which jointly manage reputation continuity and privacy through trust-aware pseudonym rotation.

Chapter 4

Reinforcement Learning-Based Trust Management in Vehicular Networks

In VEC, AVs continuously generate vast amounts of data through their onboard sensors, including traffic conditions, road hazards, and environmental information. This data holds significant value for ITS, and its sharing among vehicles and infrastructure is essential for enhancing road safety and driving efficiency [253], [254]. Data exchange in vehicular networks takes various forms depending on the context and incentive structure. In cooperative scenarios, vehicles voluntarily share data with nearby vehicles and RSUs to support collective safety and awareness. Alternatively, data sharing emerges as an incentive-driven mechanism where vehicles and RSUs act as data sellers and data requesters. Regardless of the data exchange model, a fundamental challenge remains: ensuring the trustworthiness of both the communication paths and the participating vehicles.

As discussed in Chapter 3, trust management in vehicular networks can be classified according to the primary focus of the trust evaluation into four categories: vehicle-centric, infrastructure-centric, link-centric, and data-centric [2]. This chapter addresses two of these complementary trust perspectives. The first is link-centric trust, which assesses the trustworthiness of communication links between vehicle pairs to ensure that data is forwarded securely across reliable paths. The second perspective is vehicle-centric trust, which evaluates the trustworthiness of individual vehicles based on their behavior and interactions within the network. When edge

servers request data from AVs through data sharing, the challenge of identifying credible vehicles arises.

These two trust perspectives are complementary and both critical for reliable data dissemination in VEC. A trusted communication link is provided ineffective if the data transmitted through it originates from a malicious or unreliable vehicle. Conversely, a credible data provider cannot deliver useful data if the communication path is compromised by attackers. This observation motivates two trust approaches in which both the trustworthiness of the communication path and the credibility of the vehicles are addressed.

As highlighted in Chapter 2, RL provides an effective solution to address both challenges. In response to the first research question, *"How can RL be tailored to address trust management challenges in vehicular networks (RQ1b)?"*, this chapter presents two RL-based approaches that enhance trustworthiness in vehicular network. Section 4.1 presents the Q-Learning-based Trustworthy Routing Technique (QL-TRT), a link-centric approach for trustworthy routing in VANETs. Section 4.2 presents the Dynamic Selection of Trusted Sellers using DDPG (DSTS-DDPG), a vehicle-centric approach for trust-aware data sharing in VEC.

4.1 Link-Centric Trust Management: Trustworthy Routing Using Q-Learning

VANETs are a type of ITS for CAVs. As mentioned before, there are multiple types of VANET-based V2X communications [255]. VANETs consider vehicles to be nodes with high mobility, allowing networks to form spontaneously and their topology to rapidly change. The characteristics of these networks make them vulnerable to a variety of attacks.

This section presents the proposed algorithm for link-centric trust management in VANETs. In multi-hop vehicular networks, routing protocols such as the Ad-hoc On-demand Distance Vector (AODV) are vulnerable to black hole and gray hole attacks. In a black hole attack, a malicious node advertises that it possesses the shortest route to the destination, causing the source to direct messages through it; the attacker then drops all received packets. In a gray hole attack, a malicious node selectively drops some packets whilst forwarding others, making detection particularly challenging as the attacker behaves normally for a proportion of the time. These attacks increase the packet drop rate and can cause significant damage to the vehicular infrastructure. To address these vulnerabilities, Q-learning algorithm is proposed as a trust-aware

alternative to the original AODV routing protocol. The approach evaluates the trustworthiness and Quality of Service (QoS) of directed links between vehicle pairs by considering factors such as packet forwarding ratio, energy consumption, and expected transmission time. Q-learning is then employed to learn the trust values of these links and select the most reliable route from source to destination whilst avoiding malicious nodes.

4.1.1 System Model

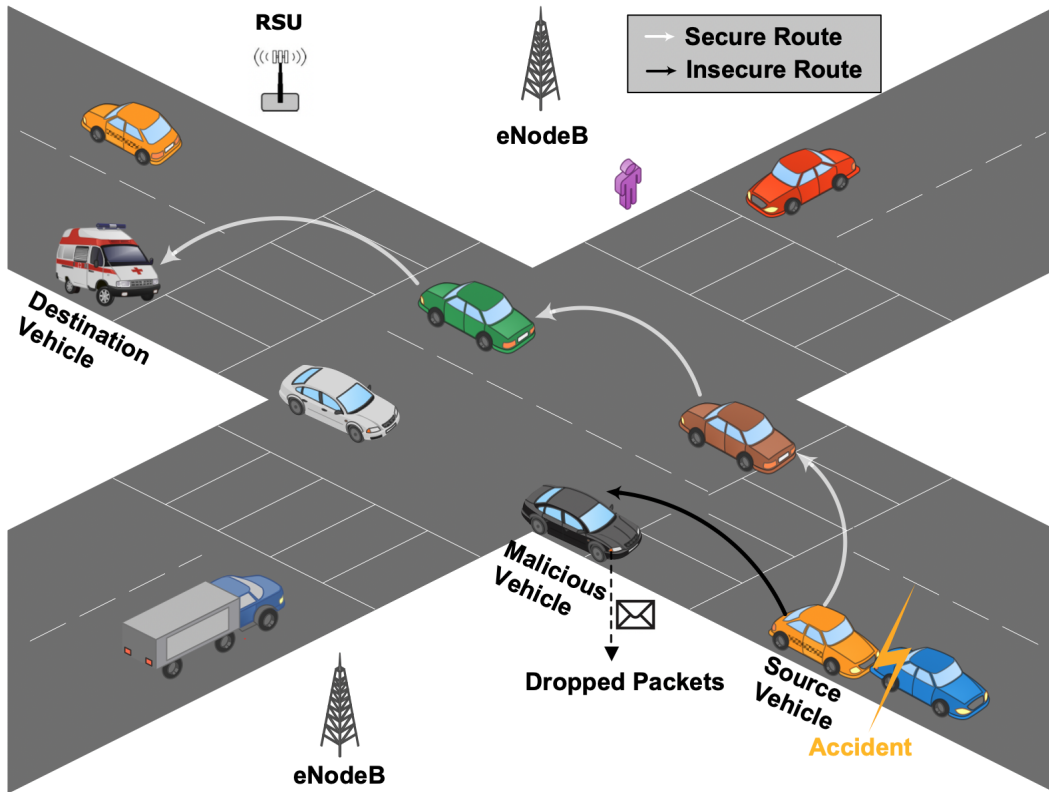


Figure 4.1: Black/gray hole attack on VANET

Figure 4.1 depicts a VANET comprising source and destination vehicles, intermediate vehicles, and RSUs. Through V2V and V2I communications, these vehicles and RSUs exchange cooperative awareness messages (CAMs), control and data packets, and emergency notifications whilst within radio range. Let $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ denote the set of N vehicles in the network. A directed link between a vehicle pair (v_i, v_j) , where $v_i, v_j \in \mathcal{V}$, indicates that v_i is the sender, v_j is the receiver, and v_j is within the transmission range of v_i . The set of neighboring vehicles of v_i , denoted

$\mathcal{N}_i = \{v_j \in \mathcal{V} \mid v_j \text{ is within the transmission range of } v_i\}$, represents the candidate next-hop vehicles to which v_i can forward packets.

Frequently, messages must be transmitted over a wide area that cannot be covered by a single hop of communication. Multi-hop routing addresses this limitation by forwarding messages from one vehicle to another until they reach the intended destination, enabling transmission over wider distances. In VANETs, where vehicles exhibit high mobility and the network topology changes rapidly, multi-hop routing is particularly essential [256]. Without loss of generality, it is assumed that there are no RSUs within close proximity of the source node; therefore, intermediate vehicles are utilized to forward packets to the destination vehicle, which is assumed to be beyond the direct transmission range of the source. This assumption represents a more challenging scenario, as the absence of nearby infrastructure necessitates that all routing decisions are made in a fully distributed manner, relying entirely on intermediate vehicles for multi-hop packet forwarding without the support of trusted infrastructure nodes. For instance, in an emergency scenario, a vehicle involved in an accident (source) may need to transmit messages to an approaching ambulance (destination) that is located well beyond single-hop range.

In ad-hoc networks, AODV is a widely adopted routing protocol. However, AODV lacks built-in trust mechanisms and is therefore vulnerable to attacks. In the AODV route discovery process, when a source node needs to communicate with a destination node and no route exists in the routing table, it floods route request (RREQ) messages to its neighbors until a RREQ reaches either the destination node or an intermediate node that holds a valid route to the destination. A route reply (RREP) message is then unicasted back along the established route [257]. When a malicious vehicle acts as an intermediate node, it can send a fabricated RREP without consulting the routing table, thereby attracting traffic. The malicious vehicle then drops the received packets instead of forwarding them to the destination. To address this vulnerability, a trust-based QoS routing protocol is proposed that employs Q-learning to evaluate link-level trust and detect black hole and gray hole attacks in VANETs whilst ensuring improved network performance.

4.1.2 Problem Formulation

In this section, we describe the proposed Q-learning trustworthy routing technique (QL-TRT) algorithm for mitigating black hole and gray hole attacks in vehicular networks using trusted route discovery. There are differences in the process for establishing a secure path from source to destination compared to the original

AODV. In order to establish a secure and reliable path from the source to the destination, our proposed scheme involves calculating the trustworthiness of the vehicles and the QoS of the links. The behavior of each vehicle is observed by its neighbors in order to determine whether it is a trusted vehicle or a malicious one. The next sections describe the process of discovering a reliable trusted route, as well as a demonstration of how Q-learning is used.

Computation of Link Trust Value

A trust-based QoS routing process involves selecting the best relay candidate from among potential forwarders. In order to ensure reliability and QoS, we propose a trust-based QoS routing algorithm that calculates the trustworthiness and reliability of the direct links between each pair of vehicles, and selects the most reliable candidate from a list of potential candidates. In the first step, the source vehicle utilizes the proposed QL-TRT scheme to compute the trust value of links to different neighbors. Next, the source vehicle selects the neighbor with the highest trust value and unicasts the message to it. Each vehicle in each state that holds packets will repeat this process until the packet reaches its destination. The trust value of each directed link to each neighbor will be determined based on the following factors.

- *Packet Forwarding Ratio (PFR)*: The PFR is a measure of the efficiency of packet forwarding in a vehicular network. For a directed link (v_i, v_j) , vehicle v_i monitors the forwarding behavior of its next-hop neighbor v_j through a passive overhearing mechanism. Since wireless transmissions are broadcast in nature, v_i can overhear whether v_j retransmits the packet to the subsequent hop, thereby estimating v_j 's forwarding ratio without requiring a central monitoring entity [258]. A significant drop in PFR may indicate that packets are being discarded by a malicious node, suggesting a black hole or gray hole attack. The PFR of vehicle v_j as observed by vehicle v_i during time period t is formalized as:

$$PFR_t(v_i, v_j) = \frac{F_t(v_j)}{S_t(v_i, v_j)} \quad (4.1)$$

where $S_t(v_i, v_j)$ denotes the number of packets sent from vehicle v_i to v_j during time period t , and $F_t(v_j)$ denotes the number of packets overheard by v_i as successfully forwarded by v_j . Note that PFR is defined directly over the directed link rather than as a per-node metric, since the forwarding behavior of v_j is observed specifically with respect to packets received from v_i .

- *Residual Energy Ratio*: To ensure sustained and reliable packet forwarding, the energy availability of the nodes forming a link is considered. The residual energy ratio of a vehicle v_i is defined as the ratio of its remaining energy E_{res_i} to its initial energy E_{init_i} :

$$E_i = \frac{E_{res_i}}{E_{init_i}} \quad (4.2)$$

where $E_i \in [0, 1]$, with a value of 1 indicating a fully charged vehicle and a value approaching 0 indicating near energy depletion. In fact, a higher value of E_i indicates that the vehicle has more energy available to maintain its communication capabilities. Each vehicle computes its own residual energy ratio locally and includes it in the periodic beacon messages broadcast to neighboring vehicles. The residual energy ratio of a directed link (v_i, v_j) during time period t is defined as the minimum of both endpoints, reflecting that the link reliability is bounded by the node with the least available energy:

$$E_t(v_i, v_j) = \min\{E_t(v_i), E_t(v_j)\} \quad (4.3)$$

- *Expected Transmission Time (ETT)*: The ETT is a link-specific metric that evaluates the performance of the wireless link between a vehicle pair. It is defined as the expected time required to transmit a packet from vehicle v_i to a neighboring vehicle v_j [259]:

$$ETT_t(v_i, v_j) = \frac{1}{D_f(v_i, v_j) \times D_r(v_i, v_j)} \times \frac{M}{B} \quad (4.4)$$

where $D_f(v_i, v_j)$ represents the forward delivery ratio, i.e., the probability that a data packet transmitted by v_i is successfully delivered to v_j , and $D_r(v_i, v_j)$ represents the reverse delivery ratio, i.e., the probability that the corresponding acknowledgment packet is successfully received by v_i from v_j . Furthermore, M denotes the packet size and B is the data rate. These delivery ratios are estimated from historical transmission statistics on the link between the vehicle pair. A lower ETT indicates a more efficient link; therefore, the inverse $ETT_t(v_i, v_j)^{-1}$ is employed in the link trust value computation to ensure that faster links contribute positively to the overall trust score.

In order to select the most reliable link over which to forward a packet, each vehicle that holds a packet calculates the trust value of links to its directed neighbors based on the defined factors. The link trust value is formalized as:

$$TV_t(v_i, v_j) = w_1 \times PFR_t(v_i, v_j) + w_2 \times E_t(v_i, v_j) + w_3 \times \left(ETT_t(v_i, v_j)\right)^{-1} \quad (4.5)$$

$$\text{s.t. } w_1, w_2, w_3 \geq 0 \quad (4.6a)$$

$$w_1 + w_2 + w_3 = 1 \quad (4.6b)$$

where $TV_t(v_i, v_j)$ denotes the trust value of the directed link from v_i to v_j , with higher values indicating a more trustworthy and reliable link. The link trust value combines three complementary factors: a high packet forwarding ratio (indicating honest forwarding behavior), a high residual energy ratio (indicating sufficient energy to sustain communication), and a low expected transmission time (indicating efficient link performance). The weights w_1 , w_2 , and w_3 reflect the relative importance of each factor and can be adjusted based on the priority of the routing scenario. For instance, if detecting and avoiding malicious nodes is the primary concern, a higher weight may be assigned to the packet forwarding ratio.

Trustworthy Route Selection Based on Q-Learning

In this subsection, the QL-TRT routing problem is formulated as a MDP, in which each vehicle holding a packet acts as an RL agent that selects the most trustworthy next-hop neighbor based on the current network state. The MDP formulation enables the agent to make routing decisions based solely on the current state without relying on the history of previous states, making it well-suited to the dynamic topology of VANETs.

The MDP for the QL-TRT algorithm is defined as follows.

State Space: A state $s_{v_i} \in \mathcal{S}$ is defined as the vehicle v_i currently holding the packet together with its set of neighboring vehicles within transmission range. The state captures the local topology surrounding the packet holder at a given time step.

Action Space: The set of actions $\mathcal{A}$ available to the agent at state s_{v_i} is the set of neighboring vehicles to which the packet can be forwarded. Selecting action $a_{v_j} \in \mathcal{A}$ corresponds to forwarding the packet to neighboring vehicle v_j . An ϵ -greedy strategy is adopted for action selection, defined as:

$$a_{v_j} = \begin{cases} \arg \max_{a \in \mathcal{A}_i} Q(s_{v_i}, a), & \text{with probability } 1 - \epsilon \\ \text{random } a \in \mathcal{A}_i, & \text{with probability } \epsilon \end{cases} \quad (4.7)$$

where $\epsilon \in [0, 1]$ controls the trade-off between exploitation of the current best-known trustworthy neighbor and exploration of alternative forwarding paths.

Reward Function: The reward $R(s_{v_i}, a_{v_j})$ is defined as the link trust value $TV_t(v_i, v_j)$ computed in Equation (4.5). A higher trust value results in a higher reward, incentivizing the agent to forward packets through more trustworthy and reliable links.

Q-Value Initialization and Route Selection: All Q-values in the Q-table are initialized to zero. At each hop, the vehicle holding the packet evaluates its neighboring links and forwards the packet to the neighbor with the highest Q-value. This process is repeated at each intermediate vehicle until the packet reaches its destination.

The Q-value for each state-action pair is updated using the standard Q-learning update rule:

$$Q(s_{v_i}, a_{v_j}) = (1 - \alpha) \times Q(s_{v_i}, a_{v_j}) + \alpha \left(R(s_{v_i}, a_{v_j}) + \gamma \times \max_{a \in \mathcal{A}_j} Q(s_{v_j}, a) \right) \quad (4.8)$$

where $\alpha \in [0, 1]$ is the learning rate, $\gamma \in [0, 1]$ is the discount factor, $R(s_{v_i}, a_{v_j}) = TV_t(v_i, v_j)$ is the immediate reward defined by the link trust value computed in Equation (4.5), and s_{v_j} denotes the next state reached after forwarding the packet to vehicle v_j . The term $\max_{a \in \mathcal{A}_j} Q(s_{v_j}, a)$ represents the maximum Q-value over all feasible actions in the next state, capturing the expected cumulative trust along the remaining path to the destination. Each vehicle includes its current maximum Q-value in its periodic beacon messages broadcast to neighboring vehicles. Through this recursive structure, the Q-value at each hop reflects not only the immediate link trustworthiness but also the quality of the downstream path, enabling end-to-end trust-aware route selection. The specific values of α and γ are detailed in the simulation setup.

Over multiple routing episodes, each vehicle refines its Q-values based on observed link trust values. The convergence of the Q-values indicates that the vehicles have learned the forwarding actions to maximize the cumulative trust along the route,

Algorithm 4.I: QL-TRT: Q-Learning-Based Trustworthy Routing

Input: Set of vehicles $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$, source vehicle v_s , destination vehicle v_d , learning rate α , discount factor γ , exploration rate ϵ , trust value weights w_1, w_2, w_3

Output: Trusted route from v_s to v_d

Initialise Q-table: $Q(s_{v_i}, a_{v_j}) \leftarrow 0, \forall v_i, v_j \in \mathcal{V}$

for each routing episode **do**

 Set current vehicle $v_i \leftarrow v_s$

while $v_i \neq v_d$ **do**

 /* Phase 1: Link Trust Computation */

 Identify neighboring vehicles $\mathcal{N}_i$

for each neighbor $v_j \in \mathcal{N}_i$ **do**

 Compute $PFR_t(v_i, v_j)$ via Eq. (4.1) using overhearing

 Obtain $E_t(v_i, v_j)$ via Eq. (4.3) from beacon messages

 Estimate $ETT_t(v_i, v_j)$ via Eq. (4.4) from delivery ratios

 Compute link trust value $TV_t(v_i, v_j)$ via Eq. (4.5)

 /* Phase 2: Action Selection (ϵ -greedy) */

if random number $< \epsilon$ **then**

 Select random neighbor $v_j \in \mathcal{N}_i$

else

$v_j \leftarrow \arg \max_{a \in \mathcal{A}_i} Q(s_{v_i}, a)$

 /* Phase 3: Q-Value Update */

 Set reward $R(s_{v_i}, a_{v_j}) \leftarrow TV_t(v_i, v_j)$

if $v_j = v_d$ **then**

$Q(s_{v_i}, a_{v_j}) \leftarrow (1 - \alpha) \times Q(s_{v_i}, a_{v_j}) + \alpha \times R(s_{v_i}, a_{v_j})$

else

$Q(s_{v_i}, a_{v_j}) \leftarrow$

$(1 - \alpha) \times Q(s_{v_i}, a_{v_j}) + \alpha (R(s_{v_i}, a_{v_j}) + \gamma \times \max_{a \in \mathcal{A}_j} Q(s_{v_j}, a))$

 /* Phase 4: Packet Forwarding */

 Forward packet to v_j

$v_i \leftarrow v_j$

resulting in a trust-aware routing policy that selects reliable paths while avoiding malicious nodes.

Algorithm 4.I presents the pseudocode of the proposed QL-TRT routing algorithm.

4.1.3 Performance Evaluation

The performance of the proposed QL-TRT algorithm is evaluated against two baseline routing protocols: the traditional AODV routing protocol [257], which lacks

Table 4.1: Simulation Parameters for the QL-TRT Algorithm

Parameter	Value
Network size	5 km highway
Number of lanes	4 (two in each direction)
Number of vehicles	100
Speed of vehicles	20 km/h $\sim$ 80 km/h
Signal transmission range	250 m
Packet size (M)	512 bytes
Packet transmission rate	5 packets/sec
Simulation time	1000 s
Number of simulation runs	100
Percentage of malicious vehicles	0% $\sim$ 30%
Gray hole packet drop rate	0% $\sim$ 50%
Learning rate (α)	0.5
Discount factor (γ)	0.9
Exploration rate (ϵ)	0.1
w_1 , w_2 and w_3	0.5, 0.2, 0.3

any trust mechanism, and the Trust-based QoS Routing (TQR) protocol [260], which incorporates trust evaluation but does not employ RL for route selection. To evaluate the performance of the proposed approach, the simulation is implemented in MATLAB. The experimental environment follows the 3GPP TR 36.885 highway model [261], featuring a network size defined by a 5 km multi-lane highway. Within this setup, 100 vehicles are randomly distributed and travel at a constant speed in the range 20–80 km/h. The configuration parameters are presented in Table 4.1. To ensure accurate and reliable results, 100 independent simulation runs were conducted for each experiment, and the average values are reported. The evaluation metrics are defined as follows.

- *Packet Delivery Ratio (PDR)*: The ratio of the number of packets successfully received by destination vehicles to the total number of packets sent by source vehicles.
- *End-to-End Delay*: The time elapsed from the transmission of a packet by the source node to its reception at the destination node, including wireless transmission time as well as any processing and queuing delays at intermediate nodes along the route.

The simulation includes varying percentages of malicious vehicles in the network, ranging from 0% to 30% of the total number of vehicles, comprising both black hole and gray hole attackers. Black hole attackers drop all packets they receive, whilst gray hole attackers selectively drop packets at a rate ranging from 0% to 50%, exhibiting a mixture of cooperative and malicious behavior. These malicious nodes are introduced to evaluate the ability of the proposed QL-TRT algorithm to detect and avoid such attacks whilst maintaining the overall security and reliability of communication in the network.

As illustrated in Figure 4.2, the PDR of all three algorithms decreases as the percentage of malicious vehicles in the network increases. However, the proposed QL-TRT algorithm consistently outperforms both AODV and TQR across all malicious vehicle percentages. At 30% malicious vehicles, QL-TRT achieves a PDR of approximately 61%, outperforming AODV by approximately 85% and TQR by approximately 39%. Notably, QL-TRT maintains a PDR above 70% even at 20% malicious vehicles, whereas AODV drops to 40% and TQR to 58% under the same conditions.

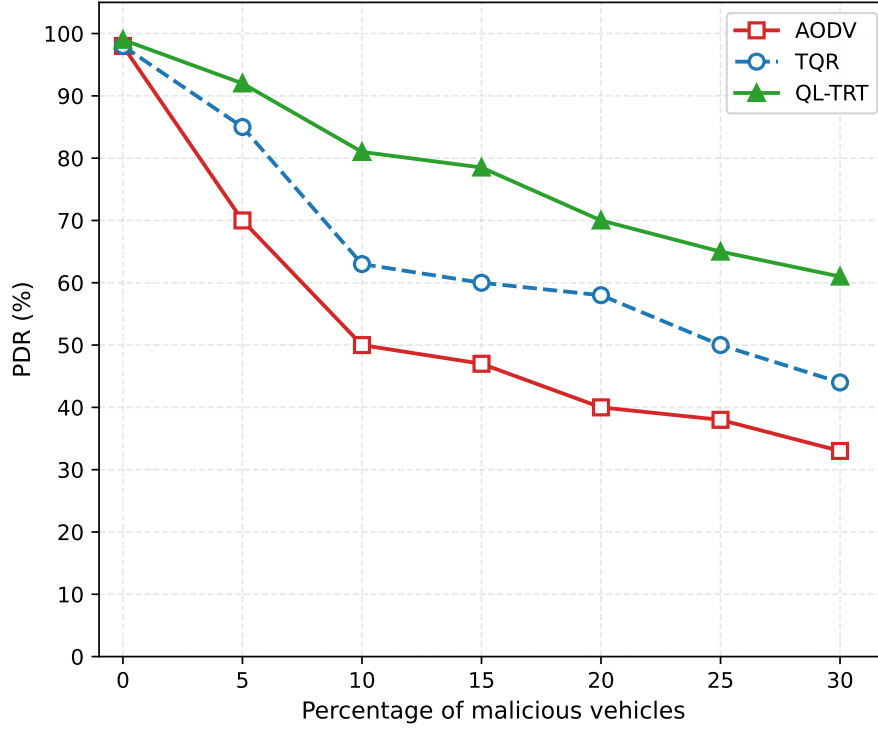


Figure 4.2: PDR with a variable number of malicious vehicles

Figure 4.3 shows the end-to-end delay as a function of the percentage of malicious vehicles, ranging from 0% to 30%. As the proportion of malicious nodes increases, the end-to-end delay in AODV increases dramatically, rising from approximately 35 ms at 0% to 350 ms at 30% malicious vehicles. This is attributed to repeated route failures, packet retransmissions, and route rediscovery processes caused by malicious nodes dropping packets. TQR exhibits a more moderate increase, reaching approximately 82 ms at 30%, as its trust evaluation mechanism avoids some malicious nodes. In contrast, QL-TRT maintains a consistently low delay, increasing only marginally from 35 ms to 48 ms across the entire range. At 30% malicious vehicles, QL-TRT reduces the end-to-end delay by approximately 86% compared to AODV.

In Figure 4.4 and Figure 4.5, the percentage of malicious vehicles is fixed at 20% of the total number of vehicles, and the performance of QL-TRT is evaluated under varying vehicle speeds ranging from 20 km/h to 80 km/h.

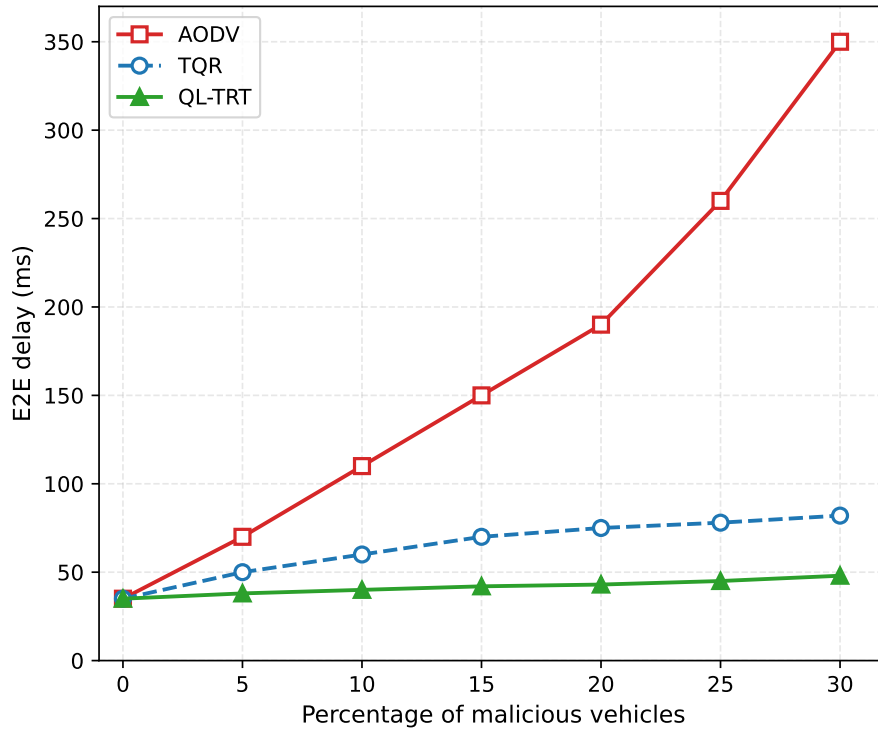


Figure 4.3: End-to-end delay with variable number of malicious nodes

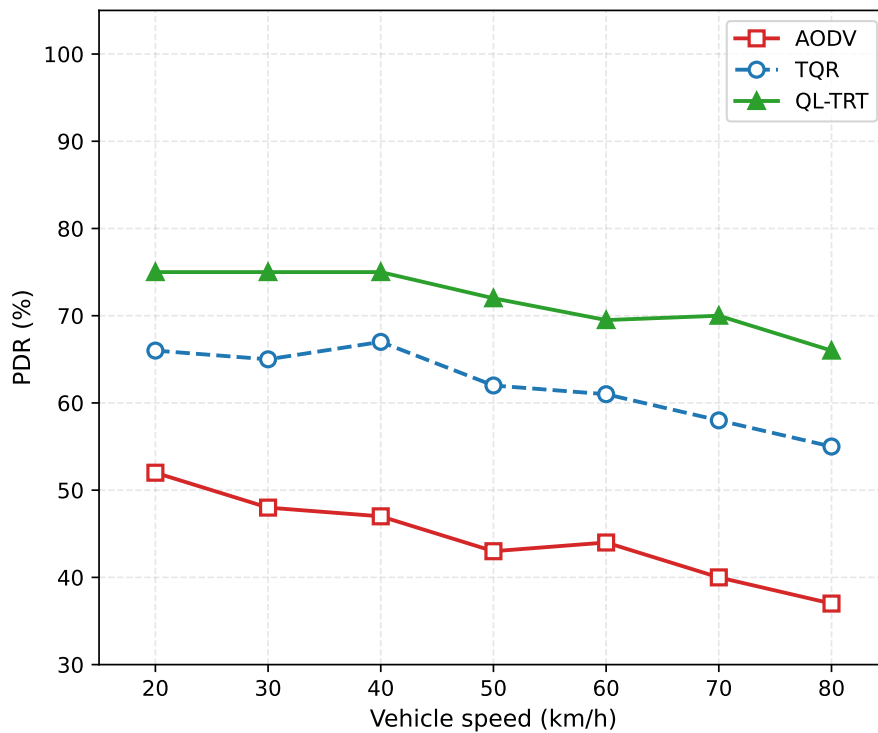


Figure 4.4: PDR as a function of vehicle speed

Figure 4.4 presents the PDR as a function of vehicle speed. The proposed QL-TRT algorithm achieves a consistently higher PDR than both AODV and TQR across all speeds. At 80 km/h, QL-TRT maintains a PDR of approximately 66%, outperforming AODV by approximately 78% and TQR by approximately 20%. Although all three algorithms exhibit a gradual decline in PDR as vehicle speed increases, due to more frequent topology changes and link disruptions, QL-TRT degrades most gracefully.

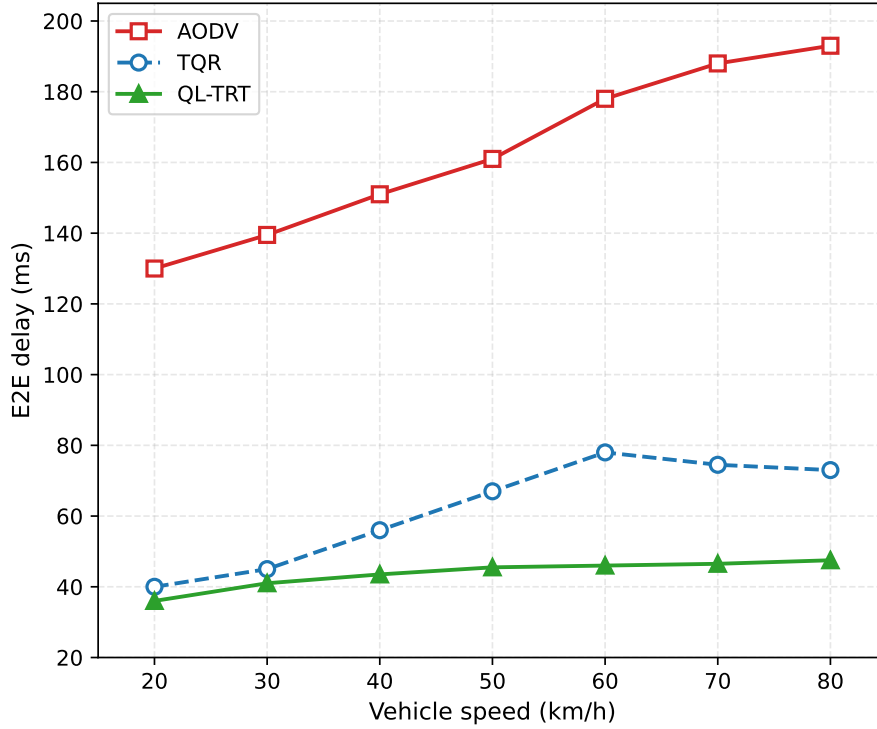


Figure 4.5: End-to-end delay as a function of vehicle speed

Figure 4.5 presents the end-to-end delay as a function of vehicle speed under the same conditions, with 20% of vehicles considered malicious. The end-to-end delay of AODV increases substantially with speed, rising from approximately 130 ms at 20 km/h to 198 ms at 80 km/h, as higher mobility exacerbates route failures through malicious nodes and necessitates frequent route rediscovery. TQR exhibits a moderate increase from 40 ms to 75 ms, whilst QL-TRT maintains a consistently low delay, increasing only marginally from 35 ms to 48 ms. At 80 km/h, QL-TRT reduces the end-to-end delay by approximately 75% compared to AODV and by approximately 35% compared to TQR.

4.1.4 Summary

The results presented demonstrate the effectiveness of the QL-TRT algorithm in establishing trustworthy routing paths in VANETs under varying attack densities and vehicle speeds. By evaluating link-level trust through packet forwarding ratio, residual energy ratio, and expected transmission time, and by employing Q-learning to adaptively select the most reliable forwarding paths, QL-TRT consistently outperforms both AODV and TQR in terms of PDR and end-to-end delay. However, QL-TRT addresses trust at the communication link level, it ensures that packets reach their destination through reliable paths, but it does not evaluate the trustworthiness of the vehicles themselves as data providers. In scenarios where edge servers request data from vehicles through incentive-driven data trading, a complementary mechanism is required to assess the credibility of participating vehicles and select the most trustworthy candidates. The following section addresses this challenge through a vehicle-centric trust management approach based on DRL.

4.2 Vehicle-Centric Trust Management: Dynamic Selection of Trusted Sellers Using DDPG

The growing importance of communication between AVs and their surroundings in the IoV, facilitated by V2X communication [33], [34], has enabled data exchange among AVs and infrastructure entities. Edge servers request data from AVs to support a variety of intelligent transportation applications, and AVs act as data providers that collect and upload the requested data to the edge server. However, the challenge of identifying credible vehicles remains a fundamental concern. Malicious or unreliable AVs may provide fabricated or low-quality data, undermining the value of the collected information. This highlights the importance of reputation management, which enables edge servers to assess the credibility of potential data providers. While the preceding section addressed trust at the link level to ensure reliable packet forwarding, this section shifts the focus to evaluating the trustworthiness of individual vehicles as data providers during data exchange in VEC. This section presents a novel approach for vehicle-centric trust management in VEC during data exchange by using DRL. Hence, Dynamic Selection of Trusted Sellers Using DDPG (DSTS-DDPG) has been proposed where the term "sellers" in the acronym refers to vehicles acting as data providers in the data exchange process. DSTS-DDPG employs a reputation management mechanism combined with DDPG to dynamically optimize the credibility score threshold for selecting trusted AVs.

The following subsections present the system model, the dynamic credibility score calculation, and the proposed DSTS-DDPG algorithm.

4.2.1 System Model

In the proposed hierarchical architecture, AVs equipped with sensors on the road constitute the **vehicle level**, RSUs integrated with edge servers constitute the **edge level**, and a central cloud server constitutes the **cloud level**. The VEC servers are assumed to be equipped with ML capabilities and are responsible for selecting trusted AVs. To ensure trustworthiness and reliability in data exchange, the edge server operates as an RL agent that learns to select AVs with high credibility scores for data exchange. Reputation management serves as an effective method to facilitate the selection of trusted AVs [262]. V2X communication links facilitate transmission between AVs and VEC servers, whereas wired backhaul links support the transmission between the VEC server and the cloud. This integrated architecture enables seamless communication among AVs and allows the cloud server to maintain credibility scores for all AVs in the network.

As illustrated in Figure 4.6, the main components of the network architecture are:

- *Data Provider/Seller*: AVs possess the capacity to collect and provide data to edge servers. Let $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ denote the set of data providers within a cluster. The data provider/seller AVs are considered to be untrustworthy by default, and some malicious providers may exist among them that could affect the security and trustworthiness of the network.
- *Data Requester/Buyer*: Data requesters are the edge servers integrated with RSUs. Let $\mathcal{J} = \{S_1, S_2, \dots, S_J\}$ denote the set of VEC servers in the network. Data requesters broadcast their tasks across the network and acquire the requested data from AVs. The data requesters are assumed to be trusted entities in the network, authorized by the trust authority.
- *Cluster*: The network is organized into distinct clusters, denoted by $\mathcal{C} = \{c_1, c_2, \dots, c_J\}$. Each cluster corresponds to the coverage area of an edge server, and all AVs within a cluster are served by the corresponding edge server.

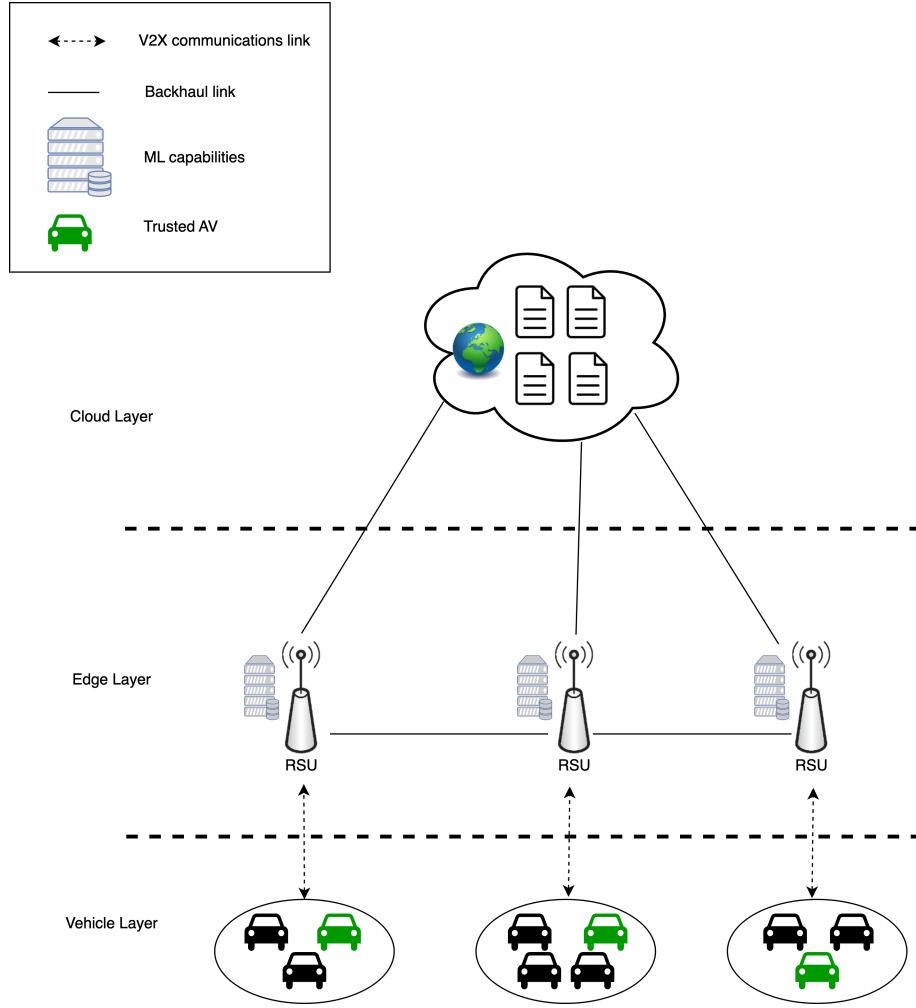


Figure 4.6: System Model

- *Transaction*: A transaction $k \in \mathcal{K}$ represents a data exchange session between a data provider and a data requester. Each AV is capable of participating in distinct transactions.

Within each cluster, the VEC server retrieves the recorded credibility value from the cloud and utilizes various parameters to calculate the credibility score of each AV through the proposed DRL-based method for each transaction. Based on these scores, the edge server selects the data provider with the highest credibility score among the available candidates. The calculated credibility scores are uploaded to the cloud, enabling score sharing across edge servers when an AV leaves one coverage area and enters another cluster.

