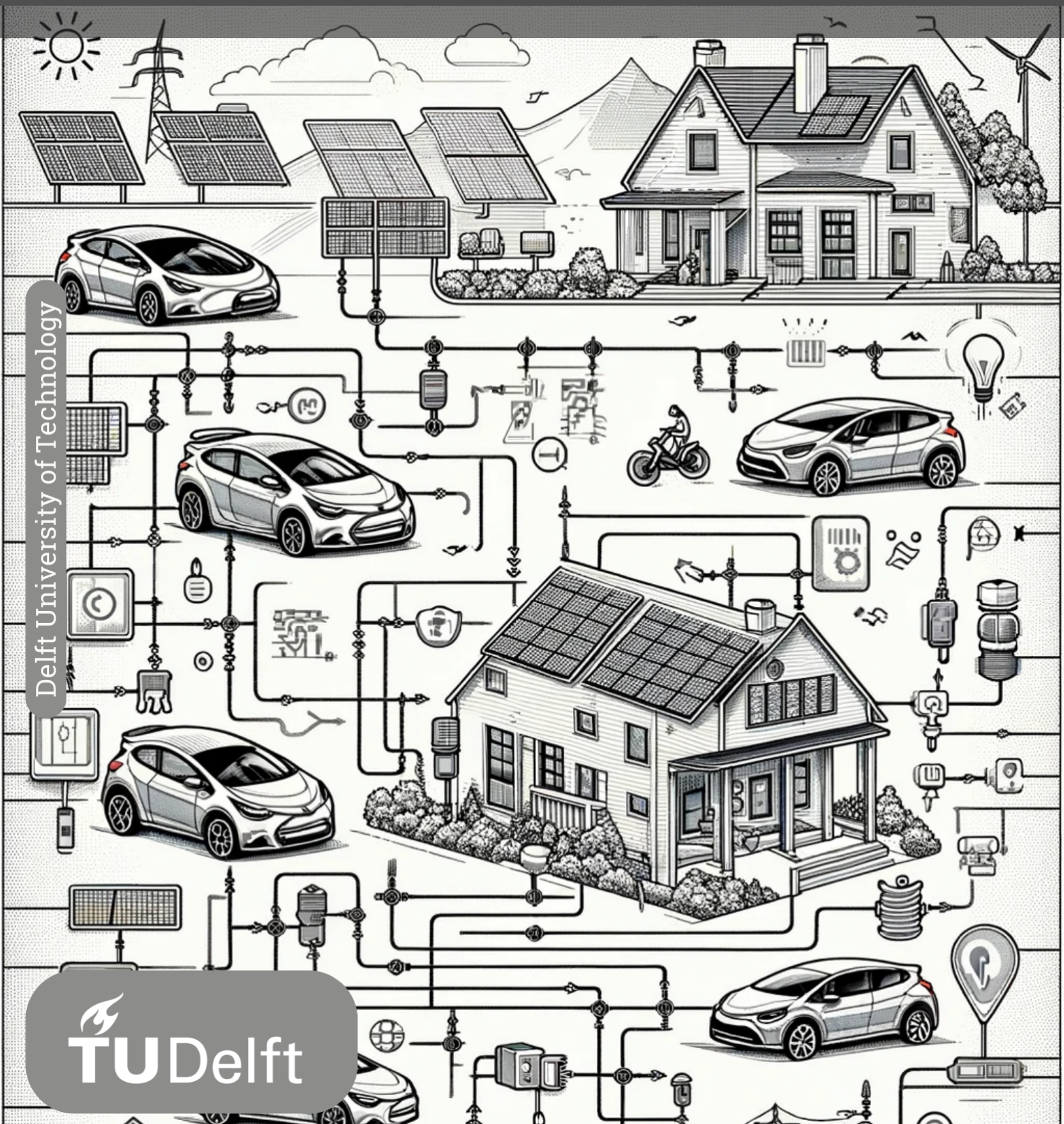
EV Charging Strategies through Power Setpoint Tracking

A Reinforcement Learning Approach

Yunus Emre Yılmaz



Delft University of Technology

TUDelft

EV Charging Strategies through Power Setpoint Tracking

A Reinforcement Learning Approach

by

Yunus Emre Yılmaz

to obtain the degree of Master of Science in Sustainable Energy Technology
at the Delft University of Technology,
to be defended publicly on Friday, May 31, 2024, at 9:00 AM.

Student number:	5859638	
Project duration:	August 28, 2023 – May 31, 2024	
Thesis committee:	Dr. PP (Pedro) Vergara Barrios	TU Delft, supervisor, EEMCS
	MSc. S. (Stavros) Orfanoudakis	TU Delft, daily co-supervisor, EEMCS
	Dr. Ir. J.L. (Jose) Rueda Torres	TU Delft, chair committee, EEMCS
	Dr. S.J. (Stefan) Pfenninger-Lee	TU Delft, committee member, TPM

TU Delft

This thesis is partly done within the Drive2X project, funded by the European Union. For more information, you can visit the corresponding webpage: https://drive2x.eu/.

Preface

This thesis presents the results of my research process, which began with learning how to conduct research and apply a novel method for optimizing EV charging, an area that was entirely new to me. During the first year of my master's program, I faced the challenge of commuting from Tilburg to Delft for classes and socializing. In retrospect, this was a demanding period. However, while writing my thesis, I had the opportunity to manage my time more effectively, allowing me to focus on my work with greater precision, which greatly motivated me.

The initial months were overwhelming as I was unfamiliar with conducting research and establishing research boundaries at the master's level. Additionally, I had to acquaint myself with Reinforcement Learning, which required an extensive learning period. Now, I can discuss EV charging for hours without hesitation, reflecting my deep passion for this topic developed through my thesis work.

I am grateful for the opportunity to engage in a research-intensive thesis, which involved an extensive literature review rather than merely applying first-year knowledge to a specific project. This process expanded my understanding of how to read research papers and identify areas needing further investigation, potentially attracting interest from relevant stakeholders. Moreover, balancing my thesis work with electives and extracurricular activities provided much-needed breaks, enabling me to approach my research with fresh perspectives from time to time.

First and foremost, I would like to thank my supervisor, Dr. Pedro P. Vergara Barrios, for his continuous guidance throughout the past year. His invaluable feedback helped me explore the field initially and conceptualize a master's-level research ultimately. I feel fortunate to have had regular progress meetings, which kept me on track weekly. Additionally, I would like to thank my daily co-supervisor, Stavros Orfanoudakis, for his support and assistance with my numerous questions, both relevant and irrelevant sometimes. Without his guidance and amazing coding skills, conducting this research would have been much more challenging. I also extend my gratitude to Dr. Ir. Jose Rueda Torres and Dr. Stefan Pfenninger-Lee for serving on the thesis committee and contributing to the evaluation of this thesis.

Lastly, I would like to thank my family, especially my aunt, for their constant encouragement and support. I also thank my friends and girlfriend for being there during both the easy and challenging times throughout this thesis journey. Have fun reading it!

Yunus Emre Yılmaz
Tilburg, Friday 31st May, 2024

Abstract

The transportation sector continues decarbonizing with the increasing number of Electric Vehicles (EVs) replacing gasoline and diesel cars every year. However, the integration of vast amounts of EVs introduces complexities in energy distribution and grid stability. Charge Point Operators (CPOs), positioned at the intersection of EVs and the grid, play a critical role in managing these complexities. They ensure that the charging infrastructure meets the needs of both EV users and the grid, highlighting the importance of smart charging strategies.

In this thesis, a smart charging approach is proposed from the point of view of a CPO. The proposed approach aims to optimize the charging schedules for EVs parked at a commercial building's parking lot. The objective of the optimization problem is to minimize the Power Setpoint Tracking (PST) error, which indicates the error between the contracted energy in the day-ahead market by the CPO and the aggregated consumption of charging stations the next day. This optimization involves complex sequential decision-making, where the uncertain nature of EV arrivals and departures demands a fast and adaptive solution. Thus, this thesis proposes a Markov Decision Process (MDP) formulation and solves it using the Deep Deterministic Policy Gradient (DDPG) algorithm to minimize the PST error by scheduling the charging of EVs. DDPG is chosen for its ability to efficiently handle complex problems with continuous state and action spaces, making it ideal, considering the uncertainties inherent to the arrival of EVs and the charging process. Additionally, DDPG's application in a commercial building's parking lot, where EV arrival and departure patterns are usually consistent, further solidifies DDPG as a strong alternative.

Evaluating the proposed DDPG approach with alternative benchmarks, such as the uncontrolled "charge as fast as possible" (CAFAP) and the optimal solution obtained through a Mixed Integer Non-Linear Programming (MINLP) formulation, signifies DDPG's superior performance in several metrics. Specifically, it outperforms the CAFAP algorithm by achieving a reduction in PST error by an average of 34% for a parking lot with 10 chargers over 12 hours of charging for a day. This highlights DDPG's efficacy in optimizing EV charging schedules over the CAFAP algorithm. Moreover, DDPG's model benefits from the ability to be trained offline with historical data and deployed online once trained. This approach allows for rapid, dynamic rescheduling of charging in real-world operations, offering speed advantages over the theoretically optimal solution, which requires prior knowledge of arrival and departure times and State of Charge (SoC) of EVs. All experiments validating these findings were conducted within the EV2Gym, a Gym environment specifically designed to simulate the EV charging scenarios.

Lastly, this thesis contributes to the field by demonstrating how RL, through the use of DDPG, can optimize PST for EV charging in a commercial building's parking lot. By offering a detailed comparison with other algorithms and showcasing the scalability and adaptability of DDPG, the research provides valuable insights for CPOs and stakeholders in the energy sector.

Contents

Preface ii

Abstract iii

Nomenclature vii

1 Introduction 1
- 1.1 EVs and Their Role Beyond Transportation . . . 2
  - 1.1.1 Smart Charging and V2G Capabilities . . . 3
- 1.2 Electricity Market Actors and Dynamics . . . 4
  - 1.2.1 EV Aggregators Role . . . 6
- 1.3 Research Objectives . . . 7
  - 1.3.1 Research Questions . . . 7
- 1.4 Thesis Outline and Methodology . . . 7

2 State of the Art EV Charging Approaches 9
- 2.1 EV Charging Optimization . . . 9
- 2.2 Reinforcement Learning Approaches . . . 10
  - 2.2.1 Reinforcement Learning . . . 10
  - 2.2.2 Reinforcement Learning in EV Charging . . . 13

3 Methodology and Model 16
- 3.1 Problem Formulation - Power Setpoint Tracking . . . 16
  - 3.1.1 Simulation Environment . . . 17
  - 3.1.2 Mathematical Model . . . 18
- 3.2 Proposed RL Formulation . . . 20
  - 3.2.1 State and Action Spaces . . . 20
  - 3.2.2 Reward Function . . . 21
  - 3.2.3 DDPG Algorithm . . . 21
  - 3.2.4 Hyperparameters . . . 24

4 Results and Discussions 28
- 4.1 Case Study - Charging at Work . . . 28
  - 4.1.1 Training and Testing Settings . . . 28
  - 4.1.2 Results . . . 32
  - 4.1.3 Transformer Capacity Limit . . . 37
  - 4.1.4 Scalability of the Algorithm . . . 37
- 4.2 Discussion . . . 41

5 Conclusion and Recommendations 43
- 5.1 Conclusion . . . 43
  - 5.1.1 Answers to the Research Questions . . . 43
  - 5.1.2 Research Objective . . . 44
- 5.2 Recommendations . . . 45

References 46

A Appendix A: Hyperparameters Tuning 51

List of Figures

1.1 EV sales and charging stations in the Netherlands between the years 2010-2022 (data retrieved from [13]) . . . 2
1.2 Dutch electricity market concept including an EV aggregator (Figure is adapted from [23] in accordance with this study) . . . 4

2.1 Markov Decision Process . . . 11

3.1 Operation of the CPO . . . 17
3.2 DDPG's operation . . . 22
3.3 Mean rewards with different action noises $\mathcal{N}$ . . . 25
3.4 Actor Loss . . . 25
3.5 Critic Loss . . . 25
3.6 DNNs architecture . . . 26

4.1 Mean rewards during training suggesting that both reward functions converged, necessitating further analysis to complete the reward function selection . . . 30
4.2 Mean rewards from 10 training sessions with selected hyperparameter set . . . 32
4.3 Power setpoints and actual power usage from one replay outputs the charging power alongside with predetermined power setpoints at each 15-minute time step throughout a day for three benchmark algorithms . . . 32
4.4 Squared tracking error throughout 100 evaluated replays . . . 33
4.5 Averages and standard deviations of squared tracking error throughout 100 evaluated replays 34
4.6 Energy tracking error throughout 100 evaluated replays . . . 34
4.7 Averages and standard deviations of energy tracking error throughout 100 evaluated replays 34
4.8 Power tracker surplus throughout 100 evaluated replays . . . 35
4.9 Averages and standard deviations of power tracker surplus throughout 100 evaluated replays 35
4.10 User satisfaction throughout 100 evaluated replays . . . 36
4.11 Averages and standard deviations of user satisfaction throughout 100 evaluated replays . . . 36
4.12 Transformer overloads throughout 100 evaluated replays . . . 37
4.13 Mean rewards for 3 chargers from 10 training sessions . . . 38
4.14 Mean rewards for 20 chargers from 10 training sessions . . . 39
4.15 Mean rewards for 50 chargers from 10 training sessions . . . 40
4.16 Energy tracking error average and standard deviations of 100 replays for all scales . . . 41

List of Tables

2.1 Literature Review of Reinforcement Learning Algorithms in EV Charging Landscape 15

3.1 List of parameters 19
3.2 Total number of registered BEVs in the Netherlands in 2023 [62, 63, 64, 65] 27

4.1 Case Study Parameters 29
4.2 Comparison Metrics 29
4.3 Reward functions evaluation 31
4.4 Best reward function search hyperparameter set 31
4.5 Selected hyperparameter set 31
4.6 Performance of Algorithms 36
4.7 Performance of algorithms for 3 chargers 38
4.8 Performance of algorithms for 20 chargers 39
4.9 Performance of algorithms for 50 chargers 41

A.1 Hyperparameter set alternatives 51

Nomenclature

Abbreviations

Abbreviation	Definition	Abbreviation	Definition
ACO	Ant Colony Optimization	PV	Photovoltaic
BEV	Battery Electric Vehicle	RES	Renewable Energy Sources
BESS	Battery Energy Storage System	RL	Reinforcement Learning
BRP	Balance Responsible Party	RVO	Rijksdienst voor Ondernemend Nederland
BSP	Balance Service Provider	SAC	Soft Actor-Critic
CAFAP	Charge As Fast As Possible	SARSA	State Action Reward State Action
CASAP	Charge As Soon As Possible	SL	Supervised Learning
CPO	Charge Point Operator	SoC	State of Charge
DDPG	Deep Deterministic Policy Gradient	SotA	State of the Art
DNN	Deep Neural Network	TRPO	Trust Region Policy Optimization
DoD	Depth of Discharge	TSO	Transmission System Operator
DQN	Deep Q-Network	UL	Unsupervised Learning
DRL	Deep Reinforcement Learning	V2B	Vehicle to Building
DSO	Distribution System Operator	V2G	Vehicle to Grid
ENTSO-e	European Network of Transmission System Operators for Electricity	V2H	Vehicle to Home
EV	Electric Vehicle	V2V	Vehicle to Vehicle
FCEV	Fuel Cell Electric Vehicle	V2X	Vehicle to Everything
ICE	Internal Combustion Engine	VPP	Virtual Power Plant
IEA	International Energy Agency		
LP	Linear Programming		
MDP	Markov Decision Process		
MILP	Mixed Integer Linear Programming		
MINLP	Mixed Integer Non-Linear Programming		
ML	Machine Learning		
NLP	Non-Linear Programming		
OU	Ornstein-Uhlenbeck		
PER	Prioritized Experience Replay		
PHEV	Plug-in Hybrid Electric Vehicle		
PPO	Proximal Policy Optimization		
PSO	Particle Swarm Optimization		
PTU	Program Time Unit		

Symbols

Symbol	Definition
A	Action space
a_t	Action at time step t
k	Charged EV
L	Loss function
M	Minibatch size
Q	Critic network
Q'	Target critic network
R	Reward functions
r_t	Reward at time step t
S	State space
s_t	State at time step t
t	Time step
α	Learning rate
γ	Discount factor
μ	Actor network
μ'	Target actor network
π	Policy
τ	Soft update
θ	Actor network parameters
φ	Critic network parameters
ε	Charged EVs set
$\mathcal{N}$	Action noise
$\mathcal{R}$	Replay buffer size
$\mathcal{U}$	Total number of episodes to train the RL agent

The symbols used to describe mathematical optimization are presented in Table 3.1 in Section 3.1.2. Similarly, symbols used in the case study are shown in Table 4.1 in Section 4.1.

1 Introduction

The goal of achieving net-zero emissions by 2050 carries significant challenges, including technological, regulatory, and societal obstacles. It requires large-scale integration of Renewable Energy Sources (RES), decarbonization of industries, and enhanced grid infrastructure, which should be more resilient and adaptive. Therefore, it can be stated that the global energy landscape will continue to evolve and will be transformed completely [1]. Parallel to the ongoing transition, the market share of electric vehicles (EVs) increases yearly. According to the International Energy Agency (IEA), EVs accounted for less than 5% of global new car sales in 2020, 9% in 2021, and rose to 14% in 2022 [2]. In addition to the continuous increase of EV sales, new regulations to ban cars with internal combustion engines (ICE) will take place in the next decade in Europe with an exception for ICEs operating on carbon-neutral fuels [3][4]. As a result, the steady growth in EV adoption will continue, and it will bring opportunities and challenges to the energy industry.

As the number of EVs increases, power grids will face challenges in meeting this demand, especially considering RES-dominated grids in the coming years. Nevertheless, there are advantages to the large-scale integration of EVs as they can act as dynamic energy storage units and can be used for grid stability and peak-load shaving through smart charging and Vehicle-to-Grid (V2G) technologies [5][6]. Furthermore, V2G technology benefits both EV users and the power grid by reducing the proportion of high-cost generators in peak times and by compensating the EV users for their services [7]. However, due to the number of uncertainties in the EV charging environment, such as EV arrival time and dynamic electricity prices [8], the stakeholders are accountable for implementing state-of-the-art (SotA) optimization techniques. As EV aggregators and Charge Point Operators (CPOs) are the intermediate entities between power grids and EV users [9], this study focuses on the problem from their point of view.

1.1 EVs and Their Role Beyond Transportation

In 2022, the transportation sector accounted for 20.7% [10] of CO_2 emissions globally, 48% of it was due to transport by cars and vans which makes them responsible for approximately 10% of global CO_2 emissions in 2022 [11]. EVs are one of the alternatives to ICE cars alongside Fuel Cell Electric Vehicles (FCEVs) for reducing emissions. However, their entry into the market has been slow over the past several decades [12].

EVs can be investigated as Battery Electric Vehicles (BEVs) and Plug-in Hybrid Electric Vehicles (PHEVs). BEVs operate with a battery pack and an electric motor. PHEVs also contain both components as BEVs, however, additionally, they have an ICE and a gas tank as well, which is why they produce CO_2. Therefore, PHEVs can be considered a transition technology from ICEs to BEVs, but they are not a promising alternative to achieving zero carbon emissions in the transportation sector. On the other hand, BEVs are the flagship of transitioning the land transportation sector to have zero emissions.