4.2.2 Dynamic Credibility Score Calculation

To select trusted sellers from a pool of available candidates, various metrics must be evaluated comprehensively. In this context, the concept of credibility is introduced to capture the reliability, trustworthiness, and reputation of data providers. Through the evaluation of different metrics, the degree of credibility among unknown AVs can be managed, thereby identifying malicious providers and ensuring that the exchanged data originates from sources with higher credibility values.

Trust and Reputation Modeling

The credibility score of an AV v_i in a cluster c_j for transaction k is represented by $\varrho_i^k = \{b_i^k, d_i^k, n_i^k, u_i^k, q_i^k, a\}$, where $b_i^k, d_i^k, n_i^k, u_i^k \in [0, 1]$ and $b_i^k + d_i^k + n_i^k + u_i^k = 1$, denoting belief, distrust, posteriori uncertainty, and priori uncertainty (without evidence), respectively. The parameter $q_i^k \in [0, 1]$ represents the successful transmission probability of the data packet, which determines the priori uncertainty u_i^k of the credibility score ϱ_i^k . The base rate $a \in [0, 1]$ is a predefined constant formed from an existing impression without solid evidence. The components b_i^k , d_i^k , n_i^k , and u_i^k are calculated from the statistics of past transaction experience as follows [263]:

$$b_i^k = (1 - u_i^k) \frac{\alpha_i^k}{\alpha_i^k + \beta_i^k + \gamma_i^k} \quad (4.9)$$

$$d_i^k = (1 - u_i^k) \frac{\beta_i^k}{\alpha_i^k + \beta_i^k + \gamma_i^k} \quad (4.10)$$

$$n_i^k = (1 - u_i^k) \frac{\gamma_i^k}{\alpha_i^k + \beta_i^k + \gamma_i^k} \quad (4.11)$$

$$u_i^k = (1 - q_i^k) \quad (4.12)$$

where α_i^k , β_i^k , and γ_i^k denote the number of positive, negative, and neutral transactions for AV v_i up to transaction k , respectively. Three types of transactions are distinguished: honest transactions correspond to cooperative data providers, dishonest transactions correspond to malicious AVs, and neutral transactions are associated with AVs that accidentally cause network failures leading to adverse effects. The expected credibility score assigned by the VEC server to AV v_i in transaction k is:

$$R_i^k = b_i^k + a \cdot n_i^k + \frac{1}{2} u_i^k \quad (4.13)$$

Familiarity

When AVs transmit data to the edge server, the edge server assesses the degree of information familiarity at each transaction k . A higher familiarity value indicates that the edge server has more prior knowledge about AV v_i , thereby increasing its trust score.

$$\Xi_i^k = b_i^k + d_i^k + n_i^k = 1 - u_i^k \quad (4.14)$$

Freshness

The credibility scores of AVs at each transaction are stored in the cloud, forming an accessible repository of historical information. However, emphasis is placed on prioritizing recent transactions, with a heightened focus on the current credibility score of AVs compared to older records. Recent interactions between an AV v_i and an edge server S_j carry greater weight than past events. The freshness of the exchanged data from AV v_i in transaction k is denoted as $\psi_i^k = \lambda^{t_k - t_{k-1}}$, where $0 < \lambda < 1$ and $t_k - t_{k-1}$ measures the time difference between two sequential transactions, providing a measure of how current the transaction is relative to the previous one.

To calculate the credibility score ($\mathcal{CSC}_i$) for AV v_i in transaction $k + 1$, the familiarity and freshness factors are considered alongside the historical reputation information as follows:

$$\chi_i^k = W_1 \Xi_i^k + W_2 \psi_i^k \quad (4.15)$$

$$\mathcal{CSC}_i^{k+1} = \frac{\sum_{l=1}^k \chi_i^l R_i^l}{\sum_{l=1}^k \chi_i^l} \quad (4.16)$$

where $W_1, W_2 \geq 0$ and $W_1 + W_2 = 1$. W_1 is the weight assigned to the familiarity factor and W_2 is the weight assigned to the freshness factor. These weights reflect the relative importance of each factor in the overall calculation of χ_i^k . The summation index $l \in \{1, 2, \dots, k\}$ iterates over all past transactions up to transaction k . The overall credibility score $\mathcal{CSC}_i^{k+1}$ for future transactions incorporates both historical reputation data (through R_i^l) and the calculated familiarity and freshness factors, weighted appropriately through χ_i^l . This score provides a robust mechanism to evaluate which AVs are trustworthy for data exchange in a VEC environment.

Based on the credibility scores computed above, a method is required to determine which AVs are sufficiently trustworthy and which should be excluded from the data exchange process. To address this, a DRL-based model is proposed to

dynamically optimize the credibility score threshold, enabling the adaptive selection of trusted AVs and transactions, as described in the following subsection.

4.2.3 DSTS-DDPG Proposed Algorithm

The process of selecting the most trustworthy AV (or AVs) from a pool of available candidates by an edge server can be effectively modeled as an MDP. Conventional credibility-based selection techniques rely on a fixed threshold: if a data provider's credibility score CSC_i falls below this threshold, the vehicle is considered untrustworthy or malicious. However, a fixed threshold cannot adapt to the dynamic behavior of AVs in vehicular networks. To overcome this limitation, DSTS-DDPG dynamically optimizes the credibility score threshold using DRL, enabling adaptive selection of trusted AVs as network conditions evolve.

It is essential to optimize this credibility score threshold dynamically to mitigate the adverse effects of data exchange with malicious AVs. Establishing a high credibility score threshold risks excluding genuinely trustworthy data providers, whilst a low threshold may permit transactions with potentially malicious AVs, resulting in the incorporation of inaccurate data into the system. Hence, the credibility score threshold should be optimally determined to enhance trustworthiness during data exchange and avoid malicious AVs. To address this challenge, a DRL-based mechanism is proposed to dynamically optimize the credibility score threshold.

State Space

In the proposed DSTS-DDPG algorithm, the edge server acts as the DRL agent. At each time step t , the state $s_t \in \mathcal{S}$ is defined as the vector of credibility scores of all candidate data providers within the cluster:

$$s_t = [CSC_1^t, CSC_2^t, \dots, CSC_N^t] \quad (4.17)$$

where CSC_i^t denotes the credibility score of the i -th candidate data provider at time step t , and N represents the total number of available data providers within the cluster. This formulation captures the variability in the number of candidate data providers, reflecting the dynamic nature of vehicular networks.

Action Space

The action space determines the credibility score threshold that governs the selection of trusted AVs. The action taken by the agent at time step t , denoted a_t , involves setting the value of a continuous threshold parameter θ_t :

$$a_t = \theta_t, \quad \text{where} \quad \theta_{\min} \leq \theta_t \leq \theta_{\max} \quad (4.18)$$

The threshold θ_t is continuously adjusted as the DDPG agent interacts with the environment. At each time step, AVs with credibility scores satisfying $\mathcal{CSC}_i^t \geq \theta_t$ are selected as trusted data providers for the data exchange, whilst those below the threshold are excluded. The continuous nature of the action space motivates the use of DDPG, which is well-suited to optimization problems with continuous action domains.

Reward Function

The reward function is designed to optimize the selection strategy by balancing two objectives: selecting highly credible AVs and maintaining threshold stability. The reward at time step t is defined as:

$$r_t = \rho \cdot \overline{\mathcal{CSC}}^t - \kappa \cdot |\Delta\theta_t| \quad (4.19)$$

where $\overline{\mathcal{CSC}}^t$ represents the average credibility score of the subset of AVs selected as trusted data providers during time step t (i.e., those satisfying $\mathcal{CSC}_i^t \geq \theta_t$). The term $\Delta\theta_t = \theta_t - \theta_{t-1}$ quantifies the change in the threshold from the previous time step, and κ serves as a penalty coefficient that discourages substantial threshold fluctuations. The parameter ρ controls the influence of the average credibility score in the reward function, subject to the constraint:

$$\rho + \kappa = 1, \quad \rho, \kappa \geq 0 \quad (4.20)$$

This reward structure promotes an equilibrium between selecting AVs with high credibility scores (through the first term) and maintaining a stable threshold strategy (through the penalty term), thereby supporting consistent and reliable trust management in the data exchange process.

As introduced in Chapter 2, DDPG is an actor-critic DRL algorithm well-suited to large-scale environments and continuous action spaces [264], [265], [266]. In the context of DSTS-DDPG, the task is to determine the optimal credibility score threshold θ_t for selecting trusted data providers. The DDPG algorithm consists of four networks: an online actor network parameterized by φ^μ , an online critic network parameterized by φ^Q , and their corresponding target networks parameterized by $\varphi^{\mu'}$ and $\varphi^{Q'}$, respectively.

The actor network maps the current state s_t to an action through a deterministic policy function $\mu(s_t|\varphi^\mu) + \mathcal{N}$, where $\mathcal{N}$ represents exploration noise added to balance exploration and exploitation. The critic network estimates the Q-value $Q(s_t, a_t|\varphi^Q)$, which represents the expected cumulative reward when taking action a_t in state s_t . The policy gradient used to update the actor network is:

$$\nabla \varphi^\mu = \mathbb{E} \left[\nabla_a Q(s, a|\varphi^Q) \Big|_{s=s_t, a=\mu(s_t)} \nabla_{\varphi^\mu} \mu(s|\varphi^\mu) \Big|_{s_t} \right] \quad (4.21)$$

The critic network is updated by minimizing the loss function L , computed over a mini-batch of samples drawn from the experience replay memory $\mathcal{D}$:

$$Q(s_t, a_t|\varphi^Q) = \mathbb{E} [r_t + \zeta Q(s_{t+1}, a_{t+1}|\varphi^Q)] \quad (4.22)$$

$$L(\varphi^Q) = \mathbb{E} \left[\left(Q(s_t, a_t|\varphi^Q) - \left(r_t + \zeta Q'(s_{t+1}, a_{t+1}|\varphi^{Q'}) \right) \right)^2 \right] \quad (4.23)$$

where $a_{t+1} = \mu'(s_{t+1}|\varphi^{\mu'})$ is the action selected by the target actor network at state s_{t+1} , and $\zeta \in [0, 1]$ is the discount factor. The target networks $\varphi^{\mu'}$ and $\varphi^{Q'}$ share the same architecture as their online counterparts but are updated gradually using a soft update factor $\tau \in (0, 1)$:

$$\begin{cases} \varphi^{\mu'} \leftarrow (1 - \tau)\varphi^{\mu'} + \tau\varphi^\mu \\ \varphi^{Q'} \leftarrow (1 - \tau)\varphi^{Q'} + \tau\varphi^Q \end{cases} \quad (4.24)$$

This soft update mechanism ensures training stability by preventing abrupt changes to the target values, thereby avoiding destructive feedback and divergence. An experience replay memory $\mathcal{D}$ is employed to store training samples. During the training process, the agent randomly selects mini-batches from $\mathcal{D}$, disrupting temporal correlations and mitigating non-stationary distribution issues inherent in sequentially collected samples.

We explain the DDPG-based credibility score threshold selection in Algorithm 4.II.

Algorithm 4.II: DSTS-DDPG: Dynamic Selection of Trusted Data Seller
Using DDPG

Input: Set of data providers $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$, learning rate of actor η_a , learning rate of critic η_c , discount factor ζ , soft update factor τ , exploration noise $\mathcal{N}$, replay memory size $|\mathcal{D}|$, batch size $\mathcal{B}$, reward parameters ρ, κ

Output: Optimized credibility score threshold θ^*

Initialize online actor network φ^μ and online critic network φ^Q with random weights

Initialize target networks: $\varphi^{\mu'} \leftarrow \varphi^\mu, \varphi^{Q'} \leftarrow \varphi^Q$

Initialize experience replay memory $\mathcal{D} \leftarrow \emptyset$

for each training episode **do**

 Observe initial state $s_0 = [\mathcal{CSC}_1^0, \mathcal{CSC}_2^0, \dots, \mathcal{CSC}_N^0]$

for each time step t **do**

 /* Phase 1: Action Selection */

 Select action $a_t = \mu(s_t | \varphi^\mu) + \mathcal{N}$

 Set threshold $\theta_t \leftarrow a_t$

 /* Phase 2: Trusted Data Provider Selection */

 Select trusted data providers: $\mathcal{V}_{\text{trusted}}^t \leftarrow \{v_i \in \mathcal{V} \mid \mathcal{CSC}_i^t \geq \theta_t\}$

 /* Phase 3: Reward Computation */

 Compute average credibility score $\overline{\mathcal{CSC}}^t$ over $\mathcal{V}_{\text{trusted}}^t$

 Compute reward $r_t = \rho \cdot \overline{\mathcal{CSC}}^t - \kappa \cdot |\Delta\theta_t|$ via Eq. (4.19)

 Observe next state s_{t+1}

 /* Phase 4: Experience Storage */

 Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}$

 /* Phase 5: Network Update */

if $|\mathcal{D}| \geq \mathcal{B}$ **then**

 Sample mini-batch of size $\mathcal{B}$ from $\mathcal{D}$

 Compute target Q-value:

$y_t = r_t + \zeta Q'(s_{t+1}, \mu'(s_{t+1} | \varphi^{\mu'}) | \varphi^{Q'})$

 Update critic by minimizing loss:

$L(\varphi^Q) = \mathbb{E} [(Q(s_t, a_t | \varphi^Q) - y_t)^2]$

 Update actor using policy gradient:

$\nabla \varphi^\mu = \mathbb{E} \left[\nabla_a Q(s, a | \varphi^Q) \Big|_{s=s_t, a=\mu(s_t)} \nabla_{\varphi^\mu} \mu(s | \varphi^\mu) \Big|_{s_t} \right]$

 /* Phase 6: Soft Update of Target Networks */

$\varphi^{\mu'} \leftarrow (1 - \tau) \varphi^{\mu'} + \tau \varphi^\mu$

$\varphi^{Q'} \leftarrow (1 - \tau) \varphi^{Q'} + \tau \varphi^Q$

4.2.4 Performance Analysis

The performance of the proposed DSTS-DDPG approach is evaluated against various baselines in the hierarchical network architecture. The evaluations are conducted through simulations using the Simulation of Urban MObility (SUMO)¹, which provides realistic vehicular network scenarios. The simulation environment models an urban road network in which each VEC server covers a coverage radius of 500 m, accommodating between 10 and 30 AVs per cluster. The simulation parameters and settings are presented in Table 4.2. The efficacy of DSTS-DDPG is measured against the following baselines:

- *Fixed Threshold Policies:* These policies employ predetermined credibility score thresholds for trustworthiness assessment. Three fixed thresholds are considered: a lower threshold of 0.1, representing minimal trust requirements, a moderate threshold of 0.5, and a higher threshold of 0.8, representing strict trust requirements. Unlike DSTS-DDPG, these thresholds remain static and cannot adapt to changing network conditions.
- *Mean Policy:* This policy computes the average credibility score across all AVs in the selected subset and uses this average as the selection criterion. While this method captures the general trust level of the network, it fails to account for the behavior of individual AVs.
- *Random Policy:* This policy randomly selects actions at each time step without considering credibility scores. It serves as a lower-bound benchmark to demonstrate the importance of intelligent decision-making in trust management.

These baseline comparisons highlight the advantages of an adaptive DRL-based approach over static or indiscriminate policies, particularly in dynamic vehicular environments where trust decisions have significant implications for data exchange reliability.

¹<https://eclipse.dev/sumo/>

Table 4.2: Simulation Parameters for the DSTS-DDPG Algorithm

Parameter	Value
Network Parameters	
Number of AVs (N)	10–30
Number of malicious data providers	3–9
Base rate (a)	0.5
Freshness parameter (λ)	0.9
Familiarity weight (W_1)	0.5
Freshness weight (W_2)	0.5
DDPG Parameters	
Replay memory size ($ \mathcal{D} $)	100,000
Batch size ($\mathcal{B}$)	64
Training optimizer	Adam
Activation function for hidden layers	ReLU
Learning rate of actor	0.001
Learning rate of critic	0.001
Discount factor (ζ)	0.99
Target update factor (τ)	0.01
Reward Parameters	
Credibility score weight (ρ)	0.7
Threshold stability penalty (κ)	0.3

Figure 4.7 presents the average reward comparison across training episodes for DSTS-DDPG and the baseline policies, where each episode consists of approximately 30 time steps. The DSTS-DDPG algorithm begins with a low average reward, as the agent initially explores the action space. The reward increases significantly between episodes 150 and 250 as the agent learns an effective threshold policy, before converging to approximately 35.

Among the baseline policies, the fixed threshold of 0.8 achieves the highest baseline reward at approximately 16, as its strict threshold selects only the most credible data providers, resulting in a high average credibility score with zero threshold instability penalty. The mean policy maintains a flat reward of approximately 9; while it selects data providers with above-average credibility scores, it incurs a slight threshold instability penalty as the average credibility changes at each time step. The fixed threshold of 0.1 achieves a reward of approximately 6; its permissive threshold

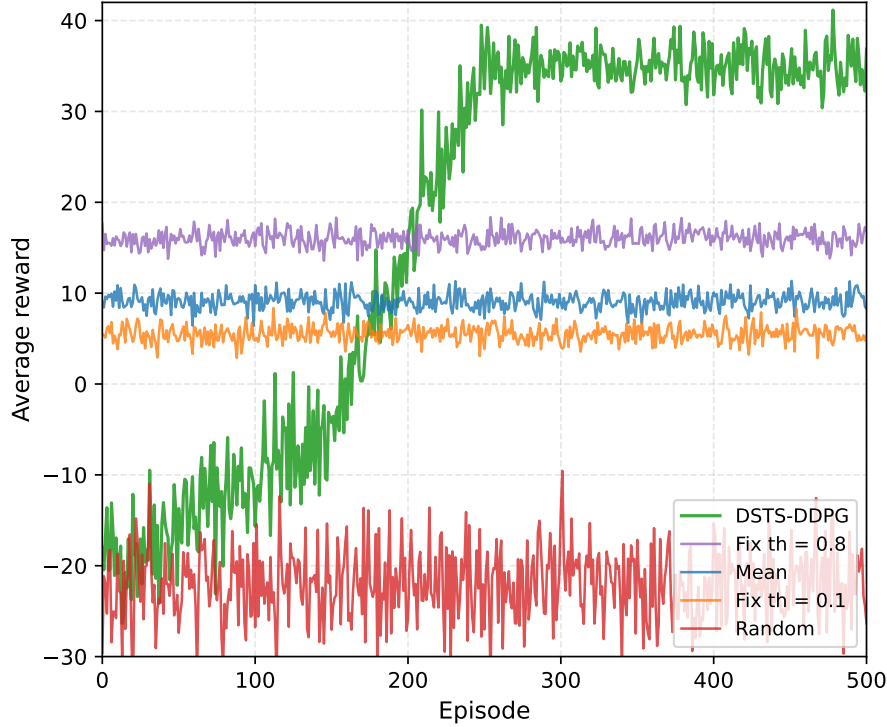


Figure 4.7: Mean reward comparison during the training process (an episode contains of about 30 steps)

admits most data providers with zero instability penalty, but the inclusion of malicious AVs lowers the average credibility of the selected subset. The random policy performs the worst, fluctuating around -22 , due to both poor AV selection and substantial threshold instability at every time step. Overall, DSTS-DDPG outperforms the best-performing baseline (fixed threshold of 0.8) by approximately 119% in terms of converged average reward.

Figure 4.8 evaluates the percentage of trustworthy transactions selected under different policies as the number of transactions increases. A scenario involving 20 AVs is considered, of which 6 are identified as malicious data providers. DSTS-DDPG consistently maintains a trustworthy transaction selection rate above 90%, achieving approximately 93% at 100 transactions and 90% at 300 transactions. The slight decline is attributed to some malicious AVs that initially behave honestly to build credibility before engaging in malicious activity, making them temporarily harder to distinguish. Nevertheless, the adaptive nature of DSTS-DDPG enables it to maintain robust performance despite this challenge. The fixed threshold policy with a threshold of 0.5 achieves approximately 78% at 100 transactions but declines to 70% at 300 transactions, as its static threshold cannot adapt to the evolving

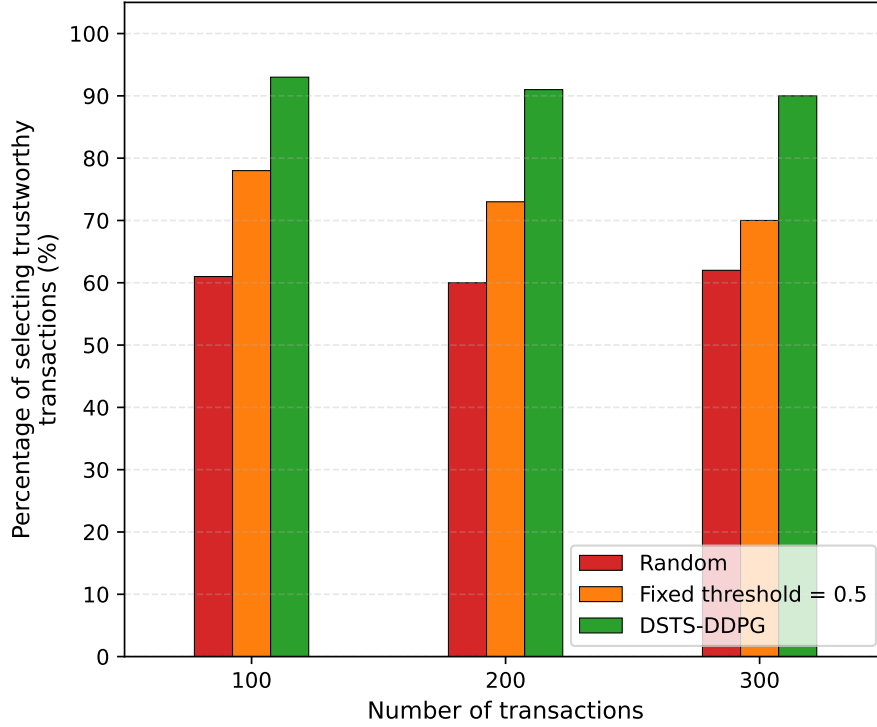


Figure 4.8: Comparison of percentage of selecting trustworthy transactions to number of transactions under different scheme.

behavior of strategic malicious AVs. The random policy remains roughly flat around 60%, as expected for a policy that does not utilize credibility information. At 300 transactions, DSTS-DDPG outperforms the fixed threshold by approximately 29% and the random policy by approximately 45%, confirming that an adaptive DRL-based mechanism is essential for reliable trust management in vehicular networks.

Figure 4.9 evaluates the scalability of the proposed DSTS-DDPG algorithm by examining the percentage of trustworthy transactions selected as the number of vehicles in the network increases from 10 to 30, with 30% of vehicles being malicious (i.e., 3 to 9 malicious data providers). While the preceding figure evaluated performance over increasing numbers of transactions, this figure assesses the impact of increasing the number of vehicles in the network while the number of transactions is fixed at 200. DSTS-DDPG maintains a consistently high selection rate, declining only marginally from 96% at 10 vehicles to 89% at 30 vehicles, remaining above 89% across all network sizes. This demonstrates the scalability of the proposed approach, as the DDPG agent effectively adapts its threshold policy to larger candidate pools containing more malicious data providers. The fixed threshold policy with a threshold of 0.50 exhibits a more pronounced decline from 82% to 65%, as

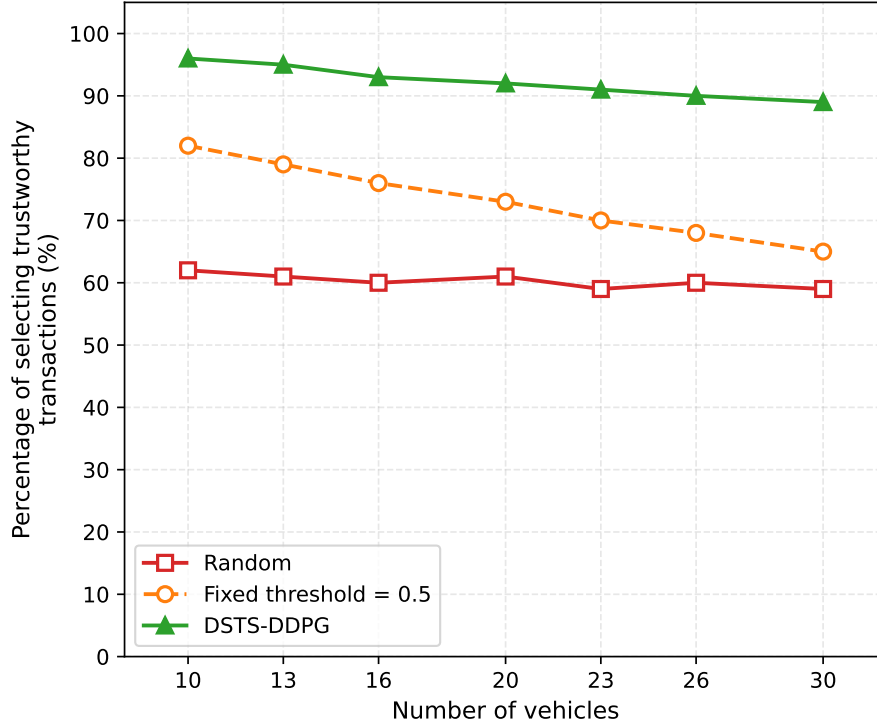


Figure 4.9: Comparison of percentage of selecting trustworthy transactions to number of transactions under different scheme.

the increasing number of malicious AVs raises the likelihood that some will possess credibility scores above the static threshold. The random policy remains roughly flat around 60%, unaffected by network size as it does not utilize credibility information. At 30 vehicles, DSTS-DDPG outperforms the fixed threshold by approximately 37% and the random policy by approximately 46%.

4.3 Conclusion

This chapter addressed the critical challenge of ensuring trustworthiness during data exchange in vehicular networks to address **RQ1b** in this thesis by presenting two complementary RL-based trust management approaches, classified according to the focus-centric taxonomy established in Chapter 3.

The first contribution, QL-TRT, addressed link-centric trust management by proposing a Q-learning-based trustworthy routing technique for VANETs. QL-TRT enables each vehicle to select the reliable forwarding path while avoiding black hole and gray hole attackers. The second contribution, DSTS-DDPG, addressed vehicle-centric trust management by proposing a DRL-based mechanism for dynamically optimizing the credibility score threshold during data exchange in VEC. By com-

binning a reputation management framework with DDPG, the approach enables edge servers to adaptively select trusted data providers, mitigating the threat posed by malicious AVs.

Chapter 5

Blockchain-Enabled Pseudonym Management in VEC-IoV

IoV represents a groundbreaking paradigm for advancing ITS, as it redefines the future of connected ecosystems. By interconnecting vehicles, infrastructure, and users, IoV enables real-time V2V and V2I communications [267], forming a critical backbone for smart urban mobility. At the core of IoV lies V2X communication, which enables seamless data sharing between CAVs and their surroundings, improving road safety, optimizing traffic flow, and enhancing user experiences [33], [34].

To support the vast data sharing and real-time processing demands of IoV, VEC has emerged as a critical enabler. By extending computational resources closer to the network edge, VEC provides low-latency, localized data processing and reduces the burden on the centralized cloud infrastructure. While such data sharing significantly improves traffic efficiency and public safety, it introduces critical privacy risks.

Privacy preservation is a fundamental concern in IoV due to the sensitivity of data sharing. For instance, location-based data and behavioral patterns shared between vehicles and infrastructure can be exploited for tracking, profiling, trajectory reconstruction or other malicious purposes. To mitigate these risks, pseudonym management has emerged as a promising privacy-preserving mechanism [268]. By using temporary identifiers, vehicles can obscure their real identities during communication, making it significantly harder for adversaries to link messages to specific vehicles or individuals [227]. Existing security standards, such as IEEE 1609.2 [82] and ETSI 102941-v2.2.1 [269], have been adopted to protect vehicle identities and prevent tracking [270]. Although frequent pseudonym changes significantly enhance

privacy, they can also result in increased communication overhead and computational demands.

Traditionally, pseudonym management, including generation, distribution, and revocation, has been handled by a centralized TA located within the cloud layer of vehicular networks [271]. However, the centralized approaches introduce considerable overhead, limited scalability, and presents a single point of failure, making it unsuitable for extensive VEC-IoV deployments [20].

To overcome these limitations, pseudonym management responsibilities can be migrated closer to the network edge. Decentralization of pseudonym management reduces dependency on cloud infrastructure, reducing communication delays, enhancing scalability, and improving overall system performance and robustness in VEC-IoV environments. Building on this decentralized foundation, the integration of permissioned blockchain technology into pseudonym management processes is investigated in this contribution. The decentralized and tamper-proof nature of blockchain ledgers [142], [160], [272] enables efficient and transparent pseudonym authentication and revocation protocols. In addition, employing DRL enables dynamic decision-making that adapts to changing conditions and supports sophisticated trade-off analysis, efficiently balancing competing objectives to enhance system performance [273], [274]. Hence, to balance privacy preservation through pseudonym management and cost efficiency, MADRL is employed, which enables interconnected agents to collaboratively learn adaptive policies by considering both environmental dynamics and the actions of other agents [267].

To address the second research question, *"What design model would effectively integrate blockchain technology with RL strategies to improve privacy in vehicular networks (RQ2a)?"*, this chapter proposes a Blockchain-enabled Pseudonym Management framework for VEC (BPMV), in which a permissioned blockchain is integrated into the VEC environment to enhance privacy preservation during data sharing in VEC-IoV. Two categories of pseudonym identities are introduced, systemic pseudonym identities (SPIDs) and communication pseudonym identities (CPIDs), with decentralized protocols for authentication and revocation. To enable dynamic and cost-aware CPID update scheduling, a social welfare function is formulated that jointly incorporates vehicle-side cost, VEC-side cost, and privacy entropy. The resulting social welfare maximization problem is modeled as a MADRL problem, allowing decentralized vehicles to update their pseudonyms in a manner that enhances privacy while remaining cost-effective.

5.1 System Model

This section presents the architecture of the BPMV framework and its components, followed by the security model.

5.1.1 Architecture of BPMV Framework

In BPMV architecture, the system is structured into three main layers: the perception layer, the blockchain-enabled edge layer, and the cloud layer. The components of the proposed blockchain-based hierarchical network model are illustrated in Figure 5.1.

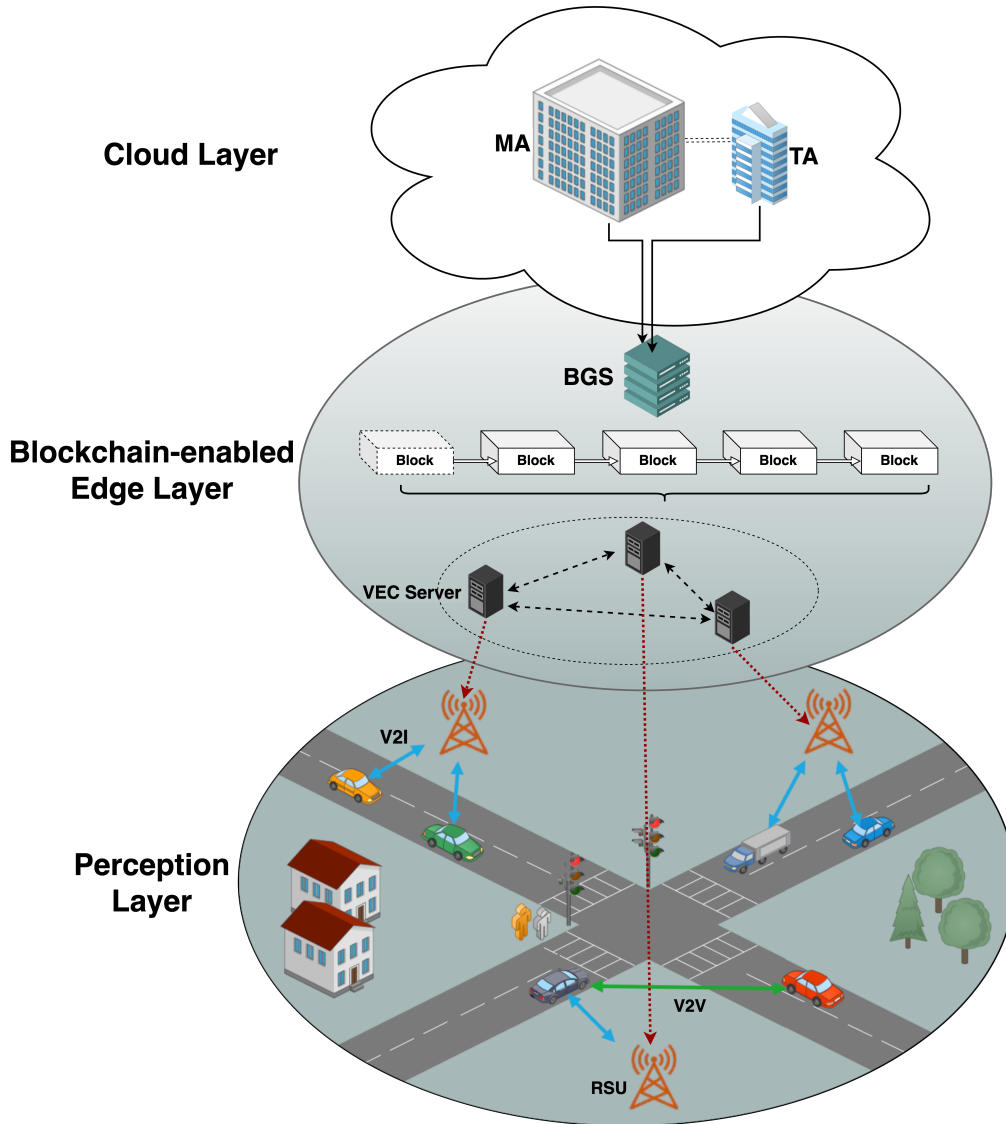


Figure 5.1: Architecture of BPMV

The **Perception Layer** consists of RSUs and CAVs which are called vehicles for simplicity, both equipped with sensors and communication modules. Each vehicle is equipped with OBUs that can generate, process, and transmit data to nearby vehicles or RSUs through V2V and V2I communication. Each RSU connects to a VEC server, providing edge computing services within its designated coverage area, strategically located close to vehicles. V_i^j represents the i th vehicle in the j th VEC server, where the set of vehicles in the coverage area of the VEC server j is denoted as $\mathcal{V}^j = \{V_1^j, V_2^j, \dots, V_i^j\}$, and $|\mathcal{V}^j|$ represents the number of vehicles within this coverage area.

Similarly, let S_j represent the j th VEC server, where the set of all VEC servers is given by $\mathcal{J} = \{1, 2, \dots, J\}$.

Each RSU is paired with a dedicated VEC server and the VEC servers are located at **Blockchain-enabled edge layer**. The TA and Misbehavior Authority (MA) are located in the **cloud layer**.

The main components of the proposed network model are described next.

- **Trusted Authority** is a fully trusted, tamper-resistant entity with robust computing and storage capabilities, which is managed by the transportation administration. TA initializes the system parameters, generates public and private keys, and manages the registration of vehicles and VEC servers.
- **Misbehavior Authority** collects and analyzes abnormal reports generated by VEC servers from the blockchain via a gateway server (BGS). When a malicious behavior is identified, the analyzed results are sent to the TA and requests the TA to add the detected vehicle pseudonym to a revocation list to prevent further misuse.
- **VEC Servers** are honest-but-curious. VEC server plays a critical role in VEC-IoV by managing pseudonym change and misbehavior reporting. These servers operate as blockchain nodes in distributed ledger. Moreover, they handle pseudonym management and participate in validating pseudonyms and recording misbehavior report transactions on the blockchain. Furthermore, each VEC server hosts an RL agent that autonomously determines pseudonym update frequencies based on the current state of the environment. The RL agent ensures that decisions dynamically adapt to real-time traffic density and cost constraints.

- **CAVs** in the IoV are assumed to be malicious and can engage in various attacks. A malicious CAV may attempt to eavesdrop on communications between the RSU and other vehicles to extract real data and associate it with a specific vehicle. Additionally, malicious CAVs may transmit false data to RSUs for their own benefit [237]. The real identities of these malicious CAVs should be accurately identified, monitored, and revoked through the collaborative efforts of VEC server reports, the MA, and the TA [238].
- **Permissioned Blockchain** Unlike fully public distributed ledgers, permissioned blockchains operate as semi-open systems, allowing only designated members to join the network. Each member is granted customized access rights based on their role and permissions [275]. In our framework only VEC servers, the TA, and the MA have access to participate in transactions in the IoV. In this system, the VEC servers act as blockchain nodes, facilitating block validation, smart contract execution, and ledger maintenance. Each block in the ledger comprises a block header and a block body, where the block header stores the hash of the previous block and a timestamp, ensuring data integrity and immutability. In addition, TA and MA can write authenticated records, such as the revocation list onto the blockchain through the BGS, which serves as a secure interface for trusted entities to interact with the ledger. They do not participate in the verification and consensus process due to the functional difference [276].
- **Lightweight Consensus Mechanism** is employed to ensure efficient, secure, and scalable agreement on pseudonym updates. Trusted VEC servers act as validators, validate transactions, and update the ledger. This decentralized approach enhances privacy, reduces reliance on a central authority, and ensures that only legitimate and verified transactions are recorded.

5.1.2 Security Model

RSUs, as semi-trusted entities, pose a potential risk to privacy, particularly in systems involving pseudonym management and data sharing. To mitigate these risks and ensure robust security and privacy within the IoV, a blockchain-enabled framework is introduced. This framework incorporates anonymous authentication and secure data sharing mechanisms. By using the decentralized and tamper-proof properties of blockchain technology, the system addresses the challenges posed by semi-trusted RSUs, with the following security and privacy requirements:

1. **Anonymity** ensures that the real identities of vehicles remain concealed during the authentication and data-sharing processes, allowing only pseudonym information to be accessible. This prevents malicious entities, including RSUs, from linking activities or communications to specific vehicles.
2. **Unlinkability** ensures that attackers, even if working together, cannot link vehicle pseudonyms to their real identities during updates or data sharing. This protects vehicle location privacy and prevents anyone, including RSUs or malicious actors, from tracking their movements.
3. **Data Integrity and Immutability** guarantees that all transactions, including pseudonym updates and data-sharing records, are securely stored on the blockchain. This ensures data cannot be tampered with or altered by an external attacker.
4. **Traceability** refers to the fact that data shared by vehicles should be associated with their pseudonyms to ensure accountability and privacy. In cases of malicious behavior, the TA, in collaboration with the MA, should be able to trace the real identity of the vehicle based on the harmful data it transmitted.
5. **Efficiency** ensures that privacy protection is sustainable and cost-effective, enabling seamless operation without imposing excessive computational and communicational overhead.

5.2 Pseudonym Management and Revocation Framework

The pseudonym authentication, management, and revocation explained in the proposed BPMV framework which is organized into distinct phases. Figure 5.2 presents the flow diagram of the framework, outlining the specific steps that are detailed next.

5.2.1 System Initialization

This chapter employs Elliptic Curve Cryptography (ECC), a cryptography technique based on elliptic curve mathematics [277]. ECC achieves a security level equivalent to that of Rivest–Shamir–Adleman (RSA) while requiring substantially smaller key sizes, which reduces computational demands and enhances overall system performance [278].

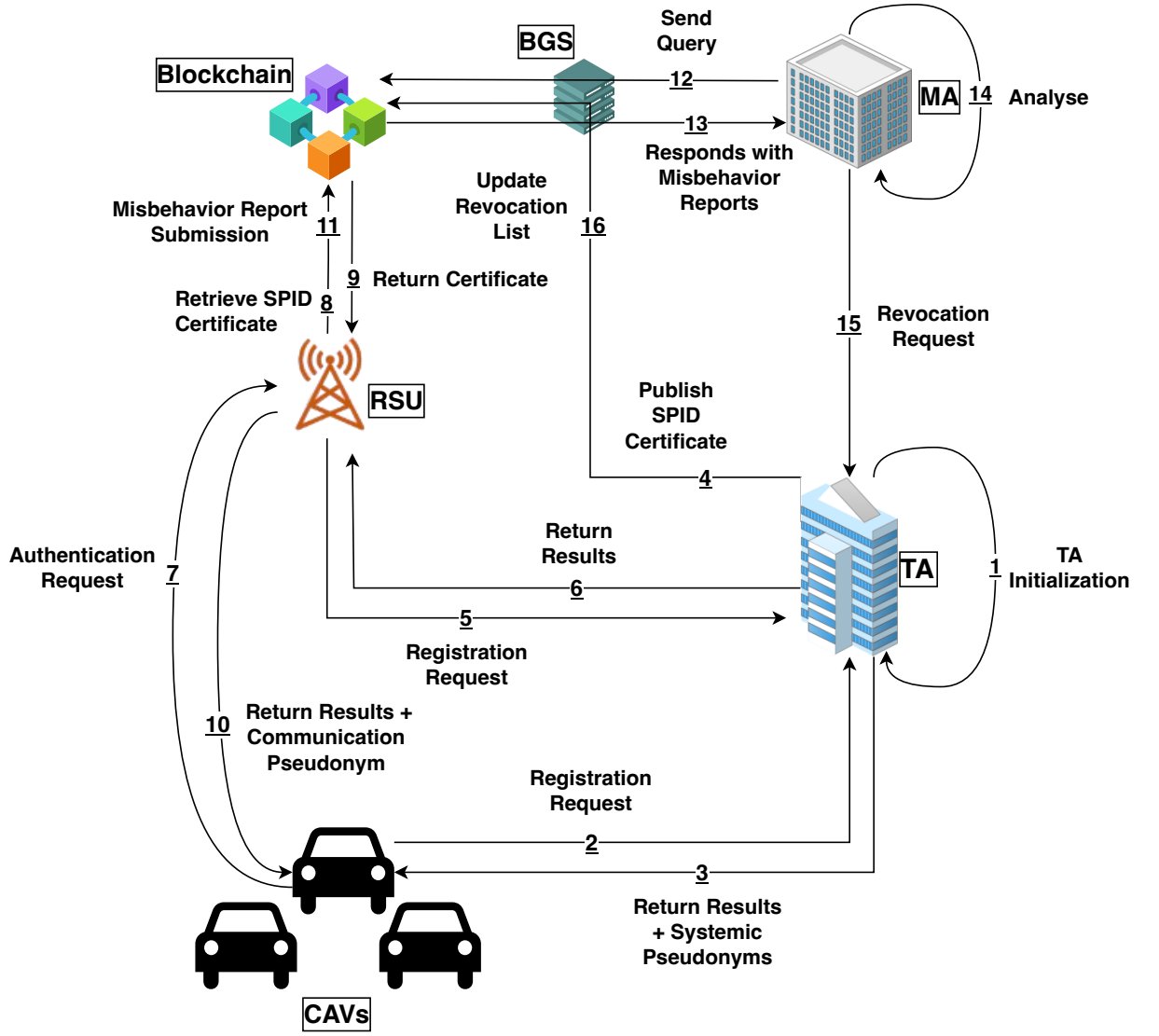


Figure 5.2: Flow Diagram

The TA is responsible for initializing the system by selecting the elliptic curve domain parameters:

$$\mathcal{D} = (p, a, b, G, n, h), \quad (5.1)$$

where p is a large prime defining the finite field $\mathbb{F}_p$, (a, b) are the coefficients of the elliptic curve $E : y^2 \equiv x^3 + ax + b \pmod{p}$, G is the base point on the curve of prime order n , and h is the cofactor such that $n \cdot h$ determines the number of points on the curve [278]. In addition, the TA selects a cryptographic hash function

$H : \{0, 1\}^* \rightarrow \mathbb{Z}_n^*$ for secure identity verification [233]. The TA generates its private-public key pair as follows:

$$sk_{TA} \xleftarrow{\$} [1, n-1], \quad PK_{TA} = sk_{TA} \cdot G. \quad (5.2)$$

The public system parameters are published as:

$$\mathcal{P}_{\text{pub}} = \{\mathcal{D}, H, PK_{TA}\}, \quad (5.3)$$

while sk_{TA} remains confidential to the TA.

5.2.2 Vehicle and VEC Server Registration

Vehicle Registration

Each vehicle V_i must register with the TA to obtain authentication credentials and a systemic pseudonym (SPID) for secure communication [232]. The vehicle generates its key pair based on the Elliptic Curve Digital Signature Algorithm (ECDSA) by selecting a uniformly random private key $sk_{V_i} \xleftarrow{\$} [1, n-1]$ and computing the corresponding public key $PK_{V_i} = sk_{V_i} \cdot G$. The vehicle sends its real identity RID_{V_i} and public key PK_{V_i} to the TA via a secure channel. If RID_{V_i} is not previously registered, the TA generates the systemic pseudonym:

$$SPID_i = H(RID_{V_i} \parallel \eta_i), \quad (5.4)$$

where $\eta_i \xleftarrow{\$} \{0, 1\}^{128}$ is a cryptographically secure random nonce. The TA creates a certificate binding $SPID_i$ to PK_{V_i} by computing an ECDSA signature:

$$\sigma_{TA,i} = \text{Sign}_{sk_{TA}}(H(SPID_i \parallel PK_{V_i} \parallel T_i^{\text{start}} \parallel T_i^{\text{exp}})) \quad (5.5)$$

where the signature is generated following the standard ECDSA signing algorithm over the elliptic curve E , T_i^{start} is the issuance time, and T_i^{exp} is the expiration time. The TA securely transmits $\{SPID_i, \sigma_{TA,i}\}$ to the vehicle via the encrypted channel, and stores the private mapping $SPID_i \rightarrow RID_{V_i}$ in its encrypted database for potential revocation. The TA publishes the certificate $\{SPID_i, PK_{V_i}, \sigma_{TA,i}, T_i^{\text{start}}, T_i^{\text{exp}}, \text{status: valid}\}$ to the blockchain.

VEC Server Registration

Each VEC server S_j registers with the TA to participate in the authentication and misbehavior tracking. The VEC server selects a random private key $sk_{S_j} \in$

$[1, n - 1]$ and computes its public key $PK_{S_j} = sk_{S_j} \cdot G$. It submits its public key PK_{S_j} and organizational identifier ID_{S_j} to the TA via a secure authenticated channel. Upon verification of the VEC legitimacy, the TA issues a certificate by computing:

$$\sigma_{TA,S_j} = \text{Sign}_{sk_{TA}}(H(ID_{S_j} \parallel PK_{S_j} \parallel T_{S_j}^{\text{start}} \parallel T_{S_j}^{\text{exp}})) \quad (5.6)$$

where $T_{S_j}^{\text{start}}$ is the issuance time and $T_{S_j}^{\text{exp}}$ is the expiration time. The TA transmits $\{ID_{S_j}, \sigma_{TA,S_j}, T_{S_j}^{\text{start}}, T_{S_j}^{\text{exp}}\}$ to the VEC server.

The system initialization, vehicle registration, and VEC server registration procedures are summarized in Protocol 5.I.

Protocol 5.I: System Initialization and Entity Registration

Phase I: System Initialization

/ TA initializes cryptographic parameters */*

TA selects domain parameters $\mathcal{D} = (p, a, b, G, n, h)$ and hash function

$$H : \{0, 1\}^* \rightarrow \mathbb{Z}_n^*$$

$$sk_{TA} \xleftarrow{\$} [1, n-1], \quad PK_{TA} \leftarrow sk_{TA} \cdot G$$

Publish $\mathcal{P}_{\text{pub}} \leftarrow \{\mathcal{D}, H, PK_{TA}\}$

Store sk_{TA} securely

Phase II: Vehicle Registration

Input: Vehicle V_i registers with the TA

/ Vehicle side */*

$$sk_{V_i} \xleftarrow{\$} [1, n-1], \quad PK_{V_i} \leftarrow sk_{V_i} \cdot G$$

Send $\{RID_{V_i}, PK_{V_i}\}$ to TA via secure channel

/ TA side */*

if RID_{V_i} *is not previously registered* **then**

$$\eta_i \xleftarrow{\$} \{0, 1\}^{128}$$

$$SPID_i \leftarrow H(RID_{V_i} \parallel \eta_i)$$

$$\sigma_{TA,i} \leftarrow \text{Sign}_{sk_{TA}}(H(SPID_i \parallel PK_{V_i} \parallel T_i^{\text{start}} \parallel T_i^{\text{exp}}))$$

Publish $\{SPID_i, PK_{V_i}, \sigma_{TA,i}, T_i^{\text{start}}, T_i^{\text{exp}}, \text{valid}\}$ to blockchain

Store $SPID_i \rightarrow RID_{V_i}$ in encrypted database

Send $\{SPID_i, \sigma_{TA,i}, T_i^{\text{start}}, T_i^{\text{exp}}\}$ to V_i

Phase III: VEC Server Registration

Input: VEC server S_j registers with the TA

/ VEC side */*

$$sk_{S_j} \xleftarrow{\$} [1, n-1], \quad PK_{S_j} \leftarrow sk_{S_j} \cdot G$$

Send $\{ID_{S_j}, PK_{S_j}\}$ to TA via secure channel

/ TA side */*

Verify S_j credentials

$$\sigma_{TA,S_j} \leftarrow \text{Sign}_{sk_{TA}}(H(ID_{S_j} \parallel PK_{S_j} \parallel T_{S_j}^{\text{start}} \parallel T_{S_j}^{\text{exp}}))$$

Send $\{ID_{S_j}, \sigma_{TA,S_j}, T_{S_j}^{\text{start}}, T_{S_j}^{\text{exp}}\}$ to S_j

5.2.3 Vehicle Authentication

When a vehicle enters the coverage area of a VEC server, it initiates authentication to obtain a CPID. To prevent cross-CPID linkability through public key correlation, the vehicle generates a fresh key pair for each new CPID session.

Authentication Request

Upon entering the coverage area of VEC server S_j , vehicle V_i holding a valid $SPID_i$ generates a fresh session-specific key pair:

$$sk_{i,k} \xleftarrow{\$} [1, n-1], \quad PK_{i,k} = sk_{i,k} \cdot G, \quad (5.7)$$

where k denotes the CPID index within the session. The vehicle constructs an authentication request:

$$\text{req} = (SPID_i, PK_{i,k}, T_q, \sigma_{auth,i}) \quad (5.8)$$

where T_q is the current timestamp and

$$\sigma_{auth,i} = \text{Sign}_{sk_{V_i}}(H(SPID_i \parallel PK_{i,k} \parallel ID_{S_j} \parallel T_q)) \quad (5.9)$$

is the ECDSA signature generated by the vehicle's SPID-bound private key sk_{V_i} , which proves ownership of $SPID_i$ and binds the fresh public key $PK_{i,k}$ to the request. The inclusion of ID_{S_j} prevents relay attacks across VEC servers.

VEC Server Verification

Upon receiving the request, S_j retrieves the vehicle's certificate $\{SPID_i, PK_{V_i}, \sigma_{TA,i}, T_i^{\text{start}}, T_i^{\text{exp}}, \text{status: valid}\}$ from the blockchain. The VEC server performs the following verification steps:

1. Verifies the TA signature $\sigma_{TA,i}$ using PK_{TA} to confirm the binding between $SPID_i$ and PK_{V_i} .
2. Checks that the certificate status is **valid** and $T_q \in [T_i^{\text{start}}, T_i^{\text{exp}}]$.
3. Validates $\sigma_{auth,i}$ using PK_{V_i} to authenticate the freshness and authenticity of the request.
4. Validates that $PK_{i,k}$ is a valid point on E .

Protocol 5.II: Vehicle Authentication and CPID Issuance

Input: Vehicle V_i enters coverage area of VEC server S_j

/ Vehicle side* **/*

$sk_{i,k} \xleftarrow{\$} [1, n-1], \quad PK_{i,k} \leftarrow sk_{i,k} \cdot G$

$\sigma_{auth,i} \leftarrow \text{Sign}_{sk_{V_i}}(H(SPID_i \parallel PK_{i,k} \parallel ID_{S_j} \parallel T_q))$

Send req $\leftarrow (SPID_i, PK_{i,k}, T_q, \sigma_{auth,i})$ to S_j

/ VEC side | Verification* **/*

Retrieve $\{SPID_i, PK_{V_i}, \sigma_{TA,i}, T_i^{\text{start}}, T_i^{\text{exp}}, \text{valid}\}$ from blockchain

Verify $\sigma_{TA,i}$ using PK_{TA} to confirm binding of $SPID_i$ and PK_{V_i}

Check status is **valid** and $T_q \in [T_i^{\text{start}}, T_i^{\text{exp}}]$

Verify $\sigma_{auth,i}$ using PK_{V_i}

Validate $PK_{i,k}$ is a valid point on E

if *any verification fails* **then**

 | Reject request and **abort**

else

 | Proceed to CPID generation

/ VEC side | CPID Generation* **/*

$\eta_{i,k} \xleftarrow{\$} \{0, 1\}^{128}$

$CPID_{i,k} \leftarrow H(SPID_i \parallel ID_{S_j} \parallel \eta_{i,k} \parallel T_{i,k}^{\text{CPID}})$

$\sigma_{S_j,i,k} \leftarrow \text{Sign}_{sk_{S_j}}(H(CPID_{i,k} \parallel PK_{i,k} \parallel T_{i,k}^{\text{start}} \parallel T_{i,k}^{\text{exp}}))$

Store $\{CPID_{i,k}, PK_{i,k}, SPID_i, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}}, \text{active}\}$ in local database

Send $\{CPID_{i,k}, \sigma_{S_j,i,k}, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}}\}$ to V_i

CPID Generation

Upon successful verification, S_j generates the k -th communication pseudonym for vehicle V_i :

$$CPID_{i,k} = H(SPID_i \parallel ID_{S_j} \parallel \eta_{i,k} \parallel T_{i,k}^{\text{CPID}}), \quad (5.10)$$

where $\eta_{i,k} \xleftarrow{\$} \{0, 1\}^{128}$ is a cryptographically secure random nonce and $T_{i,k}^{\text{CPID}}$ is the timestamp associated with the validity period of the pseudonym, subject to $T_i^{\text{start}} \leq T_{i,k}^{\text{CPID}} \leq T_i^{\text{exp}}$, ensuring that CPIDs cannot extend beyond the SPID validity window.

The VEC server issues a certificate binding the CPID to the vehicle's fresh public key:

$$\sigma_{S_j,i,k} = \text{Sign}_{sk_{S_j}}(H(CPID_{i,k} \parallel PK_{i,k} \parallel T_{i,k}^{\text{start}} \parallel T_{i,k}^{\text{exp}})). \quad (5.11)$$

The VEC server stores the session record $\{CPID_{i,k}, PK_{i,k}, SPID_i, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}}, \text{active}\}$ in its local database and securely trans-

mits $\{CPID_{i,k}, \sigma_{S_j,i,k}, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}}\}$ to the vehicle. Since each CPID uses a fresh key pair, cross-CPID linkability through public key correlation is prevented.

After the authentication and initial pseudonym issuance, a vehicle that remains within the coverage area of a VEC server must dynamically update its pseudonym over time. In this context, each VEC server acts as an RL agent, making intelligent decisions about when vehicles within its coverage should perform pseudonym updates. The detailed mechanism of this decision-making process is presented in Section 5.4. If a pseudonym update is triggered, the vehicle generates a new key pair $(sk_{i,k+1}, PK_{i,k+1})$ and submits a rotation request signed with its current CPID key $sk_{i,k}$. The VEC server verifies the request, generates a new $CPID_{i,k+1}$ with a fresh nonce, marks $CPID_{i,k}$ as **expired**, and issues the new credentials.

The vehicle authentication and CPID issuance procedure is summarized in Protocol 5.II, and the CPID rotation procedure is presented in Protocol 5.III.

Protocol 5.III: CPID Rotation

Input: RL agent triggers pseudonym update for V_i under S_j

/ Vehicle side */*

$sk_{i,k+1} \xleftarrow{\$} [1, n-1], \quad PK_{i,k+1} \leftarrow sk_{i,k+1} \cdot G$

$\sigma_{\text{rotate}} \leftarrow \text{Sign}_{sk_{i,k}}(H(SPID_i \parallel CPID_{i,k} \parallel PK_{i,k+1} \parallel T_{\text{curr}}))$

Send $\text{Req}_{\text{rotate}} \leftarrow \{SPID_i, CPID_{i,k}, PK_{i,k+1}, T_{\text{curr}}, \sigma_{\text{rotate}}\}$ to S_j

/ VEC side | Verification */*

Retrieve $PK_{i,k}$ from local database for $CPID_{i,k}$

Verify σ_{rotate} using $PK_{i,k}$

if verification fails then

 | Reject request and **abort**

else

 | Proceed to CPID rotation

/ VEC side | New CPID Generation */*

$\eta_{i,k+1} \xleftarrow{\$} \{0, 1\}^{128}$

$CPID_{i,k+1} \leftarrow H(SPID_i \parallel ID_{S_j} \parallel \eta_{i,k+1} \parallel T_{i,k+1}^{\text{CPID}})$

$\sigma_{S_j,i,k+1} \leftarrow \text{Sign}_{sk_{S_j}}(H(CPID_{i,k+1} \parallel PK_{i,k+1} \parallel T_{i,k+1}^{\text{start}} \parallel T_{i,k+1}^{\text{exp}}))$

Mark $CPID_{i,k}$ as **expired** in local database

Store $\{CPID_{i,k+1}, PK_{i,k+1}, SPID_i, T_{i,k+1}^{\text{start}}, T_{i,k+1}^{\text{exp}}, \text{active}\}$

Send $\{CPID_{i,k+1}, \sigma_{S_j,i,k+1}, T_{i,k+1}^{\text{start}}, T_{i,k+1}^{\text{exp}}\}$ to V_i

5.2.4 Message Signing and Batch Verification

In high-density IoV environments, a VEC server may receive a large number of messages within a short time interval. To efficiently handle this scenario, a Schnorr-based message signing scheme is adopted for communication messages, enabling batch verification of multiple messages simultaneously.

Message Signing

When vehicle V_i transmits a message m_i under its active $CPID_{i,k}$, it generates a signature using its per-CPID private key $sk_{i,k}$ as follows. The vehicle selects a random nonce $\mu_i \xleftarrow{\$} \mathbb{Z}_n^*$ and computes the commitment:

$$R_i = \mu_i \cdot G. \quad (5.12)$$

The vehicle then computes the challenge hash:

$$e_i = H(m_i \parallel T_{\delta_i} \parallel CPID_{i,k} \parallel R_i), \quad (5.13)$$

and generates the signature scalar:

$$s_i = \mu_i + e_i \cdot sk_{i,k} \bmod n, \quad (5.14)$$

where T_{δ_i} is the timestamp associated with message m_i , ensuring freshness and preventing replay attacks. The vehicle broadcasts the message tuple $\mathcal{M}_i = (CPID_{i,k}, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}}, R_i, m_i, s_i, T_{\delta_i})$.

Individual Verification

Any verifier can authenticate a single message tuple by retrieving the VEC-issued certificate for $CPID_{i,k}$ to obtain $PK_{i,k}$, and checking:

$$s_i \cdot G \stackrel{?}{=} R_i + e_i \cdot PK_{i,k}. \quad (5.15)$$

Batch Verification

When a VEC server receives N message tuples $\{\mathcal{M}_i\}_{i=1}^N$ from vehicles within a short time interval, it performs batch verification to authenticate all messages simultaneously. The VEC server first performs preliminary checks by confirming that each $CPID_{i,k}$ remains valid based on its certificate timestamps.

For each message tuple, the VEC server retrieves $PK_{i,k}$ from its local database and computes e_i from the message contents. To prevent rogue-key attacks, the

verifier selects small random coefficients $\rho_i \xleftarrow{\$} [1, 2^\ell]$ for each message, where ℓ is a security parameter (e.g., $\ell = 128$). The batch verification then checks:

$$\left(\sum_{i=1}^N \rho_i \cdot s_i \right) \cdot G \stackrel{?}{=} \sum_{i=1}^N \rho_i \cdot R_i + \sum_{i=1}^N (\rho_i \cdot e_i) \cdot PK_{i,k}. \quad (5.16)$$

If the equation holds, all N signatures are accepted simultaneously. Otherwise, the batch fails, and the verifier employs a binary search method to identify and discard invalid signatures.

The detailed steps of the message signing and batch verification are presented in Protocol 5.IV.

Protocol 5.IV: Message Signing and Batch Verification

Phase I: Message Signing

Input: Vehicle V_i transmits message m_i under active $CPID_{i,k}$

$\mu_i \xleftarrow{\$} \mathbb{Z}_n^*$

$R_i \leftarrow \mu_i \cdot G$

$e_i \leftarrow H(m_i \parallel T_{\delta_i} \parallel CPID_{i,k} \parallel R_i)$

$s_i \leftarrow \mu_i + e_i \cdot sk_{i,k} \bmod n$

Broadcast $\mathcal{M}_i \leftarrow (CPID_{i,k}, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}}, R_i, m_i, s_i, T_{\delta_i})$

Phase II: Batch Verification

Input: VEC server S_j receives N message tuples $\{\mathcal{M}_i\}_{i=1}^N$

/ Preliminary checks*

**/*

for *each* $\mathcal{M}_i, i = 1, \dots, N$ **do**

 Confirm $CPID_{i,k}$ is valid based on $T_{i,k}^{\text{start}}$ and $T_{i,k}^{\text{exp}}$

 Retrieve $PK_{i,k}$ from local database

$e_i \leftarrow H(m_i \parallel T_{\delta_i} \parallel CPID_{i,k} \parallel R_i)$

 Discard $\mathcal{M}_i$ if any check fails

/ Batch verification with rogue-key protection*

**/*

for *each valid* $\mathcal{M}_i$ **do**

$\rho_i \xleftarrow{\$} [1, 2^\ell]$

if $\left(\sum_{i=1}^N \rho_i \cdot s_i \right) \cdot G \stackrel{?}{=} \sum_{i=1}^N \rho_i \cdot R_i + \sum_{i=1}^N (\rho_i \cdot e_i) \cdot PK_{i,k}$ **then**

 Accept all N signatures

else

 Apply binary search to identify and discard invalid signatures

5.2.5 Revocation Report Generation

To maintain network integrity and prevent malicious activities, a decentralized revocation mechanism is implemented, involving VEC servers, MA, TA, and BGS. The BGS allows the TA and MA to interact with the blockchain [276]. The VEC servers act as blockchain nodes, responsible for ledger maintenance and authentication verification, ensuring a secure, decentralized authentication, and misbehavior logging system within the IoV ecosystem.

Each VEC server continuously monitors vehicle behavior within its coverage area. During authentication, the VEC server maintains a local session table that maps each active CPID to the corresponding SPID and session credentials, i.e., $\mathcal{T}_{S_j} : CPID_{i,k} \mapsto (SPID_i, PK_{i,k}, T_{i,k}^{\text{start}}, T_{i,k}^{\text{exp}})$. If misbehavior is detected based on a vehicle's $CPID_{i,k}$, the VEC server retrieves the corresponding $SPID_i$ directly from its session table $\mathcal{T}_{S_j}$.

The VEC server then generates a signed misbehavior report containing the $SPID_i$ of the suspected vehicle, collected evidence, and a timestamp. The VEC server submits the report to the blockchain and writes a new block, ensuring an immutable record of misbehavior incidents.

The MA periodically retrieves misbehavior reports from the blockchain, aggregating reports from multiple VEC servers. Since all reports contain $SPID_i$ rather than $CPID_{i,k}$, the MA can aggregate multiple reports about the same vehicle even if the vehicle used different communication pseudonyms across different VEC servers. If the MA detects a pattern of serious misbehavior that requires revocation, it generates a consolidated report and submits it to the TA for final verification. The TA, which maintains the confidential mapping between $SPID_i$ and real identities, verifies the aggregated evidence. Upon confirmation of serious or repeated misbehavior, the TA revokes the vehicle's active $SPID_i$ and its authorization to request future pseudonyms.

The TA updates the revocation list containing the affected pseudonym, an expiration timestamp, and a cryptographic signature for verification. The updated revocation list is published on the blockchain via the BGS, ensuring that all VEC servers have access to the updated revocation list in a secure and tamper-resistant manner.

The revocation report generation process is outlined in Protocol 5.V.

Protocol 5.V: Revocation Report Generation

Phase I: Misbehavior Detection and Reporting**Input:** VEC server S_j detects misbehavior from $CPID_{i,k}$ Retrieve $SPID_i$ from session table $\mathcal{T}_{S_j}[CPID_{i,k}]$ Generate signed misbehavior report containing $SPID_i$, collected evidence,
and timestamp $T_{\mathcal{R}}$ Submit $\mathcal{R}_{S_j}$ to blockchain**Phase II: Misbehavior Aggregation****Input:** MA periodically queries blockchain via BGS

Retrieve misbehavior reports from blockchain

Aggregate reports by $SPID_i$ across multiple VEC servers**if** *pattern of serious misbehavior detected for $SPID_i$* **then**

└ Generate consolidated report and submit to TA

Phase III: Revocation**Input:** TA receives consolidated report for $SPID_i$ Retrieve RID_{V_i} from encrypted mapping $SPID_i \rightarrow RID_{V_i}$

Verify aggregated evidence

if *misbehavior confirmed* **then**└ Revoke $SPID_i$ and authorization for future pseudonyms└ $\sigma_{\text{rev}} \leftarrow \text{Sign}_{sk_{TA}}(H(SPID_i \parallel T_{\text{rev}} \parallel T_{\text{rev}}^{\text{exp}}))$ └ Publish revocation entry $\{SPID_i, T_{\text{rev}}, T_{\text{rev}}^{\text{exp}}, \sigma_{\text{rev}}\}$ to blockchain via BGS

5.3 Problem Formulation

The previous section presented the details of the proposed protocol for pseudonym management by introducing two distinct pseudonym identities, SPID and CPID. The authentication and revocation procedures based on SPID are then described. In addition, when a vehicle remains within the coverage area of a VEC server and receives its CPID, this identifier must be periodically updated. Therefore, this section formulates the problem of dynamic CPID updating to enhance privacy preservation while considering cost efficiency.

5.3.1 Privacy Entropy in VEC-IoV Networks

Privacy entropy is a well-established quantitative metric used to assess an attacker's uncertainty when attempting to track a vehicle in a VEC-IoV network [214], [229]. It measures the effectiveness of the pseudonym management strategy in masking CAV's real identities. For an individual vehicle i within the coverage area of VEC server S_j , the time-dependent privacy entropy $H_{i,j}(t)$ at time t is defined as:

$$H_{i,j}(t) = -\log_2(p_{i,j}(t)) \cdot e^{-\lambda \Delta t}, \quad (5.17)$$

where t is the current time in seconds, $t_{i,\text{prev}}$ is the time of the last pseudonym change for vehicle i , and $\Delta t = (t - t_{i,\text{prev}})/60$ denotes the elapsed time in minutes since the last pseudonym change for vehicle i . $p_{i,j}(t) \in [0, 1]$ is the probability that an attacker successfully tracks vehicle i when it is in the coverage area of S_j , and $\lambda > 0$ denotes the privacy decay rate per minute, controlling how quickly privacy entropy diminishes over time. The probability $p_{i,j}(t)$, as proposed in [227], considers the vehicle density within the pseudonym swap area to estimate the likelihood of an attacker successfully tracking a vehicle. The probability is defined as:

$$p_{i,j}(t) = \frac{\xi}{\xi + |\mathcal{V}^j| \cdot (1 - \xi)}, \quad (5.18)$$

where $\xi \in [0.3, 1]$ represents the attack capacity of the malicious attacker and $|\mathcal{V}^j|$ denotes the number of vehicles in the coverage area of VEC server j that can perform pseudonym swapping.

5.3.2 Utility Functions for Cost-Aware Pseudonym Management

While privacy entropy quantifies an attacker's uncertainty in tracking a vehicle, frequent pseudonym updates introduce additional costs in terms of communication and computation overhead. An effective pseudonym management strategy must balance privacy preservation and cost efficiency in both vehicle-level and VEC server costs. Vehicles operate under the supervision of VEC servers, which act as RL agents responsible for optimizing privacy while considering system-wide efficiency. Although vehicles do not make independent decisions, their privacy gains and incurred costs influence the decision-making process of VEC servers. Thus, each vehicle i at time t derives a utility from the enhanced privacy, but incurs costs associated with pseudonym updates. The vehicle-level utility function is defined as follows:

$$U_i(t) = \omega_1 H_{i,j}(t) - \omega_2 C_i(t), \quad (5.19)$$

where ω_1 and ω_2 are weighting parameters for privacy entropy and cost, respectively, $H_{i,j}(t)$ (defined in Eq. 5.17) represents the privacy entropy of vehicle i within the coverage area of VEC server S_j , and $C_i(t)$ denotes the total cost incurred for a pseudonym update. The cost $C_i(t)$ is incurred only when a pseudonym update is performed; otherwise $C_i(t) = 0$. The total vehicle-side cost is defined as:

$$C_i(t) = C_{i,\text{com}}(t) + C_{i,\text{auth}}(t), \quad (5.20)$$

where $C_{i,\text{com}}(t)$ denotes the communication cost accounting for the uplink transmission delay from the vehicle to the RSU, and $C_{i,\text{auth}}(t)$ represents the authentication cost based on the message signing and verification operations within the IoV network. Each VEC server S_j is responsible for managing pseudonym updates, while optimizing the trade-off between privacy preservation and system efficiency. Although individual vehicles benefit from privacy protection, the VEC server must balance pseudonym updates with associated authentication, computational, and communication costs. Beyond calculating vehicle-level utility, each VEC server S_j also evaluates its own utility function, which incorporates both vehicle privacy gains

for all vehicles in the coverage area of S_j and costs for the VEC server S_j ($C_j(t)$). The utility function for S_j is given by:

$$U_j(t) = \sum_{i \in \mathcal{V}^j} U_i(t) - \omega_3 C_j(t), \quad (5.21)$$

Expanding $U_i(t)$ from the vehicle-level utility function yields:

$$U_j(t) = \sum_{i \in \mathcal{V}^j} [\omega_1 H_{i,j}(t) - \omega_2 C_i(t)] - \omega_3 C_j(t), \quad (5.22)$$

where $H_{i,j}(t)$ denotes the privacy entropy, $C_i(t)$ represents the total vehicle-side cost, and $C_j(t)$ is the total VEC-side cost incurred by the VEC server for managing pseudonym updates. The weighting parameters ω_1 , ω_2 , and ω_3 balance the trade-off between privacy entropy and the associated costs, where $\omega_1, \omega_2, \omega_3 > 0$ correspond to privacy entropy, vehicle-side cost, and VEC-side cost, respectively. These weights are selected through empirical tuning to balance the different scales of entropy and cost metrics. In addition, the total cost incurred by each VEC server S_j consists of multiple components, including authentication costs, blockchain-related costs, and communication overhead. The total cost function is expressed as:

$$C_j(t) = C_{j,\text{com}}(t) + C_{j,\text{auth}}(t) + C_{j,\text{block}}(t). \quad (5.23)$$

In the VEC-IoV network, each VEC server aims to improve the privacy preservation and cost efficiency of vehicles under its coverage, thereby contributing to overall system performance. To achieve this, the social welfare function is defined as the sum of all VEC servers' utility functions. The objective of the system is to maximize total social welfare while ensuring the constraints. Therefore, the total social welfare at time t is defined as:

$$SW(t) = \sum_{j \in \mathcal{J}} U_j(t) = \sum_{j \in \mathcal{J}} \left[\sum_{i \in \mathcal{V}^j} (\omega_1 H_{i,j}(t) - \omega_2 C_i(t)) - \omega_3 C_j(t) \right], \quad (5.24)$$

The optimization problem is then formulated as follows:

$$\max \sum_{j \in \mathcal{J}} \left[\sum_{i \in \mathcal{V}^j} (\omega_1 H_{i,j}(t) - \omega_2 C_i(t)) - \omega_3 C_j(t) \right], \quad \forall t \in T \quad (5.25)$$

$$\text{s.t. C1: } H_{i,j}(t) \geq H_{\min}, \quad \forall i, j, t \quad (5.25a)$$

$$\text{C2: } C_i(t) \leq C_V^{\max}, \quad \forall i, t \quad (5.25b)$$

$$\text{C3: } C_j(t) \leq C_S^{\max}, \quad \forall j, t \quad (5.25c)$$

Constraint 5.25a ensures that the privacy entropy of each vehicle remains above a predefined threshold $H_{\min}$, guaranteeing appropriate privacy protection. Constraints 5.25b and 5.25c ensure that the vehicle's cost and the VEC server's cost remain below predefined values $C_V^{\max}$ and $C_S^{\max}$ respectively, preventing excessive resource usage at both the vehicle and server levels. The specific values of $H_{\min}$, $C_V^{\max}$, and $C_S^{\max}$ are reported in the simulation setup. The formulated problem is non-convex due to the composition of the logarithmic privacy entropy function with the density-dependent tracking probability, combined with the temporal coupling introduced by the exponential decay. Additionally, it is decentralized, as each VEC server makes decisions based on local observations without requiring global information. Given these challenges, traditional centralized optimization methods are impractical. Instead, a MADRL approach is proposed to enhance privacy preservation while ensuring cost efficiency in the VEC-IoV network, as detailed in the following section.

5.4 Cost-Aware Pseudonym Management MADRL Approach

To address the decentralized and dynamic nature of pseudonym management in VEC-IoV networks, the social welfare maximization problem is transformed into an MDP formulation. The pseudonym management process is modeled as a MADRL problem, where each VEC server functions as an independent agent. The objective is to improve the utility function of each VEC server, while improving the overall social welfare. Given the decentralized decision-making process and the necessity for stable multi-agent learning, MAPPO is employed.

Figure 5.3 provides an overview of the proposed CAPM-MAPPO algorithm, which is designed to improve pseudonym management while balancing cost and privacy in vehicular networks. This approach follows the CTDE paradigm [267], [279]. In CTDE, a centralized critic has access to global information during training. However, each VEC server (agent) executes policies independently, using only its

local observations. Finally, proximal updates ensure stable policy learning across multiple agents.

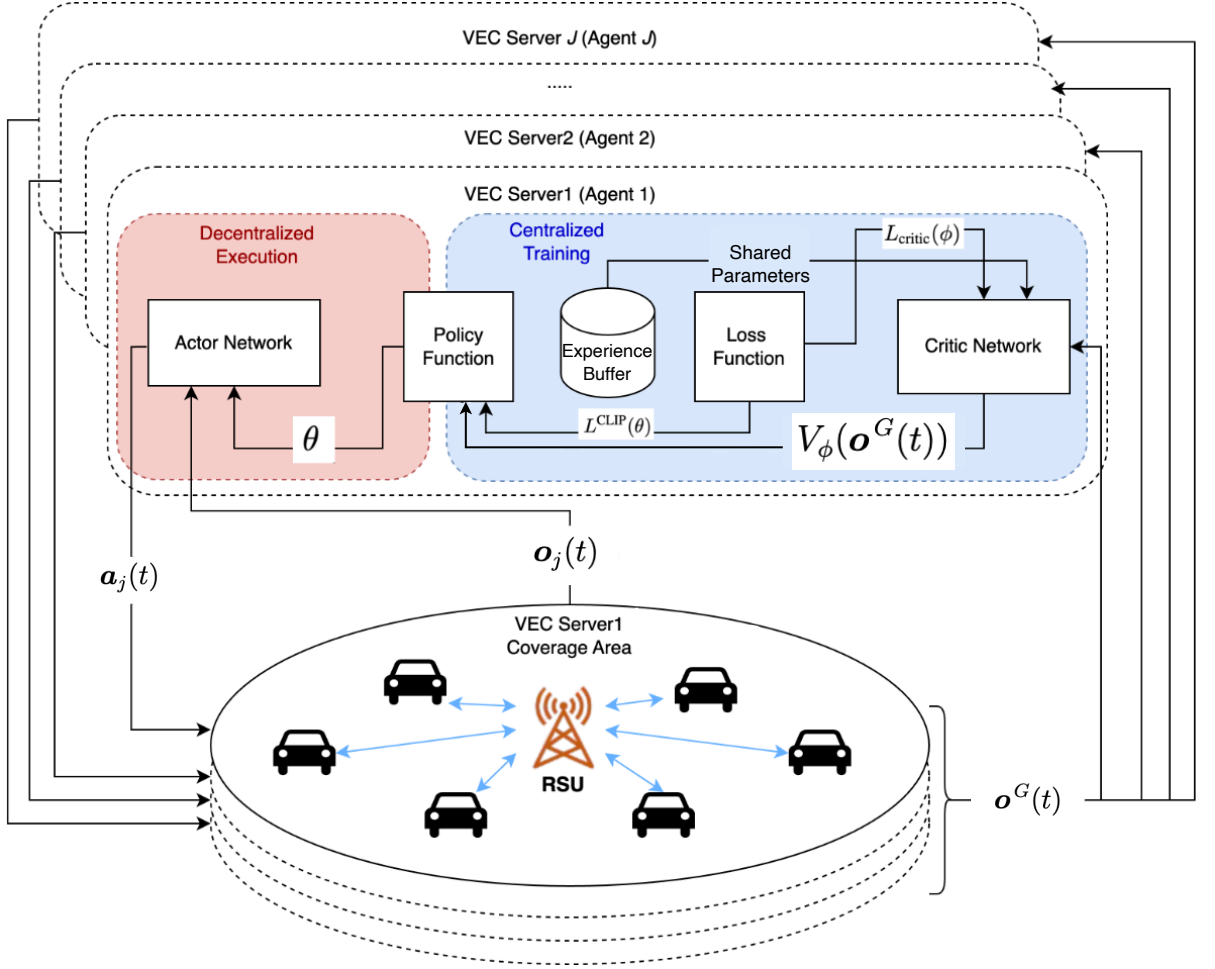


Figure 5.3: Overview of CAPM-MAPPO algorithm details

5.4.1 State Space

The state space for each VEC server j at time t is defined as a set of key system parameters that influence decision-making in pseudonym management based on its local observation of the network environment. The local observation comprises both per-vehicle features and server-level features. For each vehicle $i \in \mathcal{V}^j$, the VEC server monitors its privacy entropy $H_{i,j}(t)$ and the associated vehicle-side cost $C_i(t)$. At the server level, the observation includes the number of vehicles within coverage $|\mathcal{V}^j|$, the server-side cost $C_j(t)$, and the average utility of neighboring VEC servers $\bar{U}_{\mathcal{N}_j}(t)$,

where $\mathcal{N}_j$ denotes the set of VEC servers whose coverage areas are geographically adjacent to S_j . Formally, the local observation for each agent is expressed as:

$$\mathbf{o}_j(t) = \{\{H_{i,j}(t), C_i(t)\}_{i \in \mathcal{V}^j}, |\mathcal{V}^j|, C_j(t), \bar{U}_{\mathcal{N}_j}(t)\} \quad (5.26)$$

Additionally, the critic network requires a global view of the system to estimate the long-term reward and stabilize learning. The global state is defined as:

$$\mathbf{o}^G(t) = \{\mathbf{o}_j(t) \mid j \in \mathcal{J}\} \quad (5.27)$$

Using the global state during training, the critic provides an evaluation of the total system performance, guiding each VEC server to align its local actions with global optimization goals.

5.4.2 Action Space

Each VEC server j determines whether a vehicle within its coverage should change its pseudonym at time t . Instead of making a deterministic decision, the policy function outputs a probability distribution over actions, allowing for exploration during training. The actor network, parameterized by θ , defines a stochastic policy π_θ that maps the local observation to a probability of selecting $a_i^j(t) = 1$, which represents a pseudonym change:

$$\mathcal{P}_i^j(t) = \pi_\theta(a_i^j = 1 \mid \mathbf{o}_j(t)), \quad (5.28)$$

where $\mathcal{P}_i^j(t)$ is the probability that vehicle i under VEC server j will change its pseudonym at time t . The final action is then sampled from a Bernoulli distribution:

$$a_i^j(t) \sim \text{Bernoulli}(\mathcal{P}_i^j(t)), \quad (5.29)$$

where $a_i^j(t) = 1$ indicates that the pseudonym of vehicle i is changed, while $a_i^j(t) = 0$ means the pseudonym remains unchanged. The stochastic nature of this action selection ensures that the policy explores different decisions during training and gradually converges to an optimal pseudonym change strategy. The overall action set for VEC server j at time t is defined as:

$$\mathbf{a}_j(t) = \{a_i^j(t) \mid i \in \mathcal{V}^j\} \quad (5.30)$$

5.4.3 Reward Function

The reward function for each VEC server j is designed to balance privacy preservation and cost efficiency while ensuring that both vehicle-side and server-side costs remain within acceptable limits. The local reward is constructed as the VEC server utility augmented with penalty terms that enforce the constraints defined in the optimization problem. Specifically, penalties are applied when the privacy entropy $H_{i,j}(t)$ drops below the predefined threshold $H_{\min}$ or when the costs $C_i(t)$ or $C_j(t)$ exceed their respective maximum allowable values.

The local reward function for VEC server j at time t is given by:

$$R_j^{\text{local}}(t) = U_j(t) - \sum_{i \in \mathcal{V}^j} \Phi_H(H_{i,j}(t)) - \sum_{i \in \mathcal{V}^j} \Phi_C(C_i(t), C_j(t)), \quad (5.31)$$

where $U_j(t)$ is the VEC server utility as defined in Eq. 5.22, and the penalty functions $\Phi_H(\cdot)$ and $\Phi_C(\cdot)$ enforce the privacy and cost constraints, respectively.