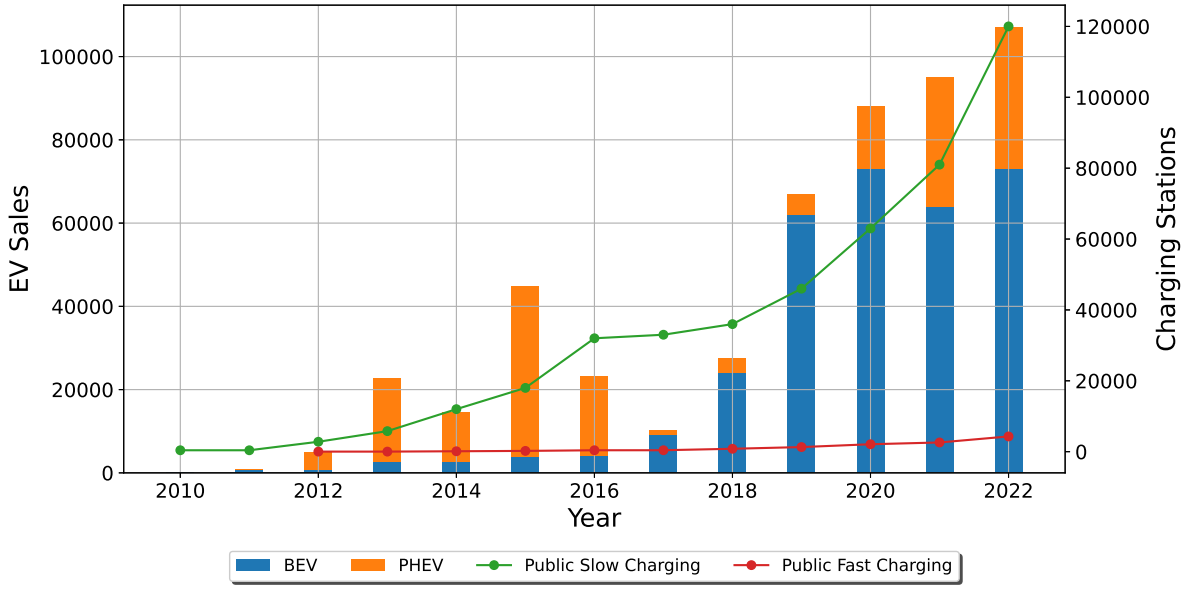


Figure 1.1: EV sales and charging stations in the Netherlands between the years 2010-2022 (data retrieved from [13])

EV sales and charging stations worldwide have been increasing in the last decade [13]. As an overview, in Figure 1.1, EV sales and cumulative installed charging station numbers can be observed for the Netherlands. It can be seen that PHEV sales were much more than BEVs in the early 2010s, however with the improvements in battery technology and building trust towards BEVs in society, BEV sales rose starting from 2017. This phenomenon can also indicate that PHEVs are the transition technology between ICEs and BEVs as mentioned before in this section. Furthermore, EV adoption has increased continuously. This increase is projected to continue by doubling the quantity in 2022 by reaching around 210,600 sales per year in 2028 [14]. As a result, EV adoption has been growing and is forecasted to grow exponentially in this decade.

Charging stations are the other side of the medallion of EV integration. Their availability is one of the main concerns when large-scale EV implementation is considered. It is worth mentioning that research conducted for Nordic countries highlights that public charging infrastructure is the third main concern against EV adoption after range and price [15]. Favourably, public charging stations have increased with EVs in the Netherlands as shown in Figure 1.1, and additionally are expected to continue increasing in the following years by almost doubling their quantity to 221,900 by 2028 [14].

It is worth noting that charging stations for EVs are not just limited to public areas. They can actually be categorized by location, such as public, home, and work. Additionally, charging stations can be classified as either fast or slow depending on their charging speed capability and the current type, whether AC or DC. AC charging is categorized into three levels based on voltage, Level 1, Level 2, and Level 3, with Level 3 having

the highest charging voltage. Level 1 and 2 chargers can be installed privately at home or workspace, whereas establishing Level 3 chargers, which necessitate distinct wiring and transformers, requires authorization from utility services and is typically done at public charging stations [16]. By considering EVs and the charging station characteristics mentioned, assumptions differ for formulating EV charging optimization problems that aim to enable the efficient use of EVs for EV owners, grid, and EV aggregators or CPOs. In the next Section 1.1.1, EVs' role beyond sustainable transportation is investigated by introducing smart charging and their V2G capabilities.

1.1.1 Smart Charging and V2G Capabilities

As the adoption of RES and EVs continues to rise, power grids will encounter vital challenges in stability and resiliency. Thus, to maintain power quality and grid resiliency, EVs' potential should be discovered. EVs have significant potential as a flexibility source for the grid, as they spend around 95% of the day parked at home or work and not in use [17]. This can be achieved through approaches such as smart charging and V2G capabilities. Thus, this section aims to explore how EVs can be integrated into the power grid efficiently by enabling the advantages of smart charging and V2G, and the challenges within are discussed.

Uncontrolled charging of EVs can lead to line overloading, so exposing the power grid to increased vulnerability [18]. This issue becomes even more important in grids with high penetration of photovoltaic systems (PVs), as they are more susceptible to disturbances [19]. In such scenarios, uncontrolled EV charging worsens the grid's fragility, increasing the risk of more significant problems and instability. To mitigate these risks, it is crucial to adopt advanced optimization techniques. These techniques can be designed not just to prevent the overloading of power lines but also to ensure the satisfaction of EV users. The objective of the problems can vary. Smart charging emerges as a key solution in this context. It involves managing the charging load of EVs in alignment with the grid's balancing needs. EV aggregators play an important role in this process by providing balancing capacity to the grid. They are examined further in Section 1.2.1.

Smart charging can be done effectively through the use of Power Setpoint Tracking (PST). This method allows EV aggregators and CPOs to allocate energy capacity in a controlled manner, instead of uncontrolled charging. In simple terms, PST is the way in which CPOs operate their EV fleet to ensure that they deliver the contracted power in the market or the power support directly contracted by the Distribution System Operators (DSOs). EV aggregators and CPOs can make these contracts in the day-ahead market, planning for the distribution of energy to EVs in the subsequent day's specific time frames. The electricity market dynamics are investigated briefly in Section 1.2. Such strategic allocation not only prevents overloads by shifting demand but also contributes to maintaining the overall stability of the grid, ensuring that the EVs' load is managed efficiently under allowed power limits.

Beyond their primary function in transportation, and additionally smart charging, EVs hold a third promising capability known as V2G. This concept expands into broader applications such as Vehicle-to-Home (V2H), Vehicle-to-Building (V2B), Vehicle-to-Vehicle (V2V), and Vehicle-to-Everything (V2X), representing the various roles that EVs can play in energy management. The specific designation of these technologies varies based on the destination and purpose of the energy and flexibility they provide. V2G allows the bi-directional energy exchange between EVs and the power grid. This technology not only includes EV charging but also enables EVs to support and stabilize the grid. By controlling the energy stored in EV batteries, bi-directional V2G significantly enhances the flexibility available to power utilities, playing a crucial role in improving the reliability and resilience of the power system [20]. However, while V2G provides greater grid flexibility compared to smart charging, it also poses challenges concerning battery health. Rapid charging and discharging cycles associated with V2G can accelerate battery degradation, as indicated in [21]. These findings suggest that the frequent use of V2G technology might need to be balanced against its long-term impact on battery health and lifespan.

To sum up, smart charging and V2G are both promising concepts for benefiting large-scale EV integration, yet they present distinct tradeoffs. Smart charging primarily focuses on optimizing the timing and rate of EV charging to align with grid demands, reducing peak loads and load valleys without discharging the vehicles. Its primary advantage lies in its simplicity and lower impact on battery health, as it avoids the ongoing

charge-discharge cycles that can shorten battery lifespan. In contrast, V2G offers a more dynamic solution by enabling bi-directional energy flow between EVs and the power grid. This allows EVs to not only draw energy for charging but also supply energy back to the grid, providing greater flexibility and support for the grid, which is crucial in a RES-dominated grid. However, this comes at the cost of decreasing the EVs' battery life due to more frequent charging and discharging cycles, potentially leading to quicker battery degradation. Therefore, while V2G offers broader benefits for grid management and energy optimization, it must be evaluated against the implications for battery lifespan, which is not a concern in the case of smart charging.

1.2 Electricity Market Actors and Dynamics

In this section, firstly, the dynamics of the Dutch electricity market are examined briefly, with an emphasis on identifying and understanding the key actors and their roles. Subsequently, the role of EV aggregators and CPOs' in the market is investigated by explaining their operational roles and their capabilities for implementing optimization techniques to benefit EV grid interaction in Section 1.2.1.

Roles within the electricity system can be categorized into three areas as physical, administrative, and market. Figure 1.2 shows an overview of the Dutch electricity market with the key actors. At the bottom of Figure 1.2, the physical layer can be observed, it is related to the direct handling of electricity production, transportation, and consumption. In the administrative area, there are entities responsible for managing the interactions between consumers, the market, and grid operators, they can be seen in the blue boxes in which black arrows are connected. These actors oversee tasks like monitoring energy consumption, production, and customer billing. Lastly, the market domain includes various market platforms that facilitate all these transactions [22], as it can be seen in the green wholesale market box in Figure 1.2.

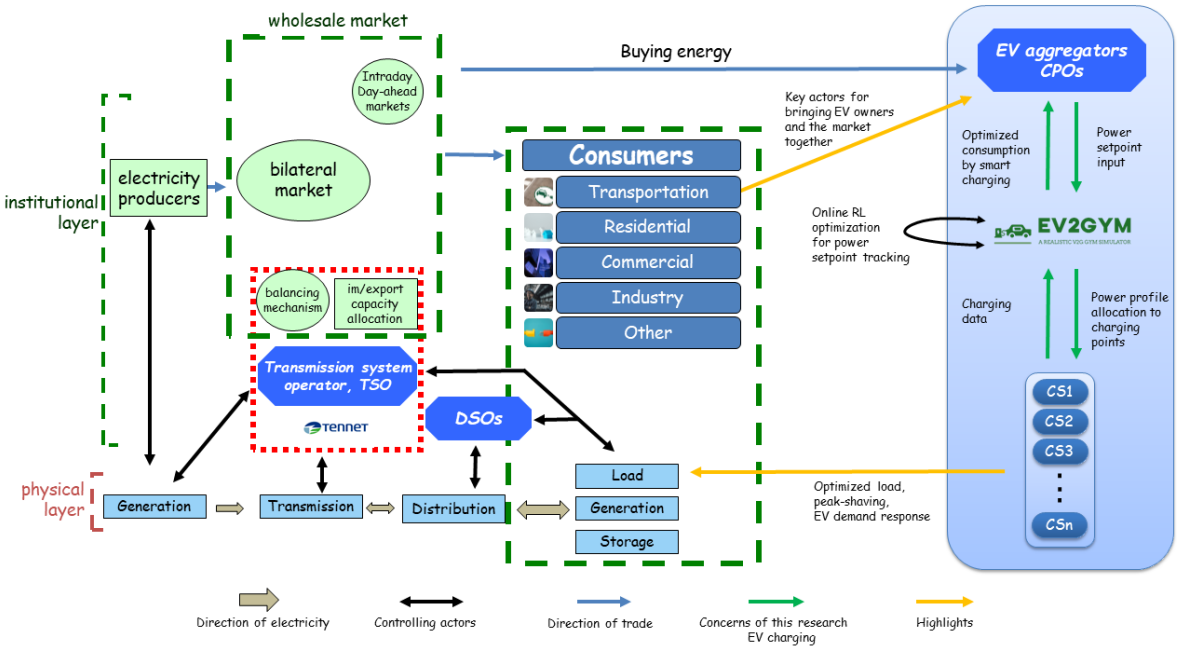


Figure 1.2: Dutch electricity market concept including an EV aggregator (Figure is adapted from [23] in accordance with this study)

In the physical layer of the electricity system, electricity producers and consumers directly interact with the grid, by creating demand and supply. As it can be observed from Figure 1.2 in light blue rectangle boxes, the physical layer includes generation, transmission, distribution, and consumer aspects such as load, generation, and storage. Key to managing these are the Transmission System Operators (TSOs) and DSOs. TenneT is the Dutch TSO and is responsible for the high-voltage grid, focusing on maintaining a balance between electricity supply and demand and inspecting the grid's planning, maintenance, and expansion. Additionally, TenneT is also responsible for connecting the Dutch grid to the European grid [24] for the import and export

of electricity. On the other hand, DSOs manage regional distribution grids. DSOs are responsible for the delivery of electricity from high-voltage to the end users and handle regional grid planning, construction, maintenance, and operation of medium and low-voltage grids. They also connect new grid participants and measure consumption for smaller consumers [22]. In addition to TSOs' and DSOs' physical layer responsibilities, their job description and influence on the grid also make them actors in the administrative area.

In the administrative area, Balance Responsible Parties (BRPs) hold financial responsibility for ensuring balance in their energy portfolios and interacting with balancing markets for grid stability. Furthermore, electricity suppliers deal with consumer contracts and participate in various markets to meet supply needs. Additionally, Balance Service Providers (BSPs) support grid balance and congestion management. Aggregators, meanwhile, gather small-scale production and consumption for larger market engagement [22], which makes aggregators crucial players in demand management within their capabilities.

Market area, can be divided into four categories as forward, day-ahead, intraday, and balancing markets according to the duration between the contracted time of capacity and the actual delivery time of electricity.

In the forward market, consumer companies are contracting and reserving a capacity for their future in weeks, months, or years. They sign bilateral contracts with the suppliers, and this is usually profitable and applicable for large consumer companies in sectors such as chemicals and steel production, which require high amounts of energy, and usually, these consumers can foresee how much energy they will need.

Secondly, in the **day-ahead market**, power capacity is contracted for the next day, 24 hours, with a distinct price for each hour. This trade is done by auction. Supplier and consumer companies bid in the market for each hour of the next day. According to the offered energy amount and bid prices, market clearing happens before the next day, thus the price is determined marginally. It is worth mentioning that, EV aggregators are highly active actors in the day-ahead market for reserving the next day's energy amount with a lower price in comparison to the price they might get in the intraday market. That is where price forecasting for the next day becomes a vital aspect for bidding precisely to prevent getting higher electricity price offers.

Consecutively, **in the intraday market**, participants have the flexibility to modify their spot market positions until five minutes before the actual delivery of electricity. Generally, purchasing electricity in the intraday market tends to be costlier compared to the day-ahead market. The intraday market operates every hour of the week, so it allows participants to change their positions immediately in response to new information [24].

Lastly, **the balancing market** is where three actors of the electricity system, the TSO, BRPs and BSPs work together to keep the power system stable. The frequency of the system is the indicator of its stability. However, the level of frequency varies according to the standards of different grids around the world, in this regard, the European grid requires 50 Hertz, that is why the balancing market aims to keep the grid frequency at 50 Hertz steadily in the Netherlands. The balancing market is built on three main pillars: balancing responsibility before the generation, providing balancing services during the operation, and settling imbalances after the system is online [25].

The balance responsibility is about planning and scheduling how much electricity will be produced for each Program Time Unit (PTU) on the delivery day. Deviations from the scheduled program would result in recompenses during the imbalance settlement step [25]. Consequently, BRPs communicate their production plans to TSOs with the objective of adhering closely to these plans, thereby mitigating the potential for incurring additional costs associated with imbalances.

Furthermore, the provision of balancing services involves offering and utilizing services to achieve real-time energy balance in the system. These services come in various forms, including balancing energy and reserve capacity. For instance, balancing energy services may involve upward or downward regulation, depending on whether there is a need for more or less power. Conversely, an example of reserve capacity is the use of EV aggregators. These aggregators can manage vast amounts of EVs by employing them as Virtual Power Plants (VPPs) to contribute to the system's balance.

The final pillar, imbalance settlement, is about sorting out the costs when there is a difference between planned and actual electricity use. When these differences are a result of deviations from the BRPs' schedules, they are financially accountable to the TSO, as the TSO is the main buyer of these balancing services from BSPs. Additionally, it is worth mentioning that, the price for this difference, called the imbalance price, is determined for each PTU [25].

Lastly, fixing imbalances involves three stages as primary, secondary, and tertiary reserves, and they are distinguished by their activation times and functions. Primary reserves respond within seconds to initial supply-demand mismatches to prevent frequency deviations. If imbalances continue, secondary reserves are activated for a duration of one to several minutes to restore the system frequency. For longer-lasting deviations, tertiary reserves are employed, which are capable of being activated for several minutes to hours to return the operation to schedule.

This section presented the actors involved in the Dutch electricity market and how they cooperate. The focus is on key actors like TSOs, DSOs, BRPs and BSPs. Various electricity markets are covered, from long-term contracts in forward markets to real-time adjustments in intraday markets. The balancing market's critical role in maintaining grid stability is also discussed. The following section will provide more detailed information on the specific roles and functions of EV aggregators and CPOs within this market framework.

1.2.1 EV Aggregators Role

EV aggregators are the intermediate actors between the market and the end-user, which is why they hold a crucial position for integrating EVs into the power grid. Numerous research findings, including those by Lund et al. [26], indicate that end users typically show minimal enthusiasm for actively managing their assets. In addition to that, according to the EV aggregator model of Kempton et al. [27], individual EV owners are unable to participate in bidding within the electricity market or conduct transactions with electrical utilities, primarily due to their lower power capacity. Thus, EV aggregators occur as a solution. Another proposition for EV aggregators is highlighted in [28], EV aggregator compiles the driving profiles of users to establish a VPP. This system predicts the number of vehicles likely to be connected to the grid at various times throughout the day and also predicts the available electrical energy and power capacity; thus, the capacity of connected EVs can be used for grid services. Additional research [16] suggests that to integrate a substantial number of EVs into the grid effectively, the introduction of EV aggregators is essential due to the market impact of a single EV being minor and unpredictable, but this can be enhanced by collectively operating EVs through an aggregator, similar to Kempton's model. To sum up, it can be concluded that one of the primary purposes of aggregators is their capability to link their customers' assets to the market with minimal transaction costs. As de Vries [29] states, the essential strategy lies in employing advanced automated and optimized solutions to capitalize on the flexibility offered by customers and doing so without compromising their operational efficiency.

Next to EV aggregators, CPOs also play an essential role in EV-grid integration. They both serve distinct yet similar services. CPOs are primarily focused on the infrastructure, installing, operating, and maintaining EV charging stations. A key aspect of their role is to manage the hardware of charging stations, along with the related software for user authentication, billing, and remote monitoring. EV aggregators, on the other hand, are more active in market interactions. Despite these distinct roles, there is potential overlap, particularly in areas of smart charging and grid management. Just like EV aggregators, CPOs are also capable of implementing EV smart charging concepts.

In this study, the concepts of smart charging will be explored under the assumption that a smart charging solution is implemented by a CPO controlling a charging station located in a commercial building's parking lot. The concerning area inside the Dutch Electricity Market framework in Figure 1.2 is shown in the light-blue box, which includes the CPOs and the EV2Gym Simulator [30]. Consequently, the subsequent sections will frequently use the term "CPO" to refer to both CPOs and EV aggregators since they are practically similar under the assumptions of this study.

1.3 Research Objectives

In this section, the main research question is outlined, accompanied by several questions. These questions were created to explore different aspects of the objective and to address the challenges in the field through a structured approach.

The uncertainties in EV charging caused by unpredictable driving behaviours and fluctuating electricity prices pose significant challenges for EV charging optimization problems. Reinforcement Learning (RL) algorithms are recognized as effective approaches for addressing complex sequential decision-making problems. Furthermore, CPOs are capable of directly controlling the charging process of EVs and applying these optimization methods, taking into account the power grid's constraints and the EV users' requirements. The goal of this study is to tackle the problem of scheduling EV charging sessions in a parking lot located in a workplace using RL algorithms. The smart charging algorithm's aim is to optimize the EV charging schedules in real-time to meet the power set point specified by the contracted power capacity of a CPO. Therefore, the main research question can be formulated as follows:

How to effectively optimize the charging schedules of EVs to meet the CPO's contracted power setpoint in a workplace setting using RL algorithms?

1.3.1 Research Questions

1. What are the key characteristics and constraints of the model-free online EV charging problem in the context of a workplace parking lot?

2. What are the key factors influencing the performance of the Deep Deterministic Policy Gradient (DDPG) algorithm in optimizing power setpoint tracking (PST) for EV smart charging?

3. How do RL-based smart charging methods improve upon or differ from mathematical optimization methods used for smart charging in managing the energy demands and grid interactions of EVs?

4. How does the applied RL method scale with the varying number of EV chargers?

1.4 Thesis Outline and Methodology

This thesis presents a comprehensive exploration of EV charging approaches from the CPOs' point of view, focusing on RL methods while comparing them to classic optimization techniques. Chapter 2 delves into the SotA EV charging optimization approaches. It begins with an examination of Classic and Metaheuristic optimization methods in Section 2.1, discussing their application and capabilities in EV charging. The chapter then continues with RL approaches in Section 2.2, initially providing an overview of RL principles before specifically addressing its application in the context of EV charging in Section 2.2.2.

Chapter 3 outlines the methodology and modeling approaches used in this study. It describes the simulation environment for testing and analysis, followed by a detailed discussion of the DDPG algorithm in Section 3.2, highlighting its advantages and implementation in the context of this dissertation.

The paper then progresses to Chapter 4, which starts with a case study, "Charging at Work". This case study is designed to provide practical insights and real-world applicability of the discussed EV charging strategies, particularly showcasing the implementation and outcomes of the DDPG approach. This chapter analyzes the data obtained from the case studies, comparing the performance of the proposed DDPG algorithm with mathematical optimization and an uncontrolled "charge as fast as possible" (CAFAP) scenario. This analysis aims to conclude the efficacy and practicality of the RL approach in EV smart charging. Chapter 4 ends with the discussion part in Section 4.2.

Finally, Chapter 5 concludes the thesis, summarizing the key findings and offering recommendations based

on the research. This chapter aims to provide a clear understanding of the potential and limitations of the studied EV charging approaches and suggests directions for future research.

2

State of the Art EV Charging Approaches

This chapter provides an overview of the current EV charging/discharging optimization techniques. Firstly in Section 2.1, EV charging optimization techniques are examined briefly, in specific classic and metaheuristic approaches. Following that in Section 2.2, RL approaches of the EV charging scheduling problem are investigated in detail and there is an overview of the literature at the end of the chapter in Table 2.1.

2.1 EV Charging Optimization

Optimization is described as the mathematical process of identifying the inputs for functions with variable values, which are aimed to be either maximized or minimized, and are subject to a range of constraints [31]. In EV charging, the objective can vary from profit maximization of charging stations to minimizing peak load or maximization of renewable usage to minimization of charging costs. In this regard, various optimization strategies have been employed to solve EV charging scheduling problems. These strategies can be examined under three categories: Classic, Metaheuristic, and Machine Learning (ML), each with distinct strengths and weaknesses based on their capabilities and the characteristics of the problems they aim to solve.

Various studies have shown that the optimization strategies mentioned above can yield promising results. Starting with Classic Optimization, which is usually based on solvers that rely on the estimation of the gradient of the objective function, in [32], the problem was formulated as a Mixed Integer Linear Programming (MILP) problem. The MILP formulation was used to optimize the profits of a public charging station with PV and a Battery Energy Storage System (BESS) by forecasting the PV output from historical energy generation of PVs in a specific time frame and EV arrival times by normalized EV power demand data. As a result of the optimization and integration of BESS, the daily profits of the station increased by 82.8%. Similarly, in [33], EV charging/discharging rate and schedule were formulated as a MILP problem. Differently, this problem was formulated for a community forming a microgrid with PVs and a BESS. It was found that 33.4% operational cost reduction is possible for the system by using the proposed optimization technique. Furthermore, van der Meer et al. [34] achieved 118% charging cost reductions turning costs into profits, in comparison to uncontrolled charging, for one charging point at a workplace by formulating the EV charging scheduling as a MILP problem for profit maximization and PV utilization. Besides the MILP formulation, Ioakimidis et al. [35] formulated an EV charging/discharging scheduling problem as Linear Programming (LP) for peak load shaving and valley filling of the consumption of a non-residential building. As a result, peak power consumption was decreased, varying from 4% to 20% according to the number of participating cars. These studies highlight how Classic optimization with different problem formulations, like MILP and LP, improves profits, reduces costs, and manages energy use in EV charging. However, due to the constantly changing demands for EV charging and the variability of RES, Classic Optimization may not always provide the most effective solution. In this case, RL can be a valuable alternative by offering a flexible framework that can adapt to real-time changes in EV charging and fluctuating RES. By utilizing a trial-and-error learning mechanism, RL can optimize charging schedules in an environment where traditional methods may struggle

to account for the complexities and uncertainties inherent to EV charging optimization problems.

Nevertheless, Metaheuristic Optimization is widely used, similar to Classic Optimization, for solving EV charging scheduling problems. Metaheuristics sample solutions and evolve them using various approaches to develop higher-quality solutions. However, in contrast to Classic Optimization, Metaheuristics do not guarantee convergence or optimality. Wang et al. [36], introduced an Ant Colony Optimization (ACO) for 500 EVs to fill load valleys at the transformer level. This resulted in a decrease in the peak valley difference by 74.8%. Moreover, Celli et al. [37], deployed a Particle Swarm Optimization (PSO) to reduce peak load and losses in the grid by controlling charging/discharging schedules of 200 EVs through an EV aggregator. As a result, peak load was reduced by around 10% and losses by 3%. Considering the scales of these problems, it can be stated that Metaheuristic Optimization methods, like ACO and PSO, are effective in large-scale scenarios. However, Metaheuristics may not always adapt efficiently to the uncertainties in the EV charging environment. On the other hand, RL has the potential to offer a better approach to EV charging optimization. RL can adapt and learn continuously and adjust strategies in real-time based on energy supply and demand data. This makes RL a promising alternative for dealing with the complexities of EV charging optimization.

As an outcome, Classic Optimization of EV charging with problem formulations such as MILP, LP, Non-Linear Programming (NLP) and their variants are designed to provide optimal solutions. However, they often face limitations, particularly in terms of scalability and their ability to handle highly complex problems due to uncertainties such as EVs' arrival time, departure time, and electricity prices. There also occurs a trade-off between solving a problem faster or with higher accuracy. In this regard, Classic Optimization can require high memory capacity and longer times to solve a complex problem. In contrast, metaheuristic approaches offer more flexibility and are generally better suited for complex, large-scale problems; however, they usually can not guarantee convergence or finding the global optimal solutions. This trade-off makes Metaheuristics particularly valuable in situations where an optimum solution is less critical than a faster and reasonably effective one.

The third category, ML Optimization, particularly RL techniques, is showing potential in providing solutions to the challenges associated with EV charging. In real-world scenarios, accurately modelling randomness is difficult due to various external factors, such as difficulties in modelling driving behaviours and fluctuating electricity prices. As Qui et al. [38] state, RL addresses the challenges in modelling randomness by adopting a data-driven approach; thus, it does not rely on exact models of uncertainties but learns from experiences by getting a vast amount of interactions within the created simulation environments to train the RL models. Furthermore, in RL, the model learns the best actions through exploration and exploitation, allowing it to manage and optimize charging schedules with respect to the algorithm's objective. The objective can be maximizing the use of RES, reducing costs, or minimizing PST error, which is the proposed objective in this thesis. In addition to the ability to have variable objectives, RL can balance different goals, such as reducing costs while managing peak demand, thanks to its multi-objective optimization capability. This capability makes RL a more comprehensive approach to solving EV charging problems. Moreover, RL's adaptability and learning ability make it a promising alternative for modelling uncertainties inherent in EV charging. The next Section 2.2, is focused on these RL techniques to improve the approach to solving EV charging problems.

2.2 Reinforcement Learning Approaches

In this section, firstly ML techniques are explained briefly in Section 2.2.1. Following that the mathematical framework of RL, Markov Decision Process (MDP) is examined. Consecutively, value-based and policy-based RL methods are discussed briefly. Finally, the advantages and limitations of RL methods for EV charging are investigated further in Section 2.2.2. This chapter ends with Table 2.1, concluding the literature review by highlighting SotA RL approaches in the literature which represents an outlook on EV charging with RL techniques.

2.2.1 Reinforcement Learning

RL is one of the three branches of Machine Learning (ML) with Supervised (SL) and Unsupervised Learning (UL). RL differs from the remaining ML branches in several aspects. In SL, models are trained with labeled

data sets, with the purpose of accurate classification or prediction of results. These models can measure their performance and learn by comparing labelled inputs and outputs to their results. [39]. Consecutively, in UL, there are no labelled data sets; in fact, they are used for grouping and examining data sets without labels. Thus, their goal is to recognize patterns in data sets [39]. Unlike these methods, in RL, an agent takes actions in an environment to maximize its cumulative reward, so the agent is not told which action to take, on the contrary, it must learn which actions are and will be the most rewarding by trial and error [40]. Therefore, RL is considered a model-free method and can find policies and strategies by utilizing itself through trial and error in various environments without prior knowledge. MDP is the mathematical ground of RL, and it is explained under Section 2.2.1 to show how an MDP can be formed.

Markov Decision Process

MDP is a classic approach to sequential decision-making, where actions impact both immediate and future rewards, necessitating a balance between short-term gains and long-term benefits [40]. It can also be described as the mathematical framework of RL.

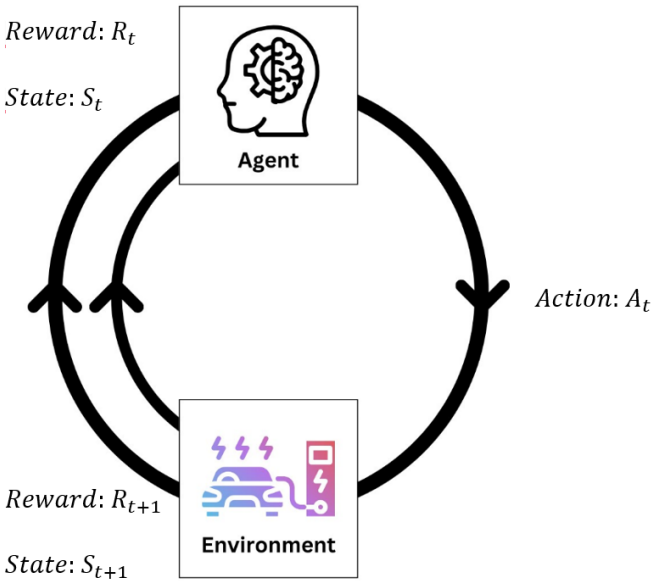


Figure 2.1: Markov Decision Process

Figure 2.1 demonstrates the flow of an MDP. The agent is the decision maker. It interacts with the environment through actions, (a_t), which in turn change the agent's state from s_t to s_{t+1} and rewards the agent based on a predefined reward function (r_t). In the simulation's following episodes or next-time steps, the agent retains its previous state (s_t), the taken action (a_t), the received reward (r_t), and a state transition function, $p(s_{t+1}|s_t, a_t)$. The state transition function is used by the environment to produce the subsequent states, and eventually, it serves as a link that maps the combination of the current state (s_t) and the taken action (a_t) to a resulting new state (s_{t+1}). Through these interactions, the agent learns to find an optimal policy which $\pi_t(a|s)$ is the probability that $a_t = a$ and $s_t = s$, creating a probability distribution across all possible actions (a) in the state (s). While applying this policy, MDP continues until the defined step or episode limit.

Additionally, the agent also stores a discount factor $\gamma \in [0, 1]$, which is used to tune the amount of rewards that the agent can get in the later episodes of the simulation to find a balance between exploration and exploitation in the environment. The cumulative discounted reward is calculated with Equation (2.1) below:

$$G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \tag{2.1}$$

Having the discount factor value 1 indicates that the future rewards are as valuable as the present rewards. Thus, the agent is more prone to exploring than exploiting. On the other hand, if the agent exploits too soon, with a small discount factor, then the result might converge without exploring the environment enough, which might result in not obtaining the targeted results. Overall, the agent's goal is to maximize its total discounted rewards by acting upon the environment and ultimately seeking to discover an optimal policy that maximizes this reward. Following that, a state-value function determines how good it is for the agent to be in a given state and is defined to represent the expected return using policy π with Equation (2.2). $\mathbb{E}_\pi$ denotes the expected value given that the agent follows policy π.

$$V_\pi(s) = \mathbb{E}_\pi[G_t|S_t = s] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| S_t = s \right], \quad \text{for all } s \in \mathcal{S} \tag{2.2}$$

Besides the state-value function, the action-value function determines how good it is to perform a given action in a given state. It shows the value of taking an action a starting from state s, by following a policy π considering the expected return. It is denoted as $Q_\pi(s, a)$ and can be observed from Equation (2.3).

$$Q_\pi(s, a) = \mathbb{E}_\pi[G_t|S_t = s, A_t = a] = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \middle| S_t = s, A_t = a \right] \tag{2.3}$$

MDPs are utilized in various RL approaches. These approaches can be divided into two categories as value-based and policy-based. While value-based methods maximize the action-value functions, policy-based methods directly optimize the policy itself, thus they are categorized by a fundamental difference. RL techniques within these categories are briefly discussed in the following sections.

Value Based Methods

Value-based methods focus on estimating the state-value and action-value functions in an environment to determine the best policy for an agent. In the value-based category, Q-learning is an algorithm for action-value function learning. It directly approximates the optimal action-value function. In addition to Q-learning, the Deep Q-Network (DQN) uses Deep Neural Networks (DNNs) [41] to approximate the action-value function. Furthermore, this category also includes algorithms like State-Action-Reward-State-Action (SARSA) and fitted Q-iteration. These methods are actively used in EV charging problems, as can be observed in Table 2.1.

One crucial difference between value-based and policy-based RL techniques is their ability to operate in both discrete and continuous state and action spaces. Discrete and continuous action spaces can be summed up with a simple analogy question. For discrete action spaces, the question is "Which direction should I move?" on the other hand, for continuous action spaces, it is "How fast should I move?". Q-learning becomes highly ineffective when applied to tasks requiring continuous action spaces, as it is significantly affected by the curse of dimensionality [41]. DQN overcomes Q-learning's discrete state space limitation by using DNNs with parameters to approximate the Q-value function in a continuous state space [38][41]. Additionally, in such cases, alternative approaches like policy-based methods or actor-critic methods might be more suitable.

Policy Based Methods

Policy-based methods offer a different approach compared to value-based methods. These methods, directly optimize the policy that an agent follows to make decisions. As Qui et al. [38] state the main advantage of the policy-based methods is that they focus on directly achieving the desired outcome, unlike value-based methods, which indirectly optimize performance through self-consistency equations by training the action-value function $Q_\pi(s, a)$. This indirect approach often leads to instability and various failure modes. On the other hand, when value-based methods are successful, they tend to be more sample efficient, reusing data

more effectively than policy-based methods [38].

Methods in the policy-based category include Actor-Critic methods that combine policy-based and value-based strategies. Soft Actor-Critic (SAC) is an off-policy algorithm that optimizes a stochastic policy. Furthermore, Proximal Policy Optimization (PPO) and Trust Region Policy Optimization (TRPO) both aim to balance policy updates with stability in learning. Additionally, DDPG is effective for high-dimensional problems with continuous state and action spaces [42], thus it is a promising alternative to experiment on EV charging problems.

2.2.2 Reinforcement Learning in EV Charging

EV charging problems mainly focus on finding more efficient and profitable ways to charge EVs, taking into account various constraints such as user satisfaction and grid compliance. These problems can be categorized into three main areas. The first one is mobility, which considers whether the EVs are moving or connected to a charging station. If the EVs are parked, the associated challenges are referred to as the static problem. In contrast, if they are on the road, the challenges are referred to as the dynamic problem. The second area is optimization, which distinguishes between online and offline approaches. The offline approach assumes that the operator has complete knowledge of the system, while the online approach has to deal with uncertainties in the environment. In terms of RL, in the offline approach, the RL model is not trained during the operation; on the contrary, the RL model is trained continuously in the online approach. The third area is the control method, which determines whether the charging process is managed centrally or in a decentralized manner. In a centralized manner, CPOs can solve the EV charging optimization problem for several charging stations, while in a decentralized manner, each station finds the optimal solution for its location.

Nevertheless, these characteristics of the EV charging problem cause various dynamics and lots of uncertainties. Numerous studies [43][44][45] have been conducted to model the behaviour of EV users to solve the dynamic problem concerning the mobility category of EV charging problems. These efforts aim to address the dynamic problem by more accurately predicting the potential for charging and discharging. However, the significant uncertainties involved in dynamic charging in contrast to static charging, such as traffic conditions and modelling user behaviour, greatly complicate the problem-solving process.

RL, a model-free and online learning method, can capture various uncertainties through numerous interactions with the environment and adapt to state conditions in real time [46]. As a result, using advanced RL algorithms to solve various EV charging optimization problems has attracted attention in recent years, leading to many outstanding research papers and important findings. In Table 2.1, there is an overview of the studies in the literature about RL applications for solving the EV charging problem.

The objectives of studies in the literature vary from maximizing the profits of either CPOs or EV owners to meeting the scheduled target load. Wan et al. [47], for instance, formulated a problem to tackle the challenge of maximizing EV owner profits also by exploring V2G capabilities. A proposed Deep RL (DRL) method was used to solve the problem for a residential EV charger, and it learned to charge one EV when the electricity prices are low and discharge when they are high, which maximizes the EV owner's profit and helps load flattening. However, in this study, discrete charging/discharging levels were used. Thus, it does not represent the EV charging problem completely. Additionally, another research by Hao et al. [48] proposed a DQN algorithm to minimize EV owners' costs, similar to the previous study by Wan et al.. Differently, the charging schedule was made with information on the real-world driving patterns of EV users. It was found out that the EV users could pay 98% less compared to the charge as soon as possible (CASAP) scenario. However, also in this study, discrete action spaces were used, and thus, it does not completely represent the continuous action spaces of the EV charging problem. As Mnih et al. [41] stated, in contrast to value-based RL, policy-based RL methods like DDPG show great potential to overcome continuous action problems and result in better scores on real-time tasks.

The last research included in Table 2.1 for residential settings, by Qui et al. [49], explores profit maximization for EV aggregators by combining DDPG and a strategy called Prioritized Experience Replay (PER), as they named PDDPG. By using this method, they aimed to model the problem in multidimensional continuous

state and action spaces which outperformed the DQN, DDPG, and Q-learning methods for a scale of 1000 residential EV chargers.