To enforce the privacy constraint, the penalty function $\Phi_H(H_{i,j}(t))$ applies a negative reward whenever $H_{i,j}(t)$ falls below the minimum threshold $H_{\min}$:

$$\Phi_H(H_{i,j}(t)) = \begin{cases} \psi_H (H_{\min} - H_{i,j}(t)), & \text{if } H_{i,j}(t) < H_{\min} \\ 0, & \text{otherwise,} \end{cases} \quad (5.32)$$

where $\psi_H > 0$ is a penalty weight controlling the magnitude of privacy constraint violations.

Similarly, to ensure cost constraints, the penalty function $\Phi_C(C_i(t), C_j(t))$ penalizes excessive vehicle-side and VEC server-side costs:

$$\Phi_C(C_i(t), C_j(t)) = \zeta_1 \max(0, C_i(t) - C_V^{\max}) + \zeta_2 \max(0, C_j(t) - C_S^{\max}), \quad (5.33)$$

where $\zeta_1, \zeta_2 > 0$ are penalty weights that control the magnitude of vehicle-side and server-side cost violations, respectively. The weighting parameters ω_1 , ω_2 , and ω_3 within $U_j(t)$ balance the trade-off between privacy entropy and the associated costs, while the penalty weights ψ_H , ζ_1 , and ζ_2 enforce the constraint boundaries.

The global reward function, used by the centralized critic, is based on social welfare:

$$R^G(t) = \frac{SW(t)}{|\mathcal{J}|} = \frac{\sum_{j \in \mathcal{J}} U_j(t)}{|\mathcal{J}|} \quad (5.34)$$

This enables the critic network to learn the system-wide efficiency, while each VEC server independently enhances its own local utility.

5.4.4 MAPPO Training with CTDE

The MAPPO framework follows an actor-critic architecture, where the actor network, parameterized by θ , is responsible for policy estimation, and the critic network, parameterized by ϕ , evaluates the expected return to guide training. The actor network maps the local observation $\mathbf{o}_j(t)$ to a stochastic policy function:

$$\pi_{\theta}(\mathbf{a}_j(t) \mid \mathbf{o}_j(t)), \quad (5.35)$$

which outputs a probability distribution over possible actions. The critic network estimates the expected cumulative reward by evaluating the global state $\mathbf{o}^G(t)$, ensuring that local actions align with system-wide optimization:

$$V_{\phi}(\mathbf{o}^G(t)) = \mathbb{E} \left[\sum_{t'=0}^{\infty} \gamma^{t'} R_j^G(t+t') \mid \mathbf{o}^G(t) \right], \quad (5.36)$$

where $\gamma \in [0, 1)$ is the discount factor that determines the relative importance of future rewards, and t' denotes the number of time steps into the future.

The training process follows the PPO framework, where the policy is updated iteratively using a clipped surrogate loss function. The advantage function A_t , estimated by the critic using the global state, quantifies how much better an action is compared to the expected value:

$$A_t = R^G(t) + \gamma V_{\phi}(\mathbf{o}^G(t+1)) - V_{\phi}(\mathbf{o}^G(t)) \quad (5.37)$$

To ensure stable learning, the policy update is performed using the PPO clipped objective:

$$L^{\text{CLIP}}(\theta) = \mathbb{E} [\min(r_t(\theta) A_t, \text{clip}(r_t(\theta), 1 - \epsilon, 1 + \epsilon) A_t)] + \Lambda \mathcal{H}(\pi_{\theta}), \quad (5.38)$$

where $r_t(\theta)$ represents the probability ratio between the new and old policies:

$$r_t(\theta) = \frac{\pi_{\theta}(\mathbf{a}_j \mid \mathbf{o}_j)}{\pi_{\theta_{\text{old}}}(\mathbf{a}_j \mid \mathbf{o}_j)}, \quad (5.39)$$

and A_t is the advantage function defined in Eq. 5.37. The clipping function prevents large policy updates, ensuring stability in learning. Additionally, an entropy

regularization term $\mathcal{H}(\pi_\theta)$ is included to encourage exploration, with Λ denoting the entropy coefficient that controls the exploration-exploitation trade-off.

The critic network is trained using a value loss function based on the Temporal Difference (TD) error:

$$L_{\text{critic}}(\phi) = \mathbb{E} \left[\left(R^G(t) + \gamma V_\phi(\mathbf{o}^G(t+1)) - V_\phi(\mathbf{o}^G(t)) \right)^2 \right], \quad (5.40)$$

ensuring that the value function accurately estimates future rewards.

During execution, each VEC server applies the learned policy in a fully decentralized manner. Since the per-vehicle actions are conditionally independent given the observation, the final action for each vehicle is selected as:

$$a_i^{j*}(t) = \arg \max_{a_i^j \in \{0,1\}} \pi_{\theta^*}(a_i^j \mid \mathbf{o}_j(t)), \quad \forall i \in \mathcal{V}^j, \quad (5.41)$$

where θ^* represents the optimized policy parameters. This enables each VEC server to make privacy-aware pseudonym change decisions while benefiting from system-wide knowledge acquired during training. During training, actions are sampled stochastically to encourage exploration, whereas at execution time, the learned policy is applied deterministically by selecting the action with the highest probability for each vehicle independently.

More details of the proposed MAPPO algorithm are provided in Algorithm 5.I.

Algorithm 5.I: CAPM-MAPPO Pseudonym Management Algorithm

Input: Number of VEC servers $|\mathcal{J}|$, number of vehicles per VEC server $|\mathcal{V}^j|$, learning rate η_{lr} , discount factor γ , clipping parameter ϵ , entropy coefficient Λ , batch size $\mathcal{B}$, and maximum training time steps T

Output: Optimized policy π_θ for pseudonym management decisions

Initialize actor network parameters θ , critic network parameters ϕ , and experience buffer $\mathcal{D}$;

foreach *training episode* **do**

Reset environment and observe initial global state $\mathbf{o}^G(0)$;

for $t = 1, \dots, T$ **do**

foreach *VEC server* $j \in \mathcal{J}$ **do**

Observe local observation $\mathbf{o}_j(t)$;

Compute action probabilities $\mathcal{P}_i^j(t) = \pi_\theta(a_i^j = 1 \mid \mathbf{o}_j(t))$ for each vehicle $i \in \mathcal{V}^j$ (Eq. 5.28);

Sample action $a_i^j(t) \sim \text{Bernoulli}(\mathcal{P}_i^j(t))$ for each vehicle $i \in \mathcal{V}^j$;

Execute joint action $\mathbf{a}_j(t) = \{a_i^j(t) \mid i \in \mathcal{V}^j\}$;

Compute local reward

$R_j^{\text{local}}(t) = U_j(t) - \sum_{i \in \mathcal{V}^j} \Phi_H(H_{i,j}(t)) - \sum_{i \in \mathcal{V}^j} \Phi_C(C_i(t), C_j(t))$;

Observe next observation $\mathbf{o}_j(t+1)$ and global state $\mathbf{o}^G(t+1)$;

Compute global reward $R^G(t) = SW(t)/|\mathcal{J}|$;

Store transition

$(\mathbf{o}_j(t), \mathbf{a}_j(t), R_j^{\text{local}}(t), R^G(t), \mathbf{o}_j(t+1), \mathbf{o}^G(t+1))$ in buffer $\mathcal{D}$;

if *size of buffer $\mathcal{D}$ reaches batch size $\mathcal{B}$* **then**

foreach *VEC server* $j \in \mathcal{J}$ **do**

Compute advantage

$A_t = R^G(t) + \gamma V_\phi(\mathbf{o}^G(t+1)) - V_\phi(\mathbf{o}^G(t))$;

Compute probability ratio $r_t(\theta)$ (Eq. 5.39);

Compute clipped actor loss $L^{\text{CLIP}}(\theta)$ (Eq. 5.38);

Update actor: $\theta \leftarrow \theta + \eta_{\text{lr}} \nabla_\theta L^{\text{CLIP}}(\theta)$;

$\theta_{\text{old}} \leftarrow \theta$; **// Store updated policy for next ratio**

Compute critic loss $L_{\text{critic}}(\phi)$ using global reward and global state (Eq. 5.40);

Update critic: $\phi \leftarrow \phi - \eta_{\text{lr}} \nabla_\phi L_{\text{critic}}(\phi)$;

Clear buffer $\mathcal{D}$;

return *Optimized policy π_θ*

5.5 Simulation Results and Discussions

This section presents the simulation-based evaluation of the proposed CAPM-MAPPO algorithm in VEC-IoV networks.

5.5.1 Simulation Setup

The simulation is implemented in Python 3.10 using the PyTorch framework [280]. The proposed CAPM-MAPPO algorithm is evaluated in a VEC-IoV network environment where each VEC server acts as an independent RL agent while training is guided by a centralized critic. The simulation environment is modeled using SUMO to ensure realistic vehicular mobility patterns. A synthetic urban grid network covering an area of approximately 5.6 km^2 is constructed, within which 10 RSUs are strategically deployed to function as VEC servers, each with a coverage radius of approximately 560 m, resulting in a per-server coverage area of approximately 1 km^2 . Adjacent VEC server coverage areas partially overlap to ensure continuous connectivity as vehicles traverse the network. To evaluate the robustness of the system across diverse operational conditions, the number of vehicles per VEC server is varied from 10 to 50, representing scenarios from sparse to congested urban traffic. Each vehicle maintains an average speed of 30–60 km/h depending on traffic density and road conditions. Each decision time step corresponds to an interval of $\tau = 1$ second of simulation time.

The proposed CAPM-MAPPO algorithm is compared with the following benchmark algorithms to evaluate the efficiency and privacy protection capacity of the algorithm:

- **Greedy Pseudonym Change:** This method greedily prioritizes immediate privacy gains by triggering a pseudonym change whenever a vehicle's privacy entropy $H_{i,j}(t)$ falls below the predefined threshold $H_{\min}$. It selects the action that maximizes short-term privacy without considering the associated communication and computational costs or long-term system utility.
- **Fixed-Interval Pseudonym Change:** This baseline follows a predefined time-based strategy, where vehicles update their pseudonyms at regular, fixed intervals regardless of the current privacy level or cost.
- **Random Pseudonym Change:** Vehicles randomly change their pseudonyms without considering privacy entropy, cost efficiency, or social welfare.

In addition, the Dynamic Pseudonym Swap Program (DPSP) [227] is included as a comparative baseline. DPSP is a state-of-the-art location privacy protection scheme designed for the IoV, which dynamically selects pseudonym swap areas based on traffic density, road topology, and vehicle mobility. The algorithm enables vehicles to exchange pseudonyms at fog-managed RSUs to enhance anonymity by increasing the size of the swap group. Unlike CAPM-MAPPO, DPSP does not consider cost-awareness or system overhead in its decision-making process and lacks dynamic learning capabilities. Since DPSP focuses primarily on enhancing anonymity through pseudonym swaps without modeling cost constraints or system utility, it is compared against the proposed algorithm only in terms of average privacy entropy versus the number of vehicles, as shown in Figure 5.6, where the evaluation context and metrics are most closely aligned.

Cost Model Parameters

To accurately evaluate the time cost associated with pseudonym updates, the costs are categorized into vehicle-side and VEC-side components, as defined in Section 5.3. The vehicle-side cost captures the uplink communication and local cryptographic operations, while the VEC-side cost captures the server processing, blockchain query, and downlink communication.

The maximum vehicle-side cost ($C_V^{\max}$) consists of two components: the uplink communication delay $C_{i,\text{com}}^{\max}$ from the vehicle to the RSU, and the local cryptographic processing cost $C_{i,\text{auth}}^{\max}$. The uplink communication delay is approximately 20 ms [229]. Following [232], the vehicle incurs 2.8 ms for signing the message and 6.88 ms for verifying received messages, resulting in a total local authentication cost of 9.68 ms. Therefore, the maximum vehicle-side cost is upper-bounded by:

$$C_V^{\max} = C_{i,\text{com}}^{\max} + C_{i,\text{auth}}^{\max} = 20 + 9.68 = 29.68 \text{ ms.} \quad (5.42)$$

The maximum VEC-side cost ($C_S^{\max}$) consists of three components: the batch authentication cost $C_{j,\text{auth}}^{\max}$, the blockchain query cost $C_{j,\text{block}}^{\max}$, and the downlink communication delay $C_{j,\text{com}}^{\max}$ from the RSU to the vehicle. Following [232], the VEC server performs batch verification for n vehicles within its coverage region, which incurs a delay of:

$$C_{j,\text{auth}}^{\max} = 1.41n + 5.46 \text{ ms.} \quad (5.43)$$

The blockchain query overhead is upper-bounded by approximately 10 ms per transaction. For the downlink communication, the 20 ms delay from [229] is adopted. Combining these terms yields the maximum VEC-side cost:

$$C_S^{\max} = (1.41n + 5.46) + 10 + 20 = 1.41n + 35.46 \text{ ms.} \quad (5.44)$$

Tables 5.1 and 5.2 summarize the vehicle-side and VEC-side cost components in detail.

Table 5.1: Vehicle-Side Cost Components

Component	Description	Time (ms)
$C_{i,\text{com}}^{\max}$	Uplink Communication Delay	20
$C_{i,\text{auth}}^{\max}$	Message Signing	2.8
	Message Verification	6.88

Table 5.2: VEC-Side Cost Components

Component	Description	Time (ms)
$C_{j,\text{auth}}^{\max}$	Batch Authentication Cost	$1.41n + 5.46$
$C_{j,\text{block}}^{\max}$	Blockchain Query Overhead	10
$C_{j,\text{com}}^{\max}$	Downlink Communication Delay	20

The MAPPO-specific training parameters and problem-specific simulation parameters are listed in Tables 5.3 and 5.4, respectively.

Table 5.3: MAPPO Algorithm Training Parameters

Parameter	Description	Value
Optimizer	Network optimizer	Adam
Activation	Activation function	ReLU
η_r	Learning rate	0.001
γ	Discount factor	0.99
ϵ	PPO clipping parameter	0.2
Λ	Entropy regularization coefficient	0.01
$\mathcal{B}$	Batch size	64
Experience buffer	Experience buffer size	4e5
Episodes	Number of training episodes	100000

Table 5.4: Simulation Parameters

Parameter	Description	Value
$ \mathcal{V}^j $	Vehicles per VEC server	10–50
τ	Decision time step interval	1 s
ω_1	Privacy entropy weight	0.5
ω_2	Vehicle-side cost weight	0.3
ω_3	VEC-side cost weight	0.2
$H_{\min}$	Minimum privacy entropy	2 bits
λ	Privacy decay rate	1.0
ξ	Attacker capability	0.3
ψ_H	Privacy penalty weight	10
ζ_1	Vehicle cost penalty weight	5
ζ_2	VEC cost penalty weight	5

5.5.2 Results and Analysis

The performance of the proposed CAPM-MAPPO algorithm is examined by comparing it against several baseline methods, including Greedy, Random, and fixed-interval schemes, and the state-of-the-art DPSP algorithm in terms of privacy entropy. First, the average reward is analyzed, which highlights the improved learning efficiency of CAPM-MAPPO over the alternatives. Next, privacy entropy and social welfare are evaluated to demonstrate how the proposed algorithm dynamically balances privacy preservation and system overhead in dynamic vehicular environments.

Figure 5.4 examines the average reward with CAPM-MAPPO. It presents a comparative analysis of the performance of different decision-making strategies, illustrating the evolution of the average episode reward over a series of training episodes. The CAPM-MAPPO algorithm demonstrates superior performance, exhibiting a smooth convergence to the highest reward values, signifying effective learning. In contrast, the fixed-interval strategy achieves a moderate reward level, but remains stable without significant improvement over time. The Greedy method, while outperforming the Random strategy, demonstrates considerable fluctuation and lacks convergence. Finally, the Random strategy performs the worst, as expected, due to the absence of structured learning. These results highlight the effectiveness of MAPPO in optimizing long-term rewards, while other approaches do not achieve similar stability and efficiency.

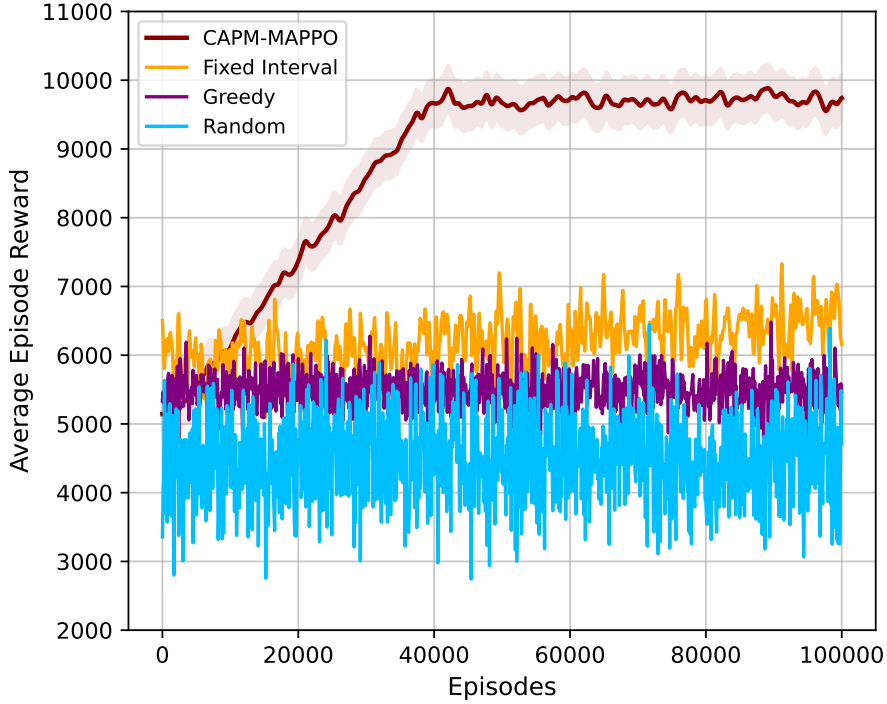


Figure 5.4: Average reward under different methods

To evaluate the effect of learning rate on the convergence behavior of the proposed CAPM-MAPPO algorithm, experiments are conducted using three distinct learning rate values. As depicted in Figure 5.5, the learning rate significantly influences both the stability and the convergence speed of the training process. A higher learning rate ($\eta_r = 0.01$) accelerates convergence during initial training episodes but introduces considerable fluctuations, potentially leading to unstable policies. In contrast, a lower learning rate ($\eta_r = 0.0001$) exhibits a smoother learning curve with reduced fluctuations, although it does not converge within the observed training range. The intermediate value ($\eta_r = 0.001$) achieves a stable balance, providing stable convergence with minimal oscillation. These results highlight the importance of selecting an appropriate learning rate in MADRL algorithms.

To evaluate privacy performance, the concept of average privacy entropy [227] across a group of vehicles within the coverage area of a VEC server is introduced. As mentioned, privacy entropy provides a quantitative measure of an attacker's

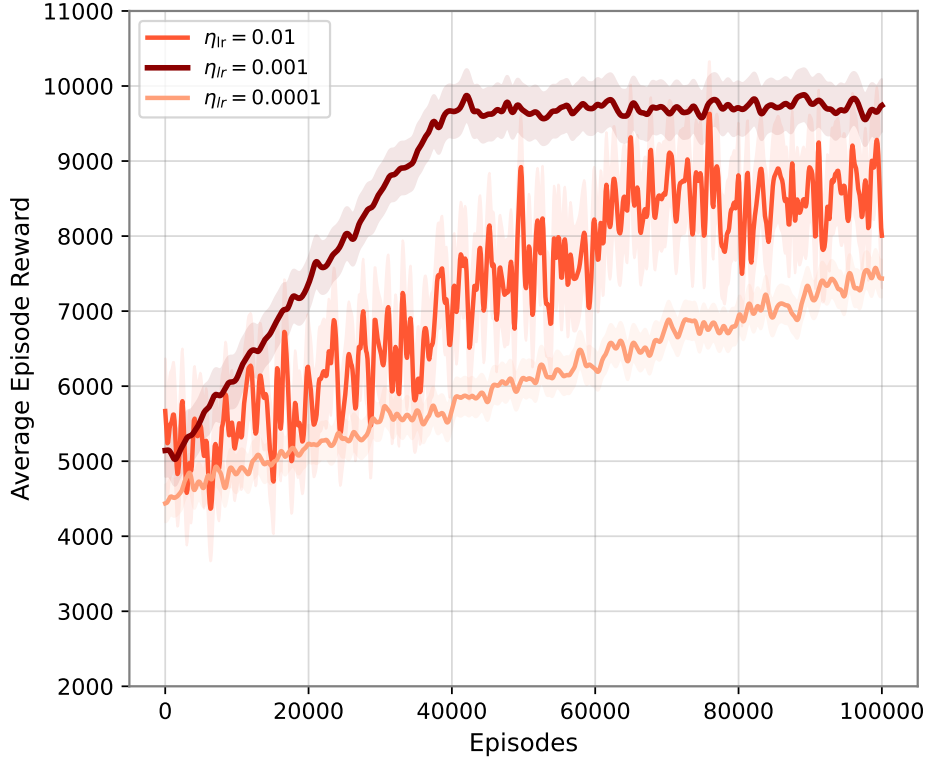


Figure 5.5: Average reward under different learning rates

uncertainty in tracking a vehicle. The average privacy entropy is defined in Eq. 5.45.

$$\bar{H}_j(t) = \frac{1}{|\mathcal{V}^j|} \sum_{i \in \mathcal{V}^j} H_{i,j}(t), \quad (5.45)$$

where $|\mathcal{V}^j|$ is the number of vehicles in the RSU's coverage area and $H_{i,j}(t)$ is the privacy entropy of the i th vehicle within the j th VEC server area at time t . This metric provides a holistic view of the system's overall privacy level, capturing the effectiveness of pseudonym management.

Figure 5.6 shows the average privacy entropy achieved by the proposed CAPM-MAPPO algorithm compared to the DPSP algorithm, fixed-interval and Random methods as the number of vehicles within the coverage area increases. As the vehicle count grows (up to 50 vehicles), the average privacy entropy improves for all methods due to the expanded anonymity set. However, the MADRL-based CAPM-MAPPO algorithm consistently outperforms the other three methods, which are not based on RL. While DPSP employs a location-based pseudonym swapping strategy driven by a predefined set of rules or conditions, CAPM-MAPPO dynamically learns

pseudonym change policies based on real-time environmental conditions. In particular, at 50 vehicles, CAPM-MAPPO achieves an average privacy entropy of 6.6 bits, representing a 32% improvement over the state-of-the-art DPSP algorithm. In addition, the mean of these average values across all vehicle densities is computed. The proposed CAPM-MAPPO algorithm achieves a mean of 4.88 bits, which is approximately 28% higher than DPSP's mean privacy entropy. When comparing CAPM-MAPPO against all baseline methods (DPSP, fixed-interval, and Random), the overall improvement averages about 66.3%. This significant gain underscores the advantage of dynamically learning pseudonym change policies based on real-time environmental conditions rather than relying on static strategies.

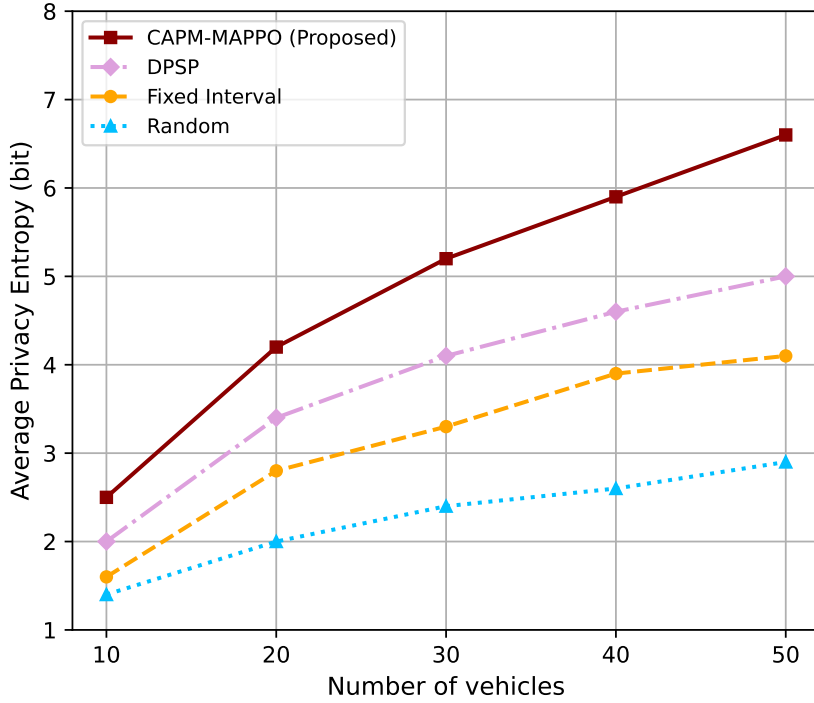


Figure 5.6: Average privacy entropy vs number of vehicles

Figure 5.7 presents the impact of increasing the number of vehicles within the VEC-IoV network on the overall social welfare, comparing the proposed CAPM-MAPPO algorithm against baseline methods. As the total number of vehicles increases from 100 to 500, distributed across 10 RSUs while each RSU serving 10–50 vehicles as VEC servers, CAPM-MAPPO consistently demonstrates improved social welfare. In contrast, the fixed-interval method displays a relatively flat trend, re-

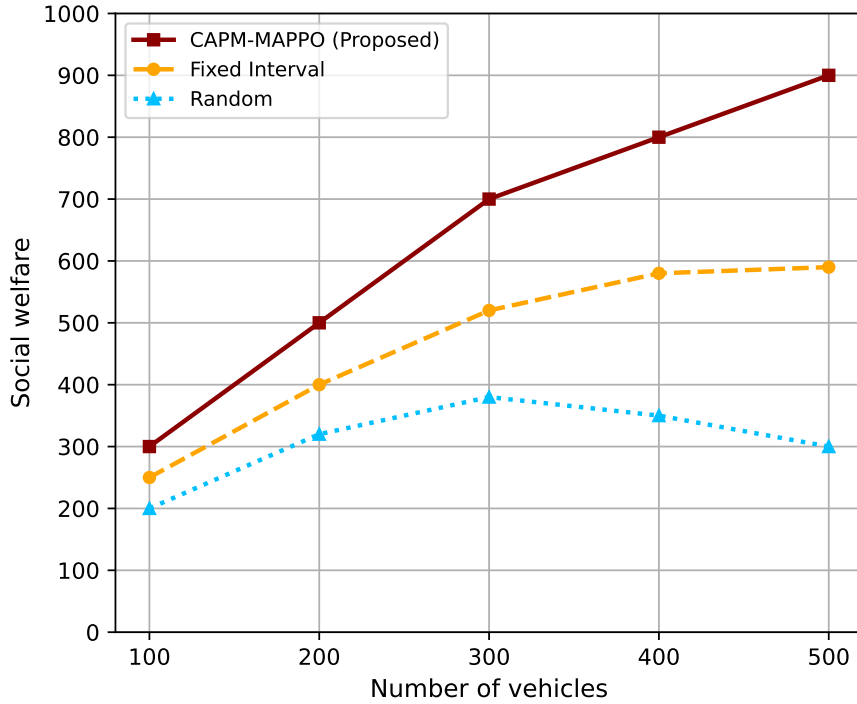


Figure 5.7: Social welfare vs number of vehicles

vealing its insensitivity to changing environmental dynamics. However, the Random method initially benefits from increased vehicle density, but eventually declines as cumulative VEC-side and vehicle-side costs take effect. In particular, when computing the mean social welfare across all vehicle densities, CAPM-MAPPO achieves a value of 640, compared to 468 for the fixed-interval method and 310 for the Random method. This translates to an average improvement of approximately 71.6% over the other methods.

The evaluation investigates how different VEC-side cost scenarios affect overall social welfare. Figure 5.8 shows the impact of VEC-side cost on social welfare when 10 vehicles are served by a single VEC server. The VEC-side cost is largely determined by batch authentication cost, which depend on the number of vehicles authenticated together. For instance, the lower bound of 58.28 ms corresponds to the cost of simultaneously authenticating 2 vehicles, while the upper bound of 229.59 ms represents the expense of authenticating all 10 vehicles as a batch. As depicted in the figure, our CAPM-MAPPO algorithm consistently outperforms the baseline strategies and achieves higher social welfare at different VEC-side costs.

Specifically, CAPM-MAPPO achieves a 19.90% improvement over the fixed-interval baseline and a 36.70% improvement over the random baseline. Although all methods show a slight decrease in social welfare, CAPM-MAPPO exhibits better adaptability and cost-awareness by dynamically adjusting pseudonym change decisions, thereby maintaining relatively high social welfare.

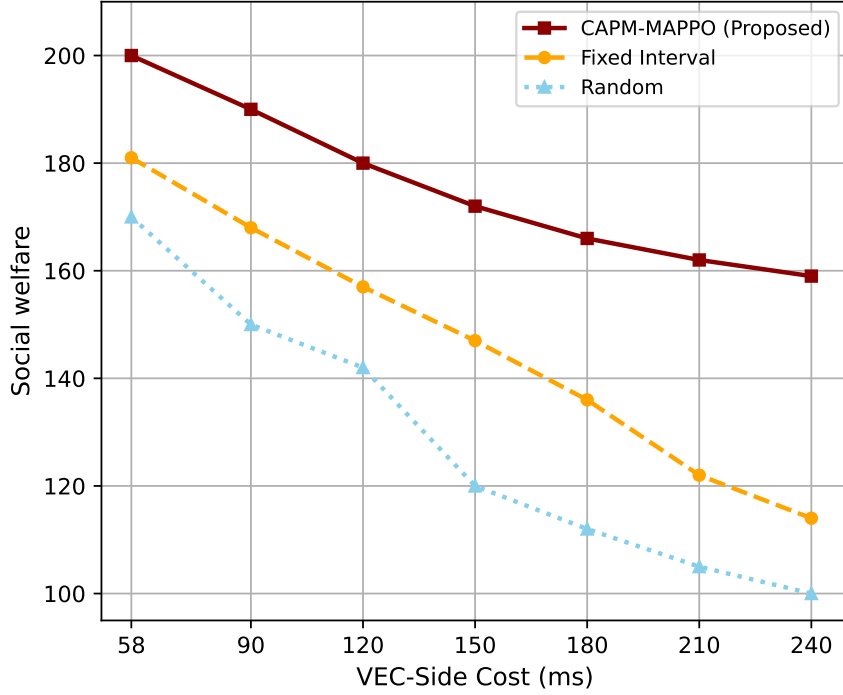


Figure 5.8: Social welfare vs VEC-side cost(ms)

Figure 5.9 demonstrates the impact of varying the privacy entropy decay rate λ on the social welfare of the VEC-IoV network. The CAPM-MAPPO algorithm exhibits a well-balanced response between privacy preservation and cost efficiency. When λ is low, privacy entropy slowly decreases, resulting in infrequent pseudonym updates and reduced privacy protection. On the other hand, higher values of λ result in rapid entropy decay, leading to more frequent pseudonym changes and incurring additional computational and communication costs. Unlike the fixed-interval and random baselines, which show limited adaptability to changing decay rates, CAPM-MAPPO dynamically adjusts its policy based on the environment's dynamics. This algorithm achieves a 49.55% improvement over the fixed-interval baseline and a 161.71% improvement over the random baseline in terms of the evaluated metric,

enabling it to maintain higher social welfare under varying conditions and demonstrating its suitability for the complexities of VEC-IoV environments.

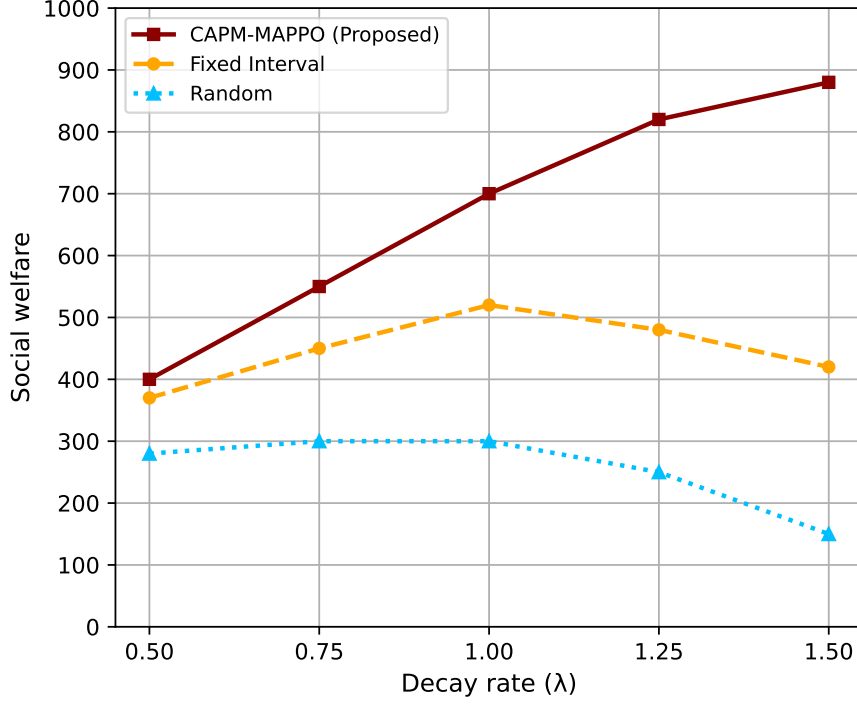


Figure 5.9: Social welfare vs decay rate

5.6 Conclusions

This chapter introduces BPMV, a blockchain-enabled pseudonym management framework for VEC in IoV networks, designed to address **RQ2a**. Specifically, the framework aims to enhance privacy preservation during data sharing in VEC-IoV by designing novel decentralized blockchain-based protocols for pseudonym authentication and revocation. Additionally, a social welfare function is formulated to enable dynamic, cost-aware pseudonym updates, and a MADRL algorithm named CAPM-MAPPO is proposed. The proposed algorithm learns to enhance social welfare through adaptive decision-making while satisfying constraints under varying network conditions. Simulation results demonstrate that CAPM-MAPPO significantly outperforms baseline methods, including Greedy, Random, and Fixed-Interval schemes, in terms of convergence speed, social welfare, and privacy entropy across diverse vehicular densities and cost scenarios. Moreover, compared to the state-of-the-art DPSP approach, our method achieves higher privacy entropy. In particular,

CAPM-MAPPO yields an average improvement of approximately 71.6% in social welfare over these baselines.

Chapter 6

Blockchain-Based Reputation Management for Privacy Preservation

As mentioned in previous chapters, the IoV has emerged as a pivotal innovation in next-generation transportation. With the ongoing development of CAVs, this paradigm has garnered substantial interest from both academia and industry. CAVs are equipped with a wide range of onboard sensors capable of collecting diverse data, including traffic flow characteristics, road conditions, and environmental information. The sharing of this sensor-generated data among vehicles and infrastructure enables more effective decision-making, predictive analysis, and cooperative driving behaviors [33], [34], [281], [282], [283].

A notable aspect of VEC-IoV is the concept of data sharing, in which vehicles share valuable data, such as traffic conditions, sensor readings, and environmental information, with VEC servers or other entities [284]. This exchange enhances system-wide efficiency. However, despite its potential, the implementation of data sharing within VEC faces critical challenges, particularly in the context of privacy preservation and trust management.

Traditional trust management in the IoV has predominantly relied on centralized authorities to issue, aggregate, and maintain trust scores for CAVs. This centralized approach increases latency and introduces single points of failure, thereby risking network security and reliability. By contrast, blockchain-based architectures distribute trust management across a network of validators. Through decentralized

consensus, these systems record updates and maintain state securely, enhancing data integrity, auditability, and resistance to tampering.

Alongside trust, privacy remains a fundamental challenge in IoV environments due to the sensitive nature of data exchanged during V2X communications. Using pseudonym management schemes mitigate these privacy risks by enabling vehicles to communicate through temporary identities, effectively concealing their true identities and enhancing the safety and reliability of ITS applications.

In the previous chapter, an algorithm was proposed to address dynamic pseudonym management, and the BPMV framework was introduced based on blockchain, where the revocation list is managed but the details of reputation score calculation are not considered. In addition, since vehicles frequently change their pseudonyms to preserve privacy, maintaining reputation continuity across pseudonym transitions, without compromising vehicle privacy, remains a critical challenge [285], [286]. Reputation must therefore reflect not only the current pseudonym but also the cumulative behavior exhibited under prior temporary identities.

In this context, and in response to the thesis's second research question, "*What design model would effectively integrate blockchain technology with RL strategies to improve privacy and trust at the same time in these networks (RQ2b)?*", this chapter proposes a Blockchain-based Reputation Management for Privacy Preservation (BRMPP) framework for VEC-IoV. BRMPP introduces a permissioned blockchain architecture which is decentralized and tamper-proof [142] to maintain vehicle reputation scores and pseudonym-identity mappings, enabling seamless reputation continuity as vehicles move across different CA regions. The framework supports two pseudonym types: a Systemic Pseudonym Identity (SPID) for long-term authentication and a Communication Pseudonym Identity (CPID) for privacy-preserving communication. Within BRMPP, the Trust-Aware Pseudonym management based on Proximal Policy Optimization (TAP-PPO) algorithm is proposed, which formulates pseudonym rotation as an MDP using DRL and employs the PPO method with a trust-aware action masking mechanism. TAP-PPO uses the reputation information maintained by BRMPP to enforce trust constraints, ensuring that only vehicles with sufficient reputation are permitted to execute pseudonym rotations, while adaptively balancing privacy entropy against rotation cost.

6.1 System Model

This section presents the architecture of the system model and security model employed by the proposed BRMPP framework.

6.1.1 System Model for VEC-enabled IoV

The system is structured into two main layers: the perception layer and the blockchain-enabled layer. The system model comprises three categories of entities: CAs, CAVs acting as data requesters or data providers, and RSU/VEC servers, as presented in Figure 6.1. Detailed definitions of each layer and entities are provided below.

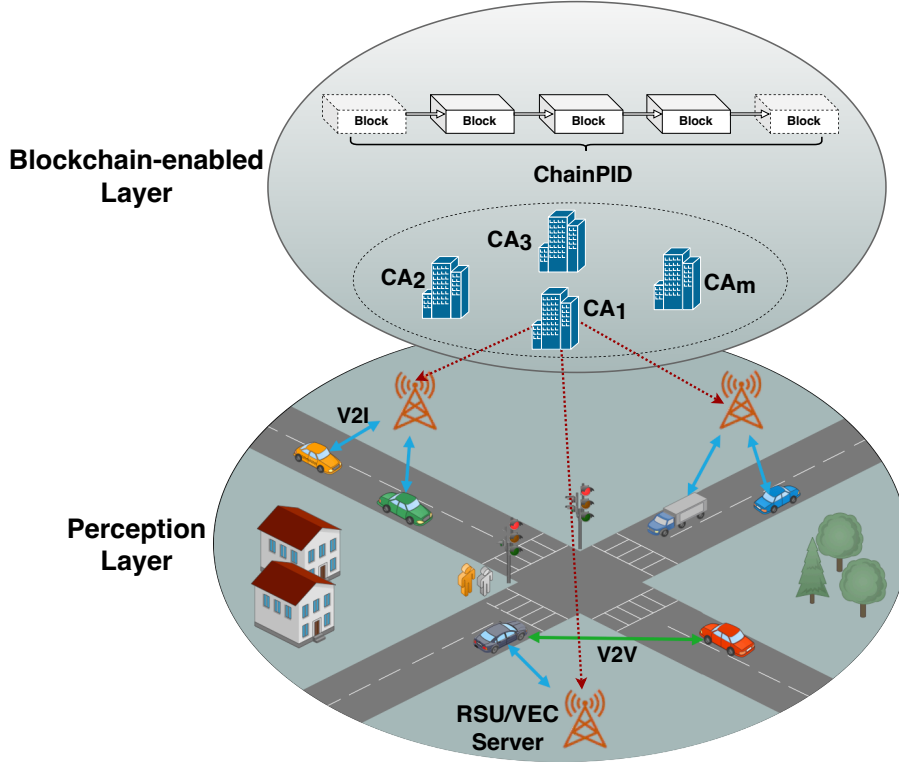


Figure 6.1: Blockchain-based system model

The **Perception Layer** comprises vehicles and RSUs. Vehicles, including CAVs, are equipped with OBUs containing sensors and communication modules. Each OBU acquires sensor data, performs local preprocessing, and transmits messages to nearby vehicles via V2V links or to RSUs via V2I communication links.

$\mathcal{S} = \{S_1, S_2, \dots, S_J\}$ denote the set of J RSUs deployed within the network that managed by each CA, where each RSU S_j functions as a VEC server capable of providing low-latency edge computing services within its local coverage area. Each RSU executes the TAP-PPO agent for pseudonym management decisions.

In addition, $\mathcal{V}_j = \{V_1, V_2, \dots, V_{N_j}\}$ denotes the set of $N_j = |\mathcal{V}_j|$ vehicles currently within the communication range of RSU S_j . Each vehicle $V_i \in \mathcal{V}_j$ is char-

acterized by a unique SPID_i within its registered Regional CA region for long-term blockchain authentication. Additionally, to facilitate privacy-preserving V2V communication, the vehicle utilizes a set of Communication Pseudonym Identities $\mathcal{C}_i = \{\text{CPID}_{i,1}, \dots, \text{CPID}_{i,c}\}$ for use within each VEC server's coverage area.

Moreover, $\mathcal{CA} = \{CA_1, CA_2, \dots, CA_M\}$ indicates the set of regional authorities. Operating with robust computational and storage resources, each CA_m initializes system parameters, generates public-private key pairs, and manages the registration of vehicles and RSUs within its region, including the issuance of SPIDs.

The **Blockchain-enabled layer** employs a permissioned blockchain which preserves the decentralized ledger model while restricting participation to authorized entities. The blockchain is named ChainPID, reflecting its primary role in managing Pseudonym IDentity records, and is maintained by all legally authorized CAs. The registration, update, and revocation of certificates are recorded as transactions on the blockchain. For ChainPID, the VEC server nodes have only read permission, while the CAs have both read and write permission. Consensus among the authorized CAs is achieved through a BFT-based consensus protocol suitable for permissioned blockchain environments with a known set of participants. Smart contracts are deployed on ChainPID to verify certificate status and to enforce reputation-based revocation policies.

6.1.2 Security Model

In the proposed BPPRM framework, VEC servers operate as semi-trusted entities that facilitate pseudonym management and local reputation assessment within their coverage areas. While VEC servers are authorized to issue and manage CPIDs, they may pose potential risks to vehicle privacy through unauthorized tracking or data correlation. Similarly, a subset of vehicles may exhibit malicious behavior to manipulate the reputation system through bad-mouthing, and on-off attacks. To address these threats, the framework employs a blockchain architecture for pseudonym management and reputation tracking. The following security and privacy requirements are established:

1. **Anonymity:** The real identity of each vehicle remains exclusively known to the issuing CA and is never exposed on the blockchain or during V2X communications. All interactions are conducted using pseudonyms, ensuring that neither VEC servers nor other vehicles can associate communication activities with a vehicle's true identity.

2. **Unlinkability:** Multiple CPIDs issued to the same vehicle under a given SPID cannot be linked by external observers, as each CPID is generated with a fresh cryptographic key pair and a unique random nonce. Since only CPIDs are used in V2X communications and the SPID-CPID mapping is known exclusively to the issuing VEC server, this approach prevents malicious vehicles from successfully conducting location-tracking attacks.
3. **Conditional Traceability:** In cases of detected misbehavior, the VEC server traces the CPID to the corresponding SPID, and the issuing CA recovers the real identity from its encrypted local database. This controlled revocation mechanism ensures that malicious vehicles can be identified and excluded from the network.
4. **Data Integrity and Immutability:** All pseudonym certificates, reputation records, and status transitions are recorded on ChainPID, using the tamper-proof properties of blockchain consensus. This guarantees that reputation scores and identity credentials cannot be altered, forged, or modified.

6.2 BRMPP Framework

In the proposed Blockchain-based Reputation Management for Privacy Preservation (BRMPP) framework, the dual goals of preserving vehicle privacy and enhancing trust through reputation management are simultaneously addressed. Privacy is safeguarded via pseudonym management, while trust is enhanced by calculating and updating reputation values associated with vehicles. Importantly, the framework enables reputation management across multiple pseudonyms while preserving user anonymity at two distinct levels: vehicles remain anonymous to VEC servers via their SPID and to peer vehicles via diverse CPID without revealing the linkage between these credentials. The following subsections detail the mechanisms for pseudonym generation and authentication, and the procedures for reputation management by calculating local and global reputation values.

6.2.1 Pseudonym Identity Issuance and Authentication

This section presents the pseudonym identity generation and authentication protocol of the proposed BRMPP framework. The protocol establishes a hierarchical identity structure comprising three layers, the Real Identity (RID) known only to the issuing CA, the systemic pseudonym or SPID valid within a CA region, and the communication pseudonym or CPID used for short-term anonymous interactions

within VEC coverage areas. Figure 6.2 illustrates the complete protocol workflow under different phases. The detailed steps of each phase are described below. In addition, Protocol 6.I and Protocol 6.II summarize these steps.

System Initialization

The system employs ECC, a public-key cryptographic scheme founded on elliptic curve mathematics [277]. During the initialization phase, the authorized CAs establish the cryptographic parameters in accordance with IEEE 1609.2 [82] and launch the ChainPID blockchain.

The CAs first agree on a common set of elliptic curve domain parameters defined as:

$$\mathcal{D} = (p, a, b, G, n, h), \quad (6.1)$$

where p is a large prime defining the finite field $\mathbb{F}_p$, (a, b) are coefficients of the elliptic curve $E : y^2 \equiv x^3 + ax + b \pmod{p}$, G is the generator point of prime order n , and h is the cofactor. The system specifies a cryptographic hash function $H : \{0, 1\}^* \rightarrow \mathbb{Z}_n^*$ [233].

Each CA_m independently generates its key pair:

$$sk_{CA_m} \xleftarrow{\$} [1, n-1], \quad PK_{CA_m} = sk_{CA_m} \cdot G. \quad (6.2)$$

To initialize the trust ledger, each CA generates a self-signed root certificate on the ChainPID blockchain:

$$\text{Cert}_{CA_m} = \{ID_{CA_m}, PK_{CA_m}, \mathcal{R}_m\}, \quad (6.3)$$

where ID_{CA_m} is the unique CA identifier, and $\mathcal{R}_m$ denotes the geographical CA_m 's region.

The public system parameters are recorded on the ChainPID ledger as:

$$\mathcal{P}_{\text{pub}} = \{\mathcal{D}, H, \{\text{Cert}_{CA_m}\}_{m=1}^M\}. \quad (6.4)$$

This immutable record enables VEC servers to retrieve and verify the public keys of regional CAs directly from the blockchain history. However, each CA_m securely stores its private key sk_{CA_m} , which is never disclosed.

Protocol 6.I: System Initialization and Entity Registration

Phase I: System Initialization

CAs agree on domain parameters $\mathcal{D} = (p, a, b, G, n, h)$ and hash function H

for *each* CA_m , $m = 1, \dots, M$ **do**

$sk_{CA_m} \xleftarrow{\$} [1, n-1]$, $PK_{CA_m} \leftarrow sk_{CA_m} \cdot G$
 Generate $\text{Cert}_{CA_m} \leftarrow \{ID_{CA_m}, PK_{CA_m}, \mathcal{R}_m\}$
 Publish Cert_{CA_m} to ChainPID

Record $\mathcal{P}_{\text{pub}} \leftarrow \{\mathcal{D}, H, \{\text{Cert}_{CA_m}\}_{m=1}^M\}$ on ChainPID

Phase II: Vehicle Registration

Input: Vehicle V_i registers with regional CA_m

/ Vehicle side* **/*

$sk_{i,m} \xleftarrow{\$} [1, n-1]$, $PK_{i,m} \leftarrow sk_{i,m} \cdot G$

Send $\{RID_{V_i}, PK_{i,m}\}$ to CA_m via secure channel

/ CA side* **/*

Verify RID_{V_i} against national registry

$\eta_{i,m} \xleftarrow{\$} \{0, 1\}^{128}$

$SPID_{i,m} \leftarrow H(RID_{V_i} \parallel \eta_{i,m} \parallel ID_{CA_m})$

if *first registration (no prior SPID exists)* **then**

$R_i^{\text{glob}} \leftarrow R_{\text{init}}$

else

/ Cross-region handover from CA_n* **/*

Retrieve $R_i^{\text{glob}}(SPID_{i,n})$ from ChainPID

$R_i^{\text{glob}}(SPID_{i,m}) \leftarrow R_i^{\text{glob}}(SPID_{i,n})$

Mark $SPID_{i,n}$ as expired on ChainPID

$\sigma_{CA_m} \leftarrow \text{Sign}_{sk_{CA_m}}(H(SPID_{i,m} \parallel PK_{i,m} \parallel T_i^{\text{exp}}))$

Publish $\text{Cert}_{i,m} \leftarrow \{SPID_{i,m}, PK_{i,m}, R_i^{\text{glob}}, T_i^{\text{exp}}, \sigma_{CA_m}\}$ to ChainPID

Store $SPID_{i,m} \rightarrow RID_{V_i}$ in encrypted private database

Send $\{SPID_{i,m}, \sigma_{CA_m}, T_i^{\text{exp}}\}$ to V_i

Phase III: VEC Server Registration

Input: VEC server S_j registers with governing CA_m

/ VEC side* **/*

$sk_{S_j} \xleftarrow{\$} [1, n-1]$, $PK_{S_j} \leftarrow sk_{S_j} \cdot G$

Send $\{ID_{S_j}, PK_{S_j}, \mathcal{Z}_j\}$ to CA_m

/ CA side* **/*

Verify S_j credentials

$\sigma_{CA_m,j} \leftarrow \text{Sign}_{sk_{CA_m}}(H(ID_{S_j} \parallel PK_{S_j} \parallel ID_{CA_m} \parallel \mathcal{Z}_j \parallel T_j^{\text{exp}}))$

Publish $\text{Cert}_{S_j} \leftarrow \{ID_{S_j}, PK_{S_j}, ID_{CA_m}, \mathcal{Z}_j, T_j^{\text{exp}}, \sigma_{CA_m,j}\}$ to ChainPID

Vehicle Registration

Each vehicle V_i registers with the regional CA_m to obtain a valid SPID within the CA's region. The vehicle generates a region-specific ECDSA key pair:

$$sk_{i,m} \xleftarrow{\$} [1, n-1], \quad PK_{i,m} = sk_{i,m} \cdot G. \quad (6.5)$$

For initial registration, the vehicle securely transmits its real identity RID_{V_i} and public key $PK_{i,m}$ to CA_m . Upon validating the identity against the national registry, CA_m generates the systemic pseudonym:

$$SPID_{i,m} = H(RID_{V_i} \parallel \eta_{i,m} \parallel ID_{CA_m}), \quad (6.6)$$

where $\eta_{i,m} \xleftarrow{\$} \{0, 1\}^{128}$ is a secure random nonce. The first CA that register the vehicle initializes the global reputation as $R_i^{\text{glob}} = R_{\text{init}}$, stores the mapping $SPID_{i,m} \rightarrow RID_{V_i}$ in its encrypted private database, and publishes the certificate to ChainPID:

$$\text{Cert}_{i,m} = \{SPID_{i,m}, PK_{i,m}, R_i^{\text{glob}}, T_i^{\text{exp}}, \sigma_{CA_m}\}, \quad (6.7)$$

where $\sigma_{CA_m} = \text{Sign}_{sk_{CA_m}}(H(SPID_{i,m} \parallel PK_{i,m} \parallel T_i^{\text{exp}}))$. The CA then transmits $\{SPID_{i,m}, \sigma_{CA_m}, T_i^{\text{exp}}\}$ to the vehicle.

When the vehicle travels to a new region governed by CA_n , reputation continuity is maintained through an inheritance mechanism. The vehicle presents its current credentials $(RID_{V_i}, SPID_{i,m}, PK_{i,m})$ to CA_n , which verifies the certificate on ChainPID and retrieves the accumulated reputation $R_i^{\text{glob}}(SPID_{i,m})$. Upon verification, CA_n generates a new pseudonym $SPID_{i,n}$ with a fresh key pair and initializes:

$$R_i^{\text{glob}}(SPID_{i,n}) \leftarrow R_i^{\text{glob}}(SPID_{i,m}). \quad (6.8)$$

The new certificate is published to ChainPID while $SPID_{i,m}$ is marked as expired.

VEC Server Registration

Each VEC server S_j registers with its governing CA_m to issue CPIDs for vehicles. The VEC generates its key pair as $sk_{S_j} \xleftarrow{\$} [1, n-1]$ and $PK_{S_j} = sk_{S_j} \cdot G$, then submits $\{ID_{S_j}, PK_{S_j}, \mathcal{Z}_j\}$ to CA_m , where $\mathcal{Z}_j$ defines the VEC server's coverage area.

Upon verification, CA_m issues:

$$\sigma_{CA_m,j} = \text{Sign}_{sk_{CA_m}}(H(ID_{S_j} \parallel PK_{S_j} \parallel ID_{CA_m} \parallel \mathcal{Z}_j \parallel T_j^{\text{exp}})), \quad (6.9)$$

and publishes Cert_{S_j} to ChainPID, where,

$$\text{Cert}_{S_j} = \{ID_{S_j}, PK_{S_j}, ID_{CA_m}, \mathcal{Z}_j, T_j^{\text{exp}}, \sigma_{CA_m, j}\}. \quad (6.10)$$

The VEC can then issue CPIDs and calculate local reputation values.

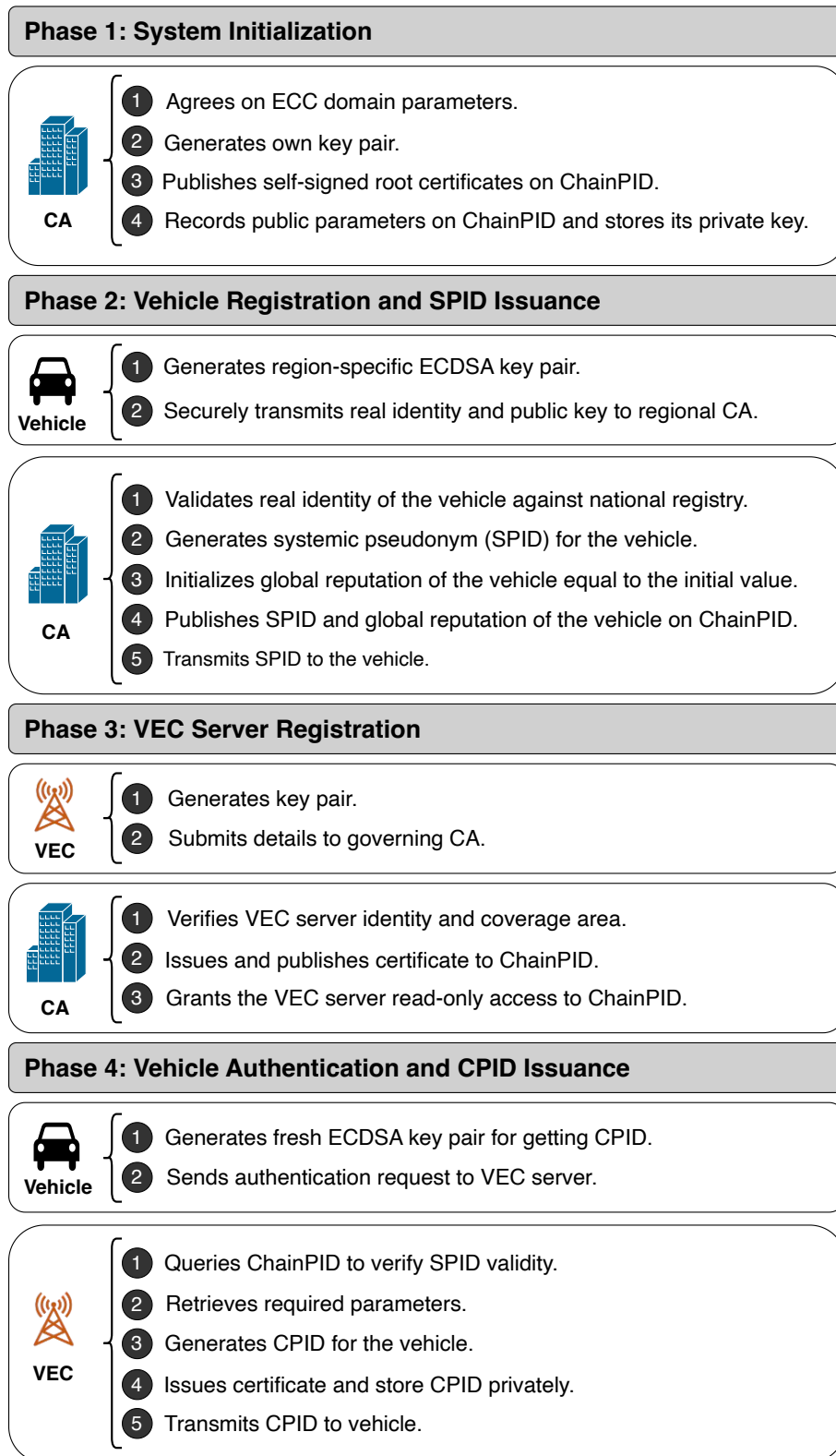


Figure 6.2: The Workflow of Pseudonym Identity Issuance and Authentication

Vehicle Authentication and CPID Issuance

When a vehicle enters a VEC server's coverage area, it must authenticate using its SPID credentials and obtain a CPID for short-term anonymous communication. To prevent cross-CPID linkability, the vehicle generates a fresh key pair for each CPID while proving SPID ownership through a separate authentication step.

Upon entering the coverage area $\mathcal{Z}_j$ of VEC server S_j , vehicle V_i holding valid $SPID_{i,m}$ generates a fresh key pair as $sk_{i,c} \xleftarrow{\$} [1, n-1]$ and $PK_{i,c} = sk_{i,c} \cdot G$, where c denotes the CPID index. This mechanism allows the vehicle to obtain multiple CPIDs under the same VEC server. The vehicle constructs an authentication request $\text{Req}_{\text{auth}} = \{SPID_{i,m}, PK_{i,c}, T_{\text{curr}}, \sigma_{\text{auth}}\}$, where T_{curr} is the current time and σ_{auth} defined as:

$$\sigma_{\text{auth}} = \text{Sign}_{sk_{i,m}}(H(SPID_{i,m} \parallel PK_{i,c} \parallel ID_{S_j} \parallel T_{\text{curr}})). \quad (6.11)$$

The VEC server S_j queries ChainPID to verify that $SPID_{i,m}$ is valid and retrieves $PK_{i,m}$ to verify σ_{auth} . Upon successful verification, S_j generates the CPID as:

$$CPID_{i,c} = H(SPID_{i,m} \parallel ID_{S_j} \parallel \eta_{i,c}), \quad (6.12)$$

where $\eta_{i,c} \xleftarrow{\$} \{0, 1\}^{128}$ is a random nonce, and VEC server issues a certificate:

$$\sigma_{S_j, i, c} = \text{Sign}_{sk_{S_j}}(H(CPID_{i,c} \parallel PK_{i,c} \parallel T_{i,c}^{\text{exp}})). \quad (6.13)$$

The VEC stores $\{CPID_{i,c}, PK_{i,c}, SPID_{i,m}, T_{i,c}^{\text{exp}}, \text{active}\}$ in its secure local database and transmits $\{CPID_{i,c}, \sigma_{S_j, i, c}, T_{i,c}^{\text{exp}}\}$ to the vehicle.

For CPID rotation, the vehicle generates a new key pair $(sk_{i,c+1}, PK_{i,c+1})$ and submits the rotation request:

$$\text{Req}_{\text{rotate}} = \{SPID_{i,m}, CPID_{i,c}, PK_{i,c+1}, T_{\text{curr}}, \sigma_{\text{rotate}}\}, \quad (6.14)$$

where:

$$\sigma_{\text{rotate}} = \text{Sign}_{sk_{i,c}}(H(SPID_{i,m} \parallel CPID_{i,c} \parallel PK_{i,c+1} \parallel T_{\text{curr}})). \quad (6.15)$$

The VEC verifies using $PK_{i,c}$ from its local database, generates $CPID_{i,c+1}$, marks $CPID_{i,c}$ as expired, and stores the new record. When the vehicle moves to a different

VEC within the same CA region, it authenticates using SPID credentials and receives a new CPID from the new VEC.

Protocol 6.II: Vehicle Authentication and CPID Management

Phase I: Vehicle Authentication and CPID Issuance**Input:** Vehicle V_i with valid $SPID_{i,m}$ enters coverage area $\mathcal{Z}_j$ of VEC S_j

/* Vehicle side */

 $sk_{i,c} \xleftarrow{\$} [1, n-1], \quad PK_{i,c} \leftarrow sk_{i,c} \cdot G$ $\sigma_{\text{auth}} \leftarrow \text{Sign}_{sk_{i,m}}(H(SPID_{i,m} \parallel PK_{i,c} \parallel ID_{S_j} \parallel T_{\text{curr}}))$ Send $\text{Req}_{\text{auth}} \leftarrow \{SPID_{i,m}, PK_{i,c}, T_{\text{curr}}, \sigma_{\text{auth}}\}$ to S_j

/* VEC side */

Query ChainPID for $\text{Cert}_{i,m}$ Verify $SPID_{i,m}$ is valid and not expiredRetrieve $PK_{i,m}$ from $\text{Cert}_{i,m}$ Verify σ_{auth} using $PK_{i,m}$ **if verification fails then**└ Reject request and **abort** $\eta_{i,c} \xleftarrow{\$} \{0, 1\}^{128}$ $CPID_{i,c} \leftarrow H(SPID_{i,m} \parallel ID_{S_j} \parallel \eta_{i,c})$ $\sigma_{S_j,i,c} \leftarrow \text{Sign}_{sk_{S_j}}(H(CPID_{i,c} \parallel PK_{i,c} \parallel T_{i,c}^{\text{exp}}))$ Store $\{CPID_{i,c}, PK_{i,c}, SPID_{i,m}, T_{i,c}^{\text{exp}}, \text{active}\}$ in local databaseSend $\{CPID_{i,c}, \sigma_{S_j,i,c}, T_{i,c}^{\text{exp}}\}$ to V_i **Phase II: CPID Rotation****Input:** Vehicle V_i requests new CPID under same VEC S_j

/* Vehicle side */

 $sk_{i,c+1} \xleftarrow{\$} [1, n-1], \quad PK_{i,c+1} \leftarrow sk_{i,c+1} \cdot G$ $\sigma_{\text{rotate}} \leftarrow \text{Sign}_{sk_{i,c}}(H(SPID_{i,m} \parallel CPID_{i,c} \parallel PK_{i,c+1} \parallel T_{\text{curr}}))$ Send $\text{Req}_{\text{rotate}} \leftarrow \{SPID_{i,m}, CPID_{i,c}, PK_{i,c+1}, T_{\text{curr}}, \sigma_{\text{rotate}}\}$ to S_j

/* VEC side */

Retrieve $PK_{i,c}$ from local database for $CPID_{i,c}$ Verify σ_{rotate} using $PK_{i,c}$ **if verification fails then**└ Reject request and **abort** $\eta_{i,c+1} \xleftarrow{\$} \{0, 1\}^{128}$ $CPID_{i,c+1} \leftarrow H(SPID_{i,m} \parallel ID_{S_j} \parallel \eta_{i,c+1})$ $\sigma_{S_j,i,c+1} \leftarrow \text{Sign}_{sk_{S_j}}(H(CPID_{i,c+1} \parallel PK_{i,c+1} \parallel T_{i,c+1}^{\text{exp}}))$ Mark $CPID_{i,c}$ as **expired** in local databaseStore $\{CPID_{i,c+1}, PK_{i,c+1}, SPID_{i,m}, T_{i,c+1}^{\text{exp}}, \text{active}\}$ Send $\{CPID_{i,c+1}, \sigma_{S_j,i,c+1}, T_{i,c+1}^{\text{exp}}\}$ to V_i

Systemic Pseudonym Revocation

Each CA maintains the global reputation score R_i^{glob} for vehicles within its region, periodically updated based on the aggregated local reputation from VEC servers. If a vehicle's R_i^{glob} falls below the revocation threshold θ_{rev} , the vehicle is classified as malicious. Consequently, its SPID is marked as revoked on ChainPID, allowing VEC servers to verify its validity during authentication. Upon revocation, the CA may recover RID_{V_i} from its encrypted database for future investigations.

6.2.2 Reputation Management Under Different Pseudonyms

To achieve these aims, BRMPP employs a permissioned blockchain-based architecture.

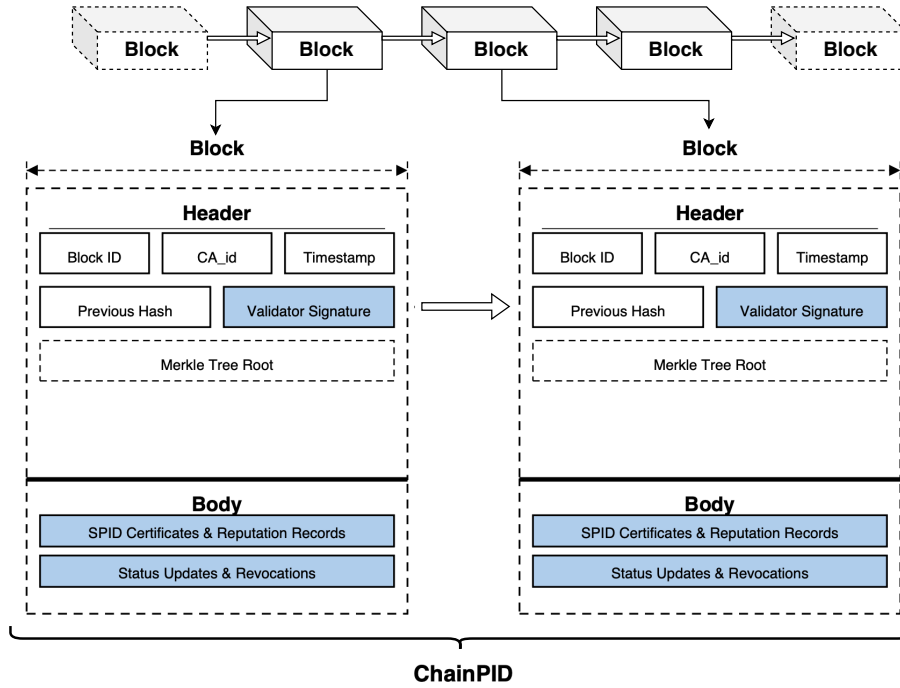


Figure 6.3: The block and blockchain structure of ChainPID

The ChainPID, as illustrated in Figure 6.3, is administered by the CAs, which possess both read and write privileges, and it records two critical categories of information: (i) the validity status of each SPID for authentication and revocation control, and (ii) the global reputation values of vehicles associated with valid SPIDs. ChainPID serves as the shared trust infrastructure enabling reputation continuity across CA regions. When a vehicle moves from one region to another, the receiving CA retrieves the vehicle's accumulated global reputation from ChainPID, ensuring

seamless trust transfer without requiring direct inter-CA communication. Access to the ChainPID layer is restricted; VEC servers possess read-only privileges, allowing them to retrieve global reputation data for informed local decision-making without compromising user privacy.

Furthermore, each VEC server computes the local reputation of every SPID within its coverage area and securely transmits these values to its regional CA. The CA then aggregates the local reputations reported by multiple VEC servers to derive the corresponding global reputation.

The following subsections provide a detailed description of the calculation and maintenance of local and global reputation values within the BRMPP framework.

Local Reputation Calculation

In the VEC-enabled architecture, vehicles interact with other vehicles using short-term CPIDs to preserve privacy. To capture the reputation, each VEC server maintains sequence of feedback events for every active $CPID_{i,c}$ within its coverage area. Let n denote the total number of feedback events received for $CPID_{i,c}$. For the f -th feedback, a signed feedback weight δ_f is defined, which encodes whether the observed behavior is considered trustworthy or malicious. The aging-weighted interaction value after n feedback events up to time step t is given by [242]:

$$r_{i,c}(n_c(t)) = \left(\sum_{f=1}^{n_c(t)} \gamma^{n_c(t)-f} \delta_f \right) \ln(N_{\text{peers}}(t)), \quad (6.16)$$

where $n_c(t)$ is the number of feedback events up to time step t for active $CPID_{i,c}$, $\gamma \in (0, 1)$ is an aging factor that discounts historical feedback, and $N_{\text{peers}}(t)$ denotes the number of distinct vehicles that have provided feedback for this pseudonym. The parameter δ_f is defined as:

$$\delta_f = \begin{cases} \delta_{\text{pos}}, & \text{for positive feedback,} \\ \delta_{\text{neg}}, & \text{for negative feedback,} \end{cases} \quad (6.17)$$

with $\delta_{\text{pos}} > 0$ and $\delta_{\text{neg}} < 0$.

The local reputation of $CPID_{i,c}$ after n feedback events is mapped to a normalized trust interval via a Gompertz function [242]:

$$R_{i,c}^{\text{loc}}(n_c(t)) = a \cdot e^{-b \cdot e^{-d \cdot r_{i,c}(n_c(t))}}, \quad (6.18)$$

where a , b , and d are configuration parameters controlling the upper asymptote, the displacement, and the growth rate, respectively. With $a = 1$, this formulation guarantees that the local reputation is normalized within the interval $[0, 1]$. It exhibits sigmoid growth, where trust increases gradually through consistent positive feedback, while ensuring a sharp decline when penalties are applied for detected misbehavior.

Each VEC server maintains a secure mapping from each vehicle's long-term $SPID_i$ to its currently active short-term $CPID_{i,c}$. Since only one CPID is active at any time step, the local reputation is:

$$R_i^{\text{loc}}(t) = R_{i,c}^{\text{loc}}(n_c(t)), \quad (6.19)$$

where $c(t)$ denotes the active CPID index at time step t . The VEC server periodically transmits $R_i^{\text{loc}}(t)$ to the corresponding CA via a secure channel.

Global Reputation Calculation

Upon receiving local reputation assessments, the CA calculates the global reputation for each active SPID and publishes the updated value to ChainPID.

Let $R_i^{\text{glob}}(t-1)$ denote the global reputation of vehicle V_i at time step $t-1$. Upon receiving $R_i^{\text{loc}}(t)$, the CA updates the global reputation using an asymmetric Exponentially Weighted Moving Average (EWMA) mechanism designed to enforce the hard-to-earn, easy-to-lose principle:

$$R_i^{\text{glob}}(t) = (1 - \beta)R_i^{\text{glob}}(t-1) + \beta R_i^{\text{loc}}(t). \quad (6.20)$$

To satisfy asymmetric trust requirements, the aggregation factor β is defined as:

$$\beta = \begin{cases} \beta_{\text{gain}}, & \text{if } R_i^{\text{loc}}(t) \geq R_i^{\text{glob}}(t-1) \\ \beta_{\text{punish}}, & \text{if } R_i^{\text{loc}}(t) < R_i^{\text{glob}}(t-1) \end{cases} \quad (6.21)$$

subject to $0 < \beta_{\text{gain}} < \beta_{\text{punish}} \leq 1$. A larger β_{punish} ensures malicious behavior leads to rapid trust decline, while a smaller β_{gain} ensures reputation recovery only through sustained honest participation.

6.3 Problem Formulation

In VEC networks, privacy preservation and trust management present inter-dependent requirements. While pseudonym rotation enhances vehicle privacy by

reducing linkability, the decision to grant such privacy protection should be dependent on the vehicle's reputation. This section formulates the privacy entropy metric that quantifies vehicle unlinkability, introduces the reputation discrepancy metric that captures behavioral inconsistencies across network regions, and defines a constrained optimization problem where pseudonym rotation is permitted only for vehicles whose global reputation exceeds a minimum threshold and whose local-global reputation discrepancy remains within an acceptable bound.

6.3.1 Privacy Entropy Formulation

The privacy entropy [287] for vehicle V_i operating under its c -th CPID quantifies the uncertainty faced by an external attacker attempting to track the vehicle. Let t denote the current time step and $t_{i,c}^{\text{start}}$ denote the time step when $CPID_{i,c}$ was issued. The pseudonym age is defined as:

$$\Delta t_{i,c} = t - t_{i,c}^{\text{start}}. \quad (6.22)$$

The privacy entropy for vehicle V_i operating under its c -th CPID at time t is formulated as:

$$H_i^{\text{priv}}(t) = -\log_2(P_{i,c}^{\text{track}}(t)) \cdot e^{-\lambda^{\text{priv}} \Delta t_{i,c}}, \quad (6.23)$$

where $\lambda^{\text{priv}} > 0$ is the privacy decay rate that captures the attacker's traffic analysis capability, and $P_{i,c}^{\text{track}}(t) \in (0, 1]$ is the attacker's tracking probability defined as [227]:

$$P_{i,c}^{\text{track}}(t) = \frac{\xi}{\xi + |K(t_i^{\text{last}})| \cdot (1 - \xi)}, \quad (6.24)$$

where $\xi \in [0.3, 1]$ represents the attacker's prior capability of identifying the target vehicle before observing network traffic, and $|K(t_i^{\text{last}})|$ denotes the number of vehicles that simultaneously rotated their CPIDs at the time of vehicle V_i 's most recent rotation, and t_i^{last} denotes time of last rotated for V_i .

6.3.2 Reputation Discrepancy Formulation

To detect spatially inconsistent behavior where a vehicle may act honestly within one VEC domain while behaving maliciously in others, the reputation discrepancy metric is introduced to measure the difference between local and global reputation values.

For vehicle V_i at time step t , the reputation discrepancy is defined as:

$$\Delta_i^{\text{rep}}(t) = \left| R_i^{\text{loc}}(t) - R_i^{\text{glob}}(t) \right|, \quad (6.25)$$

where $R_i^{\text{loc}}(t)$ is the local reputation from the current active CPID (Eq. (6.18)), and $R_i^{\text{glob}}(t)$ is the global reputation maintained by the CA (Eq. (6.20)).

Since both reputations are normalized to $(0, 1]$ via the Gompertz function and EWMA aggregation respectively, the discrepancy is inherently bounded:

$$\Delta_i^{\text{rep}}(t) \in [0, 1). \quad (6.26)$$

A high discrepancy $\Delta_i^{\text{rep}}(t)$ of roughly 1 indicates substantial behavioral divergence between local observation and global history, signaling potential malicious or regionally inconsistent behavior.

6.3.3 Optimization Problem Formulation

The trust-aware pseudonym management task is formulated as a sequential decision-making problem where the VEC server coordinates the CPID rotations of all N_j vehicles within its coverage area. The objective is to balance individual privacy needs against costs while strictly enforcing trust constraints. Time is divided into slots indexed by $t \in \{0, 1, \dots, T_{\max}\}$, where each slot corresponds to rotation decision epoch and $T_{\max}$ denotes the episode horizon, the maximum number of decision steps within one episode of the optimization problem. The utility function for vehicle V_i at time step t is defined as:

$$U_i(t) = \omega_p H_i^{\text{priv}}(t+1) - \omega_c C_i^{\text{cost}}(t), \quad (6.27)$$

where $H_i^{\text{priv}}(t+1)$ is the privacy entropy achieved after the rotation decision $a_i(t)$ is executed, $C_i^{\text{cost}}(t) = a_i(t) \in \{0, 1\}$ is the binary rotation cost, and $\omega_p, \omega_c > 0$ are weighting coefficients that govern the privacy-cost trade-off.

The VEC server seeks to learn an optimal decision policy that, at each time step t , produces the joint rotation decision $\mathbf{A}_j(t) = [a_1(t), \dots, a_{N_j}(t)]$ to maximize the cumulative system utility over the episode horizon ($T_{\max}$):

$$\max_{\{a_i(t)\}} \sum_{t=0}^{T_{\max}} \frac{1}{N_j} \sum_{i=1}^{N_j} U_i(t) \quad (6.28)$$

$$\text{s.t. C1: } a_i(t) \in \{0, 1\}, \quad \forall i, t, \quad (6.28a)$$

$$\text{C2: } R_i^{\text{glob}}(t) < \theta_{\text{trust}} \implies a_i(t) = 0, \quad \forall i, t, \quad (6.28b)$$

$$\text{C3: } \Delta_i^{\text{rep}}(t) > \Delta_{\text{max}} \implies a_i(t) = 0, \quad \forall i, t. \quad (6.28c)$$

where θ_{trust} is the trust threshold for global reputation, and Δ_{max} is the maximum allowable reputation discrepancy.

Constraint (6.28a) restricts the decision to a binary choice per vehicle for its pseudonym rotation. Constraints (6.28b) and (6.28c) enforce that vehicles with insufficient global reputation or inconsistent reputation history are strictly prohibited from rotating pseudonyms, thereby mitigating whitewashing attack and identity abuse.

Solving (6.28) directly is intractable for three reasons. First, the state space is continuous and high-dimensional, comprising the privacy entropy, reputation, and timing information of all vehicles. Second, the joint action space grows exponentially as 2^{N_j} with the number of vehicles, which classical methods cannot enumerate efficiently. Third, the transition dynamics governing vehicle mobility and traffic conditions are unknown and stochastic. These characteristics make classical optimization methods insufficiently adaptive, motivating the use of deep reinforcement learning to approximate the optimal solution, as developed in the following section.

6.4 Deep Reinforcement Learning Approach

To solve the optimization problem defined in (6.28) while satisfying constraints **C1–C3**, TAP-PPO has been proposed. This algorithm frames the decision process as an MDP with constraint enforcement through action masking, which guarantees that trust and discrepancy constraints are satisfied by construction at every time step.

Unlike traditional optimization methods that require complete knowledge of system dynamics, TAP-PPO enables the VEC server to learn directly from interactions with the vehicular environment. At each time step t , the VEC server observes the state $\mathbf{S}_j(t)$, executes an effective action $\tilde{\mathbf{A}}_j(t)$, and receives a reward signal based on the utility function $U_i(t)$.

6.4.1 MDP Formulation

The optimization problem in (6.28), is formulated as an MDP, in which the policy $\pi(\tilde{\mathbf{A}}_j(t)|\mathbf{S}_j(t))$ maps observed states to effective actions, and a discount factor $\gamma \in [0, 1)$ is introduced to ensure stable training over the multi-step decision horizon.

The objective becomes:

$$\max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{T_{max}} \gamma^t \frac{1}{N_j} \sum_{i=1}^{N_j} U_i(t) \mid \mathbf{S}_j(0) \right], \quad (6.29)$$

where the expectation accounts for the stochasticity in vehicle mobility, traffic dynamics, and policy sampling. Constraints (6.28b) and (6.28c) are enforced through action masking, as defined in the following.

State Space

At each time step t , the VEC server observes the state of each vehicle V_i within its coverage area. The per-vehicle state is defined as:

$$s_i(t) = \left[H_i^{\text{priv}}(t), R_i^{\text{glob}}(t), \Delta_i^{\text{rep}}(t), \Delta t_{i,c} \right], \quad (6.30)$$

where $H_i^{\text{priv}}(t)$ is the current privacy entropy of vehicle V_i , $R_i^{\text{glob}}(t)$ is the global reputation value, $\Delta_i^{\text{rep}}(t)$ represents the reputation discrepancy and $\Delta t_{i,c}$ denotes the time elapsed since the last pseudonym rotation for vehicle V_i . The aggregated state for VEC server S_j is the concatenation of individual vehicle states:

$$\mathbf{S}_j(t) = [s_1(t), s_2(t), \dots, s_{N_j}(t)]. \quad (6.31)$$

Action Space

The decision space for each vehicle V_i is binary:

$$a_i(t) \in \{0, 1\}, \quad (6.32)$$

where $a_i(t) = 1$ indicates a request to grant CPID rotation and $a_i(t) = 0$ indicates rotation is denied.

To satisfy constraints C2 and C3 defined in (6.28b) and (6.28c), an Action Masking mechanism is implemented that transforms the network's raw output a into the effective action $\tilde{a}$. This ensures that pseudonym changes are strictly blocked for

untrusted vehicles ($R_i^{\text{glob}}(t) < \theta_{\text{trust}}$) or those exhibiting suspicious reputation discrepancies ($\Delta_i^{\text{rep}}(t) > \Delta_{\text{max}}$).

The discrepancy mask validates whether vehicle V_i maintains consistency between its local and global reputations:

$$m_i^{\text{dis}}(t) = \begin{cases} 1, & \text{if } \Delta_i^{\text{rep}}(t) \leq \Delta_{\text{max}}, \\ 0, & \text{otherwise,} \end{cases} \quad (6.33)$$

where $\Delta_i^{\text{rep}}(t)$ is defined in (6.25) and Δ_{max} represents the maximum allowable discrepancy threshold. The reputation mask verifies if vehicle V_i meets the global trust requirements:

$$m_i^{\text{rep}}(t) = \begin{cases} 1, & \text{if } R_i^{\text{glob}}(t) \geq \theta_{\text{trust}}, \\ 0, & \text{otherwise,} \end{cases} \quad (6.34)$$

where $R_i^{\text{glob}}(t)$ denotes the global reputation and θ_{trust} is the predefined trust threshold.

The composite validity mask is derived as:

$$m_i(t) = m_i^{\text{dis}}(t) \cdot m_i^{\text{rep}}(t), \quad (6.35)$$

which evaluates to 1 if and only if the vehicle is both trusted and behaviorally consistent.