Furthermore, for public charging, Wang et al. [50], aimed for profit maximization of a public charging station using Future Reserving SARSA. Different from most of the other methods, the proposed method does not need distributional information, and it showed 138.5% higher profit than methods such as Greedy policy or SAA-based Monte Carlo sampling techniques. Furthermore, Zhang et al. [51], proposed a method called charging control DDPG (CDDPG) which aimed to minimize EV users' charging costs while satisfying the desired battery capacity. The proposed method outperformed DDPG and DQN. Similarly, Li et al. [52], proposed a DDPG algorithm with additional adjustments. The proposed recurrent DDPG (RDDPG) method showed promising results in terms of scalability, and it was claimed that the method can be applied to larger scales without retraining the model, which is a significant outcome.

As an example of applied value-based methods, Lee et al. [53], proposed a DQN method to reduce charging costs while flattening the load of a specific charger. It was assumed that a charging pattern should be extracted from a specific charger, highlighting that EV user patterns vary according to different locations. Hence, it might not be very effective to use a trained model within a specific location for another location. On the other hand, finding a charging pattern for each charger would be more effective for finding local optimal solutions [53]. However, by this decentralized approach, the grid service capabilities of large-scale EVs can not made use of.

In contrast to the mentioned research, Sadeghianpourhamami et al. [54], did not focus on profits or costs but on meeting the target load schedule by proposing a fitted Q-learning method. Thus it is much more similar to the proposed method PST in this study. Furthermore, Sadeghianpourhamami et al. also targeted optimizing several charging stations with different characteristics in a centralized and scalable manner.

In summary, the research detailed in Table 2.1 points to a common goal in the EV charging field, making the process more profitable and cost-effective. These studies show that smarter systems capable of adjusting to the inherent uncertainties of the EV charging problem are required. Value-based RL methods served as a starting point; however, the trend is shifting towards more complex methods such as DDPG and its variants due to their ability to better emulate the EV charging problem by handling the continuous state and action spaces, considering the continuous charging levels of EVs. The adoption of these advanced methods is crucial for addressing the uncertain nature of EV charging.

Table 2.1: Literature Review of Reinforcement Learning Algorithms in EV Charging Landscape

Author	Objective	Charger Location and number	Method	Performance Evaluation	Mobility	V2G
Wan et al., 2018 [47]	Maximization of EV owner profits and satisfaction	Residential, 1	DRL	Fitted Q iteration, Theoretical limit(YALMIP)	Static	Yes
Wang et al., 2018 [50]	Profit maximization of a public charging station	Public	Future Reserving SARSA	SAA-based on Monte Carlo sampling techniques-Greedy policy	Static	No
Hao et al., 2023 [48]	Minimize EV owner charging costs	Residential	DQN	Theoretical optimum, Greedy, DQN with known departure time, CASAP(Charging asap)	Dynamic	Yes
Li et al., 2023 [52]	Reduce EVs charging cost for the CPO	Public	Recurrent - DDPG	Uncontrolled, Day-ahead scheduling, DQN, DDPG, MA-DDPG	Static	Yes
Sadeghianpourhamami et al., 2020 [54]	Meeting target load schedule(DR) within groups of charging stations, PSPT	Public, 10 stations, 50 stations	Fitted Q-iteration	Theoretical optimum, uncontrolled charging	Static	No
Lee et al. 2020 [53]	Reducing charging costs and increasing load-shifting capability of a specific charger	Public, 1	DQN with KDE	Uncontrolled	Static	Yes
Zhang et al., 2021 [51]	Minimizing user's charging expense	Public, 1	CDDPG	DDPG, DQN	Static	Yes
Qui et al., 2020 [49]	Profit maximization of EV aggregator	Residential, 1000	PDDPG	DQN, DDPG, Q-learning	Dynamic	Yes

3

Methodology and Model

This chapter delves into the problem formulation first in Section 3.1, including a description of the simulation environment and the mathematical model of the PST problem. Consecutively, the proposed RL method, DDPG, and how it was used for the specific PST problem is explained in Section 3.2, also including state and reward functions and the hyperparameters of the DDPG algorithm.

3.1 Problem Formulation - Power Setpoint Tracking

PST is a crucial way of scheduling EVs' charging by a CPO. CPO is responsible for buying energy to maintain the operation of its respective chargers. The energy is usually bought in the day ahead market by a CPO and allocated to EVs the next day via EV chargers. Thus, CPOs run their optimization algorithms to find out how much energy they should contract in the day ahead market for various objectives such as maximization of profits or EV owner's satisfaction. According to the amount of contracted energy, CPOs determine a power setpoint for a specific time frame of the next day, which is disaggregated to EVs by scheduling their charging process. It is essential to disaggregate the energy carefully to avoid exceeding the power setpoints. If the CPO exceeds the power setpoints, then the CPO has to make a tough decision: either purchase additional energy at a higher cost in the intra-day or balancing market or accept the risk of unsatisfied EV users due to their EVs not being fully charged. This situation highlights the necessity of finding a way to minimize the PST error by scheduling EV charging.

Minimization of the PST error is the second optimization that CPOs need to do after the determination of power setpoints to maintain profitable operations. Furthermore, minimizing the PST error can prevent any imbalance, allowing money to be saved by charging EVs when electricity prices are lower and avoiding charging during peak price times. It is important to note that optimizing the amount of power to be traded in the energy market is outside the scope of this thesis. The focus here is solely on the disaggregation of energy once such a power setpoint is defined.

Figure 3.1 illustrates the formulation of this study. In Figure 3.1, a CPO is shown that contracts energy in the day ahead electricity market. This CPO is responsible for 10 EV chargers in a commercial building's parking lot concerning a workplace. Given the location of this parking lot, it is assumed that EVs will be arriving after 6 am and departing before 6 pm on weekdays only, aligning with typical office hours and days. This assumption is made to tailor the optimization more closely to a workplace setting.

In order to train the RL algorithm and test different scenarios for the described EV charging problem, a digital environment is required. For this research, a realistic V2G Gym simulator called EV2Gym [30] was used to obtain results. The logo of EV2Gym can also be seen between the CPO and the EV chargers in Figure 3.1. The following Section 3.1.1 will describe the EV2Gym simulator and explain how the specific PST problem for a workplace is implemented in this simulator.

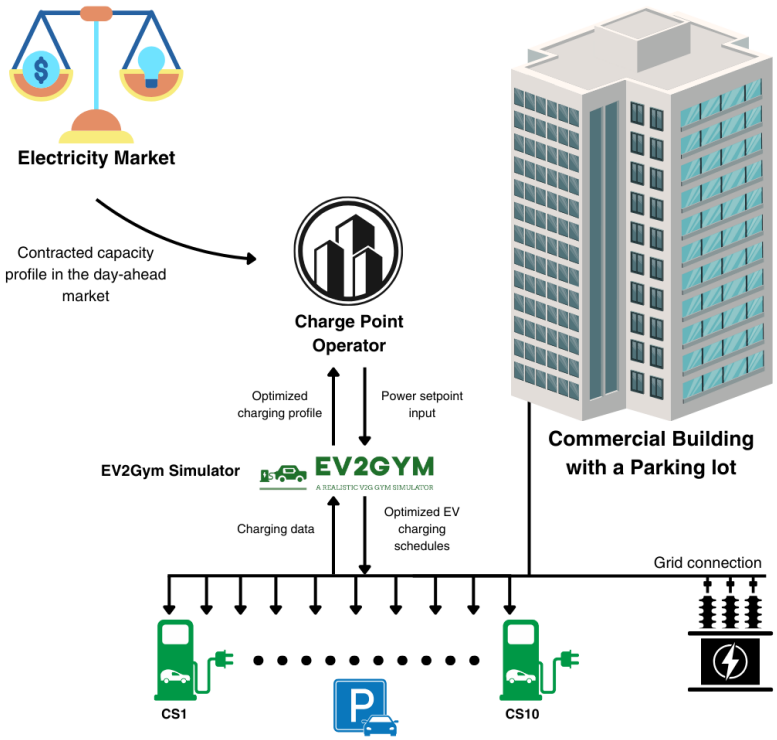


Figure 3.1: Operation of the CPO

3.1.1 Simulation Environment

EV2Gym is a flexible simulator designed to test different algorithms, focusing on how well they perform for EV charging/discharging problems. As can be understood from its name, it is also capable of evaluating V2G scenarios. It is built as a Gym environment [55], which is a toolkit of OpenAI utilized for RL algorithms. Furthermore, EV2Gym stands out due to its adjustability for employing different algorithms for various scenarios. The configuration settings of EV charging simulations, such as the number of chargers and transformers, EV and charger specifications, and the transformer current limits, can be adjusted according to the characteristics of scenarios. Moreover, the environment uses only open source data; however, custom data like electricity prices or EV behaviour can be implemented in the EV2Gym Simulator to customise EV charging/discharging problems. Additionally, it has the capability of saving the replays to make comparisons for evaluating the performance of the deployed algorithm by comparing it with other alternative algorithms or with the optimal solutions obtained by using the Gurobi Solver [56].

Power Setpoints

The calculation of power setpoints is a crucial part of the PST problem. The accuracy of power setpoints is essential for optimally solving the PST problem and training the RL agent by introducing flexibility for the charging of EVs. It is worth noting that determining power setpoints is the initial optimization challenge faced by CPOs, who must predict and secure the necessary amount of energy to contract in the day ahead electricity market. Although this optimization is beyond the scope of this study, mentioning it highlights the practical importance of precise power setpoint calculations. In response to this challenge, this research proposes a method designed to minimize the PST error with realistic power setpoints generated for each step of the simulation's duration.

The calculation of power setpoints is shown in the Pseudocode 1. To explain the process in detail, the calculation begins by creating an array of power setpoints, initialized to zero and intended to represent the

required power for each timestep in the simulation. Consecutively, for every timestep of the simulation, the arrival of new EVs is checked, and the needed additional energy is calculated based on the difference between the battery capacity of each EV and their SoC, which is called Depth of Discharge (DoD) of EVs. Then, the calculated required energy for each EV is multiplied by a flexibility factor (κ) introduced in the simulation to enable EV charging flexibility. In the next step, the calculated amount of energy with the flexibility factor is randomly distributed throughout each EV's duration of stay by using the normalized electricity prices as weights. This approach aims to get realistic power setpoints to be used to train the RL agent but does not represent an optimization methodology to define optimal power setpoints. For each EV, the function also respects the minimum and maximum charging powers allowed by both the EV's specifications and the charging station's limits. If any of the power setpoints fall outside these limits, a redistribution of power is carried out across the EVs' duration of stay, ensuring that every point adheres to the minimum and maximum limits of each EV and charging station.

Lastly, the generated power setpoints undergo median smoothing to mitigate abrupt variations, resulting in a charging schedule that is both realistic and feasible for implementation. In the next Section 3.1.2, the mathematical model of the formulated PST error minimization problem is explained in detail. The created mathematical model is also used to solve the PST error minimization problem optimally by forming a Mixed Integer Non-Linear Programming (MINLP) problem.

Algorithm 1 Calculation of Power Setpoints

```
Initialize an array of size number of time steps to represent power setpoints
for time step = 1 to T do
    Check for new EV arrivals
    for each EV arrival do
        Required energy ← Calculate the required energy for each EV for charging to the desired SoC level (80%)
        Adjusted energy ← Required energy ×κ
        for Remaining staying time = time step to departure time do
            Distribute adjusted energy to time steps using electricity prices as weights.
        end for
    end for
end for
Apply median smoothing to power setpoints
```

3.1.2 Mathematical Model

A mathematical model should be formulated to show the objective and constraints of the PST error minimization problem by EV charging scheduling. Table 3.1 presents the list of parameters used in the problem formulation, and these terms will be referenced throughout the rest of this Chapter 3.

The PST problem for the formulated scenario, charging at a workplace, simulation length T consists of 48 discrete t time steps, representing 15-minute intervals, leading to 12 hours in total. The problem was designed to optimize charging between 6 am and 6 pm, aiming to resemble real-world conditions for a workplace. Additionally, i represents each charging station for EVs to connect and is part of the set of charging stations C. The set of charging stations (C) are connected to a transformer to introduce power limits ($\underline{P}^{\text{tr}}, \overline{P}^{\text{tr}}$) to the simulation. Furthermore, there is a set of EVs indicated by H set and each EV by j, to have a distribution of different EVs according to their proportion in the total number of registrations in the Netherlands by 2023. The EV specifications are shown in Table 3.2 at the end of this Chapter 3. Lastly, the binary variable u is introduced to show if an EV is connected to a charging station i at time step t.

Table 3.1: List of parameters

Model	Parameters	Symbol	Range - Value
Simulation	Timescale	Δt	1 Step - 15 minutes
	Simulation Length	T	48 Steps - 12 hours
	Power Setpoint Flexibility Factor	κ	5%
EV	Min. & Max. Charging Power (kW)	$\underline{P}^{\text{ch}}, \overline{P}^{\text{ch}}$	0, 22
	Min. & Max. Battery Capacity (kWh)	$\underline{E}, \overline{E}$	32, 75
	Charge Efficiency	η	90%
	Battery Capacity at Arrival (kWh)	E^{arr}	
	SoC (%)		0 - 1
	Time of Arrival & Time of Departure (t)	$t_{\text{arr}}, t_{\text{dep}}$	6:00 am - 6:00 pm
	Set of EVs	H_j	
Charging Station	Min. & Max. Charging Station Current (A)	$\underline{I}^{\text{ch}}, \overline{I}^{\text{ch}}$	0, 56
	Charging Power (kW)	P^{ch}	
	Voltage (V) & Phases	V, ϕ	230, 3
Transformer	Min. & Max. Power (kW)	$\underline{P}^{\text{tr}}, \overline{P}^{\text{tr}}$	0, 60
	Set of Connected Charging Stations	C_{i}	10
Charging Related Parameters	Power Setpoints (kW)	P^{set}	
	Total Charging Power (kW)	P^{tot}	
	Binary Variable for Charging	$u_{i,t}$	0, 1

Objective Function

The objective function of the PST problem is to minimize the squared difference between the power setpoints (P_t^{set}) and the total charging power (P_t^{tot}) at each time step. The algorithm controls the charging current $I_{i,t}^{\text{ch}}$ at each time step t and charging station i, while $t \in T$ and $i \in C$.

$$\min_{I_{i,t}^{\text{ch}}} \sum_{t \in T} \left(P_t^{\text{set}} - P_t^{\text{tot}}\right)^2 \tag{3.1}$$

Constraints

The power at each time step t and charging station i is calculated by the Equation (3.2). The controlling current $I_{i,t}^{\text{ch}}$ is multiplied by the voltage V, phases ϕ, charging efficiency η, and with the binary variable $u_{i,t}$ to indicate if an EV is connected to a charging station i.

$$P_{i,t}^{\text{ch}} = I_{i,t}^{\text{ch}} \cdot V \cdot \sqrt{\phi} \cdot \eta \cdot u_{i,t} \quad \forall i, \forall t \tag{3.2}$$

The total charging power P_t^{tot} at step t is the sum of charging power at each charging station i, shown with the Equation (3.3).

$$P_t^{\text{tot}} = \sum_{i \in C} \left(P_{i,t}^{\text{ch}}\right) \quad \forall i, \forall t \tag{3.3}$$

The calculated total charging power P_t^{tot} at step t complies with the transformer's lower and upper limits, $\underline{P}_t^{\text{tr}}, \overline{P}_t^{\text{tr}}$ as shown by the Equation (3.4).

$$\underline{P}_t^{\text{tr}} \leq P_t^{\text{tot}} \leq \overline{P}_t^{\text{tr}} \quad \forall t \tag{3.4}$$

The charging current of each connected EV ($I_{i,t}^{\text{ch}}$) complies with the minimum ($\underline{I}_i^{\text{ch}}$) and maximum ($\overline{I}_i^{\text{ch}}$) current limits of each charging station i, shown with the Equation (3.5).

$$\underline{I}_i^{\text{ch}} \leq I_{i,t}^{\text{ch}} \leq \overline{I}_i^{\text{ch}} \quad \forall i, \forall t \tag{3.5}$$

Next to the constraints related to charging stations and the transformer, there are several EV-related constraints and equations. The energy inside each EV ($E_{i,j,t}$) connected at charging station i at step t complies

with each EV's varying lower and upper battery capacity constraints, $\underline{E}_i, \overline{E}_i$, with respect to its model j as shown in Equation (3.6).

$$\underline{E}_{i,j} \leq E_{i,j,t} \leq \overline{E}_{i,j} \quad \forall i, \forall t \tag{3.6}$$

The batteries' energy, SoC, changes according to the Equation (3.7). The charging power $P^{\text{ch}}_{i,t}$ is multiplied with the Δt and added to the energy level in the previous time step indicated with $E_{i,j,t-1}$.

$$E_{i,j,t} = E_{i,j,t-1} + \left(P^{\text{ch}}_{i,t}\right) \cdot \Delta t \quad \forall i, \forall t \tag{3.7}$$

The charging power at charging station i at time step t complies with each EV model's different minimum and maximum charging power limits, $\underline{P}^{\text{ch}}_j$, $\overline{P}^{\text{ch}}_j$.

$$\underline{P}^{\text{ch}}_j \leq P^{\text{ch}}_{i,j,t} \leq \overline{P}^{\text{ch}}_j \quad \forall i, \forall j, \forall t \tag{3.8}$$

The arrival time of EVs is known. Therefore, the energy level of each EV at its arrival at a charging station i is indicated by $E^{\text{arr}}_{i,t}$ in Equation (3.9).

$$E_{i,t} = E^{\text{arr}}_{i,t} \quad \forall i, \forall t | t = t^{\text{arr}}_i \tag{3.9}$$

The binary variable $u_{i,t}$ at Equation (3.10) indicates if an EV is connected to a charging station i at time step t.

$$u_{i,t} \in \{0, 1\} \quad \forall i, \forall t \tag{3.10}$$

3.2 Proposed RL Formulation

This section outlines the learning and implementation processes of the proposed DDPG algorithm within the EV2Gym environment [30]. Initially, the design of the state and action space is detailed in Section 3.2.1, highlighting the components of the state function and the implementation of continuous actions in the environment. Following this, the alternative reward functions are introduced in Section 3.2.2. The DDPG algorithm's structure, its operational workflow, and learning performance are then elaborated in Section 3.2.3. While the design of the state and reward function is critical for training the DDPG algorithm, the hyperparameters are equally important, which is explored in Section 3.2.4. The hyperparameters section includes an examination of essential hyperparameters such as the learning rate (α), target network update rate (τ), action noise ($\mathcal{N}$), discount factor (γ), mini-batch size (M), replay buffer size ($\mathcal{R}$), and the dimensions of the utilized DNNs.

3.2.1 State and Action Spaces

The design of the state and action spaces is critical for the success of the algorithms in RL. It is essential to include only the variables that are relevant and useful in the state function, which enables the RL agent to learn more efficiently, considering the reduced complexity of its input variables. Keeping the size of the state vector in balance is also essential. If too many variables are included in the state vector, it may cause complexity again and result in longer computation times. On the other hand, if too few variables are included, it may not provide enough information for the agent to learn. Therefore, the state function variables should include only the minimum number of essential variables that directly influence the agent's ability to predict future states and make decisions to find an optimal policy.

After extensive experimentation with various state representations the one presented here provided the best results. The proposed state of the PST problem has 3 variables regardless of the number of charging stations and an additional 3 variables which initialize when an EV is connected to a charger i. When an EV is not connected, zeros replace these 3 variables to keep the state vector size constant. The total size of the state vector is $3 + 3C$ for each time step t, C being the total number of chargers. The 3 variables, regardless of the number of charging stations, are the environment's normalized time step t/T, power setpoint P^{set}_t, and the total power usage P^{tot}_{t-1}. The remaining 3 variables that concern the number of charging stations C are EVs' normalized arrival and departure times $t^{arr}/T, t^{dep}/T$ and SoC_t of each EV at step t. It is aimed to

simplify the state space and improve the consistency of time perception by utilizing normalized time in the state vector, enhancing the DDPG agent's ability to learn optimal policies effectively. Additionally, it should be noted that using normalized time as a component of the state representation is a novel contribution of this study. Finally, these variables together form the state vector:

$$\mathbf{s}_t = [\tfrac{t}{T}, P_t^{\text{set}}, P_{t-1}^{\text{tot}}, \tfrac{t_i^{arr}}{T}, \tfrac{t_i^{dep}}{T}, SoC_{i,t}] \in S, i \in C$$

Actions inside the environment are taken with respect to the constraints mentioned between the Equations (3.2) and (3.10). The charging of EVs is controlled by the charging current $I_{i,t}^{\text{ch}}$. The actions take continuous values between 0 and 1 to adjust the charging levels for each charging station, with 0 being not charging at all and 1 being charging at full power, leading to an action vector $\mathbf{a}_t \in [0, 1]$ for each charger. Hence, the action vector of the environment is the size of the total number of chargers, C, formulating the action vector:

$$\mathbf{a}_t = [0, 1]^{\text{C}} \in A$$

3.2.2 Reward Function

RL agents aim to identify the best policy for actions based on the rewards they get from interacting with the environment. These agents focus on enhancing their total discounted reward over time, representing their overall success. As such, the structure of the reward function plays a significant role in the RL agent's learning process. The objective function of the PST problem, as mentioned in Equation (3.1), is formulated for the RL agent's reward function. The first reward function R_1 in Equation (3.11) is very similar to the objective function of the mathematical optimization problem. It calculates the negative squared difference between the power setpoints and the total power usage at each time step of the simulation. The reward is negative because the agent tries to maximize its total discounted reward throughout the simulation.

$$R_1 = -(P_{t-1}^{\text{set}} - P_{t-1}^{\text{tot}})^2 \tag{3.11}$$

The second proposed reward function introduces an additional factor named "charge power potential", with the goal of incorporating the charging requirements of EVs into consideration. $I_{i,t}^{\text{pot}}$ represents the total potential of charging based on the number of EVs connected, considering their SoCs and both the current and power limitations of the connected EVs and charging station i. The calculation of the charge power potential is performed using Equation (3.12).

$$P_t^{\text{pot}} = \sum_{i \in C} (I_{i,t}^{\text{pot}}) \cdot V \cdot \sqrt{\phi} \cdot \eta \quad \forall i, \forall t \tag{3.12}$$

Given that the actual charging demand may fall below the calculated power setpoints, this function seeks to minimize the difference between actual power usage and the power setpoints, or the charge power potential, whichever is lower. This approach ensures a more efficient matching of charging power to charging demand. The proposed reward function is shown with the Equation (3.13).

$$R_2 = -(min(P_{t-1}^{\text{pot}}, P_{t-1}^{\text{set}}) - P_{t-1}^{\text{tot}})^2 \tag{3.13}$$

The performance of both reward functions is investigated in Section 4.1. The remaining experiments are done with the selected reward function that yielded the best outcome, reward function R_2.

3.2.3 DDPG Algorithm

The DDPG algorithm is a model-free, off-policy actor-critic algorithm designed for environments with continuous state and action spaces. Using discrete action spaces for EV charging problems was mentioned in the literature as a limitation in several researches [47][48]. Therefore, the DDPG algorithm is considered a more promising alternative for solving EV charging problems than RL algorithms such as DQN and Q-learning. Furthermore, DDPG offers a deterministic approach to finding the best policy. As Silver states [42], deterministic policy gradient algorithms can outperform their counterparts, which use stochastic policy algorithms, such as SAC. Hence, the DDPG algorithm is selected among other RL algorithms to solve the

PST problem.

Figure 3.2 shows the formulation of the PST problem and the interactions the RL agent takes in the environment. Here is a detailed explanation of how the algorithm operates:

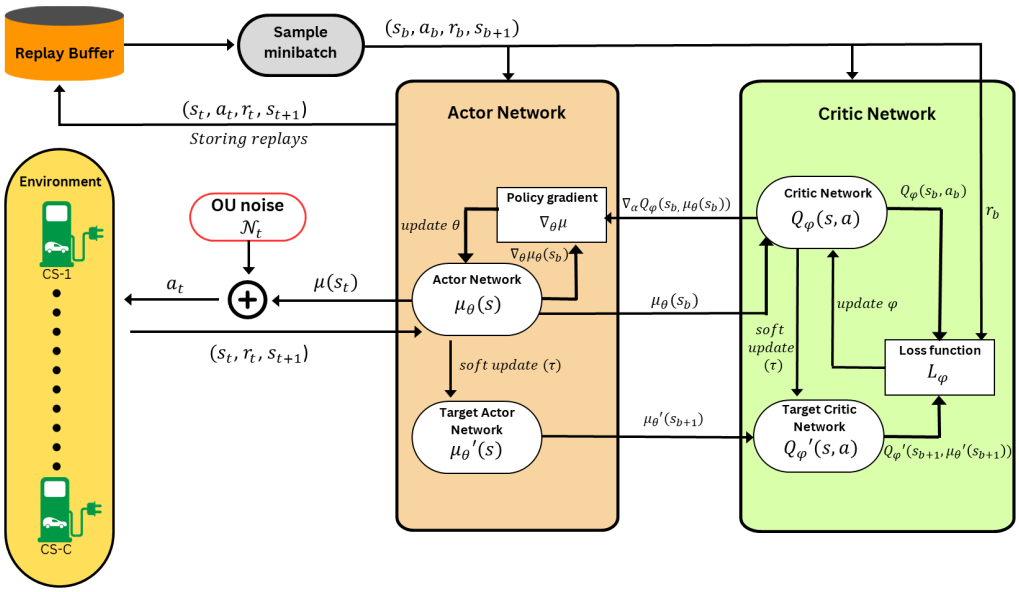


Figure 3.2: DDPG's operation

Step 1: Initialization
Initialize the actor-network $\mu_\theta(s)$, parameterised by θ that maps states to actions, $S \rightarrow A$, and the critic network $Q_\varphi(s,a)$, parameterised by φ that estimates the Q-value of state-action pairs, $Q_\varphi(s,a)$. The actor outputs a deterministic action $\mu_\theta(s)$ for the current state s_t, whereas the critic does the policy evaluation task. The critic assesses the actor policy by estimating the Q-value function, $Q_\varphi(s,a)$. This estimation is made by minimizing the loss function, L_φ shown with the Equation (3.14). By continuously minimizing the loss function and updating the actor-network parameters θ, the actor policy π improves, leading to better results. However, this updating process takes part in the learning step.

$$L(\varphi) = (r_t + \gamma Q_\varphi(s_{t+1}, \mu(s_{t+1})) - Q_\varphi(s_t, a_t))^2 \tag{3.14}$$

Initialization continues with target networks $\mu'_\theta(s)$ and $Q'_\varphi(s,a)$ for the actor and critic to provide stable targets. Then, a replay buffer to store transition tuples (s_t, a_t, r_t, s_{t+1}), for breaking the correlation between samples, is initialized.

Step 2: Exploration and Experience Collection
For each time step t, the agent selects an action $\mu_\theta(s_t)$ using the actor-network for the current state s_t and adds an Ornstein–Uhlenbeck (OU) noise [57] $\mathcal{N}_t$ for exploration. As Hollenstein et al. [58] state, introducing an action noise and its type and scale have a crucial effect on RL agents for learning. After adding the noise, the algorithm executes action a_t in the environment to observe the next state s_{t+1} and receive the reward r_t. Following that, the algorithm stores the transition (s_t, a_t, r_t, s_{t+1}) in the replay buffer $\mathcal{R}$. The parameter weights for the actor and critic networks update only after collecting a sufficient number of experience replays in the replay buffer. In our experiments, the RL agent begins its learning process once 100 transitions have been collected in the proposed DDPG algorithm.

Step 3: Learning
The replay buffer $\mathcal{R}$ has the capacity to store a number of transitions. In the proposed approach, this capacity is selected as 10^6. When the replay buffer is full, the oldest samples are discarded so the agent can learn from the latest experiences. Because DDPG is an off-policy algorithm, the replay buffer can be large, allowing the algorithm to benefit from learning across a set of uncorrelated transitions [59]. Then, the algorithm samples a mini-batch of a total of M transitions (s_b, a_b, r_b, s_{b+1}) from the replay buffer $\mathcal{R}$ to update actor and critic network weights. Then, for each sampled transition, the algorithm calculates the target Q-value as shown in Equation (3.15), where γ is the discount factor. The discount factor affects the importance of immediate and future rewards.

$$y_b = r_b + \gamma Q'_{\varphi}(s_{b+1}, \mu'_{\theta}(s_{b+1})) \tag{3.15}$$

Consecutively, the algorithm updates the critic network by minimizing the loss between its predicted Q-values, $Q_{\varphi}(s_b, a_b)$ and the target Q-values $Q'_{\varphi}(s_{b+1}, \mu'_{\theta}(s_{b+1}))$ shown with the Equation (3.16).

$$L_{\varphi} = \frac{1}{M} \sum_{b} (y_b - Q_{\varphi}(s_b, a_b))^2 \tag{3.16}$$

Later, the actor policy is updated using the sampled policy gradient aimed at actions that maximize the critic's predicted Q-values according to Equation (3.17).

$$\nabla_{\theta}\mu = \frac{1}{M} \sum_{b} \nabla_a Q_{\varphi}(s_b, \mu_{\theta}(s_b)) \nabla_{\theta}\mu_{\theta}(s_b) \tag{3.17}$$

After the actor policy update, the weights for both the main and target of the actor and critic networks are updated. Here, α^{μ} represents the learning rate in the gradient ascent process for updating the actor network and α^{Q} denotes the learning rate for the gradient descent process applied to the online critic network in Equations (3.18) and (3.19).

$$\theta \leftarrow \theta + \alpha^{\mu} \nabla_{\theta}\mu \tag{3.18}$$

$$\varphi \leftarrow \varphi - \alpha^{Q} \nabla_{\varphi} L(\varphi) \tag{3.19}$$

Lastly, the weights of the target networks μ'_{θ} and Q'_{φ} are updated towards the main networks using a soft update τ as shown by the Equations (3.20) and (3.21).

$$\theta' \leftarrow \tau\theta + (1 - \tau)\theta' \tag{3.20}$$

$$\varphi' \leftarrow \tau\varphi + (1 - \tau)\varphi' \tag{3.21}$$

Step 4: Repeat
Repeat the process for each time step and episode, allowing the policy to converge towards the optimal policy that maximizes the expected return, which is mean rewards. Finally, the algorithm saves the trained RL model after the defined number of episodes is reached. The total number of episodes indicated by $\mathcal{U}$ to train the RL agent is selected as 25,000.

Pseudocode of the DDPG Algorithm

The explained consecutive steps of the DDPG algorithm are represented in Pseudocode 2 below:

Algorithm 2 DDPG algorithm - Training Process

Initialize actor μ and critic network Q with weights θ and φ.
Initialize target networks Q' and μ' with weights $\theta' \leftarrow \theta$, $\varphi' \leftarrow \varphi$
Initialize replay buffer $\mathcal{R}$
for episode $= 1, \ldots, \mathcal{U}$ **do**
  Receive initial observation state s_1, including normalized time step, power setpoints, SoC of connected EVs, and EV arrival and departure times.
  Initialize a random OU noise $\mathcal{N}$ for action exploration
  for $t = 1, \ldots, T$ **do**
    Select action $a_t = \mu_\theta(s_t) + \mathcal{N}_t$ according to the current policy and exploration noise
    Execute action a_t and observe reward r_t and observe new state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{R}$
    Sample a random minibatch of M transitions (s_b, a_b, r_b, s_{b+1}) from $\mathcal{R}$
    Set $y_b = r_b + \gamma Q'_\varphi(s_{b+1}, \mu'_\theta(s_{b+1})$
    Update critic by minimizing the loss: $L_\varphi = \frac{1}{M}\sum_b (y_b - Q_\varphi(s_b, a_b))^2$
    Update the actor policy using the sampled policy gradient:
    $\nabla_\theta \mu = \frac{1}{M}\sum_b \nabla_a Q_\varphi(s_b, \mu_\theta(s_b)) \nabla_\theta \mu_\theta(s_b)$
    Update the actor and critic networks:
    $\theta \leftarrow \theta + \alpha^\mu \nabla_\theta \mu$
    $\varphi \leftarrow \varphi - \alpha^Q \nabla_\varphi L(\varphi)$
    Update the target networks:
    $\theta' \leftarrow \tau\theta + (1-\tau)\theta'$
    $\varphi' \leftarrow \tau\varphi + (1-\tau)\varphi'$
  end for
end for
Save the trained model after reaching $\mathcal{U}$ episodes

3.2.4 Hyperparameters

Hyperparameters are a set of parameters that determine the learning process of the RL agent. They play a vital role in how the agent learns to make decisions that maximize its long-term rewards by finding an optimal policy. Utilized hyperparameters and their effect on the learning process of the RL agent are explained in this section.

Learning Rate (α): The learning rate of an RL agent refers to the speed at which it updates both the main and target networks of its actor and critic. A higher learning rate means that the agent learns faster and makes more significant updates. However, this can result in overshooting or missing an optimal policy. Conversely, a lower learning rate means that learning is steadier but slower, and more time may be required to converge on the optimal policies.

Replay Buffer Size ($\mathcal{R}$): The replay buffer size determines how many of the past replays the RL agent can store to learn from. A larger replay buffer allows the agent to learn from a wider range of its tuple, including past states, actions, rewards and past subsequent states.

Action Noise ($\mathcal{N}$): Introducing noise to the actions increases the environment exploration of the RL agent. This increased exploration can help the agent discover optimal policies that would not have been discovered otherwise. It is important to note that the type and amount of noise introduced can have a significant impact on the degree of exploration undertaken by the agent.

Figure 3.3: Mean rewards with different action noises $\mathcal{N}$

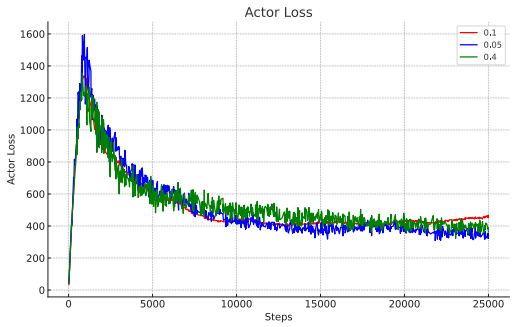


Figure 3.4: Actor Loss

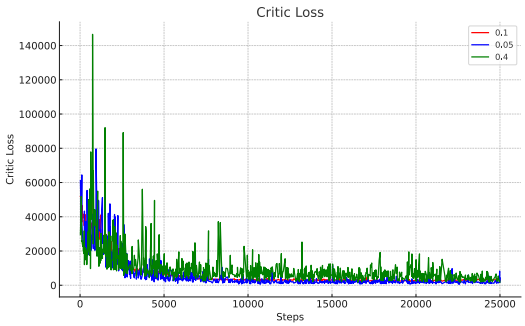


Figure 3.5: Critic Loss

In the graphs presented in Figures 3.3, 3.4, and 3.5, the results of experiments conducted with three different action noise ($\mathcal{N}$) values can be observed. The green line, which represents the highest noise level, shows that the most variation in the agent's rewards, especially in critic loss. It can be deduced that a higher degree of variation indicates that the agent is more prone to exploration.

Deep Neural Networks Architecture: The actor and critic networks are used by the agent to make decisions and evaluate its actions. Thus, the size of the DNNs plays a vital role in the agent's ability to learn the dynamics and complexity of the environment. If the size of the DNNs is small, they may not capture the complexity of the environment. Conversely, if they are large, they may not capture the relationships between states and actions. In this research, the DNNs' sizes implemented can be seen in Figure 3.6. The actor network has 128 neurons in each of its main and target networks. Figure 3.6 also shows that the input of the actor network is the state vector, which consists of $|3+3C|$ variables as shown in Section 3.2.1. After passing through the 128-128 sized main and target networks, the actor network outputs the actions, consisting of C neurons, each neuron indicating the action taken for each EV charger. The state and action vectors are then combined and used as the input of the critic network. These two inputs pass through the critic network to form the Q-value function. As a result, the size of the DNNs is a crucial hyperparameter that significantly impacts the RL agent's learning process.

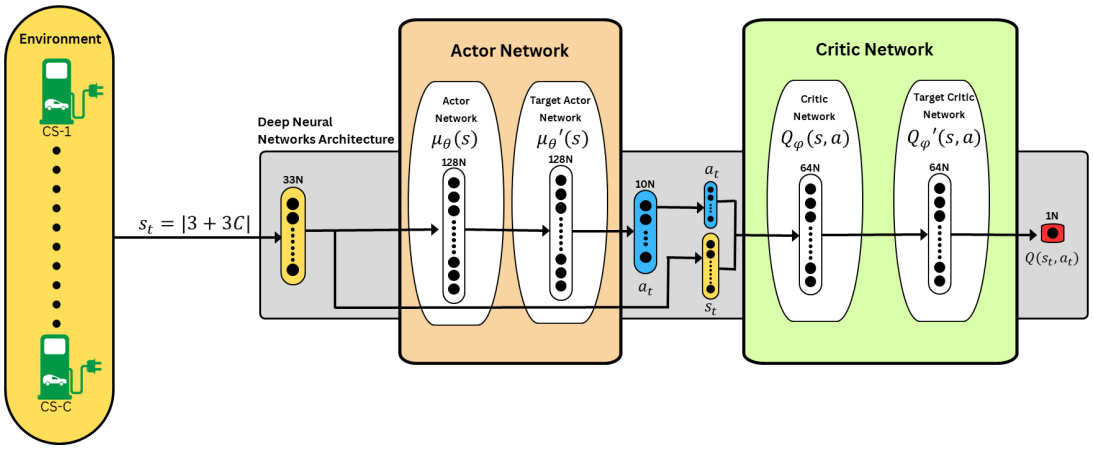


Figure 3.6: DNNs architecture

Minibatch Size (M): The minibatch size is the number of experiences that the agent uses to update its parameters (θ, φ). Using a larger batch size can make learning more stable by averaging out noise and outliers in the replay buffer. However, it also increases the risk of exploiting a sub-optimal solution because the agent may not explore the environment sufficiently.

Soft Update (τ): The soft update parameter, denoted by τ, plays a crucial role in ensuring stable learning. It determines the pace at which the target networks are updated with the weights from the actor and critic networks. A smaller value of τ leads to incremental changes, which helps maintain training stability by avoiding sudden shifts in policy that could result from more aggressive updates. The value of τ lies between 0 and 1.

Discount Factor (γ): The discount factor, denoted by γ, decides the worth of future rewards in comparison to immediate ones. Its value ranges between 0 and 1. A discount factor of 1 means that each reward during the simulation is equally valuable for the agent's learning process.

Hyperparameters are crucial for training an RL agent to learn an optimal policy and produce accurate test results. Although some hyperparameter values lead to convergence of the return, the mean reward curve, they may not perform well during testing due to reaching sub-optimal policies. Therefore, it is important to check if the agent learned an optimal policy during training and test the saved RL model with evaluation replays to check if it is capable of achieving reliable results. It is worth noting that some hyperparameter values may not lead to any convergence in the mean reward curve of the agent. The evaluation of results achieved with various hyperparameter sets is shown in Appendix A. Lastly, the testing of the hyperparameters with actual scenarios is done in the next Chapter 4 in Section 4.1 by first explaining the case study parameters in Table 4.1 and comparison metrics in Table 4.2.