Finally, the effective action executed by the system is defined as:

$$\tilde{a}_i(t) = a_i(t) \cdot m_i(t) = \begin{cases} a_i(t), & \text{if } m_i(t) = 1, \\ 0, & \text{if } m_i(t) = 0. \end{cases} \quad (6.36)$$

Consequently, when $m_i(t) = 0$, the rotation is overridden to 0 (denied) regardless of the agent's intended action $a_i(t)$.

Table 6.1 summarizes the trust-aware action masking mechanism. At each step t , the RL agent proposes a raw action $a_i(t) \in \{0, 1\}$ for every vehicle in the zone. The mask $m_i(t)$ filters these decisions, producing the effective action $\tilde{a}_i(t) = a_i(t) \cdot m_i(t)$. Consequently, only vehicles that are both trusted and selected by the agent undergo CPID rotation (row 4), while untrusted vehicles are blocked regardless of the agent's decision (row 2). Although excluding untrusted vehicles from rotation reduces the effective anonymity set size, it results in significant cost savings by avoiding unnecessary rotations. This prevents the agent from wasting rotation resources on

Table 6.1: Action Masking Outcomes for Vehicle V_i at Time t

$m_i(t)$	$a_i(t)$	$\tilde{a}_i(t)$	Outcome
0	0	0	Malicious, not selected, no rotation
0	1	0	Malicious, selected but blocked by mask
1	0	0	Trusted, not selected, no rotation
1	1	1	Trusted and selected, rotation executed

potentially malicious vehicles, ensuring that every executed rotation contributes to the privacy of trusted participants.

The joint effective action vector for VEC server S_j is:

$$\tilde{\mathbf{A}}_j(t) = [\tilde{a}_1(t), \tilde{a}_2(t), \dots, \tilde{a}_{N_j}(t)]. \quad (6.37)$$

In addition, the effective anonymity set size is computed as $|Z(t)| = \sum_{i=1}^{N_j} \tilde{a}_i(t)$. This count determines the tracking probability $P_{i,c}^{\text{track}}(t)$ in Eq. (6.24) for all vehicles that rotate at step t , directly linking the agent's rotation decisions to the resulting privacy entropy $H_i^{\text{priv}}(t+1)$.

Reward Function

The reward function is designed to directly maximize the system utility formulated in (6.27). At each time step t , the reward $r(t)$ is the average utility across all N_j vehicles in the coverage area:

$$r(t) = \frac{1}{N_j} \sum_{i=1}^{N_j} U_i(t) = \omega_p \bar{H}^{\text{priv}}(t) - \omega_c \bar{C}^{\text{cost}}(t), \quad (6.38)$$

where $\omega_p, \omega_c > 0$ are the weighting coefficients. The term $\bar{H}^{\text{priv}}(t)$ represents the average privacy level of the vehicles in coverage area, and $\bar{C}^{\text{cost}}(t)$ represents the normalized system load.

Privacy Component ($\bar{H}^{\text{priv}}$): Consistent with the objective in (6.28), the network-wide average privacy entropy across all trusted vehicles is:

$$\bar{H}^{\text{priv}}(t) = \frac{1}{|N_j^{\text{trusted}}|} \sum_{i \in N_j^{\text{trusted}}} H_i^{\text{priv}}(t+1), \quad (6.39)$$

where $N_j^{\text{trusted}} \subseteq N_j$ and $N_j^{\text{trusted}} = \{V_i : m_i(t) = 1\}$ denotes the set of trusted vehicles at time t .

Cost Component ($\bar{C}^{\text{cost}}$): The cost term captures the resource consumption (blockchain and communication overhead) associated with pseudonym changes. Since the effective action $\tilde{a}_i(t) \in \{0, 1\}$ is binary, the average cost reflects the percentage of vehicles rotating simultaneously:

$$\bar{C}^{\text{cost}}(t) = \frac{1}{N_j} \sum_{i=1}^{N_j} \tilde{a}_i(t). \quad (6.40)$$

A value of $\bar{C}^{\text{cost}} = 1.0$ implies maximum system load (all vehicles rotating), while 0.0 implies minimal overhead.

Trade-off Analysis: The ratio $\frac{\omega_c}{\omega_p}$ defines the *Marginal Rate of Substitution* (MRS) for the system. It quantifies the minimum privacy entropy (H) the system must maintain to justify the cost of a pseudonym change. The impact of different cost scenarios (varying ω_c) on agent behavior is investigated in Section 6.5.

Actor-Critic Architecture

An actor-critic architecture is employed where both networks utilize a shared feature extraction backbone. This design is crucial for handling the variable number of vehicles N_j within the VEC server's coverage while maintaining permutation invariance—ensuring that the ordering of vehicles in the input list does not affect the policy outcome.

State Encoding Given the aggregated state $\mathbf{S}_j(t)$ containing N_j vehicle states, each per-vehicle state is first encoded through a shared encoder network:

$$\mathbf{z}_i(t) = f_{\text{enc}}(s_i(t); \boldsymbol{\theta}_{\text{enc}}), \quad (6.41)$$

where $f_{\text{enc}}(\cdot)$ is a multi-layer perceptron (MLP) with parameters $\boldsymbol{\theta}_{\text{enc}}$, and $\mathbf{z}_i(t) \in \mathbb{R}^{d_z}$ is the latent representation of vehicle V_i 's state. For notational convenience, the combined actor parameters are denoted as $\boldsymbol{\theta} = \{\boldsymbol{\theta}_{\text{enc}}, \boldsymbol{\theta}_{\text{actor}}\}$, which are jointly optimized during training. The global context representation is obtained by aggregating individual encodings:

$$\mathbf{z}_{\text{glob}}(t) = \frac{1}{N_j} \sum_{i=1}^{N_j} \mathbf{z}_i(t), \quad (6.42)$$

which captures the overall state of vehicles within the VEC server's coverage area.

Actor Network The actor network π_{θ} generates a stochastic policy, mapping the vehicle state encodings to action probabilities. To capture both local requirements and global system constraints, the network takes the concatenated local and global features as input. For vehicle V_i , the raw rotation probability is computed as:

$$p_i(t) = \sigma(f_{\text{actor}}([\mathbf{z}_i(t), \mathbf{z}_{\text{glob}}(t)]; \boldsymbol{\theta}_{\text{actor}})), \quad (6.43)$$

where $\sigma(\cdot)$ is the sigmoid activation function, $[\cdot, \cdot]$ denotes vector concatenation, and $\boldsymbol{\theta}_{\text{actor}}$ represents the trainable weights of the actor MLP.

The base policy for vehicle V_i is modeled as a Bernoulli distribution:

$$\pi_{\theta}(a_i | s_i(t)) = p_i(t)^{a_i} \cdot (1 - p_i(t))^{1-a_i}. \quad (6.44)$$

To enforce trust and consistency constraints, a masked policy is defined as $\tilde{\pi}$. When the validity mask $m_i(t) = 0$, the policy is forced to a deterministic denial ($a_i = 0$):

$$\tilde{\pi}_{\theta}(a_i | s_i(t), m_i(t)) = \begin{cases} \pi_{\theta}(a_i | s_i(t)), & \text{if } m_i(t) = 1, \\ \mathbb{I}(a_i = 0), & \text{if } m_i(t) = 0, \end{cases} \quad (6.45)$$

where $\mathbb{I}(\cdot)$ is the indicator function (1 if the condition holds, 0 otherwise).

Assuming conditional independence given the state, the joint policy for all N_j vehicles managed by server S_j factorizes as:

$$\tilde{\pi}_{\theta}(\tilde{\mathbf{A}}_j(t) | \mathbf{S}_j(t)) = \prod_{i=1}^{N_j} \tilde{\pi}_{\theta}(\tilde{a}_i(t) | s_i(t), m_i(t)). \quad (6.46)$$

Critic Network The critic network V_{ϕ} estimates the state-value function, representing the expected cumulative discounted reward from the current aggregated state $\mathbf{S}_j(t)$. It operates solely on the global context embedding to evaluate the system-level performance:

$$V_{\phi}(\mathbf{S}_j(t)) = f_{\text{critic}}(\mathbf{z}_{\text{glob}}(t); \boldsymbol{\phi}), \quad (6.47)$$

where $f_{\text{critic}}(\cdot)$ is an MLP with parameters $\boldsymbol{\phi}$ that maps the global context to a scalar value baseline.

To reduce variance in the policy gradient estimates, the advantage function is computed using Generalized Advantage Estimation (GAE). The advantage $\hat{A}(t)$ at time step t is defined as:

$$\hat{A}(t) = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \delta(t+l), \quad (6.48)$$

where $\gamma \in [0, 1)$ is the discount factor, $\lambda \in [0, 1]$ is the GAE smoothing parameter, and $\delta(t)$ denotes the Temporal Difference (TD) error:

$$\delta(t) = r(t) + \gamma V_\phi(\mathbf{S}_j(t+1)) - V_\phi(\mathbf{S}_j(t)). \quad (6.49)$$

PPO Training Objective

The PPO algorithm optimizes the policy using a clipped surrogate objective to ensure stable updates. The probability ratio between the new and old masked policies is defined as:

$$\rho(t) = \frac{\tilde{\pi}_\theta(\tilde{\mathbf{A}}_j(t)|\mathbf{S}_j(t))}{\tilde{\pi}_{\theta_{\text{old}}}(\tilde{\mathbf{A}}_j(t)|\mathbf{S}_j(t))}, \quad (6.50)$$

where $\tilde{\mathbf{A}}_j(t)$ is the joint effective action vector and θ_{old} denotes the policy parameters before the update.

The clipped surrogate objective is formulated to penalize large policy updates:

$$\mathcal{L}^{\text{clip}}(\theta) = \mathbb{E}_t \left[\min \left(\rho(t) \hat{A}(t), \text{clip}(\rho(t), 1 - \epsilon, 1 + \epsilon) \hat{A}(t) \right) \right], \quad (6.51)$$

where ϵ is the clipping hyperparameter and $\hat{A}(t)$ is the advantage estimate defined in (6.48).

To ensure the critic accurately predicts expected returns, the value function loss minimizes the mean squared error (MSE) between the estimated value and the target returns:

$$\mathcal{L}^{\text{value}}(\phi) = \mathbb{E}_t \left[(V_\phi(\mathbf{S}_j(t)) - V_{\text{target}}(t))^2 \right], \quad (6.52)$$

where $V_{\text{target}}(t) = \hat{A}(t) + V_{\phi_{\text{old}}}(\mathbf{S}_j(t))$ represents the target return.

To prevent premature convergence to sub-optimal deterministic policies, an entropy bonus is incorporated based on the masked policy distribution:

$$\mathcal{L}^{\text{ent}}(\theta) = \mathbb{E}_t \left[\frac{1}{N_j} \sum_{i=1}^{N_j} H(\tilde{\pi}_\theta(\cdot|s_i(t), m_i(t))) \right], \quad (6.53)$$

where $H(\cdot)$ denotes the entropy of the effective policy distribution. When the action mask $m_i(t) = 0$, the masked policy reduces to a deterministic distribution, and its entropy therefore becomes zero:

$$H(\tilde{\pi}_{\theta}(\cdot \mid s_i(t), m_i(t))) = 0. \quad (6.54)$$

Consequently, the entropy bonus exclusively encourages exploration among eligible vehicles, preventing the agent from wasting exploratory capacity on trust-constrained decisions. Given the shared feature extraction architecture, a joint objective function is optimized that simultaneously improves the policy and the value estimation:

$$\mathcal{L}^{\text{total}} = \mathcal{L}^{\text{clip}}(\theta) - c_v \mathcal{L}^{\text{value}}(\phi) + c_e \mathcal{L}^{\text{ent}}(\theta), \quad (6.55)$$

where $c_v, c_e > 0$ are coefficients balancing value function fitting and entropy regularization, respectively.

The network parameters (including the shared encoder) are updated via gradient ascent to maximize $\mathcal{L}^{\text{total}}$:

$$\{\theta, \phi\} \leftarrow \{\theta, \phi\} + \alpha \nabla_{\{\theta, \phi\}} \mathcal{L}^{\text{total}}, \quad (6.56)$$

where α is the learning rate.

Algorithm 6.I summarizes the TAP-PPO training procedure. The training runs for E_{max} episodes, each consisting of T_{max} time steps. At each time step, the VEC agent encodes individual vehicle states through the shared encoder, applies the validity masks to enforce trust constraints via the masked policy $\tilde{\pi}_{\theta}$ defined in Eq. (6.45), and executes the resulting joint action. The collected transitions, including the log-probabilities required for importance sampling, are stored in buffer $\mathcal{D}$. The policy parameters are updated every T_{up} steps using the accumulated experience, where $T_{\text{up}} \ll T_{\text{max}}$ is the rollout length that controls the update frequency within each episode. Moreover, within each update, the policy is refined over K_{opt} optimization epochs using mini-batches of size B sampled from the buffer, following the standard PPO multi-epoch update scheme.

6.5 Simulation Results and Discussions

This section presents the simulation-based evaluation of the proposed algorithm in VEC-IoV networks.

Algorithm 6.I: TAP-PPO: Trust-Aware Pseudonym Management

Input: Max episodes $E_{\max}$, episode horizon $T_{\max}$, rollout length T_{up} , batch size B , optimization epochs K_{opt} , clipping parameter ϵ , learning rate α .

Initialize: Actor parameters $\theta = \{\theta_{\text{enc}}, \theta_{\text{actor}}\}$, Critic parameters ϕ , Buffer $\mathcal{D} \leftarrow \emptyset$;

for episode $e = 1$ **to** $E_{\max}$ **do**

 Reset environment, observe $\mathbf{S}_j(0)$;

for time step $t = 0$ **to** $T_{\max}$ **do**

/ Phase 1: State Encoding & Masked Action Selection */*

for each vehicle V_i in coverage **do**

 Observe per-vehicle state $s_i(t)$ via Eq. (6.30);

 Encode: $\mathbf{z}_i(t) \leftarrow f_{\text{enc}}(s_i(t); \theta_{\text{enc}})$ via Eq. (6.41);

 Compute masks $m_i^{\text{dis}}(t)$, $m_i^{\text{rep}}(t)$ via Eq. (6.33), (6.34);

 Compute composite mask $m_i(t)$ via Eq. (6.35);

 Compute global context $\mathbf{z}_{\text{glob}}(t)$ via Eq. (6.42);

for each vehicle V_i in coverage **do**

 Compute $p_i(t)$ from $[\mathbf{z}_i(t), \mathbf{z}_{\text{glob}}(t)]$ via Eq. (6.43);

 Sample $\tilde{a}_i(t) \sim \tilde{\pi}_{\theta}(\cdot | s_i(t), m_i(t))$ via Eq. (6.45);

 Record $\log \tilde{\pi}_{\theta}(\tilde{a}_i(t) | s_i(t), m_i(t))$;

/ Phase 2: Environment Interaction */*

 Execute joint action $\tilde{\mathbf{A}}_j(t)$, observe reward $r(t)$ via Eq. (6.38) and next state $\mathbf{S}_j(t+1)$;

 Compute value estimate $V_{\phi}(\mathbf{S}_j(t))$ via Eq. (6.47);

 Store $(\mathbf{S}_j(t), \tilde{\mathbf{A}}_j(t), r(t), \mathbf{S}_j(t+1), \log \tilde{\pi}_{\theta}, V_{\phi}(\mathbf{S}_j(t)))$ in $\mathcal{D}$;

/ Phase 3: PPO Policy Update */*

if $|\mathcal{D}| \geq T_{\text{up}}$ **then**

$\{\theta_{\text{old}}, \phi_{\text{old}}\} \leftarrow \{\theta, \phi\}$;

 Compute TD errors $\delta(t)$ via Eq. (6.49);

 Compute GAE advantages $\hat{A}(t)$ via Eq. (6.48) and returns $V_{\text{target}}(t) = \hat{A}(t) + V_{\phi_{\text{old}}}(\mathbf{S}_j(t))$;

for epoch $k = 1$ **to** K_{opt} **do**

 Sample mini-batch of size B from $\mathcal{D}$;

 Compute $\mathcal{L}^{\text{total}}$ via Eq. (6.55);

 Update $\{\theta, \phi\}$ via Eq. (6.56);

 Clear buffer: $\mathcal{D} \leftarrow \emptyset$;

6.5.1 Simulation Setup

The proposed TAP-PPO algorithm is implemented using Python 3.10 and the PyTorch deep learning library. The evaluation is conducted within a simulated

VEC-enabled IoV environment, where pseudonym changes are managed during active data-sharing sessions. To ensure mobility, the environment is modeled using the SUMO suite. A synthetic urban grid network covering an area of approximately 5.6 km^2 is constructed to represent a region governed by a single CA entity. Within the CA region, 10 RSUs are strategically deployed to function as VEC servers, each with a coverage radius of approximately 560 m and a coverage area of approximately 1 km^2 . Vehicles are initialized at random entry points and follow dynamic trajectories across multiple VEC coverage zones. To evaluate the robustness of the system across diverse operational conditions, traffic density is varied from 20 to 100 vehicles/ km^2 , representing scenarios from sparse to highly congested urban traffic. Each vehicle maintains an average speed of 30–60km/h depending on traffic density and road conditions. Each decision time step corresponds to $\Delta t = 1$ second of simulation time. The simulation environment incorporates a variable ratio of malicious vehicles (0% to 50%) to test the Action Masking layer. Malicious vehicles execute two attack strategies: bad-mouthing attacks, where false negative reputation feedback is provided about honest vehicles; and on-off attacks, where vehicles alternate between honest and malicious behavior to avoid detection. Table 6.2 summarizes the key simulation settings in our proposed approach.

To comprehensively evaluate the performance of TAP-PPO, it is compared against the following baseline and state-of-the-art approaches:

- **Random Pseudonym Rotation:** At each decision step, each vehicle independently rotates its pseudonym with a fixed probability $p_{\text{rot}} = 0.5$, regardless of its privacy entropy, rotation cost, or reputation status. It is referred to as *Random* in the experimental results presented in this chapter.
- **Periodic with fixed-interval Pseudonym Change:** This baseline follows a pre-defined time-based strategy with a fixed rotation interval of $N_{\text{rot}} = 5$ decision steps, where all vehicles update their pseudonyms at regular intervals regardless of traffic conditions, vehicle trustworthiness, or network state. This baseline (denoted *Periodic* hereafter) represents a fixed-rule strategy.
- **Standard PPO:** This variant implements the same reward function as TAP-PPO, incorporating privacy entropy and rotation cost, but excludes the action masking layer that enforces reputation constraints. All vehicles are permitted to rotate pseudonyms regardless of their reputation scores, allowing us to isolate the impact of trust-aware action.

Table 6.2: Simulation Settings

Category	Parameter	Value
Training Config	Max Episodes ($E_{\max}$)	2000
	Episode Horizon ($T_{\max}$)	500 steps
	Rollout Length (T_{up})	128 steps
	Optimization Epochs (K_{opt})	10
	Mini-batch Size (B)	64
	Learning Rate (α)	1×10^{-3}
PPO Parameters	Discount Factor (γ)	0.99
	GAE Parameter (λ)	0.95
	Clipping Parameter (ϵ)	0.2
	Value Loss Coeff. (c_v)	0.5
	Entropy Coeff. (c_e)	0.01
Network Architecture	Encoder Hidden Dim (d_z)	64
	Actor Hidden Layers	[64, 64]
	Critic Hidden Layers	[64, 64]
Reward Weights	Privacy Weight (ω_p)	1.0
	Cost Weight (ω_c)	0.3
Privacy Model	Attacker Capability (ξ)	0.3
	Privacy Decay Rate (λ^{priv})	0.01
Constraints	Trust Threshold (θ_{trust})	0.5
	Max Discrepancy ($\Delta_{\max}$)	0.35

- Dynamic Pseudonym Swap Program (DPSP)[227]: DPSP dynamically selects pseudonym swap areas based on traffic density, road topology, and vehicle mobility patterns. In DPSP the threshold used for swap zone activation where the The minimum swap group threshold is set to 6 vehicles. Unlike TAP-PPO, DPSP does not consider rotation cost or vehicle reputation in its decision-making process and lacks adaptive learning capabilities. TAP-PPO compares against DPSP specifically in terms of average privacy entropy versus the number of vehicles, as shown in Figure 6.6, where the evaluation context and metrics are most closely aligned.
- Cooperative Pseudonym Change based on Number of Neighbors (CPN) [288]: CPN is a distributed, neighbor-based cooperative approach where vehicles coordinate pseudonym changes based on the density of nearby vehicles to create mix zones. CPN has a neighbor count threshold which is set to 3 in this simulation. It operates without adaptive learning mechanisms, relying solely on local neighbor discovery through V2V communication.

6.5.2 Results and Analysis

Figure 6.4 illustrates the convergence behavior of TAP-PPO compared with three baselines: Standard PPO, Periodic, and Random approaches. All algorithms are evaluated over 10^6 environment steps. TAP-PPO achieves the highest average episode reward of approximately 3250 and converges within 2.5×10^5 steps. Standard PPO converges to approximately 2900, requiring 4.5×10^5 steps. This demonstrates a 12.1% reward improvement and 44.4% faster convergence for TAP-PPO. The improvement stems from two factors. First, by masking invalid actions for untrusted vehicles, TAP-PPO reduces the effective exploration space from 2^{N_j} to $2^{N_j^{\text{trusted}}}$, enabling faster policy learning. Second, Standard PPO wastes rotation budget on malicious vehicles that contribute minimally to system-wide privacy entropy while still incurring the full rotation cost; TAP-PPO eliminates this inefficiency by construction, allowing it to allocate rotations exclusively to honest vehicles where the privacy gain is meaningful. The non-learning baselines remain flat throughout training. The Periodic strategy achieves approximately 1400, while Random achieves approximately 900. TAP-PPO outperforms the Periodic and Random baselines by 132.1% and 261.1%, respectively, demonstrating that adaptive, state-dependent rotation decisions substantially outperform static policies.

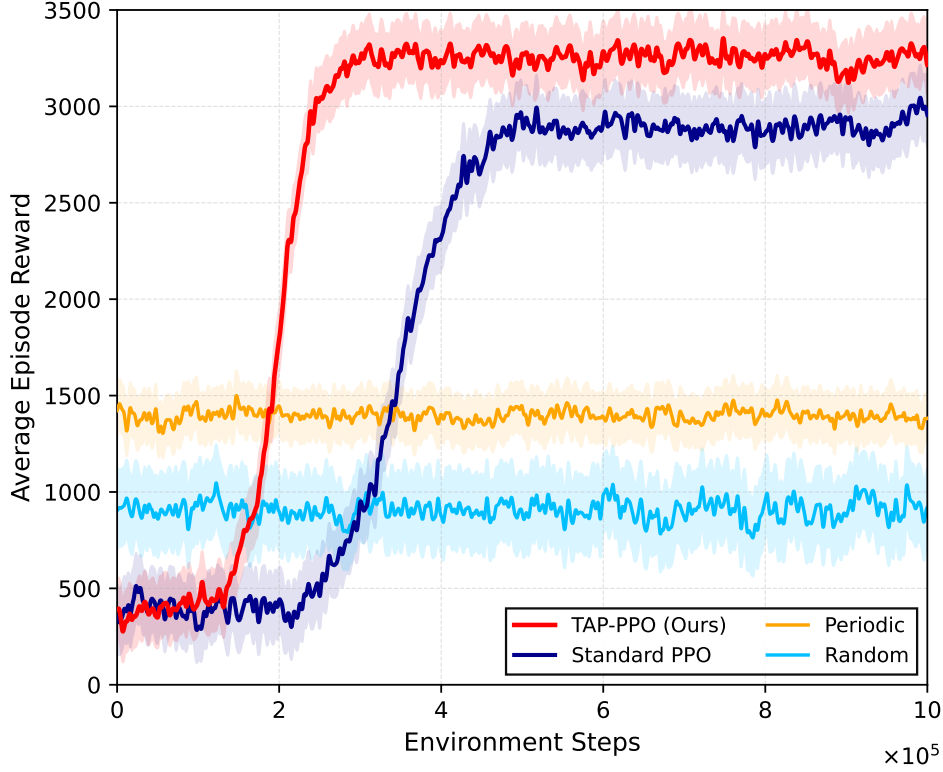


Figure 6.4: Average reward under different methods

Figure 6.5 analyzes TAP-PPO under three learning rates: $\alpha \in \{10^{-2}, 10^{-3}, 10^{-4}\}$.

The selected learning rate $\alpha = 1 \times 10^{-3}$ achieves the highest final reward of approximately 3250 with stable convergence within 2.5×10^5 steps, representing the trade-off between learning speed and policy stability. A higher learning rate ($\alpha = 1 \times 10^{-2}$) exhibits the fastest initial rise, converging within approximately 1.2×10^5 steps. However, the large gradient steps cause the policy to overshoot optimal action probabilities, resulting in persistent oscillation and a substantially lower final reward of approximately 2400, a 35.4% degradation compared to $\alpha = 10^{-3}$. The wider confidence band further confirms that $\alpha = 10^{-2}$ produces inconsistent performance across episodes, as the PPO clipping mechanism alone is insufficient to stabilize updates of this magnitude. A lower learning rate ($\alpha = 1 \times 10^{-4}$) yields the most stable learning trajectory but converges slowly, requiring over 6×10^5 steps to reach approximately 2900. This represents a 140% increase in convergence time, with $\alpha = 10^{-3}$ outperforming it by 12.1% in final reward. In dynamic vehicular environments where traffic conditions evolve rapidly, such slow adaptation would limit the system's ability to respond to changing privacy requirements. These results validate $\alpha =$

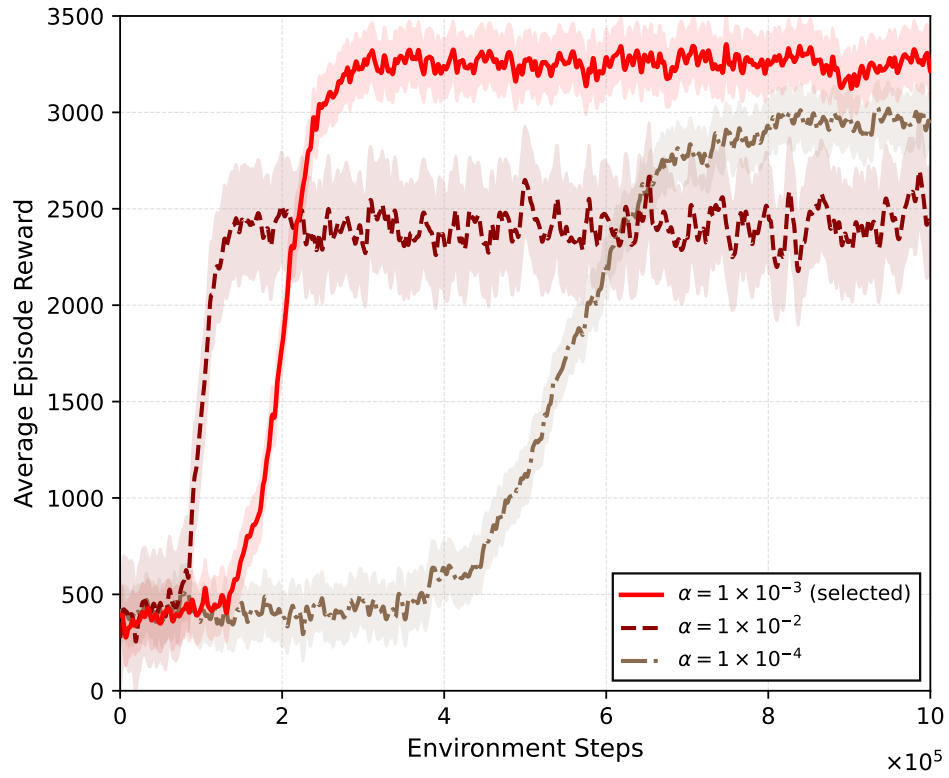


Figure 6.5: Average episode reward under different learning rates

1×10^{-3} as the selected learning rate, achieving the highest reward with practical convergence speed.

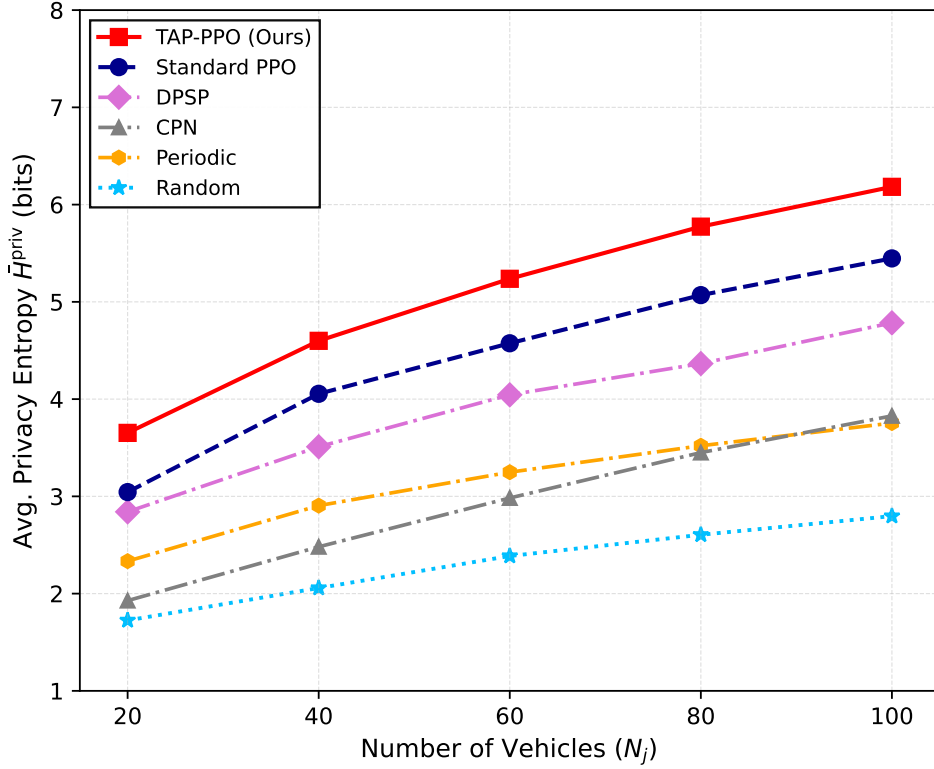


Figure 6.6: Average privacy entropy versus number of vehicles

Figure 6.6 evaluates the average privacy entropy as a function of the number of vehicles participating in coordinated pseudonym rotation within a VEC coverage area. TAP-PPO is compared against five baselines and state-of-the-art approaches: Standard PPO (without trust-aware action masking), DPSP [227], CPN [288], Periodic, and Random rotation. The simulation assumes 30% of vehicles exhibit malicious behavior to evaluate the robustness of the action masking mechanism under adversarial conditions. Results are averaged over 50 simulation runs with 95% confidence intervals. TAP-PPO achieves the highest privacy entropy across all vehicle densities, reaching approximately 6.2 bits at $N_j = 100$ vehicles. Standard PPO achieves 5.4 bits at the same density, representing a 14.8% improvement by TAP-PPO. This performance gain confirms that trust-aware action masking effectively eliminates wasteful rotations on malicious vehicles. Among the rule-based state-of-the-art approaches, DPSP achieves the highest entropy at 4.80 bits with 100 vehicles, which TAP-PPO outperforms by 29.2% due to static privacy mechanism. CPN reaches 3.85 bits, which TAP-PPO outperforms by 61.0%. Periodic and Random baselines achieve 3.80 and 2.80 bits, which TAP-PPO outperforms by 63.2% and 121.4%, respectively.

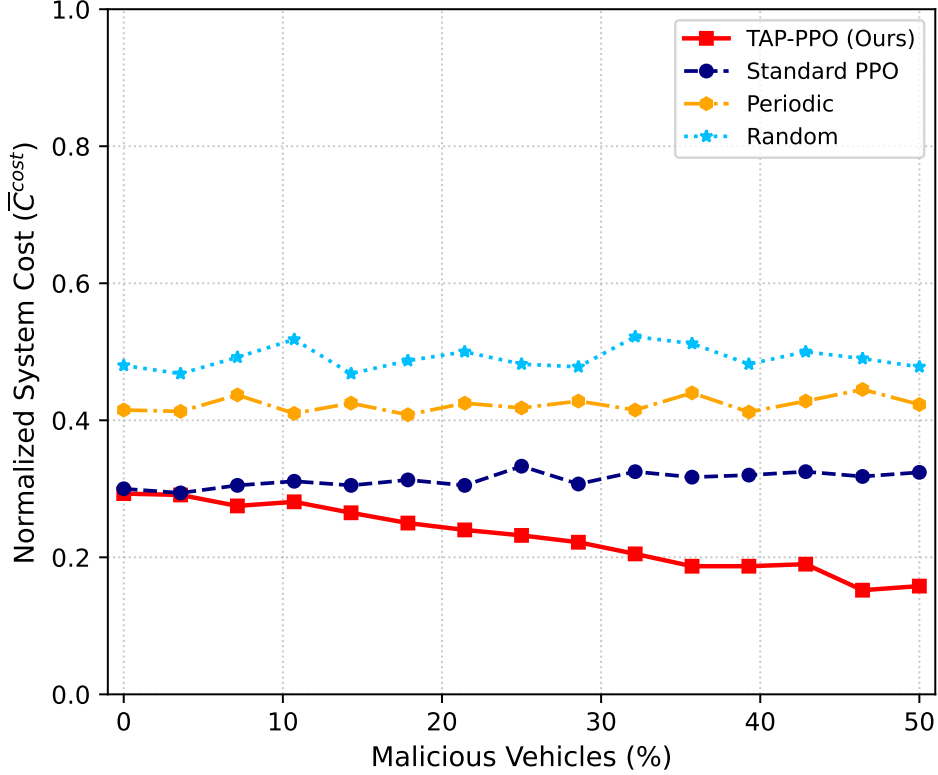


Figure 6.7: Cost Analysis across Malicious Percentages

Figure 6.7 depicts the comparative analysis of the normalized system cost, $\bar{C}^{\text{cost}}$, as the percentage of malicious vehicles in the network increases from 0% to 50%. With the reward function weight ratio configured at $\omega_c/\omega_p = 0.3$, the learning objective prioritizes privacy maximization over cost minimization. The Random and Periodic baseline algorithms exhibit the highest operational costs across all malicious ratios due to their lack of adaptive feedback mechanisms. Specifically, Random rotation maintains a high cost of approximately 0.50, which is 257% higher than TAP-PPO at a 50% malicious ratio, while Periodic rotation fluctuates near 0.42, or 200% higher than the proposed approach. In contrast, both RL-based approaches demonstrate significantly higher efficiency, initiating at a shared cost baseline of approximately 0.31 under no-attack conditions (0% malicious). This convergence confirms that both Standard PPO and TAP-PPO successfully learn to minimize expenses when trust constraints are inactive. However, at a 50% malicious vehicle ratio, the cost for Standard PPO remains relatively stagnant near 0.31 because it cannot differentiate node trustworthiness. TAP-PPO significantly reduces these expenditures to approximately 0.14, achieving a 121.4% cost reduction compared to

Standard PPO. This performance divergence proves that trust-aware action masking effectively eliminates wasteful rotations among untrusted vehicles.

Figure 6.8 examines how the marginal rate of substitution $\frac{\omega_c}{\omega_p}$ affects system behavior under a fixed configuration of 100 vehicles with 30% malicious ratio. The MRS defines the minimum entropy gain required to justify an additional rotation, offering a tunable parameter for the privacy-cost trade-off.

Panel (a) in Figure 6.8 reveals that TAP-PPO consistently outperforms Standard PPO in average privacy entropy across all MRS values. At an MRS of 0.1, TAP-PPO achieves approximately 6.2 bits compared to approximately 5.2 bits for Standard PPO, representing a 19.2% improvement attributable to trust-aware masking. Conversely, Periodic and Random rotations remain relatively constant at approximately 3.9 bits and 2.7 bits respectively, failing to adapt to operator preferences.

Panel (b) in Figure 6.8 highlights the cost advantage of action masking; at low MRS values, TAP-PPO incurs a cost of approximately 0.25 compared to Standard PPO's 0.33, achieving a 32.0% cost reduction by eliminating wasteful rotations on untrusted vehicles. As the MRS increases, both reinforcement learning agents successfully reduce costs toward zero, while Periodic and Random baselines maintain fixed expenditures near 0.40 and 0.48, respectively. These results confirm that TAP-PPO provides an effective policy lever to smoothly trade privacy for cost savings, a capability absent in non-adaptive baselines.

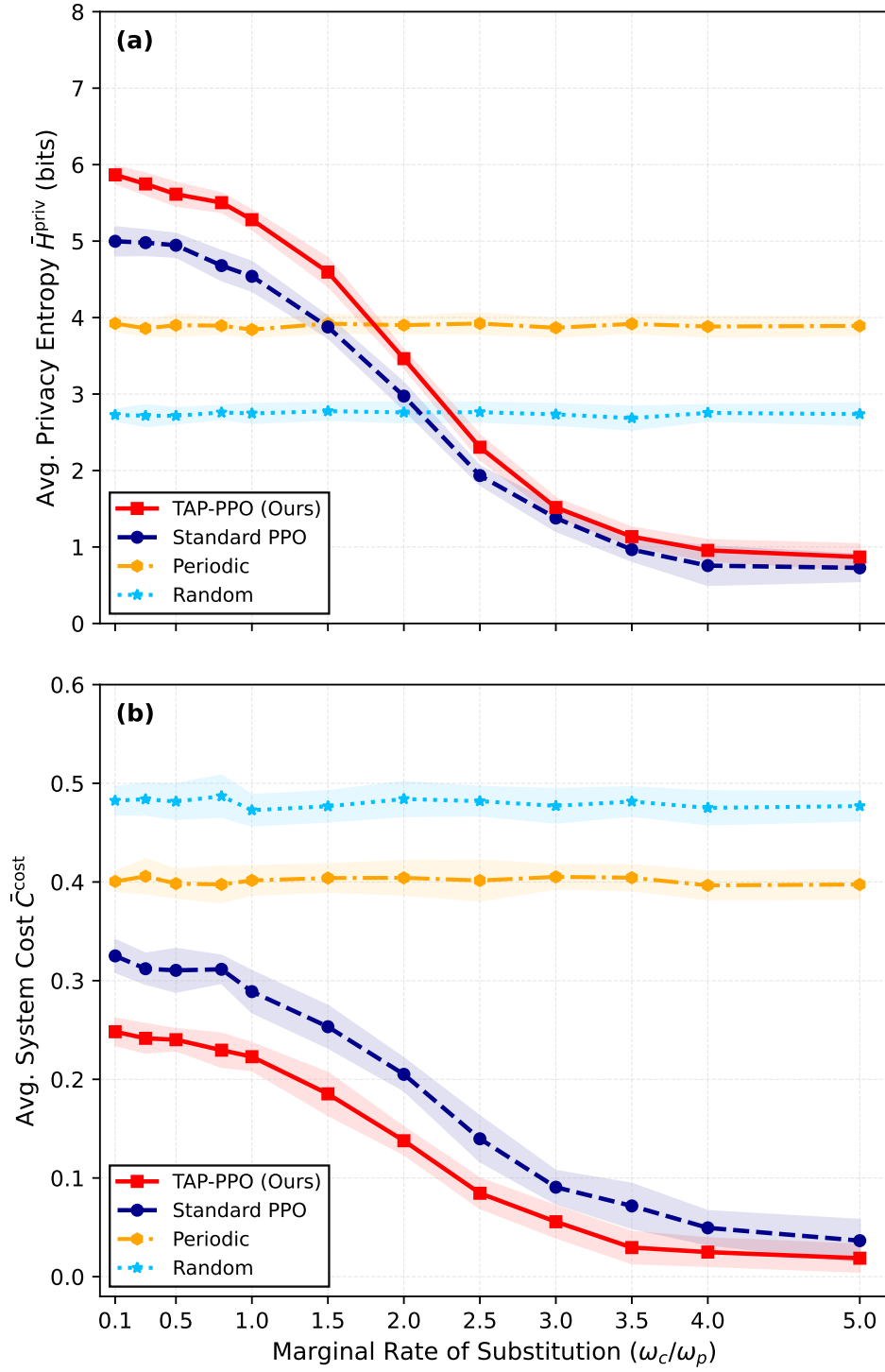


Figure 6.8: Sensitivity to Privacy-Cost Weighting

6.6 Conclusions

This chapter has addressed **RQ2b**, which seeks a design model that effectively integrates blockchain technology with RL strategies to improve privacy and trust simultaneously in vehicular networks. In response, the proposed BRMPP framework integrates a permissioned blockchain with a multi-agent RL strategy to jointly address reputation continuity and trust-aware pseudonym management across pseudonym rotations in VEC-enabled IoV environments. Building on the trust management mechanisms developed in Chapter 4 and the blockchain-based pseudonym infrastructure introduced in Chapter 5, BRMPP employs a hierarchical pseudonym structure with SPIDs and CPIDs to decouple long-term trust management from short-term privacy protection. The introduced permissioned blockchain, ChainPID, maintains immutable pseudonym identity records and reputation scores, enabling reputation continuity as vehicles move across CA regions. The proposed TAP-PPO algorithm uses the two-level asymmetric reputation mechanism to enforce trust constraints through action masking, ensuring that only reputable vehicles are granted pseudonym rotations while adaptively balancing privacy entropy against rotation cost. Extensive simulations demonstrated that TAP-PPO outperforms privacy entropy over all baselines and state-of-the-art approaches, with improvements ranging from 14.8% over standard PPO to 121.4% over random rotation. Furthermore, the Random baseline incurs 257% higher pseudonym rotation cost than TAP-PPO under 50% malicious vehicle conditions, confirming the effectiveness of integrating trust-aware decision-making into pseudonym management.