Open Source Data

For the PST workplace problem, the EV spawn rate, time of stay and energy required were determined using distributions derived from the ElaadNL data [60] on EVs in the Netherlands. Only weekday data was used to ensure a more realistic and specific analysis of EV behaviours in a workplace context. Additionally, to determine the power setpoints, electricity prices were obtained from open source data of the European Network of Transmission System Operators for Electricity (ENTSO-e) [61] between the years 2015-2023.

Furthermore, based on consecutive surveys conducted by Rijksdienst voor Ondernemend Nederland (RVO), the total number of registered BEVs in the Netherlands in 2023 are obtained as shown in Table 3.2 with their specifications such as battery capacity and maximum charging power. The distribution of EVs charging in the parking lot is based on their proportion in the total number of registered BEVs in the Netherlands. By combining EV specifications with actual electricity pricing and real EV usage patterns for a workplace setting, the simulation becomes more realistic and applicable to real-life situations.

Table 3.2: Total number of registered BEVs in the Netherlands in 2023 [62, 63, 64, 65]

BEV Model	Registrations 2023 NL	Battery Capacity (kWh)	Max AC Charge Power (kW)
Tesla Model 3	45545	57.5	11
Kia Niro	23105	64.8	11
Volkswagen ID.3	19950	58	11
Hyundai Kona	17752	64	11
Tesla Model Y	16186	57.5	11
Skoda Enyaq	16165	58	11
Peugeot 208	14017	46.3	7.4
Renault Zoe	14008	52	22
Volkswagen ID.4	13283	77	11
Volvo XC40	12520	66	11
Nissan Leaf	11977	39	3.6
Tesla Model S	10899	75	11
Volkswagen Golf	10019	32	7.2

In Chapter 3 the formulation of the PST minimization problem specific to a workplace was introduced, and the corresponding mathematical model was detailed in Section 3.1.2. Subsequently, the design of the state and action spaces was delved into in Sections 3.2.1 and 3.2.2. The DDPG algorithm's functionality at each step of the process was then described. This was followed by an explanation of the hyperparameters used, as outlined in Section 3.2.4. Consecutively, the open-source data utilized to train the RL agent was described before delving into the case study in the next Chapter 4.

4

Results and Discussions

In this chapter, the performance of the proposed DDPG algorithm for solving the PST error minimization problem from the CPO's perspective is presented. A case study is conducted in Section 4.1, where the PST problem is formulated for a workplace parking lot with a total of 10 EV chargers, as explained in Chapter 3, in the Problem Formulation Section 3.1. DDPG's performance is then investigated in terms of complying with transformer power limits in Section 4.1.3, followed by an investigation of the proposed algorithm's scalability with the number of decision variables (chargers) in Section 4.1.4. Finally, the chapter ends with a discussion of the proposed DDPG algorithm's capabilities and limitations considering the achieved results in Section 4.2.

4.1 Case Study - Charging at Work

The problem's properties are explained in detail in Chapter 3, in the Problem Formulation Section 3.1. In short, the charging of EVs is controlled by a CPO to ensure profitable operation by PST. The CPO buys energy in advance from the day-ahead market and allocates it for the following day's EV charging sessions. Once the energy is contracted, the CPO assigns a power setpoint for each time step of the upcoming day. For this case study, the time step was chosen as 15 minutes, considering the wholesale market energy contracting time steps. This power setpoint must be adhered to by scheduling the EV charging sessions accordingly. Deviating from these power setpoints can lead to buying additional energy at higher rates in the intraday market or risking customer dissatisfaction due to partially charged EVs. To prevent such scenarios, the CPO tries to minimize the PST error by scheduling EV charging sessions.

4.1.1 Training and Testing Settings

Table 4.1 displays the parameters of the case study. The time step, represented by t, refers to the duration of each simulation step. The simulation length, denoted by T, is the duration of one replay, which is 12 hours in total. This is due to the fact that EV arrival and departure times are constrained between 06:00 and 18:00. This limitation aims to simulate real-world conditions and improve the RL agent's learning by training the agent with similar EV patterns.

Furthermore, data from ElaadNL [60] gives the EV arrival patterns, duration of stay, and charging needs for a workplace, while electricity prices are drawn from ENTSO-e's open-source data [61]. This information, combined with specific EV data from the RVO's surveys shown in Table 3.2 in the previous chapter, ensures to provide practical information on the PST problem in the workplace.

Additionally, the power setpoints are determined with a 5% flexibility margin, which is denoted by κ. This means that the CPO contracts 5% more energy in the day-ahead market than the total charging demand for the subsequent day for each replay.

Table 4.1: Case Study Parameters

Parameters	Symbol	Value	Range
Time step	t	1	15 minutes
Simulation length - One replay	T	48	720 minutes - 12 hours
EV arrival and departure times		ElaadNL data [60]	06:00-18:00
Electricity prices	c	ENTSO-e [61]	
EV models	H_j	BEVs in NL in 2023 from Table 3.2	
Number of chargers	C_i	10	
Power setpoint flexibility	κ	5%	
Transformer limit	$\overline{P}^{\text{tr}}$	60 kW	

The upper limit for the transformer power, denoted as $\overline{P}^{\text{tr}}$, is determined by multiplying the highest achieved level of power setpoints in the replays by 2. These power setpoints are determined with respect to the total charging demand and are described in detail in Chapter 3 Section 3.1. This approach aims to ensure the reliable operation of the EV chargers, given that they are located in a commercial building and are connected to the same transformer as the building's loads.

Baseline Algorithms

To evaluate the performance of the proposed DDPG algorithm, it is crucial to compare its results against those of other established approaches. For this purpose, two benchmark algorithms have been selected. The first benchmark is an optimal solution that relies on precise information regarding the arrival and departure times of EVs as well as their SoC. It is worth mentioning that while the optimal algorithm will achieve theoretically optimal results, practically, it is not possible to achieve these results due to uncertainties such as arrival and departure times and SoC of EVs. The second is a baseline scenario labelled "charge as fast as possible" (CAFAP), in which EVs initiate charging immediately upon connection and drawing the maximum power permitted by either the EV or the connected EV charger.

Table 4.2: Comparison Metrics

Metric	**Symbol**	**Equation**
Squared Tracking Error	ϵ^{tr}	$\sum_{t\in T}(P_t^{set} - P_t^{tot})^2$
Energy Tracking Error (kWh)	$\lvert\epsilon^{tr}\rvert$	$\sum_{t\in T}\lvert P_t^{set} - P_t^{tot}\rvert \cdot \Delta t$
User Satisfaction (%)	ϵ^{usr}	$\frac{1}{\lvert\mathcal{E}\rvert} \cdot \sum_{k\in\mathcal{E}} \frac{SoC_k}{SoC_k^*}$
Power Tracker Surplus (kW)	ϵ^{sur}	$\sum_{t\in T} \max((P_t^{tot} - P_t^{set}), 0)$
Transformer Overload (kW)	ϵ^{ov}	$\sum_{t\in T} \max((P_t^{tr} - \overline{P}^{tr}), 0)$

The comparative analysis utilizes various metrics, which are detailed in Table 4.2. One key metric is the squared tracking error (ϵ^{tr}). This metric is not only the objective function for the mathematical optimization problem formulation but also similar to the reward function for the proposed DDPG algorithm with a slight difference. The agent is designed to maximize the negative value of this reward, which underscores its significance as a comparison metric. Another metric, the energy tracking error, symbolised by $|\epsilon^{tr}|$, determines the aggregate energy error for a replay, a day of charging, by multiplying the PST error at each time step with the error duration and summing the resulting energy error in kWh units.

Furthermore, the user satisfaction metric, denoted by ϵ^{usr}, reflects the increase in EVs' SoC and shows how much of the EV's battery is full with respect to the desired SoC level (SoC_k^*) which is determined as 80%. This threshold ensures that the charging process does not adversely impact battery lifespan. Ultimately, if the battery is not charged until the desired SoC (80%) before the EV's departure, user satisfaction decreases to the ratio of the EV's SoC (SoC_k) at departure divided by the desired SoC (SoC_k^*). The average sum of this calculated value for each served EV k gives the user satisfaction for one charging session. Additionally, the power tracker surplus metric (ϵ^{sur}) measures the extent to which actual charging exceeds the power

setpoints at each simulation step, with the result expressed in kW units.

The final metric is the transformer overload metric (ϵ^{ov}), which indicates the total amount of charging power that exceeds the determined transformer power limit. The results obtained with respect to the transformer overload metric are investigated explicitly in Section 4.1.3.

The RL agent is trained using the case study parameters listed in Table 4.1. To train the agent, a sufficient number of replays (episodes) are determined as 1,200,000 time steps of 15 minutes each, summing up to 25,000 episodes. This forms a total of 25,000 days consisting of 12 hours each. The agent's learning performance is evaluated based on its return, which means the mean reward's convergence and maximization. However, this value alone is insufficient to choose the hyperparameter values and the reward function due to the possible convergence of the agent by a sub-optimal policy. Therefore, a set of 100 random replays consisting of only weekdays are created with the same case study parameters listed in Table 4.1. These 100 random replays are then tested using three benchmark algorithms: the trained RL model named DDPG, an uncontrolled CAFAP algorithm, and the Optimal result achieved by a MINLP formulation of the PST problem that assumes complete knowledge of the problem, such as EV arrivals and departures, which is not realistic. However, it can provide an experimental optimal boundary for the best performance. The results obtained from these approaches are then evaluated based on the comparison metrics listed in Table 4.2.

Reward Function Determination

Two reward functions come forward as promising alternatives for the PST problem. R_1, as shown in Equation (3.11), is the negative squared difference between power setpoints and total power usage in the previous step of the simulation. On the other hand, the second proposed reward function R_2 utilizes one additional term named charge power potential in its formula as presented in Equation (3.13). The reward is the negative squared difference between the power usage and the minimum of power setpoint and the charge power potential in the previous step. As can be seen from Figure 4.1, both of the reward functions maximized the mean reward and converged in the training duration.

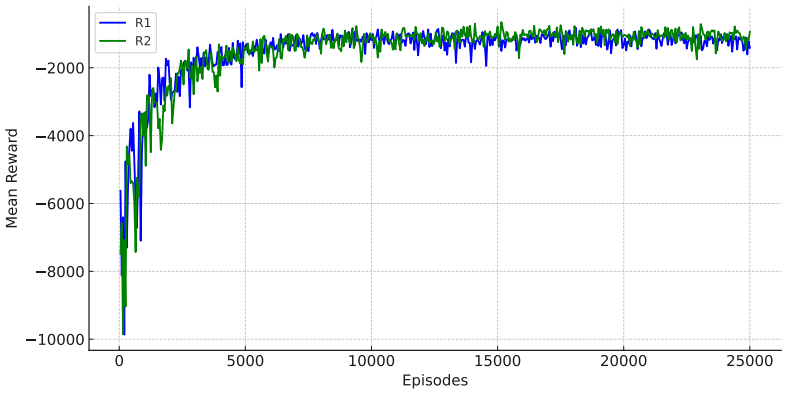


Figure 4.1: Mean rewards during training suggesting that both reward functions converged, necessitating further analysis to complete the reward function selection

After training, the trained models should be evaluated based on their performance using comparison metrics. The performances of R_1 and R_2 are shown in Table 4.3, where the reward functions are compared with CAFAP and MINLP optimal charging for a total of 100 replays. The use of R_2 results in significantly lower squared tracking error and energy tracking error compared to R_1. Moreover, R_2 performed 74% better over R_1 in terms of power tracker surplus metric. However, R_1 performed better than R_2 in terms of user satisfaction metric. Yet, having an average user satisfaction of 86% and improving all other metrics by reducing the PST error makes R_2 a more acceptable choice compared to a slightly higher user satisfaction obtained by R_1. Therefore, R_2 is selected as the reward function of the DDPG algorithm for the rest of this

experimental results chapter.

Table 4.3: Reward functions evaluation

Algorithm	**Squared Tracking Error** (kW^2) ϵ^{tr}		**Energy Tracking Error** (kWh) $\lvert\epsilon^{tr}\rvert$		**User Satisfaction** $(\%)$ ϵ^{usr}		**Power Tracker Surplus** (kW) ϵ^{sur}	
	Average	*Std*	*Average*	*Std*	*Average*	*Std*	*Average*	*Std*
CAFAP	12045.21	3689.83	150.19	23.40	1.00	0.00	290.20	45.56
DDPG - R_1	10201.09	3724.10	139.65	25.13	0.94	0.02	208.32	46.57
DDPG - R_2	6311.55	2165.49	110.43	20.24	0.86	0.03	53.68	24.87
Optimal - MINLP	194.75	110.58	12.63	4.66	0.98	0.01	0.00	0.00

After the determination of the reward function with the hyperparameters shown in Table 4.4, the model's hyperparameters are tuned further. The RL model is trained on more than 100 varied hyperparameter sets, and the tested hyperparameter sets are shown in Appendix A to find the best set. Furthermore, convergence of the mean reward is observed in several of these tests. However, convergence of the mean reward does not always mean that the agent has found an optimal policy; convergence can also happen due to finding a sub-optimal policy. Therefore, the performance of the trained model is evaluated on 100 randomly generated replays, according to the defined comparison metrics shown in Table 4.2. The results are shown in the next Section 4.1.2.

Table 4.4: Best reward function search hyperparameter set

Replay Buffer $(\mathcal{R})$	**Minibatch** (M)	**Discount factor** (γ)	**Soft update** (τ)	**Learning rate** (α)	**Noise** $(\mathcal{N})$	**Actor N** (μ_θ)	**Critic N** (Q_φ)
1000000	64	0.99	0.001	0.001	0.1	128-128	64-64

After testing and training with over 100 hyperparameter sets, the best results are obtained with the hyperparameters listed in Table 4.5. There is a slight difference in the chosen hyperparameter set compared to the ones in Table 4.4 because a slight decrease in the PST error was obtained by utilizing the hyperparameter set in Table 4.5. This is achieved by increasing the action noise $(\mathcal{N})$ to increase the agent's exploration rate and decreasing the soft update (τ) to prevent updating the target networks too quickly with increased exploration noise. This resulted in a better balance between the action noise and target update.

Table 4.5: Selected hyperparameter set

Replay Buffer $(\mathcal{R})$	**Minibatch** (M)	**Discount factor** (γ)	**Soft update** (τ)	**Learning rate** (α)	**Noise** $(\mathcal{N})$	**Actor N** (μ_θ)	**Critic N** (Q_φ)
1000000	64	0.99	0.0005	0.001	0.2	128-128	64-64

In Figure 4.2, the mean rewards achieved by using the selected hyperparameter set can be observed. It can be seen that the mean rewards start to converge after the 5000th episode; however, to guarantee convergence, the training lasted 25,000 episodes. Additionally, it can be observed that the mean rewards converged to a higher number in comparison to the level achieved in Figure 4.1. This result also suggests that the selected hyperparameter set might outperform the training hyperparameter set.

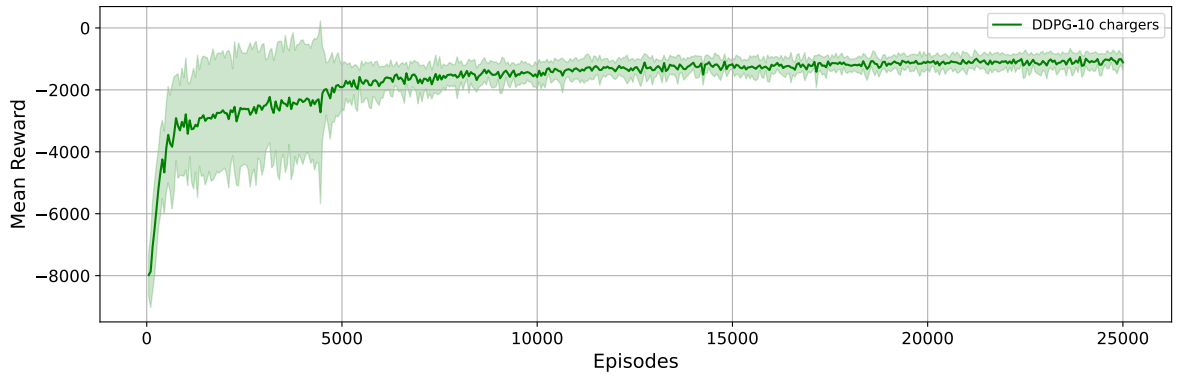


Figure 4.2: Mean rewards from 10 training sessions with selected hyperparameter set

4.1.2 Results

The trained RL model was evaluated using 100 replays to compare its performance with the selected CAFAP and Optimal-MINLP charging algorithms. Comparison metrics were used for the evaluation. Figure 4.3 shows the variations in charging approaches among all the algorithms. It is important to note that Figure 4.3 represents data from only one replay, while the evaluation of the hyperparameters and the DDPG algorithm is based on average values derived from 100 replays. Table 4.6 at the end of this section summarizes the obtained average values and their standard deviations for 100 replays.

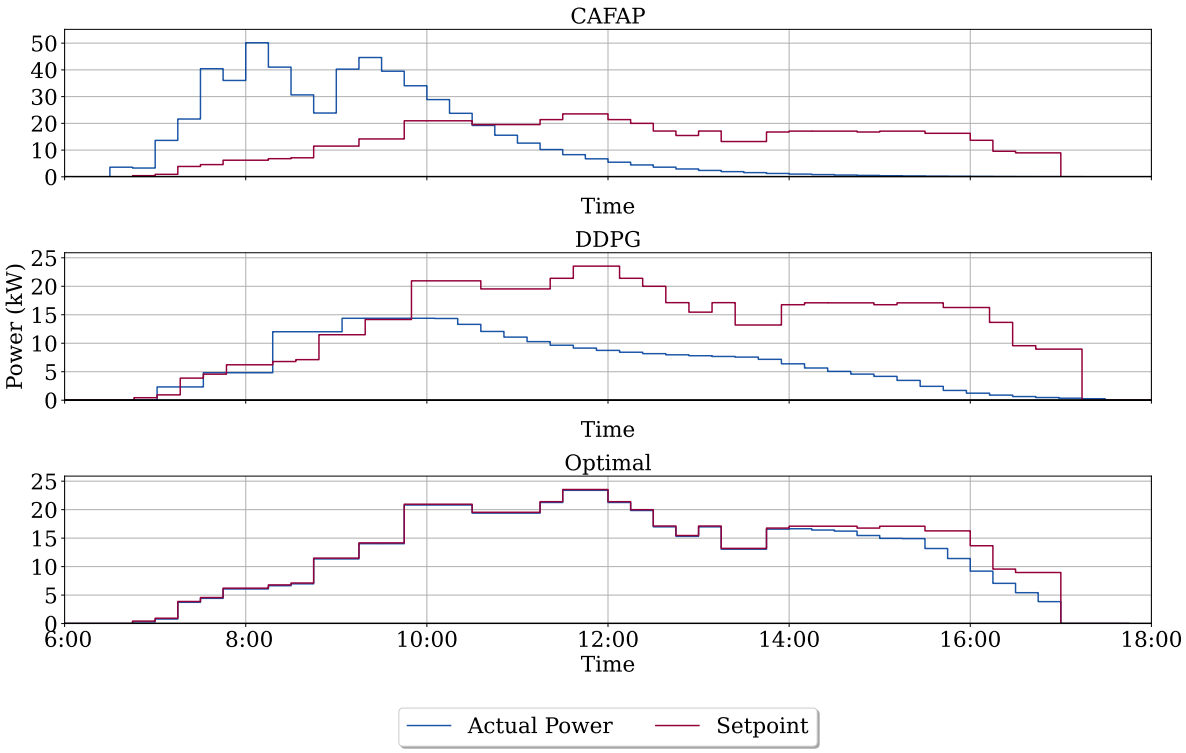


Figure 4.3: Power setpoints and actual power usage from one replay outputs the charging power alongside with predetermined power setpoints at each 15-minute time step throughout a day for three benchmark algorithms

Regarding Figure 4.3, the CAFAP algorithm is designed to charge EVs as quickly as possible. However, this approach tends to overshoot the power setpoints and inefficiently utilizes the contracted energy, as shown in the first plot. This overshooting not only leads to higher costs for the CPO when buying energy in the intraday market, but it may also pose risks to the transformer of the commercial building. On the other

hand, the DDPG algorithm aims to minimize the PST error by extending the charging duration rather than charging immediately. This strategy optimizes energy usage and reduces potential costs and risks associated with power overshoots.

Lastly, the Optimal charging algorithm, which is calculated using the Gurobi Solver, is capable of achieving the theoretical optimum in terms of minimizing the PST error for the formulated problem setup. However, it is worth noting that due to the inherent uncertainties in EV charging, such optimal outcomes are not possible to achieve in practical scenarios.

To thoroughly evaluate algorithm performances, it is essential to consider multiple replays, typically around 100, and analyze the average values followingly. This approach is necessary to make an accurate assessment using the comparison metrics. It is also crucial to understand how the RL model, once trained, behaves in randomly generated replays within its training environment.

The first metric, which is the squared tracking error, is compared across all algorithms according to their performance for each replay in Figure 4.4. This metric is also the objective function that the Optimal algorithm aims to minimize, and it is very similar to the reward function of the DDPG. Therefore, it is important to make a comparison using this metric. The results indicate that the DDPG algorithm outperformed the CAFAP algorithm consistently in almost all replays. However, the Optimal algorithm performed better than both DDPG and CAFAP in terms of performance. This is an expected result due to the Optimal algorithm, the theoretical optimal for the formulated problem setup.

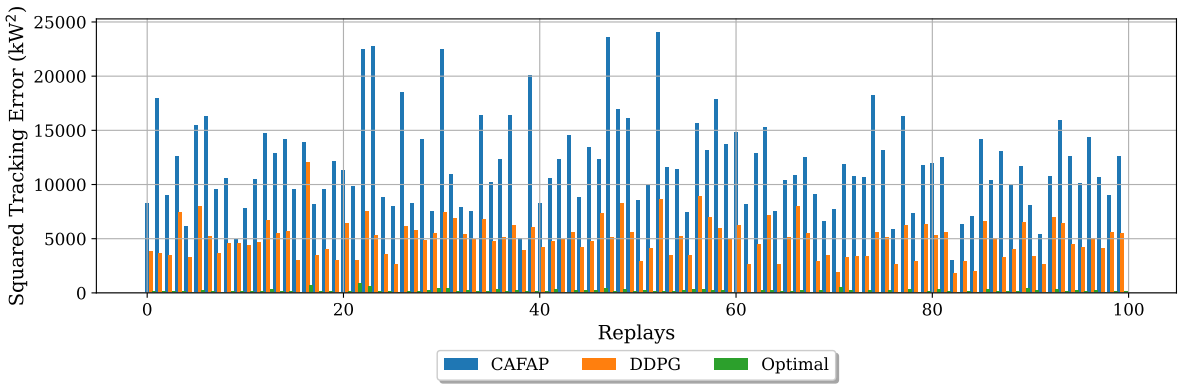


Figure 4.4: Squared tracking error throughout 100 evaluated replays

Following the evaluation of each replay, the average squared tracking error and its standard deviation obtained from 100 replays are compared in Figure 4.5. It can be observed that the DDPG algorithm outperformed CAFAP with a significant difference in their average performance. The highest value of DDPG, which is an outlier, is at the same level as the average of CAFAP. On the other hand, the Optimal performed better than both as expected.

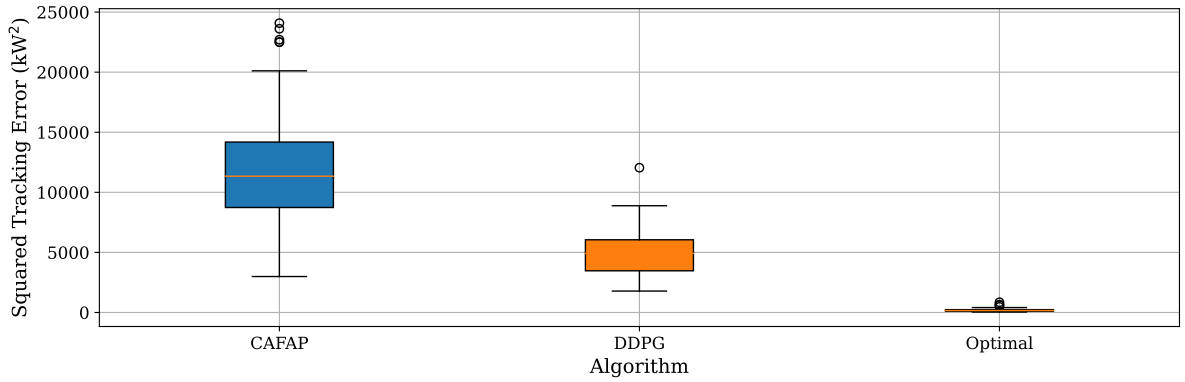


Figure 4.5: Averages and standard deviations of squared tracking error throughout 100 evaluated replays

After comparing the squared tracking error, another important metric, energy tracking error was analyzed for all three algorithms in Figures 4.6 and 4.7. This metric shows the exact deviation from the contracted power setpoints, and thus, it is particularly important. Similar to the squared tracking error results, the proposed DDPG algorithm outperformed the CAFAP algorithm significantly in almost every replay.

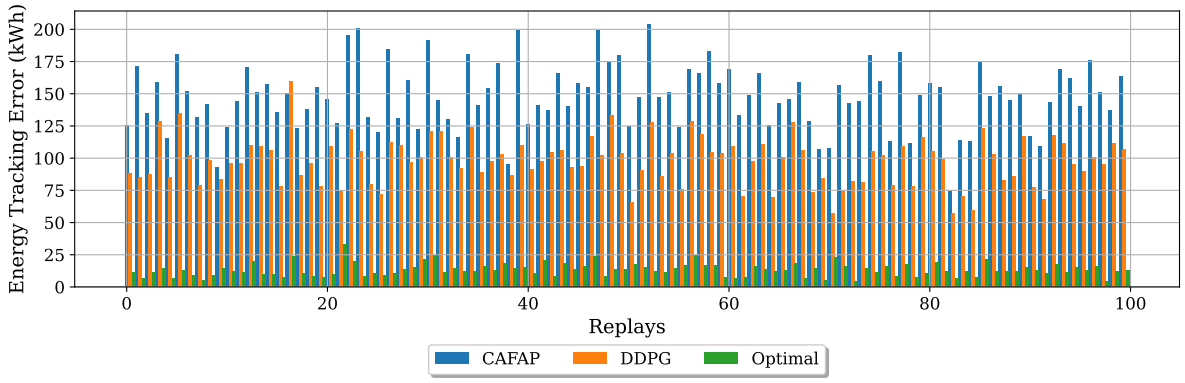


Figure 4.6: Energy tracking error throughout 100 evaluated replays

Furthermore, the DDPG achieved much lower energy tracking errors than CAFAP, averaging around 50 kWh less per replay, as shown in Figure 4.7 and in Table 4.6. Once again, the Optimal benchmark algorithm demonstrated superior performance, surpassing both DDPG and CAFAP in minimizing deviations from power setpoints.

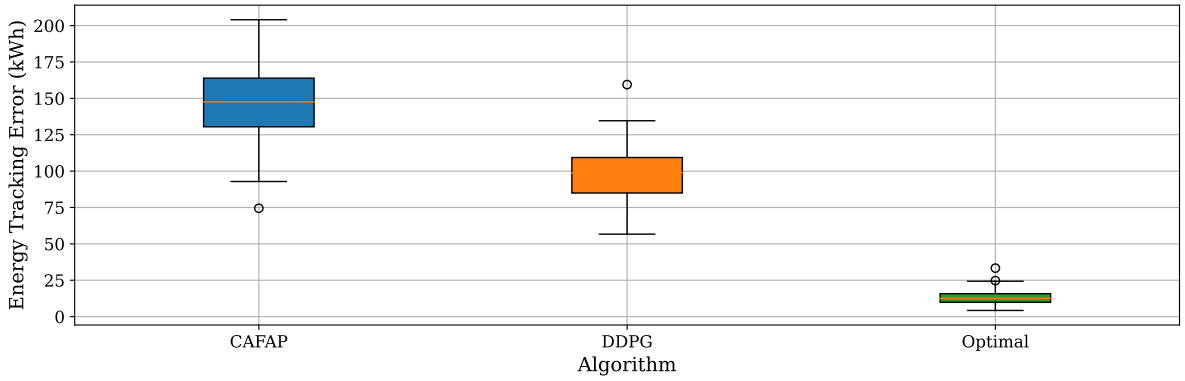


Figure 4.7: Averages and standard deviations of energy tracking error throughout 100 evaluated replays

It is important to evaluate how well each algorithm manages to stay within or exceed the power setpoints. The power tracker surplus is one such comparison metric. The power setpoints represent the planned power levels for charging EVs, and exceeding them will result in profit loss for the CPO and may also cause capacity issues for the transformers in commercial buildings. Figure 4.8 shows that for every replay the DDPG algorithm managed to perform much better than CAFAP in terms of adhering to the power setpoints. This result was expected because CAFAP prioritizes charging as fast as possible without considering the power setpoints.

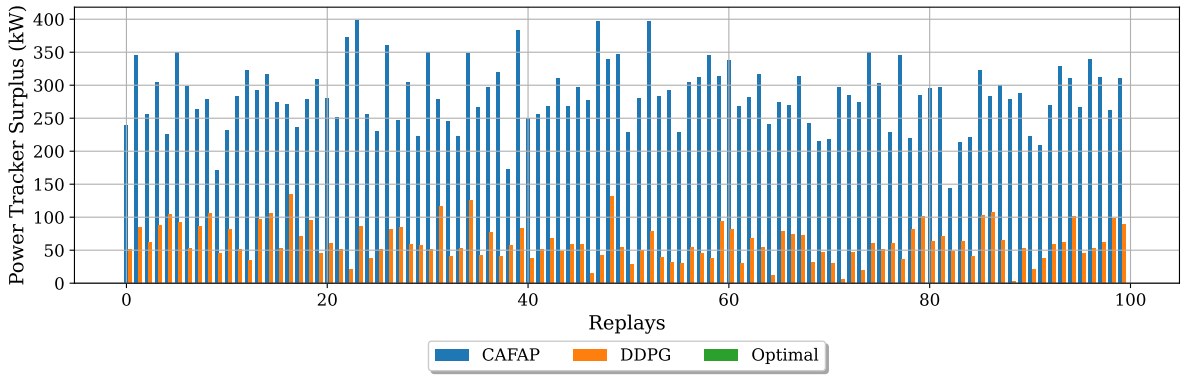


Figure 4.8: Power tracker surplus throughout 100 evaluated replays

Moreover, it is worth noting that the lowest power tracker surplus obtained by CAFAP is actually an outlier and it exceeds the maximum value obtained by the DDPG, as can be seen from Figure 4.9. On the other hand, the lowest tracker surplus of DDPG is equivalent to the average of the Optimal. This indicates that DDPG outperforms CAFAP by a significant margin, and in some instances, it even performs almost as well as the Optimal.

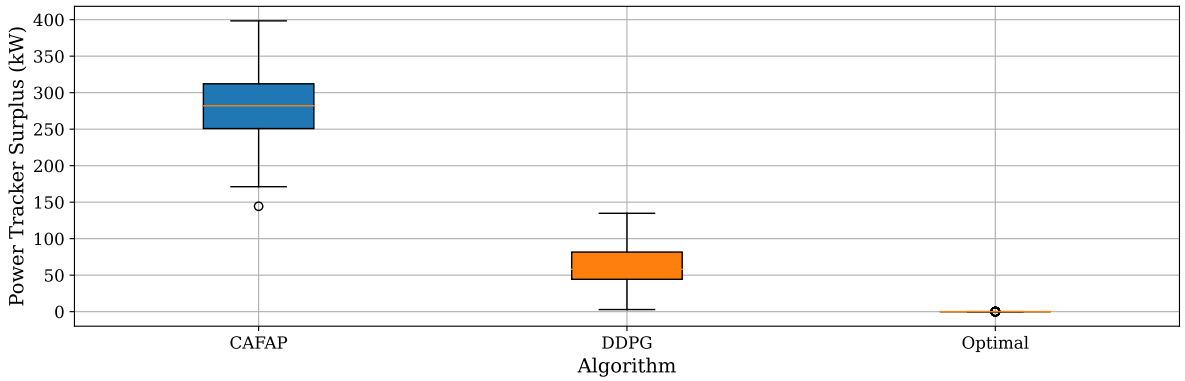


Figure 4.9: Averages and standard deviations of power tracker surplus throughout 100 evaluated replays

Table 4.6 provides further quantification of this difference. The DDPG algorithm exceeded the power setpoints by an average of 61.73 kW per replay, while the CAFAP algorithm significantly surpassed this, with an average of 283.77 kW per replay. This contrast highlights a key benefit of the proposed DDPG algorithm, which focuses on managing power usage within planned limits, unlike CAFAP's approach, which prioritizes speed over efficiency.

The user satisfaction metric was selected to evaluate how satisfied EV users are upon departure. As depicted in Figure 4.10, both the CAFAP and Optimal algorithms enabled EVs to be charged to almost their full battery capacity. This outcome is expected due to the fast charging strategy of the CAFAP and the Optimal algorithm's status as a benchmark.

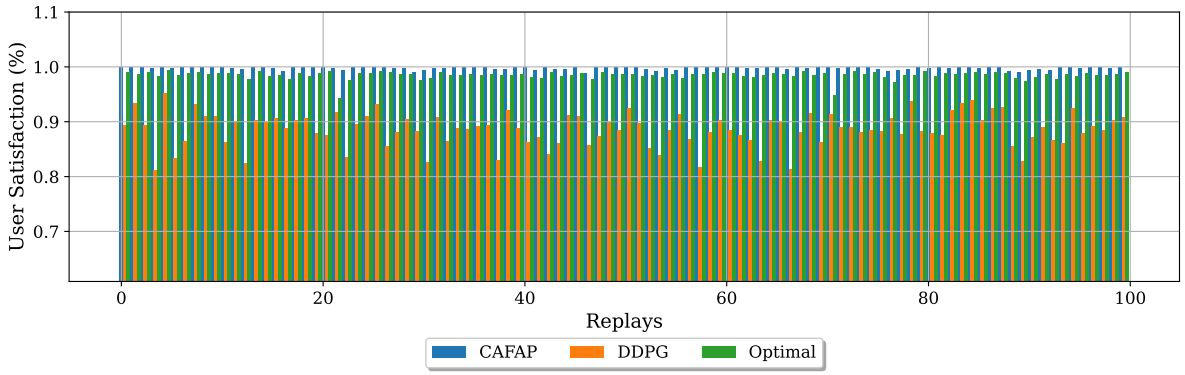


Figure 4.10: User satisfaction throughout 100 evaluated replays

However, the DDPG algorithm did not perform as well as the other benchmark algorithms in this metric by achieving an average user satisfaction rate of approximately 88.6% across 100 replays as shown in Figure 4.11 and Table 4.6. This suggests that while DDPG excels in reducing metrics such as energy tracking error and power tracker surplus, it does so at the expense of user satisfaction.

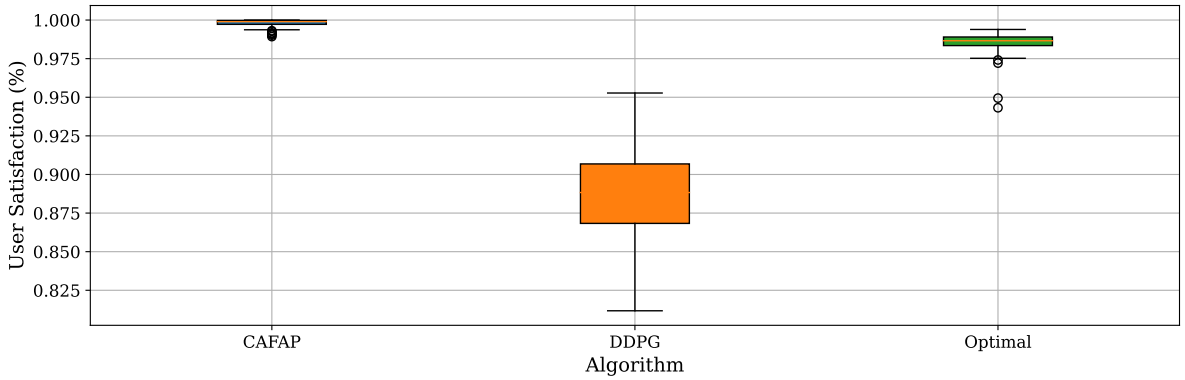


Figure 4.11: Averages and standard deviations of user satisfaction throughout 100 evaluated replays

In the context of this study, specifically in a workplace setting where charging can be scheduled during work hours, a compromise in user satisfaction is considered acceptable. The observed minimum level of user satisfaction from Figure 4.11 was approximately 81%, which indicates that even though DDPG may sacrifice some user satisfaction, it still maintains a reasonably high baseline level. This decision demonstrates a strategic trade-off between minimizing the PST error and maximizing the EV user experience. Yet, it is worth noting that reaching almost 90% user satisfaction on average while decreasing the energy tracking error by 34% on average in comparison to CAFAP is considered a significant outcome for the DDPG algorithm.

Furthermore, despite the Optimal algorithm outperforming DDPG in all comparison metrics, DDPG stood out in terms of speed by optimizing charging in 10 seconds for 100 replays compared to the Optimal algorithm, which required 35 seconds. The difference between the calculation speeds is expected to grow with the increasing number of chargers, which is investigated in Section 4.1.4.

Table 4.6: Performance of Algorithms

Algorithm	**Squared Tracking Error** (kW^2) ϵ^{tr}		**Energy Tracking Error** (kWh) $\lvert\epsilon^{tr}\rvert$		**User Satisfaction** $(\%)$ ϵ^{usr}		**Power Tracker Surplus** (kW) ϵ^{sur}	
	Average	*Std*	*Average*	*Std*	*Average*	*Std*	*Average*	*Std*
CAFAP	11862.31	4278.17	147.68	25.38	0.997963	0.002	283.77	49.30
DDPG	4972.03	1753.99	97.62	18.65	0.886139	0.030	61.73	27.42
Optimal - MINLP	189.56	129.06	13.16	5.06	0.985195	0.007	0.000257	0.000582

4.1.3 Transformer Capacity Limit

In this section, the focus is on the transformer overload metric, which gains importance in the context of the PST error minimization problem for a workplace, considering that EV chargers are connected to the commercial building's transformer. Given that these EV chargers share the same transformer as the commercial building, it is important to monitor and manage the power loads to avoid exceeding the transformer's capacity, which can lead to significant operational issues.