Chapter 7

Conclusion

7.1 Contributions: Overview & Limitations

This thesis has investigated blockchain-enabled RL methods for advancing privacy preservation and trust management in vehicular networks. Motivated by the rapid expansion of the IoV and the inherent challenges posed by high mobility, dynamic network topologies, and adversarial behavior, this research has integrated RL and blockchain technologies to develop a suite of algorithms and frameworks that address privacy and trust challenges. In particular, the work directly addresses two fundamental research questions. The first research question (RQ1) examines the key challenges in ensuring security, privacy, and trust within vehicular networks (RQ1a), and investigates how RL can be tailored to address trust management challenges in this context (RQ1b). The second research question (RQ2) explores what design model would effectively integrate blockchain with RL strategies to improve privacy (RQ2a), and what design model would effectively integrate blockchain with RL strategies to improve privacy and trust simultaneously (RQ2b).

By formulating trust management as an RL-driven link-level and vehicle-level decision problem, the thesis has developed algorithms that adaptively evaluate the trustworthiness of communication paths and data providers in vehicular environments. Additionally, the privacy challenges inherent in pseudonym management have been tackled through blockchain-enabled frameworks and multi-agent RL algorithms that jointly enhance privacy entropy while considering the cost through social welfare optimization under dynamic network conditions. Building on these foundations, a joint privacy-and-trust framework has been designed in which blockchain-verified reputation scores are incorporated into the RL policy space through a trust-aware action-masking mechanism, thereby preserving reputation continuity across

pseudonym rotations. Although security is not a primary target of optimization, the proposed contributions indirectly contribute to more secure vehicular networks by defending against routing-layer attacks, enforcing decentralized authentication of pseudonym identities, and providing tamper-evident records of reputation history. This progression from trust to privacy, and finally to their joint provisioning, demonstrates that proactive, learning-based mechanisms supported by blockchain infrastructure are capable of adapting to adversarial behavior in ways that static, rule-based approaches cannot.

Chapter 4 introduced two RL-based approaches that enhance trustworthiness in vehicular networks from two distinct perspectives. This was in response to RQ1b, namely how RL can be tailored to address trust management challenges in vehicular networks. The first approach, QL-TRT, is a link-centric approach in which the trustworthiness and quality of service of directed links between neighboring vehicles are evaluated through packet forwarding ratio, residual energy ratio, and expected transmission time. Q-learning is employed to learn the resulting link trust values and to select the most reliable forwarding paths from source to destination, thereby defending VANETs against black-hole and gray-hole attacks that compromise routing-layer availability. The second approach, DSTS-DDPG, is a vehicle-centric approach in which edge servers dynamically select a credibility score threshold for selecting trustworthy data providers during data sharing in VEC. Together, these approaches form a layered trust-aware architecture in which QL-TRT secures the communication path at the link level and DSTS-DDPG ensures the credibility of the participating vehicles at the edge level.

Limitations: While QL-TRT performs effectively in moderate-density VANETs, the tabular Q-table representation limits scalability to large-scale, high-density networks. DSTS-DDPG mitigates this through a deep function approximation, although the training cost grows as the number of vehicles and interactions increases.

Chapter 5 presents the BPMV framework, addressing RQ2a through a blockchain-enabled pseudonym management system integrated with a multi-agent RL algorithm for privacy preservation in VEC-IoV. BPMV employs a hierarchical pseudonym structure based on SPIDs and CPIDs, and introduces decentralized protocols for authentication and revocation. CAPM-MAPPO, a novel multi-agent PPO algorithm, governs dynamic CPID updates by optimizing a social welfare function that jointly incorporates vehicle-side and VEC-side costs with privacy entropy, thereby enabling adaptive, cost-aware pseudonym rotation under dynamic network conditions. Simulation-based evaluations show that CAPM-MAPPO outperforms base-

lines in convergence, cost efficiency, and social welfare, while achieving higher privacy entropy compared to the state-of-the-art method across diverse vehicular densities.

Limitations: The CAPM-MAPPO algorithm, as a MADRL algorithm, is limited to a fixed number of VEC servers acting as agents; however, by incorporating permutation-invariant mechanisms, the algorithm can be extended to dynamically adapt to variable VEC server populations in real-world deployments.

Chapter 6 presents the BRMPP framework, addressing RQ2b through a blockchain-based reputation management system integrated with a trust-aware RL algorithm for joint privacy preservation and trust management in VEC-IoV. BRMPP employs a hierarchical pseudonym structure comprising SPIDs for long-term reputation management and CPIDs for short-term privacy-preserving interactions, enabling reputation continuity across CA regions. A permissioned blockchain is integrated for immutable pseudonym identity and reputation management, and a two-level reputation mechanism is employed in which local reputation is assessed at VEC servers and aggregated into a global reputation at the CA level. TAP-PPO, a novel PPO-based algorithm, governs CPID rotation decisions by formulating a constrained utility maximization problem that balances privacy entropy against rotation cost, and employs a trust-aware action-masking mechanism that enforces reputation constraints within the policy space. Simulation results in a SUMO-based urban environment show that TAP-PPO consistently outperforms baseline and state-of-the-art approaches in privacy entropy across varying vehicle densities.

Limitations: The permissioned blockchain relies on CAs as consensus nodes that are geographically dispersed across different administrative regions, which may introduce latency and coordination overhead in deployments involving cross-region reputation updates; however, the framework can be extended to support such deployments through hierarchical consensus mechanisms that reduce cross-region synchronization latency.

7.2 Future Work

Beyond the contributions presented in this thesis, several directions emerge as promising avenues for future research. Some of these directions arise directly from the limitations identified in the individual contribution chapters, while others reflect broader research opportunities in the area of privacy-preserving and trust-aware vehicular networks. These directions are outlined below.

1. Both the BPMV and BRMPP frameworks rely on permissioned blockchain architectures operated by VEC servers and certificate authorities. Given that permissionless blockchains offer greater decentralization and resilience to single points of failure, future efforts should focus on incorporating public blockchain architectures into vehicular privacy and trust frameworks. Because access-control-based confidentiality does not extend to globally readable ledgers, such a migration requires combining public blockchains with cryptographic techniques such as commitments, zero-knowledge proofs, and selectively decryptable encryption schemes to preserve the privacy of pseudonym and reputation records. In addition, addressing the inherent scalability and latency challenges of public blockchains further necessitates exploring advanced techniques such as DRL-driven intelligent sharding and cross-shard coordination protocols to efficiently manage reputation and pseudonym management operations in vehicular network deployments.
2. Although the BPMV and BRMPP frameworks already address pseudonym-based privacy preservation, future extensions should focus on strengthening the cryptographic and administrative layers of pseudonym management. Promising directions include exploring privacy-preserving authentication methods based on zero-knowledge proofs, which would enable vehicles to prove possession of valid credentials without revealing identifying information, and developing distributed revocation mechanisms that efficiently propagate revocation decisions across CAs serving different administrative regions. These enhancements would further strengthen user identity privacy, reduce reliance on trusted authorities, and ensure compliance with emerging data protection regulations in vehicular environments.
3. The RL-based rotation policies in CAPM-MAPPO and TAP-PPO currently operate on per-vehicle decisions made independently by each VEC server. Future work should explore extending the decision space from per-vehicle to group-level rotation, in which a VEC server synchronizes rotation decisions across groups of vehicles within its coverage area to maximize the anonymity set at the moment of rotation. A further extension is inter-VEC coordination across boundary regions, in which neighboring VEC servers jointly negotiate rotation decisions for vehicles in overlapping coverage areas through explicit communication or a lightweight consensus protocol. Such coordination mech-

anisms would enhance the unlinkability guarantees of pseudonym rotation and improve robustness under complex deployment topologies.

4. Multi-agent deep reinforcement learning algorithms, including the CAPM-MAPPO algorithm, rely on a fixed number of agents during training and evaluation. Vehicular network deployments inherently display a dynamic number of VEC servers, as infrastructure density and operational availability fluctuate under real-world conditions. Future work should therefore focus on developing more adaptive MADRL architectures that accommodate this variability. In particular, permutation-invariant approaches such as graph neural networks and attention-based mechanisms can be employed to dynamically adjust to varying agent counts, enhancing the robustness and scalability of RL-based privacy management algorithms in large-scale vehicular deployments.
5. Future work should explore the integration of large language models (LLMs) with RL for privacy and trust management in vehicular networks. LLM-guided RL frameworks could provide high-level contextual guidance to the policies developed in this thesis, improving sample efficiency and generalization to unseen deployment scenarios. At the same time, the deployment of LLMs in vehicular networks raises new privacy concerns, including data leakage during LLM offloading to edge infrastructure and identity exposure through query patterns, which future research should address.

7.3 Final Remarks

This thesis has established a comprehensive foundation for addressing the privacy and trust challenges that arise in vehicular networks as they evolve into large-scale, dynamic IoV environments. By integrating RL and blockchain technologies, the proposed algorithms and frameworks provide effective mechanisms for enabling trustworthy interactions among vehicles and VEC servers, preserving vehicle identity and location privacy, and sustaining reputation continuity across pseudonym rotations. Together, the contributions offer practical approaches for VEC-IoV deployments and are designed to scale with the evolving demands of next-generation ITS. As emerging technologies such as LLMs become integrated into the connected vehicular ecosystem, the proactive, learning-based mechanisms presented in this thesis are well-positioned to support the next generation of privacy-preserving and trust-aware vehicular services. The research directions outlined for future work are

expected to extend these foundations further, strengthening decentralization, scalability, and robustness in increasingly complex vehicular environments.

7.4 Acknowledgment

This work was funded by Taighde Éireann – Research Ireland through the Research Ireland Centre for Research Training in Machine Learning (18/CRT/6183) and 13/RC/2106_P2 (ADAPT Centre). This work was also supported by the NEXT project funded from the European Commission’s Erasmus+ Programme call Capacity building in the field of higher education (ERASMUS-EDU-2023-CBHE) under Project ID 101129022.

Bibliography

- [1] M. Chowdhury and K. Dey, “Intelligent transportation systems-a frontier for breaking boundaries of traditional academic engineering disciplines [education],” *IEEE Intelligent Transportation Systems Magazine*, vol. 8, no. 1, pp. 4–8, 2016. DOI: 10.1109/MITS.2015.2503199.
- [2] E. Mohammadzadeh Mianji, G.-M. Muntean, and I. Tal, “Enhancing vehicular network security, privacy, and trust through reinforcement learning: A comprehensive survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 26, no. 12, pp. 21 393–21 426, 2025. DOI: 10.1109/TITS.2025.3612202.
- [3] R. Das, L. Zhou, S. Bi, T. Wang, and T. Hou, “Towards the safety of intelligent transportation: A survey on the security challenges and mitigations in internet of vehicles (ioV),” *ACM Trans. Sen. Netw.*, vol. 22, no. 3, Apr. 2026, ISSN: 1550-4859. DOI: 10.1145/3786592. [Online]. Available: <https://doi.org/10.1145/3786592>.
- [4] SAE International. “SAE Updates J3016 Levels of Automated Driving Graphic to Reflect Evolving Standard.” [Online]. Available: <https://www.sae.org/news/2019/01/sae-updates-j3016-automated-driving-graphic>.
- [5] V. S. R. Kosuru and A. K. Venkitaraman, “Advancements and challenges in achieving fully autonomous self-driving vehicles,” *World Journal of Advanced Research and Reviews*, vol. 18, no. 1, pp. 161–167, 2023.
- [6] G. Bendiab, A. Hameurlaine, G. Germanos, N. Kolokotronis, and S. Shiaeles, “Autonomous vehicles security: Challenges and solutions using blockchain and artificial intelligence,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 4, pp. 3614–3637, 2023. DOI: 10.1109/TITS.2023.3236274.
- [7] Statista. “Projected sales of autonomous vehicles worldwide from 2019 to 2030.” [Online]. Available: <https://www.statista.com/statistics/1230733/projected-sales-autonomous-vehicles-worldwide/>.
- [8] Statista. “Global autonomous car market size,” Accessed: Apr. 21, 2026. [Online]. Available: <https://www.statista.com/statistics/428692/projected-size-of-global-autonomous-vehicle-market-by-vehicle-type/>.

- [9] Statista. “Connected cars worldwide.” Accessed: April 21, 2026, Accessed: Apr. 21, 2026. [Online]. Available: <https://www.statista.com/topics/1918/connected-cars/>.
- [10] Statista. “Size of the global autonomous car market in 2021, with a forecast through 2026. ”[Online]. Available: <https://www.statista.com/statistics/428692/projected-size-of-global-autonomous-vehicle-market-by-vehicle-type/>.
- [11] C. Nobles, D. N. Burrell, S. L. Burton, and T. Waller, “Driving into cybersecurity trouble with autonomous vehicles,” in *Handbook of Research on Cybersecurity Risk in Contemporary Business Systems*, IGI Global, 2023, pp. 255–273.
- [12] Z. Lu, G. Qu, and Z. Liu, “A survey on recent advances in vehicular network security, trust, and privacy,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 760–776, 2018.
- [13] S. S. Shinde, A. Bozorgchenani, D. Tarchi, and Q. Ni, “On the design of federated learning in latency and energy constrained computation offloading operations in vehicular edge computing systems,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 2, pp. 2041–2057, 2022. DOI: 10.1109/TVT.2021.3135332.
- [14] Y. Ju et al., “Joint secure offloading and resource allocation for vehicular edge computing network: A multi-agent deep reinforcement learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 5, pp. 5555–5569, 2023. DOI: 10.1109/TITS.2023.3242997.
- [15] H. Xiao, L. Cai, J. Feng, Q. Pei, and W. Shi, “Resource optimization of mab-based reputation management for data trading in vehicular edge computing,” *IEEE Transactions on Wireless Communications*, vol. 22, no. 8, pp. 5278–5290, 2023. DOI: 10.1109/TWC.2022.3233035.
- [16] X. Zhang, W. Wu, Z. Zhao, J. Wang, and S. Liu, “Rmddqn-learning: Computation offloading algorithm based on dynamic adaptive multi-objective reinforcement learning in internet of vehicles,” *IEEE Transactions on Vehicular Technology*, pp. 1–16, 2023. DOI: 10.1109/TVT.2023.3270967.
- [17] Y. Chen et al., “An entropy-based privacy-preserving federated deep reinforcement learning framework for task offloading in vehicular edge computing networks,” *IEEE Internet of Things Journal*, vol. 13, no. 5, pp. 8011–8024, 2026. DOI: 10.1109/JIOT.2025.3602798.
- [18] J. Kang et al., “Blockchain for secure and efficient data sharing in vehicular edge computing and networks,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4660–4670, 2019. DOI: 10.1109/JIOT.2018.2875542.

- [19] Z. Yang, K. Yang, L. Lei, K. Zheng, and V. C. M. Leung, "Blockchain-based decentralized trust management in vehicular networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1495–1505, 2019. DOI: 10.1109/JIOT.2018.2836144.
- [20] S. Bao et al., "Pseudonym management through blockchain: Cost-efficient privacy preservation on intelligent transportation systems," *IEEE Access*, vol. 7, pp. 80 390–80 403, 2019. DOI: 10.1109/ACCESS.2019.2921605.
- [21] N. Aung, T. Kechadi, T. Zhu, S. Zerdoumi, T. Guerbouz, and S. Dhelim, "Blockchain application on the internet of vehicles (iov)," in *2022 IEEE 7th International Conference on Intelligent Transportation Engineering (ICITE)*, 2022, pp. 586–591. DOI: 10.1109/ICITE56321.2022.10101404.
- [22] Q. Xu, L. Zhang, Y. Liu, and Z. Li, "Enhancing trust management system for connected autonomous vehicles using machine learning methods: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 27, no. 4, pp. 3862–3893, 2026. DOI: 10.1109/TITS.2025.3647284.
- [23] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.
- [24] Y. Xu, M. Lei, M. Li, M. Zhao, and B. Hu, "A new anti-jamming strategy based on deep reinforcement learning for manet," in *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, IEEE, 2019, pp. 1–5.
- [25] D. Zhang, F. R. Yu, R. Yang, and L. Zhu, "Software-defined vehicular networks with trust management: A deep reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1400–1414, 2022. DOI: 10.1109/TITS.2020.3025684.
- [26] D. Zhang, F. R. Yu, and R. Yang, "A machine learning approach for software-defined vehicular ad hoc networks with trust management," in *2018 IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6. DOI: 10.1109/GLOCOM.2018.8647426.
- [27] Q. Wu, Y. Zhao, Q. Fan, P. Fan, J. Wang, and C. Zhang, "Mobility-aware cooperative caching in vehicular edge computing based on asynchronous federated and deep reinforcement learning," *IEEE Journal of Selected Topics in Signal Processing*, vol. 17, no. 1, pp. 66–81, 2023. DOI: 10.1109/JSTSP.2022.3221271.
- [28] H. H. R. Sherazi, R. Iqbal, F. Ahmad, Z. A. Khan, and M. H. Chaudary, "Ddos attack detection: A key enabler for sustainable communication in internet of vehicles," *Sustainable Computing: Informatics and Systems*, vol. 23, pp. 13–20, 2019.
- [29] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, "Vehicular ad hoc networks (vanets): Status, results, and challenges," *Telecommunication Systems*, vol. 50, no. 4, pp. 217–241, 2012.

- [30] M. A. Al-Shareeda, M. Anbar, I. H. Hasbullah, and S. Manickam, "Survey of authentication and privacy schemes in vehicular ad hoc networks," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 2422–2433, 2021. DOI: 10.1109/JSEN.2020.3021731.
- [31] P. K. Singh, S. K. Nandi, and S. Nandi, "A tutorial survey on vehicular communication state of the art, and future research directions," *Vehicular Communications*, vol. 18, p. 100 164, 2019.
- [32] R. Gasmi and M. Aliouat, "Vehicular ad hoc networks versus internet of vehicles - a comparative view," in *2019 International Conference on Networking and Advanced Systems (ICNAS)*, 2019, pp. 1–6. DOI: 10.1109/ICNAS.2019.8807870.
- [33] M. Fardad, G.-M. Muntean, and I. Tal, "Latency-aware v2x operation mode coordination in vehicular network slicing," in *IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, 2023, pp. 1–6. DOI: 10.1109/VTC2023-Spring57618.2023.10200069.
- [34] M. Fardad, E. M. Mianji, G.-M. Muntean, and I. Tal, "A fast and effective graph-based resource allocation and power control scheme in vehicular network slicing," in *2022 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 2022, pp. 1–6. DOI: 10.1109/BMSB55706.2022.9828750.
- [35] O. Kaiwartya et al., "Internet of vehicles: Motivation, layered architecture, network model, challenges, and future aspects," *IEEE Access*, vol. 4, pp. 5356–5373, 2016. DOI: 10.1109/ACCESS.2016.2603219.
- [36] D. Jiang and L. Delgrossi, "Ieee 802.11p: Towards an international standard for wireless access in vehicular environments," in *VTC Spring 2008 - IEEE Vehicular Technology Conference*, 2008, pp. 2036–2040. DOI: 10.1109/VETECS.2008.458.
- [37] S. Chen et al., "Vehicle-to-everything (v2x) services supported by lte-based systems and 5g," *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 70–76, 2017. DOI: 10.1109/MCOMSTD.2017.1700015.
- [38] J. B. Kenney, "Dedicated short-range communications (dsrc) standards in the united states," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011. DOI: 10.1109/JPROC.2011.2132790.
- [39] A. Festag, "Cooperative intelligent transport systems standards in europe," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 166–172, 2014. DOI: 10.1109/MCOM.2014.6979970.
- [40] R. Sedar, C. Kalalas, F. Vázquez-Gallego, L. Alonso, and J. Alonso-Zarate, "A comprehensive survey of v2x cybersecurity mechanisms and future research paths," *IEEE Open Journal of the Communications Society*, vol. 4, pp. 325–391, 2023. DOI: 10.1109/OJCOMS.2023.3239115.

- [41] L. Silva et al., “Computing paradigms in emerging vehicular environments: A review,” *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 3, pp. 491–511, 2021. DOI: 10.1109/JAS.2021.1003862.
- [42] W. Chen, X. Qiu, T. Cai, H.-N. Dai, Z. Zheng, and Y. Zhang, “Deep reinforcement learning for internet of things: A comprehensive survey,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1659–1692, 2021. DOI: 10.1109/COMST.2021.3073036.
- [43] N. Nomikos, S. Zoupanos, T. Charalambous, and I. Krikidis, “A survey on reinforcement learning-aided caching in heterogeneous mobile edge networks,” *IEEE Access*, vol. 10, pp. 4380–4413, 2022. DOI: 10.1109/ACCESS.2022.3140719.
- [44] L. Bréhon-Grataloup, R. Kacimi, and A.-L. Beylot, “Mobile edge computing for v2x architectures and applications: A survey,” *Computer Networks*, vol. 206, p. 108797, 2022.
- [45] J. Liu et al., “RI/drl meets vehicular task offloading using edge and vehicular cloudlet: A survey,” *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8315–8338, 2022. DOI: 10.1109/JIOT.2022.3155667.
- [46] L. Hou, M. A. Gregory, and S. Li, “A survey of multi-access edge computing and vehicular networking,” *IEEE Access*, vol. 10, pp. 123436–123451, 2022. DOI: 10.1109/ACCESS.2022.3224032.
- [47] B. Mao, J. Liu, Y. Wu, and N. Kato, “Security and privacy on 6g network edge: A survey,” *IEEE Communications Surveys & Tutorials*, vol. 25, no. 2, pp. 1095–1127, 2023. DOI: 10.1109/COMST.2023.3244674.
- [48] M. Fardad, G.-M. Muntean, and I. Tal, “Decentralized vehicular edge computing framework for energy-efficient task coordination,” in *2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring)*, 2024, pp. 1–7. DOI: 10.1109/VTC2024-Spring62846.2024.10683387.
- [49] P. Agbaje, A. Anjum, A. Mitra, E. Oseghale, G. Bloom, and H. Olufowobi, “Survey of interoperability challenges in the internet of vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 22838–22861, 2022. DOI: 10.1109/TITS.2022.3194413.
- [50] Y. Yu, L. Guo, Y. Liu, J. Zheng, and Y. Zong, “An efficient sdn-based ddos attack detection and rapid response platform in vehicular networks,” *IEEE Access*, vol. 6, pp. 44570–44579, 2018. DOI: 10.1109/ACCESS.2018.2854567.
- [51] M. Kim, I. Oh, K. Yim, M. Sahlabadi, and Z. Shukur, “Security of 6g-enabled vehicle-to-everything communication in emerging federated learning and blockchain technologies,” *IEEE Access*, vol. 12, pp. 33972–34001, 2024. DOI: 10.1109/ACCESS.2023.3348409.

- [52] S. I. Boucetta and Z. C. Johanyák, “Survey on security attacks in software defined vanets,” in *IEEE 16th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, 2022, pp. 185–190. DOI: 10.1109/SACI55618.2022.9919595.
- [53] H. Menouar, I. Guvenc, K. Akkaya, A. S. Uluagac, A. Kadri, and A. Tuncer, “Uav-enabled intelligent transportation systems for the smart city: Applications and challenges,” *IEEE Communications Magazine*, vol. 55, no. 3, pp. 22–28, 2017. DOI: 10.1109/MCOM.2017.1600238CM.
- [54] R. Khan, P. Kumar, D. N. K. Jayakody, and M. Liyanage, “A survey on security and privacy of 5g technologies: Potential solutions, recent advancements, and future directions,” *IEEE Communications Surveys & Tutorials*, vol. 22, no. 1, pp. 196–248, 2020. DOI: 10.1109/COMST.2019.2933899.
- [55] T. Alladi, V. Chamola, N. Sahu, V. Venkatesh, A. Goyal, and M. Guizani, “A comprehensive survey on the applications of blockchain for securing vehicular networks,” *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1212–1239, 2022. DOI: 10.1109/COMST.2022.3160925.
- [56] N. H. Hussein, C. T. Yaw, S. P. Koh, S. K. Tiong, and K. H. Chong, “A comprehensive survey on vehicular networking: Communications, applications, challenges, and upcoming research directions,” *IEEE Access*, vol. 10, pp. 86 127–86 180, 2022.
- [57] A. Talpur and M. Gurusamy, “Machine learning for security in vehicular networks: A comprehensive survey,” *IEEE Communications Surveys & Tutorials*, vol. 24, no. 1, pp. 346–379, 2021.
- [58] H. Hasrouny, A. E. Samhat, C. Bassil, and A. Laouiti, “Vanet security challenges and solutions: A survey,” *Vehicular Communications*, vol. 7, pp. 7–20, 2017.
- [59] M. Raya and J.-P. Hubaux, “Securing vehicular ad hoc networks,” *Journal of computer security*, vol. 15, no. 1, pp. 39–68, 2007.
- [60] P. Bagga, A. K. Das, M. Wazid, J. J. P. C. Rodrigues, and Y. Park, “Authentication protocols in internet of vehicles: Taxonomy, analysis, and challenges,” *IEEE Access*, vol. 8, pp. 54 314–54 344, 2020. DOI: 10.1109/ACCESS.2020.2981397.
- [61] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, “Vehicular ad hoc networks (vanets): Status, results, and challenges,” *Telecommunication Systems*, vol. 50, no. 4, pp. 217–241, 2012.
- [62] F. Sakiz and S. Sen, “A survey of attacks and detection mechanisms on intelligent transportation systems: Vanets and iov,” *Ad Hoc Networks*, vol. 61, pp. 33–50, 2017.
- [63] R. Minhas and M. Tilal, “Effects of jamming on ieee 802.11 p systems,” M.S. thesis, 2010.

- [64] S. B. Hakim, M. Adil, A. Ali, A. Farouk, and H. H. Song, "Internet of vehicles security threats, countermeasures, open challenges with future research directions," *IEEE Internet of Things Journal*, vol. 12, no. 22, pp. 46 347–46 374, 2025. DOI: 10.1109/JIOT.2025.3605066.
- [65] M. S. Al-Kahtani, "Survey on security attacks in vehicular ad hoc networks (vanets)," in *2012 6th international conference on signal processing and communication systems*, IEEE, 2012, pp. 1–9.
- [66] M. Azees, P. Vijayakumar, and L. Jegatha Deborah, "Comprehensive survey on security services in vehicular ad-hoc networks," *IET Intelligent Transport Systems*, vol. 10, no. 6, pp. 379–388, 2016.
- [67] E. Khezri, H. Hassanzadeh, R. O. Yahya, and M. Mir, "Security challenges in internet of vehicles (ioV) for ITS: A survey," *Tsinghua Science and Technology*, vol. 30, no. 4, pp. 1700–1723, 2025. DOI: 10.26599/TST.2024.9010083.
- [68] J. Grover, V. Laxmi, and M. S. Gaur, "Sybil attack detection in vanet using neighbouring vehicles," *International Journal of Security and Networks*, vol. 9, no. 4, pp. 222–233, 2014.
- [69] Z. Zhou, A. Gaurav, B. B. Gupta, M. D. Lytras, and I. Razzak, "A fine-grained access control and security approach for intelligent vehicular transport in 6G communication system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9726–9735, 2022. DOI: 10.1109/TITS.2021.3106825.
- [70] P. Ranaweera, A. D. Jurcut, and M. Liyanage, "Survey on multi-access edge computing security and privacy," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1078–1124, 2021. DOI: 10.1109/COMST.2021.3062546.
- [71] G. Manogaran, B. S. Rawal, V. Saravanan, M. K. Priyan, Q. Xin, and P. Shakeel, "Token-based authorization and authentication for secure Internet of Vehicles communication," *ACM Trans. Internet Technol.*, vol. 22, no. 4, pp. 1–20, 2023.
- [72] A. Dhamgaye and N. Chavhan, "Survey on security challenges in vanet," *International Journal of Computer Science and Network*, vol. 2, 2013.
- [73] F. Ahmad, A. Adnane, V. N. Franqueira, F. Kurugollu, and L. Liu, "Man-in-the-middle attacks in vehicular ad-hoc networks: Evaluating the impact of attackers' strategies," *Sensors*, vol. 18, no. 11, p. 4040, 2018.
- [74] F. Ahmad, F. Kurugollu, A. Adnane, R. Hussain, and F. Hussain, "Marine: Man-in-the-middle attack resistant trust model in connected vehicles," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3310–3322, 2020. DOI: 10.1109/JIOT.2020.2967568.
- [75] N.-W. Lo and H.-C. Tsai, "Illusion attack on vanet applications—a message plausibility problem," in *2007 IEEE Globecom Workshops*, IEEE, 2007, pp. 1–8.

- [76] J. Li, H. Lu, and M. Guizani, “Acpn: A novel authentication framework with conditional privacy-preservation and non-repudiation for vanets,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 938–948, 2015. DOI: 10.1109/TPDS.2014.2308215.
- [77] D. Manivannan, S. S. Moni, and S. Zeadally, “Secure authentication and privacy-preserving techniques in vehicular ad-hoc networks (vanets),” *Vehicular Communications*, vol. 25, p. 100 247, 2020.
- [78] C. Lai, R. Lu, D. Zheng, and X. Shen, “Security and privacy challenges in 5g-enabled vehicular networks,” *IEEE Network*, vol. 34, no. 2, pp. 37–45, 2020. DOI: 10.1109/MNET.001.1900220.
- [79] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, “Caravan: Providing location privacy for vanet,” *Proceedings of Embedded Security in Cars (ESCAR)*, vol. 8, 2005.
- [80] A. Haddaji, S. Ayed, and L. C. Fourati, “Artificial intelligence techniques to mitigate cyber-attacks within vehicular networks: Survey,” *Computers and Electrical Engineering*, vol. 104, p. 108 460, 2022.
- [81] H. El-Sayed, M. Chaqfeh, H. El-Kassabi, M. A. Serhani, and H. Alexander, “Trust enforcement in vehicular networks: Challenges and opportunities,” *IET Wireless Sensor Systems*, vol. 9, no. 5, pp. 237–246, 2019.
- [82] “Ieee standard for wireless access in vehicular environments—security services for application and management messages,” *IEEE Std 1609.2-2022 (Revision of IEEE Std 1609.2-2016)*, pp. 1–349, 2023. DOI: 10.1109/IEEESTD.2023.10075082.
- [83] “Ieee standard for wireless access in vehicular environments (wave) - certificate management interfaces for end entities,” *IEEE Std 1609.2.1-2022 (Revision of IEEE Std 1609.2.1-2020)*, pp. 1–261, 2022. DOI: 10.1109/IEEESTD.2022.9810154.
- [84] B. Brecht et al., “A security credential management system for v2x communications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 12, pp. 3850–3871, 2018. DOI: 10.1109/TITS.2018.2797529.
- [85] *Intelligent transport systems (ITS); security; ITS communications security architecture and security management*, 2018.
- [86] *Intelligent transport systems (ITS); security; security header and certificate formats*, 2021.
- [87] *Intelligent transport systems (ITS); security; pre-standardization study on pseudonym change management*, 2025.
- [88] Y. Xin et al., “Machine learning and deep learning methods for cybersecurity,” *IEEE Access*, vol. 6, pp. 35 365–35 381, 2018. DOI: 10.1109/ACCESS.2018.2836950.

- [89] O. Avatefipour et al., “An intelligent secured framework for cyberattack detection in electric vehicles’ can bus using machine learning,” *IEEE Access*, vol. 7, pp. 127 580–127 592, 2019. DOI: 10.1109/ACCESS.2019.2937576.
- [90] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.
- [91] R. S. Sutton, A. G. Barto, et al., *Introduction to reinforcement learning*. MIT press Cambridge, 1998, vol. 135.
- [92] S. Kim and A. S. Ibrahim, “Byzantine-fault-tolerant consensus via reinforcement learning for permissioned blockchain-empowered v2x network,” *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 1, pp. 172–183, 2023. DOI: 10.1109/TIV.2022.3168575.
- [93] C. J. C. H. Watkins and P. Dayan, “Q-learning,” *Machine Learning*, vol. 8, no. 3, pp. 279–292, May 1992. DOI: 10.1007/BF00992698.
- [94] M. Bowling and M. Veloso, “Rational and convergent learning in stochastic games,” in *International joint conference on artificial intelligence*, Lawrence Erlbaum Associates Ltd, vol. 17, 2001, pp. 1021–1026.
- [95] Z. Zhu and H. Zhao, “A survey of deep rl and il for autonomous driving policy learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14 043–14 065, 2021.
- [96] S. Levine. “Deep reinforcement learning.” Image extracted from Lecture 8 slides. [Online]. Available: <https://rail.eecs.berkeley.edu/deeprlcourse/deeprlcourse/static/slides/lec-8.pdf>.
- [97] B. R. Kiran et al., “Deep reinforcement learning for autonomous driving: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 6, pp. 4909–4926, 2021.
- [98] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press, 1957.
- [99] S. Aradi, “Survey of deep reinforcement learning for motion planning of autonomous vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 740–759, 2022. DOI: 10.1109/TITS.2020.3024655.
- [100] X. Lu et al., “Reinforcement learning-based physical cross-layer security and privacy in 6g,” *IEEE Communications Surveys & Tutorials*, vol. 25, no. 1, pp. 425–466, 2023. DOI: 10.1109/COMST.2022.3224279.
- [101] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, “Dueling network architectures for deep reinforcement learning,” in *International conference on machine learning*, PMLR, 2016, pp. 1995–2003.
- [102] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.
- [103] T. P. Lillicrap et al., “D.(2019),” *Continuous control with deep reinforcement learning*,

- [104] T. Li et al., “Applications of multi-agent reinforcement learning in future internet: A comprehensive survey,” *IEEE Communications Surveys & Tutorials*, vol. 24, no. 2, pp. 1240–1279, 2022.
- [105] A. Feriani and E. Hossain, “Single and multi-agent deep reinforcement learning for ai-enabled wireless networks: A tutorial,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1226–1252, 2021.
- [106] J. Hu and M. P. Wellman, “Nash q-learning for general-sum stochastic games,” *Journal of machine learning research*, vol. 4, no. Nov, pp. 1039–1069, 2003.
- [107] G. Thomas, Y. Luo, and T. Ma, “Safe reinforcement learning by imagining the near future,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 13 859–13 869, 2021.
- [108] S. Gu et al., “A review of safe reinforcement learning: Methods, theory and applications,” *arXiv preprint arXiv:2205.10330*, 2022.
- [109] J. Garcia and F. Fernández, “A comprehensive survey on safe reinforcement learning,” *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1437–1480, 2015.
- [110] L. Zhang, R. Zhang, T. Wu, R. Weng, M. Han, and Y. Zhao, “Safe reinforcement learning with stability guarantee for motion planning of autonomous vehicles,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 12, pp. 5435–5444, 2021.
- [111] X. Lu, L. Xiao, G. Niu, X. Ji, and Q. Wang, “Safe exploration in wireless security: A safe reinforcement learning algorithm with hierarchical structure,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 732–743, 2022.
- [112] C. Dai, L. Xiao, X. Wan, and Y. Chen, “Reinforcement learning with safe exploration for network security,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3057–3061. DOI: 10.1109/ICASSP.2019.8682983.
- [113] F. da Silva and A. Costa, “Accelerating multiagent reinforcement learning through transfer learning,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 31, 2017.
- [114] W. Zhao, J. P. Queralta, and T. Westerlund, “Sim-to-real transfer in deep reinforcement learning for robotics: A survey,” in *2020 IEEE symposium series on computational intelligence (SSCI)*, IEEE, 2020, pp. 737–744.
- [115] M. E. Taylor and P. Stone, “Transfer learning for reinforcement learning domains: A survey,” *Journal of Machine Learning Research*, vol. 10, no. 7, 2009.
- [116] F. L. Da Silva and A. H. R. Costa, “A survey on transfer learning for multi-agent reinforcement learning systems,” *Journal of Artificial Intelligence Research*, vol. 64, pp. 645–703, 2019.

- [117] Z. Zhu, K. Lin, A. K. Jain, and J. Zhou, “Transfer learning in deep reinforcement learning: A survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [118] S. Pateria, B. Subagdja, A.-h. Tan, and C. Quek, “Hierarchical reinforcement learning: A comprehensive survey,” *ACM Computing Surveys (CSUR)*, vol. 54, no. 5, pp. 1–35, 2021.
- [119] M. Al-Emran, “Hierarchical reinforcement learning: A survey,” *International journal of computing and digital systems*, vol. 4, no. 02, 2015.
- [120] T. D. Kulkarni, K. Narasimhan, A. Saeedi, and J. Tenenbaum, “Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation,” *Advances in neural information processing systems*, vol. 29, 2016.
- [121] P. Dayan and G. E. Hinton, “Feudal reinforcement learning,” *Advances in neural information processing systems*, vol. 5, 1992.
- [122] A. S. Vezhnevets et al., “Feudal networks for hierarchical reinforcement learning,” in *International Conference on Machine Learning*, PMLR, 2017, pp. 3540–3549.
- [123] M. A. Ferrag, O. Friha, L. Maglaras, H. Janicke, and L. Shu, “Federated deep learning for cyber security in the internet of things: Concepts, applications, and experimental analysis,” *IEEE Access*, vol. 9, pp. 138 509–138 542, 2021. DOI: 10.1109/ACCESS.2021.3118642.
- [124] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, and H. Vincent Poor, “Federated learning for internet of things: A comprehensive survey,” *IEEE Communications Surveys & Tutorials*, vol. 23, no. 3, pp. 1622–1658, 2021. DOI: 10.1109/COMST.2021.3075439.
- [125] J. Wang, J. Hu, J. Mills, G. Min, M. Xia, and N. Georgalas, “Federated ensemble model-based reinforcement learning in edge computing,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 34, no. 6, pp. 1848–1859, 2023. DOI: 10.1109/TPDS.2023.3264480.
- [126] J. Qi, Q. Zhou, L. Lei, and K. Zheng, “Federated reinforcement learning: Techniques, applications, and open challenges,” *arXiv preprint arXiv:2108.11887*, 2021.
- [127] J. Hua, L. Zeng, G. Li, and Z. Ju, “Learning for a robot: Deep reinforcement learning, imitation learning, transfer learning,” *Sensors*, vol. 21, no. 4, p. 1278, 2021.
- [128] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, “Safe model-based reinforcement learning with stability guarantees,” *Advances in neural information processing systems*, vol. 30, 2017.
- [129] L. Kaiser et al., “Model-based reinforcement learning for atari,” *arXiv preprint arXiv:1903.00374*, 2019.

- [130] H. Lee, N. Kim, and S. W. Cha, "Model-based reinforcement learning for eco-driving control of electric vehicles," *IEEE Access*, vol. 8, pp. 202 886–202 896, 2020. DOI: 10.1109/ACCESS.2020.3036719.
- [131] C.-V. Pal and F. Leon, "Brief survey of model-based reinforcement learning techniques," in *2020 24th International Conference on System Theory, Control and Computing (ICSTCC)*, 2020, pp. 92–97. DOI: 10.1109/ICSTCC50638.2020.9259716.
- [132] K. Chatzilygeroudis, V. Vassiliades, F. Stulp, S. Calinon, and J.-B. Mouret, "A survey on policy search algorithms for learning robot controllers in a handful of trials," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 328–347, 2020. DOI: 10.1109/TR0.2019.2958211.
- [133] T. Wang et al., "Benchmarking model-based reinforcement learning," *arXiv preprint arXiv:1907.02057*, 2019.
- [134] D. Silver, G. Lever, N. Heess, T. Degris, D. Wierstra, and M. Riedmiller, "Deterministic policy gradient algorithms," in *International conference on machine learning*, Pmlr, 2014, pp. 387–395.
- [135] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," *Advances in neural information processing systems*, vol. 12, 1999.
- [136] Y. Ren, X. Chen, S. Guo, S. Guo, and A. Xiong, "Blockchain-based vec network trust management: A drl algorithm for vehicular service offloading and migration," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 8, pp. 8148–8160, 2021. DOI: 10.1109/TVT.2021.3092346.
- [137] H. Yang, J. Zhao, Z. Xiong, K.-Y. Lam, S. Sun, and L. Xiao, "Privacy-preserving federated learning for uav-enabled networks: Learning-based joint scheduling and resource management," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 10, pp. 3144–3159, 2021. DOI: 10.1109/JSAC.2021.3088655.
- [138] M. B. Mollah et al., "Blockchain for the internet of vehicles towards intelligent transportation systems: A survey," *IEEE Internet of Things Journal*, vol. 8, no. 6, pp. 4157–4185, 2021. DOI: 10.1109/JIOT.2020.3028368.
- [139] E. A. Shammar, A. T. Zahary, and A. A. Al-Shargabi, "A survey of iot and blockchain integration: Security perspective," *IEEE Access*, vol. 9, pp. 156 114–156 150, 2021. DOI: 10.1109/ACCESS.2021.3129697.
- [140] E. Bellini, Y. Iraqi, and E. Damiani, "Blockchain-based distributed trust and reputation management systems: A survey," *IEEE Access*, vol. 8, pp. 21 127–21 151, 2020. DOI: 10.1109/ACCESS.2020.2969820.
- [141] R. Jabbar et al., "Blockchain technology for intelligent transportation systems: A systematic literature review," *IEEE Access*, vol. 10, pp. 20 995–21 031, 2022. DOI: 10.1109/ACCESS.2022.3149958.

- [142] X. Wang, H. Zhu, Z. Ning, L. Guo, and Y. Zhang, “Blockchain intelligence for internet of vehicles: Challenges and solutions,” *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 2325–2355, 2023. DOI: 10.1109/COMST.2023.3305312.
- [143] S. Nakamoto, *Bitcoin: A peer-to-peer electronic cash system*, White paper, 2008. [Online]. Available: <https://bitcoin.org/bitcoin.pdf>.
- [144] M. Castro, B. Liskov, et al., “Practical byzantine fault tolerance,” in *OsDI*, vol. 99, 1999, pp. 173–186.
- [145] P. Surapaneni, S. Bojjagani, V. C. Bharathi, M. Kumar Morampudi, A. Kumar Maurya, and M. Khurram Khan, “A systematic review on blockchain-enabled internet of vehicles (bioV): Challenges, defenses, and future research directions,” *IEEE Access*, vol. 12, pp. 123 529–123 560, 2024. DOI: 10.1109/ACCESS.2024.3453433.
- [146] M. Fardad, “Blockchain-enabled and latency-aware resource management for vehicular networks,” Ph.D. dissertation, 2025.
- [147] S. W. Turner, M. Karakus, E. Guler, and S. Uludag, “A promising integration of sdn and blockchain for iot networks: A survey,” *IEEE Access*, vol. 11, pp. 29 800–29 822, 2023. DOI: 10.1109/ACCESS.2023.3260777.
- [148] C. Campajola, R. Cristodaro, F. M. D. Collibus, T. Yan, N. Vallarano, and C. J. Tessone, *The evolution of centralisation on cryptocurrency platforms*, 2023. arXiv: 2206.05081 [physics.soc-ph]. [Online]. Available: <https://arxiv.org/abs/2206.05081>.
- [149] E. Androulaki et al., “Hyperledger fabric: A distributed operating system for permissioned blockchains,” in *Proceedings of the Thirteenth EuroSys Conference*, ser. EuroSys ’18, Porto, Portugal: Association for Computing Machinery, 2018, ISBN: 9781450355841. DOI: 10.1145/3190508.3190538. [Online]. Available: <https://doi.org/10.1145/3190508.3190538>.
- [150] R. G. Brown, J. Carlyle, I. Grigg, and M. Hearn, “Corda: An introduction,” *R3 CEV, August*, vol. 1, no. 15, p. 14, 2016.
- [151] P. M. Rao, S. Jangirala, S. Pedada, A. K. Das, and Y. Park, “Blockchain integration for iot-enabled v2x communications: A comprehensive survey, security issues and challenges,” *IEEE Access*, vol. 11, pp. 54 476–54 494, 2023. DOI: 10.1109/ACCESS.2023.3281844.
- [152] S. K. Dwivedi, R. Amin, A. K. Das, M. T. Leung, K.-K. R. Choo, and S. Vollala, “Blockchain-based vehicular ad-hoc networks: A comprehensive survey,” *Ad Hoc Netw.*, vol. 137, p. 102 980, 2022. DOI: 10.1016/j.adhoc.2022.102980.
- [153] L. Benarous, S. Bitam, and A. Mellouk, “Blockchain-based privacy-aware pseudonym management framework for vehicular networks,” in *Security in Vehicular Networks: Focus on Location and Identity Privacy*. 2022, pp. 181–209. DOI: 10.1002/9781394172610.ch9.

- [154] J. Meijers et al., “Blockchain for V2X: Applications and architectures,” *IEEE Open Journal of Vehicular Technology*, vol. 3, pp. 193–209, 2022. DOI: 10.1109/OJVT.2022.3172709.
- [155] Y. Yuan and F.-Y. Wang, “Towards blockchain-based intelligent transportation systems,” in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, 2016, pp. 2663–2668. DOI: 10.1109/ITSC.2016.7795984.
- [156] M. Fardad, G.-M. Muntean, and I. Tal, “A blockchain-enabled vehicular edge computing framework for secure performance-oriented v2x service delivery,” *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 13 853–13 867, 2024. DOI: 10.1109/TVT.2024.3394150.
- [157] X. Ye, M. Li, P. Si, R. Yang, Z. Wang, and Y. Zhang, “Collaborative and intelligent resource optimization for computing and caching in IoV with blockchain and MEC using A3C approach,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 2, pp. 1449–1463, 2023. DOI: 10.1109/TVT.2022.3210570.
- [158] A. Ensor, S. Schefer-Wenzl, and I. Miladinovic, “Blockchains for IoT payments: A survey,” in *2018 IEEE Globecom Workshops (GC Wkshps)*, 2018, pp. 1–6. DOI: 10.1109/GLOCOMW.2018.8644522.
- [159] X. Deng and T. Gao, “Electronic payment schemes based on blockchain in VANETs,” *IEEE Access*, vol. 8, pp. 38 296–38 303, 2020. DOI: 10.1109/ACCESS.2020.2974964.
- [160] M. Fardad, E. M. Mianji, G.-M. Muntean, and I. Tal, “Hybrid consensus networks for scalable and secure internet of vehicles,” in *2024 IEEE 48th Annual Computers, Software, and Applications Conference (COMPSAC)*, 2024, pp. 1671–1676. DOI: 10.1109/COMPSAC61105.2024.00263.
- [161] G. Raja, K. Kottursamy, K. Dev, R. Narayanan, A. Raja, and K. B. V. Karthik, “Blockchain-integrated multiagent deep reinforcement learning for securing cooperative adaptive cruise control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9630–9639, 2022. DOI: 10.1109/TITS.2022.3168486.
- [162] T. Xu, X. Lu, L. Xiao, Y. Tang, and H. Dai, “Voltage based authentication for controller area networks with reinforcement learning,” in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019, pp. 1–5.
- [163] L. Xiao, X. Lu, T. Xu, W. Zhuang, and H. Dai, “Reinforcement learning-based physical-layer authentication for controller area networks,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2535–2547, 2021.
- [164] H. Sedjelmaci, “Attacks detection approach based on a reinforcement learning process to secure 5g wireless network,” in *IEEE International Conference on Communications Workshops (ICC Workshops)*, IEEE, 2020, pp. 1–6.

- [165] Z. Li, Y. Kong, and C. Jiang, “A transfer double deep q network based ddos detection method for internet of vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 4, pp. 5317–5331, 2023. DOI: 10.1109/TVT.2022.3233880.
- [166] Y. Deng et al., “Resource provisioning for mitigating edge ddos attacks in mec-enabled sdvn,” *IEEE Internet of Things Journal*, vol. 9, no. 23, pp. 24 264–24 280, 2022. DOI: 10.1109/JIOT.2022.3189975.
- [167] T. Zhang et al., “How to mitigate ddos intelligently in sd-iov: A moving target defense approach,” *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1, pp. 1097–1106, 2023. DOI: 10.1109/TII.2022.3190556.
- [168] R. Sedar, C. Kalalas, J. Alonso-Zarate, and F. Vázquez-Gallego, “Multi-domain denial-of-service attacks in internet-of-vehicles: Vulnerability insights and detection performance,” in *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*, 2022, pp. 438–443. DOI: 10.1109/NetSoft54395.2022.9844055.
- [169] J. Kamel, M. Wolf, R. W. van der Hei, A. Kaiser, P. Urien, and F. Kargl, “Veremi extension: A dataset for comparable evaluation of misbehavior detection in vanets,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6. DOI: 10.1109/ICC40277.2020.9149132.
- [170] W. Lu et al., “Secure transmission for multi-uav-assisted mobile edge computing based on reinforcement learning,” *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 3, pp. 1270–1282, 2023. DOI: 10.1109/TNSE.2022.3185130.
- [171] Y. Ding et al., “Online edge learning offloading and resource management for uav-assisted mec secure communications,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 17, no. 1, pp. 54–65, 2023. DOI: 10.1109/JSTSP.2022.3222910.
- [172] Z. Li, Y. Kong, C. Wang, and C. Jiang, “Ddos mitigation based on space-time flow regularities in iov: A feature adaption reinforcement learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 2262–2278, 2022. DOI: 10.1109/TITS.2021.3066404.
- [173] Y. Fan, X. Xiao, and W. Feng, “An anti-jamming game in vanet platoon with reinforcement learning,” in *2018 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW)*, IEEE, 2018, pp. 1–2.
- [174] Z. Lin, X. Yan, L. Xiao, S. Yan, Y. Tang, and J. Liu, “Environment-aware reinforcement learning based vanet communications against jamming and interference,” in *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, 2022, pp. 5717–5722. DOI: 10.1109/GLOBECOM48099.2022.10001042.

- [175] S. Feng and S. Haykin, “Cognitive risk control for anti-jamming v2v communications in autonomous vehicle networks,” *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 9920–9934, 2019. DOI: 10.1109/TVT.2019.2935999.
- [176] L. Xiao, X. Lu, D. Xu, Y. Tang, L. Wang, and W. Zhuang, “Uav relay in vanets against smart jamming with reinforcement learning,” *IEEE Transactions on Vehicular Technology*, vol. 67, no. 5, pp. 4087–4097, 2018.
- [177] Z. Lv, L. Xiao, Y. Du, G. Niu, C. Xing, and W. Xu, “Multi-agent reinforcement learning based uav swarm communications against jamming,” *IEEE Transactions on Wireless Communications*, pp. 1–1, 2023. DOI: 10.1109/TWC.2023.3268082.
- [178] R. Ding, F. Zhou, Y. Qu, C. Dong, Q. Wu, and T. Q. S. Quek, “Novel online-offline ma2c-ddpg for efficient spectrum allocation and trajectory optimization in dynamic spectrum sharing uav networks,” in *IEEE/CIC International Conference on Communications in China (ICCC)*, 2023, pp. 1–6. DOI: 10.1109/ICCC57788.2023.10233493.
- [179] N. I. Mowla, N. H. Tran, I. Doh, and K. Chae, “Afrl: Adaptive federated reinforcement learning for intelligent jamming defense in fanet,” *Journal of Communications and Networks*, vol. 22, no. 3, pp. 244–258, 2020. DOI: 10.1109/JCN.2020.000015.
- [180] Y. Yao, J. Zhao, Z. Li, X. Cheng, and L. Wu, “Jamming and eavesdropping defense scheme based on deep reinforcement learning in autonomous vehicle networks,” *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 1211–1224, 2023. DOI: 10.1109/TIFS.2023.3236788.
- [181] Y. Yao, J. Zhao, Z. Li, X. Cheng, L. Wu, and X. Li, “Cognitive risk control for anti-eavesdropping in connected and autonomous vehicles network,” in *2022 IEEE 96th Vehicular Technology Conference (VTC2022-Fall)*, 2022, pp. 1–7. DOI: 10.1109/VTC2022-Fall157202.2022.10012876.
- [182] Y. Yao, F. Shu, Z. Li, X. Cheng, and L. Wu, “Secure transmission scheme based on joint radar and communication in mobile vehicular networks,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2023. DOI: 10.1109/TITS.2023.3271452.
- [183] Y. You, R. Zhao, and H. Sun, “Deep reinforcement learning-based trajectory planning for secure uav communication,” in *2021 4th International Conference on Information Communication and Signal Processing (ICICSP)*, 2021, pp. 528–532. DOI: 10.1109/ICICSP54369.2021.9611877.
- [184] J. Zhang, “A q-learning based method for secure uav communication against malicious eavesdropping,” in *2022 14th International Conference on Computer and Automation Engineering (ICCAE)*, 2022, pp. 168–172. DOI: 10.1109/ICCAE55086.2022.9762433.

- [185] X. Lu, L. Xiao, T. Xu, Y. Zhao, Y. Tang, and W. Zhuang, "Reinforcement learning based phy authentication for vanets," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 3, pp. 3068–3079, 2020. DOI: 10.1109/TVT.2020.2967026.
- [186] H. El-Sayed, H. Alexander, P. Kulkarni, M. A. Khan, R. M. Noor, and Z. Trabelsi, "A novel multifaceted trust management framework for vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 20 084–20 097, 2022. DOI: 10.1109/TITS.2022.3187788.
- [187] C. Pu, "A novel blockchain-based trust management scheme for vehicular networks," in *2021 Wireless Telecommunications Symposium (WTS)*, 2021, pp. 1–6. DOI: 10.1109/WTS51064.2021.9433711.
- [188] T. Jing, Y. Liu, X. Wang, and Q. Gao, "A reservoir computing-based distributed trust sharing provisioning for internet of vehicle," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, 2022, pp. 3038–3043. DOI: 10.1109/ITSC55140.2022.9922018.
- [189] M. Cheng, J. Zhang, S. Nazarian, J. Deshmukh, and P. Bogdan, "Trust-aware control for intelligent transportation systems," in *2021 IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 377–384. DOI: 10.1109/IV48863.2021.9576045.
- [190] X. Fu, F. R. Yu, J. Wang, Q. Qi, and J. Liao, "Performance optimization for blockchain-enabled distributed network function virtualization management and orchestration," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6670–6679, 2020. DOI: 10.1109/TVT.2020.2985581.
- [191] S. Guleng, C. Wu, X. Chen, X. Wang, T. Yoshinaga, and Y. Ji, "Decentralized trust evaluation in vehicular internet of things," *IEEE Access*, vol. 7, pp. 15 980–15 988, 2019. DOI: 10.1109/ACCESS.2019.2893262.
- [192] S. Gyawali, Y. Qian, and R. Q. Hu, "Deep reinforcement learning based dynamic reputation policy in 5g based vehicular communication networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 6, pp. 6136–6146, 2021.
- [193] J. Guo et al., "Trove: A context-awareness trust model for vanets using reinforcement learning," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6647–6662, 2020.
- [194] H. Chai, S. Leng, K. Zhang, and S. Mao, "Proof-of-reputation based-consortium blockchain for trust resource sharing in internet of vehicles," *IEEE Access*, vol. 7, pp. 175 744–175 757, 2019. DOI: 10.1109/ACCESS.2019.2956955.
- [195] D. Zhang, F. R. Yu, and R. Yang, "A machine learning approach for software-defined vehicular ad hoc networks with trust management," in *2018 IEEE Global Communications Conference (GLOBECOM)*, IEEE, 2018, pp. 1–6.

- [196] D. Zhang, F. R. Yu, R. Yang, and H. Tang, “A deep reinforcement learning-based trust management scheme for software-defined vehicular networks,” in *Proceedings of the 8th ACM symposium on design and analysis of intelligent vehicular networks and applications*, 2018, pp. 1–7.
- [197] M. S. Sani, S. Iranmanesh, R. Raad, and F. Tubbal, “Energy-efficient routing protocol in vehicular opportunistic networks: A dynamic cluster-based routing using deep reinforcement learning,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–16, 2026. DOI: 10.1109/TITS.2026.3661465.
- [198] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, and C.-H. Hsu, “Efficient and secure routing protocol based on artificial intelligence algorithms with uav-assisted for vehicular ad hoc networks in intelligent transportation systems,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4757–4769, 2021. DOI: 10.1109/TITS.2020.3041746.
- [199] T. Zhang, C. Xu, B. Zhang, J. Shen, X. Kuang, and L. A. Grieco, “Toward attack-resistant route mutation for vanets: An online and adaptive multiagent reinforcement learning approach,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 23 254–23 267, 2022. DOI: 10.1109/TITS.2022.3198507.
- [200] W. Wang, M. Min, L. Xiao, Y. Chen, and H. Dai, “Protecting semantic trajectory privacy for vanet with reinforcement learning,” in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019, pp. 1–5.
- [201] M. Min, W. Wang, L. Xiao, Y. Xiao, and Z. Han, “Reinforcement learning-based sensitive semantic location privacy protection for vanets,” *China Communications*, vol. 18, no. 6, pp. 244–260, 2021. DOI: 10.23919/JCC.2021.06.019.
- [202] Y. Dai, D. Xu, K. Zhang, S. Maharjan, and Y. Zhang, “Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4312–4324, 2020. DOI: 10.1109/TVT.2020.2973705.
- [203] Y. Lu, X. Huang, Y. Dai, S. Maharjan, and Y. Zhang, “Federated learning for data privacy preservation in vehicular cyber-physical systems,” *IEEE Network*, vol. 34, no. 3, pp. 50–56, 2020. DOI: 10.1109/MNET.011.1900317.
- [204] Y. Lu, X. Huang, K. Zhang, S. Maharjan, and Y. Zhang, “Blockchain empowered asynchronous federated learning for secure data sharing in internet of vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4298–4311, 2020. DOI: 10.1109/TVT.2020.2973651.
- [205] X. Chen, T. Zhang, S. Shen, T. Zhu, and P. Xiong, “An optimized differential privacy scheme with reinforcement learning in vanet,” *Computers & Security*, vol. 110, p. 102 446, 2021.

- [206] C. Zhu, Z. Cheng, D. Ye, F. K. Hussain, T. Zhu, and W. Zhou, “Time-driven and privacy-preserving navigation model for vehicle-to-vehicle communication systems,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 7, pp. 8459–8470, 2023. DOI: 10.1109/TVT.2023.3248613.
- [207] S. Berri, J. Zhang, B. Bensaou, and H. Labiod, “Privacy-preserving data-prefetching in vehicular networks via reinforcement learning,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6. DOI: 10.1109/ICC40277.2020.9149372.
- [208] S. Berri, J. Zhang, B. Bensaou, and H. Labiod, “Preserving location-privacy in vehicular networks via reinforcement learning,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18 535–18 545, 2022. DOI: 10.1109/TITS.2022.3165387.
- [209] D. Wei, J. Zhang, M. Shojafar, S. Kumari, N. Xi, and J. Ma, “Privacy-aware multiagent deep reinforcement learning for task offloading in vanet,” *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–15, 2022. DOI: 10.1109/TITS.2022.3202196.
- [210] S. Wang, Q. Zhu, H. Huang, Y. Lei, W. Zhan, and H. Duan, “Deep reinforcement learning based real-time proactive edge caching in intelligent transportation system,” in *2023 8th International Conference on Cloud Computing and Big Data Analytics (ICCCBDA)*, 2023, pp. 162–166. DOI: 10.1109/ICCCBDA56900.2023.10154769.
- [211] Y. Liu and B. Mao, “On a novel content edge caching approach based on multi-agent federated reinforcement learning in internet of vehicles,” in *2023 32nd Wireless and Optical Communications Conference (WOCC)*, 2023, pp. 1–5. DOI: 10.1109/WOCC58016.2023.10139417.
- [212] Y. Lin, J. Bao, Y. Zhang, J. Li, F. Shu, and L. Hanzo, “Privacy-preserving joint edge association and power optimization for the internet of vehicles via federated multi-agent reinforcement learning,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 8256–8261, 2023. DOI: 10.1109/TVT.2023.3240682.
- [213] M. Nasr-Azadani, J. Abouei, and K. N. Plataniotis, “Distillation and ordinary federated learning actor-critic algorithms in heterogeneous uav-aided networks,” *IEEE Access*, vol. 11, pp. 44 205–44 220, 2023. DOI: 10.1109/ACCESS.2023.3273123.
- [214] J. Kang et al., “Blockchain-based pseudonym management for vehicle twin migrations in vehicular edge metaverse,” *IEEE Internet of Things Journal*, vol. 11, no. 21, pp. 34 254–34 269, 2024. DOI: 10.1109/JIOT.2024.3404559.
- [215] U. Ahmed, J. C.-W. Lin, and G. Srivastava, “Privacy-preserving deep reinforcement learning in vehicle ad hoc networks,” *IEEE Consumer Electronics Magazine*, vol. 11, no. 6, pp. 41–48, 2021.

- [216] Y. Liu, H. Wang, M. Peng, J. Guan, and Y. Wang, “An incentive mechanism for privacy-preserving crowdsensing via deep reinforcement learning,” *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 8616–8631, 2021. DOI: 10.1109/JIOT.2020.3047105.
- [217] S. Wang, S. Sun, X. Wang, Z. Ning, and J. J. P. C. Rodrigues, “Secure crowdsensing in 5g internet of vehicles: When deep reinforcement learning meets blockchain,” *IEEE Consumer Electronics Magazine*, vol. 10, no. 5, pp. 72–81, 2021. DOI: 10.1109/MCE.2020.3048238.
- [218] G. Zhang, Z. Luo, and T. Yang, “Distributed computation offloading based on deep reinforcement learning and blockchain in internet of vehicles,” in *2023 IEEE/CIC International Conference on Communications in China (ICCC)*, 2023, pp. 1–6. DOI: 10.1109/ICCC57788.2023.10233288.
- [219] Z. Li, X. Yang, C. Wang, K. Ma, and C. Jiang, “Crowd-learning: A behavior-based verification method in software-defined vehicular networks with mec framework,” *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 1622–1639, 2022. DOI: 10.1109/JIOT.2021.3107581.
- [220] H. Lin, S. Garg, J. Hu, G. Kaddoum, M. Peng, and M. S. Hossain, “Blockchain and deep reinforcement learning empowered spatial crowdsourcing in software-defined internet of vehicles,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3755–3764, 2021. DOI: 10.1109/TITS.2020.3025247.
- [221] Y. Lin et al., “Drl-based adaptive sharding for blockchain-based federated learning,” *IEEE Transactions on Communications*, pp. 1–1, 2023. DOI: 10.1109/TCOMM.2023.3288591.
- [222] E. M. Mianji, G.-M. Muntean, and I. Tal, “Trustworthy routing in vanet: A q-learning approach to protect against black hole and gray hole attacks,” in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, 2023, pp. 1–6. DOI: 10.1109/VTC2023-Spring57618.2023.10201086.
- [223] A. Boualouache, S.-M. Senouci, and S. Moussaoui, “A survey on pseudonym changing strategies for vehicular ad-hoc networks,” *IEEE Communications Surveys & Tutorials*, vol. 20, no. 1, pp. 770–790, 2018. DOI: 10.1109/COMST.2017.2771522.
- [224] Y. Li, Y. Yin, X. Chen, J. Wan, G. Jia, and K. Sha, “A secure dynamic mix zone pseudonym changing scheme based on traffic context prediction,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9492–9505, 2022. DOI: 10.1109/TITS.2021.3125744.
- [225] H. Li, Y. Lai, and Y. Chen, “Broadcast and silence period (bsp): A pseudonym change strategy,” *IEEE Transactions on Vehicular Technology*, vol. 72, no. 10, pp. 13 618–13 630, 2023. DOI: 10.1109/TVT.2023.3279121.

- [226] S. Haider, D. Gao, R. Ali, A. Hussain, and M. T. Ikram, "A privacy conserves pseudonym acquisition scheme in vehicular communication systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 9, pp. 15 536–15 545, 2022. DOI: 10.1109/TITS.2022.3141744.
- [227] W. Zhang, Z. Guo, G. Han, H. Zhu, and Q. Liu, "Dpsp: A dynamic pseudonym swap program based location privacy protection algorithm for internet of vehicles," *IEEE Transactions on Network Science and Engineering*, vol. 11, no. 5, pp. 4525–4535, 2024. DOI: 10.1109/TNSE.2024.3392709.
- [228] A. Hayat, Z. Iftikhar, M. I. Khan, A. Mehbodniya, J. L. Webber, and S. Hanif, "A novel pseudonym changing scheme for location privacy preservation in sparse traffic areas," *IEEE Access*, vol. 11, pp. 89 974–89 985, 2023. DOI: 10.1109/ACCESS.2023.3303846.
- [229] J. Kang, R. Yu, X. Huang, and Y. Zhang, "Privacy-preserved pseudonym scheme for fog computing supported internet of vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 8, pp. 2627–2637, 2018. DOI: 10.1109/TITS.2017.2764095.
- [230] A. P. Mdee, M. T. R. Khan, J. Seo, and D. Kim, "Security compliant and co-operative pseudonyms swapping for location privacy preservation in vanets," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 8, pp. 10 710–10 723, 2023. DOI: 10.1109/TVT.2023.3254660.
- [231] M. A. Al-Shareeda and S. Manickam, "A systematic literature review on security of vehicular ad-hoc network (vanet) based on veins framework," *IEEE Access*, vol. 11, pp. 46 218–46 228, 2023. DOI: 10.1109/ACCESS.2023.3274774.
- [232] J. Liu et al., "Cpahp: Conditional privacy-preserving authentication scheme with hierarchical pseudonym for 5g-enabled iov," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 7, pp. 8929–8940, 2023. DOI: 10.1109/TVT.2023.3246466.
- [233] L. Da, W. Hu, Y. Wang, Y. Ding, B. Li, and D. Zhu, "A blockchain-assisted road condition monitoring scheme with privacy-preserving for vanets," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 12, pp. 19 625–19 640, 2024. DOI: 10.1109/TVT.2024.3450078.
- [234] D. Chulerttiyawong and A. Jamalipour, "A blockchain assisted vehicular pseudonym issuance and management system for conditional privacy enhancement," *IEEE Access*, vol. 9, pp. 127 305–127 319, 2021. DOI: 10.1109/ACCESS.2021.3112013.
- [235] B. A. Mohammed et al., "Efficient blockchain-based pseudonym authentication scheme supporting revocation for 5g-assisted vehicular fog computing," *IEEE Access*, vol. 12, pp. 33 089–33 099, 2024. DOI: 10.1109/ACCESS.2024.3372390.

- [236] X. Zhang, X. Chen, S. Liu, and S. Zhong, “Anonymous authentication and information sharing scheme based on blockchain and zero knowledge proof for vanets,” *IEEE Transactions on Vehicular Technology*, vol. 73, no. 12, pp. 18 043–18 058, 2024. DOI: 10.1109/TVT.2024.3432553.
- [237] G. Cheng et al., “Conditional privacy-preserving multi-domain authentication and pseudonym management for 6g-enabled iov,” *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 10 206–10 220, 2024. DOI: 10.1109/TIFS.2023.3314211.
- [238] P. Hu et al., “Security-enhanced data sharing scheme with location privacy preservation for internet of vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 73, no. 9, pp. 13 751–13 764, 2024. DOI: 10.1109/TVT.2024.3393302.
- [239] Y. Zhou, J. Chen, B. He, and L. Lv, “Data trading for blockchain-based data market in cyber-physical-social smart systems,” in *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2021, pp. 1558–1563. DOI: 10.1109/PIMRC50174.2021.9569718.
- [240] C. Ke, F. Xiao, Y. Cao, and Z. Huang, “A group-vehicles oriented reputation assessment scheme for edge vanets,” *IEEE Transactions on Cloud Computing*, vol. 12, no. 3, pp. 859–875, 2024. DOI: 10.1109/TCC.2024.3406509.
- [241] J. Zuo et al., “A decentralized reputation management model for enhanced iov networks with 5g broadcast services,” *IEEE Transactions on Broadcasting*, vol. 71, no. 1, pp. 63–73, 2025. DOI: 10.1109/TBC.2024.3434745.
- [242] W. Chen, W. Yang, M. Xiao, L. Xue, and S. Wang, “Lbdt: A lightweight blockchain-based data trading scheme in internet of vehicles using proof-of-reputation,” *IEEE Transactions on Mobile Computing*, vol. 24, no. 4, pp. 2800–2816, 2025. DOI: 10.1109/TMC.2024.3497934.
- [243] V. Chunduri et al., “Blockchain-based secure trust management scheme for internet of vehicles over cyber-physical system,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 26, no. 9, pp. 14 067–14 076, 2025. DOI: 10.1109/TITS.2024.3485481.
- [244] Y. Sun et al., “Ppdr: A privacy-preserving dual reputation management scheme in vehicle platoon,” *IEEE Transactions on Dependable and Secure Computing*, vol. 22, no. 6, pp. 6636–6652, 2025. DOI: 10.1109/TDSC.2025.3588717.
- [245] L. Zheng, T. Feng, Z. Xie, X. Li, and C. Su, “Barm: Blockchain-assisted anonymous authentication and reputation management for mobile crowd-sensing in internet of vehicles,” *IEEE Internet of Things Journal*, vol. 12, no. 12, pp. 22 463–22 477, 2025. DOI: 10.1109/JIOT.2025.3552447.

- [246] S. Ma et al., “A cloud-assisted multi-dimensional reputation management scheme with privacy preservation for emergency message dissemination in vanets,” *IEEE Transactions on Dependable and Secure Computing*, pp. 1–18, 2026. DOI: 10.1109/TDSC.2026.3659756.
- [247] Y. Cheng, J. Ma, Z. Liu, Y. Wu, K. Wei, and C. Dong, “A lightweight privacy preservation scheme with efficient reputation management for mobile crowdsensing in vehicular networks,” *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 3, pp. 1771–1788, 2023. DOI: 10.1109/TDSC.2022.3163752.
- [248] J. Yin, Y. Xiao, J. Feng, M. Yang, Q. Pei, and X. Yi, “Didtrust: Privacy-preserving trust management for decentralized identity,” *IEEE Transactions on Dependable and Secure Computing*, vol. 22, no. 3, pp. 3105–3120, 2025. DOI: 10.1109/TDSC.2024.3524760.
- [249] Y. Liu et al., “Vrepchain: A decentralized and privacy-preserving reputation system for social internet of vehicles based on blockchain,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 12, pp. 13 242–13 253, 2022. DOI: 10.1109/TVT.2022.3198004.
- [250] X. Liu, Q. Liu, M. Luo, X. Yang, and Q. Luo, “Blockchain-based efficient and traceable data sharing scheme for vehicular networks,” *IEEE Transactions on Vehicular Technology*, vol. 74, no. 8, pp. 11 777–11 792, 2025. DOI: 10.1109/TVT.2025.3555108.
- [251] K. Yan, W. Ma, Q. Yang, S. Sun, and W. Wang, “Info-chain: Reputation-based blockchain for secure information sharing in 6g intelligent transportation systems,” *IEEE Internet of Things Journal*, vol. 11, no. 5, pp. 9198–9212, 2024. DOI: 10.1109/JIOT.2023.3323011.
- [252] D. Mukathe, W. Di, W. Ahmed, and T. Worku, “Blockchain-powered authenticated key agreement scheme with reputation-incentive mechanism for vehicle-to-vehicle communication in iov,” *IEEE Internet of Things Journal*, vol. 12, no. 13, pp. 25 500–25 515, 2025. DOI: 10.1109/JIOT.2025.3558278.
- [253] F. Belamri, S. Boulfekhar, and D. Aissani, “A survey on QoS routing protocols in vehicular ad hoc network (VANET),” *Telecommunication Systems*, vol. 78, no. 1, pp. 117–153, 2021.
- [254] I. Tal and G.-M. Muntean, “Towards reasoning vehicles: A survey of fuzzy logic-based solutions in vehicular networks,” *ACM Computing Surveys (CSUR)*, vol. 50, no. 6, pp. 1–37, 2017.
- [255] M. Fardad, E. M. Mianji, G.-M. Muntean, and I. Tal, “A fast and effective graph-based resource allocation and power control scheme in vehicular network slicing,” in *2022 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, IEEE, 2022, pp. 1–6.

- [256] H. I. Abbasi, R. C. Voicu, J. A. Copeland, and Y. Chang, "Towards fast and reliable multihop routing in VANETs," *IEEE Transactions on Mobile Computing*, vol. 19, no. 10, pp. 2461–2474, 2019.
- [257] E. M. Royer and C. E. Perkins, "Ad-hoc on-demand distance vector routing," in *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, vol. 2, 1999, pp. 90–100.
- [258] X. Li, Z. Jia, P. Zhang, R. Zhang, and H. Wang, "Trust-based on-demand multipath routing in mobile ad hoc networks," *IET information security*, vol. 4, no. 4, pp. 212–232, 2010.
- [259] S. Bindel, S. Chaumette, and B. Hilt, "F-ETX: An enhancement of ETX metric for wireless mobile networks," in *International Workshop on Communication Technologies for Vehicles*, Springer, 2015, pp. 35–46.
- [260] B. Wang, X. Chen, and W. Chang, "A light-weight trust-based QoS routing algorithm for ad hoc networks," *Pervasive and Mobile Computing*, vol. 13, pp. 164–180, 2014.
- [261] "Technical specification group radio access network; study on LTE-based V2X services; (release 14)," 3GPP, Tech. Rep. TR 36.885 V2.0.0, 2016.
- [262] Y. Liu, K. Li, Y. Jin, Y. Zhang, and W. Qu, "A novel reputation computation model based on subjective logic for mobile ad hoc networks," *Future Generation Computer Systems*, vol. 27, no. 5, pp. 547–554, 2011.
- [263] G. Liu, Q. Yang, H. Wang, X. Lin, and M. P. Wittie, "Assessment of multi-hop interpersonal trust in social networks by three-valued subjective logic," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, 2014, pp. 1698–1706. DOI: 10.1109/INFOCOM.2014.6848107.
- [264] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.
- [265] J. Chen, H. Xing, Z. Xiao, L. Xu, and T. Tao, "A drl agent for jointly optimizing computation offloading and resource allocation in mec," *IEEE Internet of Things Journal*, vol. 8, no. 24, pp. 17 508–17 524, 2021. DOI: 10.1109/JIOT.2021.3081694.
- [266] Z. Wang, Y. Wei, F. R. Yu, and Z. Han, "Utility optimization for resource allocation in multi-access edge network slicing: A twin-actor deep deterministic policy gradient approach," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 5842–5856, 2022. DOI: 10.1109/TWC.2022.3143949.
- [267] E. M. Mianji, M. Fardad, G.-M. Muntean, and I. Tal, "A survey on multi-agent reinforcement learning applications in the internet of vehicles," in *2024 IEEE 99th Vehicular Technology Conference (VTC2024-Spring)*, 2024, pp. 1–7.

- [268] S. A. Jan, N. U. Amin, M. Othman, M. Ali, A. I. Umar, and A. Basir, "A survey on privacy-preserving authentication schemes in vanets: Attacks, challenges and open issues," *IEEE Access*, vol. 9, pp. 153 701–153 726, 2021. DOI: 10.1109/ACCESS.2021.3125521.
- [269] ETSI, "Intelligent transport systems (ITS); security; trust and privacy management," European Telecommunications Standards Institute (ETSI), ETSI Technical Specification ETSI TS 102 941 V2.2.1, Nov. 2022. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/102900_102999/102941/02.02.01_60/ts_102941v020201p.pdf.
- [270] A. Boualouache, S.-M. Senouci, and S. Moussaoui, "Privanet: An efficient pseudonym changing and management framework for vehicular ad-hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3209–3218, 2020. DOI: 10.1109/TITS.2019.2924856.
- [271] A. Boualouache, S.-M. Senouci, and S. Moussaoui, "Towards an efficient pseudonym management and changing scheme for vehicular ad-hoc networks," in *2016 IEEE Global Communications Conference (GLOBECOM)*, 2016, pp. 1–7. DOI: 10.1109/GLOCOM.2016.7842339.
- [272] A. M. Rahmani, J. Tanveer, F. S. Gharehchopogh, S. Rajabi, and M. Hosseinzadeh, "A novel offloading strategy for multi-user optimization in blockchain-enabled mobile edge computing networks for improved internet of things performance," *Computers and Electrical Engineering*, vol. 119, no. Part A, p. 109 514, 2024, ISSN: 0045-7906. DOI: 10.1016/j.compeleceng.2024.109514. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0045790624004415>.
- [273] K. Moghaddasi, S. Rajabi, and F. S. Gharehchopogh, "Multi-objective secure task offloading strategy for blockchain-enabled iov-mec systems: A double deep q-network approach," *IEEE Access*, vol. 12, pp. 3437–3463, 2024. DOI: 10.1109/ACCESS.2023.3348513.
- [274] K. Moghaddasi, S. Rajabi, F. S. Gharehchopogh, and A. Ghaffari, "An advanced deep reinforcement learning algorithm for three-layer D2D-edge-cloud computing architecture for efficient task offloading in the internet of things," *Sustainable Computing: Informatics and Systems*, vol. 43, p. 100 992, 2024, ISSN: 2210-5379. DOI: 10.1016/j.suscom.2024.100992. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2210537924000374>.
- [275] Y. Zuo, J. Guo, N. Gao, Y. Zhu, S. Jin, and X. Li, "A survey of blockchain and artificial intelligence for 6g wireless communications," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 2494–2528, 2023. DOI: 10.1109/COMST.2023.3315374.
- [276] Y. Yang, L. Wei, J. Wu, C. Long, and B. Li, "A blockchain-based multidomain authentication scheme for conditional privacy preserving in vehicular ad-hoc network," *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8078–8090, 2022. DOI: 10.1109/JIOT.2021.3107443.

- [277] M. Shen, H. Lu, F. Wang, H. Liu, and L. Zhu, “Secure and efficient blockchain-assisted authentication for edge-integrated internet-of-vehicles,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 11, pp. 12 250–12 263, 2022. DOI: 10.1109/TVT.2022.3194008.
- [278] K. Lauter, “The advantages of elliptic curve cryptography for wireless security,” *IEEE Wireless Communications*, vol. 11, no. 1, pp. 62–67, 2004. DOI: 10.1109/MWC.2004.1269719.
- [279] X. Ning, M. Zeng, M. Hua, and Z. Fei, “Multiple reconfigurable intelligent surfaces aided vehicular edge computing networks: A mappo-based approach,” *IEEE Transactions on Vehicular Technology*, vol. 73, no. 11, pp. 17 496–17 509, 2024. DOI: 10.1109/TVT.2024.3419554.
- [280] A. Paszke et al., “Pytorch: An imperative style, high-performance deep learning library,” in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019.
- [281] M. Al-Emran, “Hierarchical reinforcement learning: A survey,” *International Journal of Computing and Digital Systems*, vol. 04, pp. 137–143, 2015. [Online]. Available: <https://api.semanticscholar.org/CorpusID:15038968>.
- [282] F. Berkenkamp, M. Turchetta, A. P. Schoellig, and A. Krause, “Safe model-based reinforcement learning with stability guarantees,” in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS’17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 908–919, ISBN: 9781510860964.
- [283] C. Watkins, “Learning from delayed rewards,” 1989. [Online]. Available: <https://api.semanticscholar.org/CorpusID:59809750>.
- [284] E. M. Mianji, G.-M. Muntean, and I. Tal, “Trust and reputation management for data trading in vehicular edge computing: A drl-based approach,” in *2024 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, 2024, pp. 1–7. DOI: 10.1109/BMSB62888.2024.10608229.
- [285] Z. Lu, Q. Wang, G. Qu, and Z. Liu, “Bars: A blockchain-based anonymous reputation system for trust management in vanets,” in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, 2018, pp. 98–103. DOI: 10.1109/TrustCom/BigDataSE.2018.00025.
- [286] B. Li, R. Liang, D. Zhu, W. Chen, and Q. Lin, “Blockchain-based trust management model for location privacy preserving in vanet,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3765–3775, 2021. DOI: 10.1109/TITS.2020.3035869.

- [287] E. M. Mianji, G.-M. Muntean, and I. Tal, “Dynamic pseudonym management with privacy-energy trade-offs in iov networks,” in *2025 IEEE 101st Vehicular Technology Conference (VTC2025-Spring)*, 2025, pp. 1–7. DOI: 10.1109/VTC2025-Spring65109.2025.11174685.
- [288] Y. Pan and J. Li, “Cooperative pseudonym change scheme based on the number of neighbors in vanets,” *Journal of Network and Computer Applications*, vol. 36, no. 6, pp. 1599–1609, 2013.