Typically, the power capacity of a transformer is determined based on the maximum current and voltage specifications of the connected chargers. However, in this study, a different approach is used. The transformer's power capacity limit is set to twice the maximum power setpoint recorded during the simulations. This approach is used because chargers usually do not operate at their maximum charging power simultaneously. Additionally, not all EVs require their peak charging power at the same time. Therefore, by calculating the total charging demand for each simulation replay and taking these usage patterns into consideration, it becomes feasible to implement the mentioned calculation of the transformer power capacity.

Furthermore, the conventional method of calculating transformer capacities often leads to oversized transformers, based on theoretical maximums. This method ignores the actual usage and results in wasteful and excessive capacity. On the other hand, the proposed approach adopts a calculated limit that reflects the actual usage, ensuring that the transformer capacity is effectively utilized without being wasteful.

The maximum power setpoint observed throughout the simulations led to the transformer power limit being set at 60 kW. Figure 4.12 illustrates transformer overloads for each test replay. Out of the 100 replays, only the CAFAP algorithm resulted in overloads while the DDPG and Optimal algorithms did not. This indicates that the proposed DDPG algorithm can effectively reduce power overshoots, which in turn enhances the flexibility of managing other loads in the commercial building apart from just EV charging.

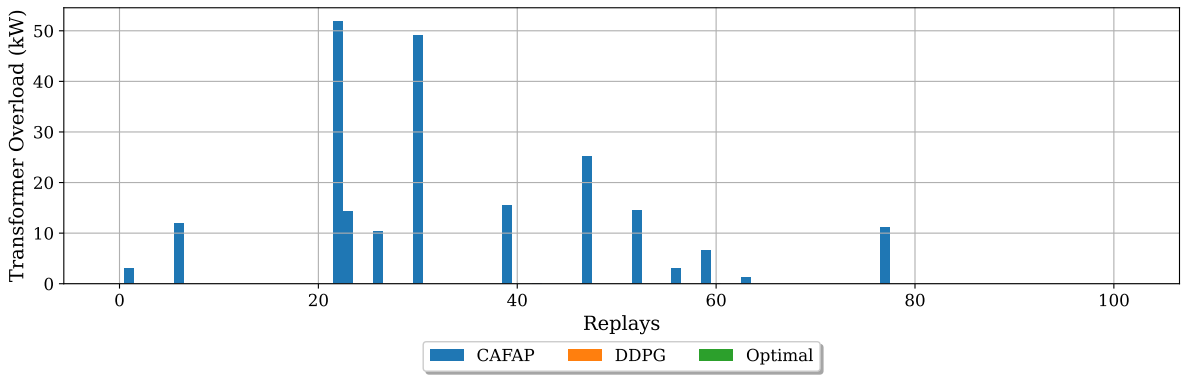


Figure 4.12: Transformer overloads throughout 100 evaluated replays

Additionally, the CAFAP algorithm may lead to significant power draw variations from the grid, as evidenced in the 22nd and 30th replays reaching around 50 kW overshoots. This variation underscores the need for a smart charging approach to avoid stressing the transformer beyond its limits.

4.1.4 Scalability of the Algorithm

In this section, the scalability of the algorithm is tested by training the same RL model with different numbers of EV chargers using the same set of hyperparameters. Each training and testing results are categorized and presented by the number of utilized EV chargers.

Initially, the model was trained with 3 chargers to determine whether the hyperparameters tuned for 10 chargers could still yield promising results. This step is crucial for assessing the adaptability of the model to smaller setups. Subsequently, the model was scaled up to 20 chargers, allowing the performance with an

increased number of chargers to be evaluated.

Finally, the model was tested with 50 chargers to test an extreme case. While the deployment of 50 EV chargers in a commercial building is not common practice today, it was recognized that the continuous rise in EV sales could make such a scenario feasible in the near future. Training the model with 50 chargers also allowed for the testing of the algorithm's ability to converge and minimize the PST error without any changes in the hyperparameters, even at this larger scale. This experiment is key in demonstrating the robustness and scalability of the RL model in accommodating future growth in EV integration to the grid.

3 Chargers

To begin with, the RL model was trained for 3 chargers, with all parameters being the same as the case study that was conducted for 10 chargers. The transformer capacity limit was set at 18 kW, due to the smaller scale of the model with only three chargers. As shown in Figure 4.13, the model's mean rewards converged around -40, which indicates that the model found a policy that maximizes its rewards. However, to confirm whether this policy is beneficial for the PST minimization target, the model needs to be tested with 100 randomly generated replays.

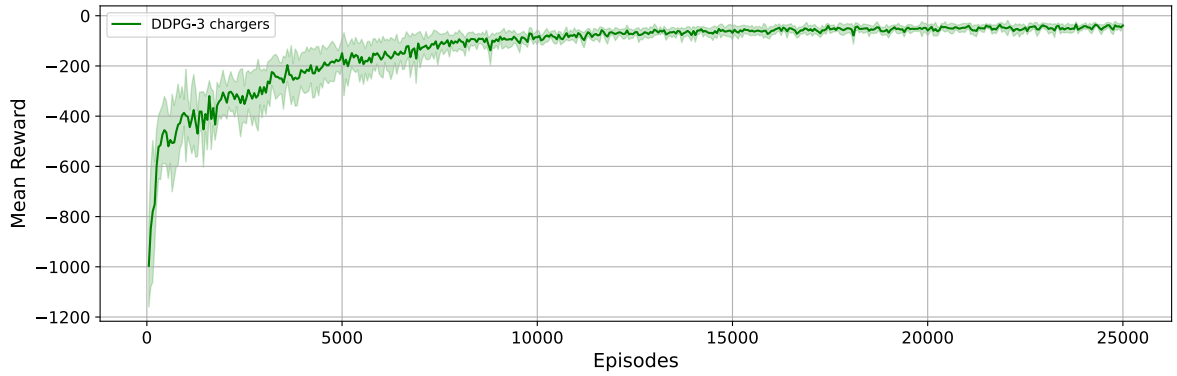


Figure 4.13: Mean rewards for 3 chargers from 10 training sessions

The model is tested with randomly generated replays after its training. The results indicate that the agent found a policy that reduced the PST error while maintaining a higher user satisfaction than the 10 chargers application. Although the agent found a sub-optimal policy, it is still effective on smaller scales. However, the energy tracking error was not reduced as much as in the case study for 10 chargers. Nonetheless, the results demonstrate the model's capability to be utilized on smaller scales. Additionally, the DDPG algorithm optimized charging schedules for 100 replays in 7 seconds, while the Optimal took 12 seconds. This is not a significant difference, however, it should be considered that only 3 chargers were deployed. As the last comparison metric, the model outperformed the CAFAP algorithm by decreasing the power tracker surplus by 44.8%. Table 4.7 provides an overview of the results.

Table 4.7: Performance of algorithms for 3 chargers

Algorithm	**Squared Tracking Error** (kW^2) ϵ^{tr}		**Energy Tracking Error** (kWh) $\lvert\epsilon^{tr}\rvert$		**User Satisfaction** $(\%)$ ϵ^{usr}		**Power Tracker Surplus** (kW) ϵ^{sur}	
	Average	*Std*	*Average*	*Std*	*Average*	*Std*	*Average*	*Std*
CAFAP	1646.925	1022.103	48.029	13.864	0.999	0.003	96.416	28.031
DDPG	1090.874	696.094	40.877	12.721	0.912	0.047	53.104	23.519
Optimal	42.631	31.338	4.439	1.868	0.972	0.013	0.001	0.001

20 Chargers

The DDPG algorithm is tested for 20 chargers to evaluate its scalability with an increasing number of EV chargers. The limit for transformer capacity was set at 120 kW, with the same scaling ratio as the total

number of chargers. Figure 4.14 shows that the RL agent was able to converge to a local optimal solution with the policy it found. However, to assess how the found policy performs, it is necessary to evaluate the results obtained from 100 replays. The mean rewards converged around -3500, which is significantly lower than the convergence level of the case study conducted for 10 chargers. This was expected since the PST error also increases with the number of chargers, causing an increase in all error metrics.

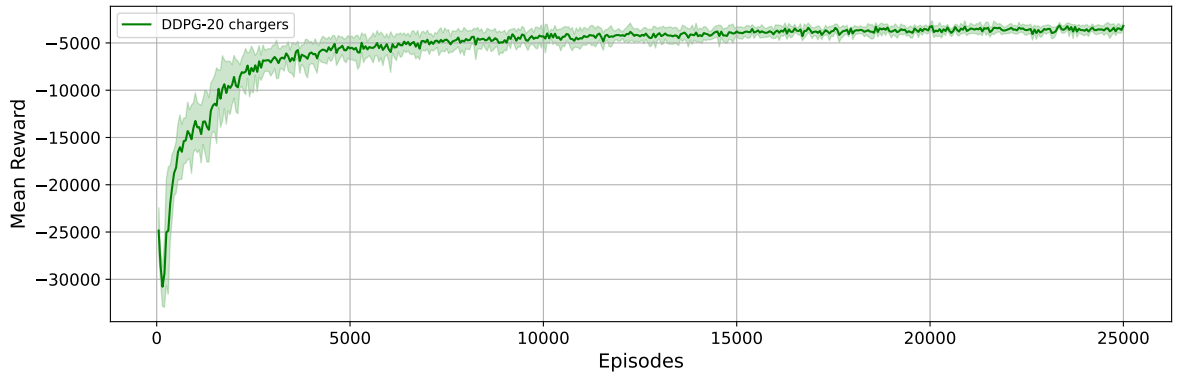


Figure 4.14: Mean rewards for 20 chargers from 10 training sessions

After testing the model across 100 replays, the results are summarized in Table 4.8. Notably, the average squared tracking error decreased significantly, with the DDPG outperforming the CAFAP by a considerable margin, even when accounting for standard deviations. Similarly, the energy tracking error metric also indicates that the model surpassed CAFAP's performance. However, the user satisfaction metric reveals a trade-off that while the model exceeds CAFAP's results, it does so at the expense of user satisfaction, with a modest decline of 15%. Despite this decrease, such a reduction is considered acceptable within the context of overall benefits.

On the power tracker surplus metric, the model significantly outperformed CAFAP, reducing power surplus by 83%. This reduction is particularly advantageous as it can lower the high costs associated with purchasing electricity in the intraday market by the CPO and also decrease the risks of overshooting the transformer power limit.

It is worth noting that optimizing charging schedules for DDPG took only 14 seconds, while it took 65 seconds for Optimal. This highlights that the Optimal algorithm takes significantly more time to solve the problem as the number of chargers increases.

Table 4.8: Performance of algorithms for 20 chargers

Algorithm	**Squared Tracking Error** (kW^2) ϵ^{tr}		**Energy Tracking Error** (kWh) $\lvert\epsilon^{tr}\rvert$		**User Satisfaction** $(\%)$ ϵ^{usr}		**Power Tracker Surplus** (kW) ϵ^{sur}	
	Average	*Std*	*Average*	*Std*	*Average*	*Std*	*Average*	*Std*
CAFAP	47441.286	12245.394	304.328	37.773	0.998	0.002	579.625	76.324
DDPG	27134.415	6252.681	231.359	28.300	0.846	0.023	98.225	39.395
Optimal	728.261	221.944	28.645	5.340	0.986	0.003	0.000	0.000

The combined results across these metrics suggest that the algorithm successfully applied a policy for the implementation of 20 chargers that is similar to the policy used in the earlier case study with 10 chargers. Consequently, it can be concluded that the model is scalable, maintaining its promises with the same parameters as it transitions from 10 to 20 chargers.

50 Chargers

For the final scalability test, the DDPG algorithm was evaluated using a setup of 50 chargers. For this extreme case, the transformer capacity limit was set to 300 kW. The mean reward for the algorithm is

illustrated in Figure 4.15. Unlike the mean reward curves from other scalability tests, it is clear that the algorithm could not find a policy that would converge to a local optimal. This result suggests that the RL agent can not learn the greater amount of complexities due to the significantly increased number of chargers when utilizing the same hyperparameter set. The greater complexity poses more significant challenges for the DDPG algorithm in assessing and identifying an optimal policy. In this regard, to improve the performance of the DDPG algorithm, hyperparameters such as action noise ($\mathcal{N}$) and the size of DNNs can be adjusted. Lowering the action noise reduces the amount of exploration the agent does, which increases the chances of finding a local optimal solution. On the other hand, increasing the size of DNNs can help the RL agent capture more intricate patterns and complexities. However, it is crucial to experiment with various hyperparameter configurations to find the optimal set that can effectively learn the complexities of 50 chargers.

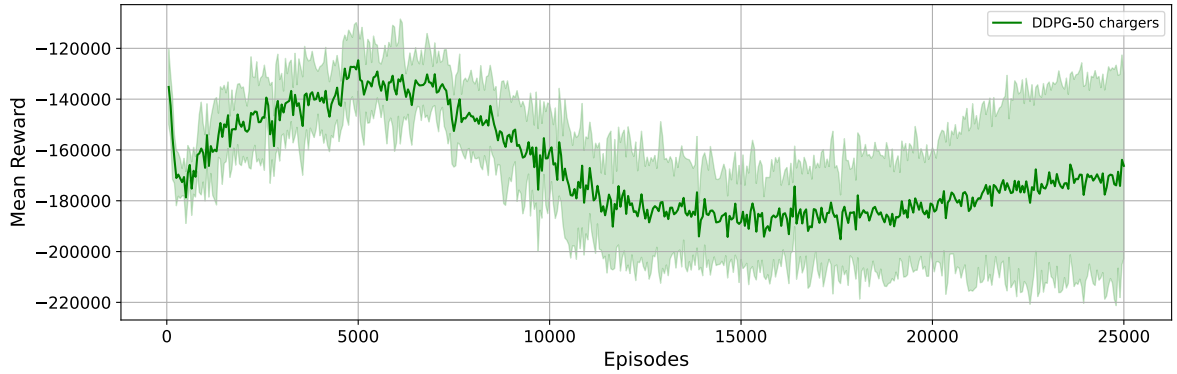


Figure 4.15: Mean rewards for 50 chargers from 10 training sessions

Following the training phase, the model was applied to 100 replays to evaluate its performance using the comparison metrics, despite the poor learning performance of the DDPG algorithm. The outcomes of this evaluation are summarized in Table 4.9, which details the results obtained from these replays.

It has been noticed that the squared tracking error decreased, but the decrease was not as significant as in the previous case studies for 10 and 20 chargers. This is especially true when considering the standard deviations alongside the average values. In addition to the squared tracking error, there was also a reduction in the energy tracking error. However, this improvement came at the cost of a more substantial decrease in user satisfaction when compared to the earlier tests with 10 and 20 chargers.

It is worth noting that the power tracker surplus metric has significantly decreased by 97%. This improvement indicates that there has been a conscious effort to avoid charging above the predetermined power setpoint level. However, despite these positive trends in certain metrics, the overall results suggest that the DDPG algorithm tends to avoid charging the EVs rather than risk exceeding the power setpoint.

The outcome for 50 chargers highlights the increased complexity faced by the DDPG algorithm as the number of chargers increases, which affects its ability to efficiently learn and implement optimal charging policies with the selected hyperparameter set. In addition, the Optimal algorithm is also hindered by increased complexity. Optimizing 100 replays took the Optimal algorithm 300 seconds, significantly increasing the calculation time. Conversely, the DDPG algorithm optimized charging schedules in just 20 seconds. This result suggests that DDPG can scale very well in terms of speed, while the Optimal algorithm suffers more from increased complexities.

Table 4.9: Performance of algorithms for 50 chargers

Algorithm	**Squared Tracking Error** (kW^2) ϵ^{tr}		**Energy Tracking Error** (kWh) $\lvert\epsilon^{tr}\rvert$		**User Satisfaction** $(\%)$ ϵ^{usr}		**Power Tracker Surplus** (kW) ϵ^{sur}	
	Average	*Std*	*Average*	*Std*	*Average*	*Std*	*Average*	*Std*
CAFAP	268536.942	40664.881	745.762	55.351	0.998	0.001	1408.680	110.592
DDPG	204358.383	30494.091	640.144	48.869	0.788	0.018	41.051	33.907
Optimal	4591.651	1106.559	76.738	10.074	0.986	0.002	0.000	0.000

Finally, the box plot in Figure 4.16 shows the average and standard deviation of 100 replays for all scales of the problem to represent the results of each scale together. As the number of EV chargers increases, the difference between the averages also increases. Higher amounts of EV chargers also result in an increased standard deviation, which is expected due to the larger problem scale. It can be observed that as the problem scale increases, the upper standard deviation level of DDPG aligns with the lower standard deviation of CAFAP, except for the 3 chargers scale. This suggests that in rare cases, DDPG and CAFAP can produce similar results in terms of energy tracking error, and, therefore, PST error.

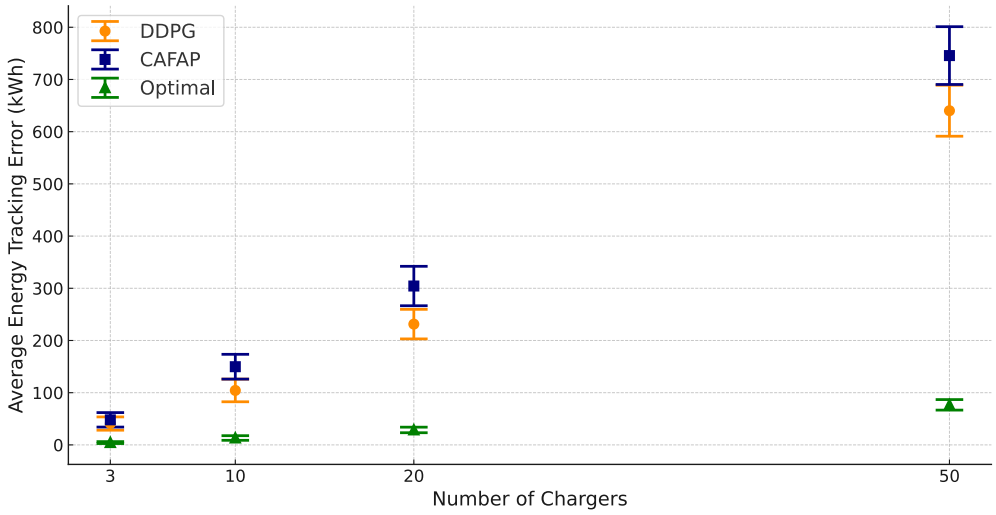


Figure 4.16: Energy tracking error average and standard deviations of 100 replays for all scales

4.2 Discussion

This study introduces an RL approach aimed at minimizing the PST error, a frequent optimization problem for CPOs during their operational activities. By integrating the characteristics of the EV charging environment, a DDPG algorithm was deployed to optimize the charging schedules of EVs to minimize PST error by meeting predetermined power setpoints.

The proposed approach was compared against two benchmark algorithms, CAFAP and a theoretical optimal derived from a MINLP formulation of the PST minimization problem. The simulation environment replicates a commercial building's parking lot equipped with 10 EV chargers in the Netherlands. Both the EV charging load and the building's other electrical loads share a common transformer, thereby necessitating a limit on the transformer's power capacity for charging EVs.

The training data for the DDPG algorithm comprised actual EV charging schedules from a workplace, reflecting the real-world settings of the problem. The algorithm did training over 25,000 episodes, corresponding to an equivalent number of days, utilizing these real-world data. To assess the effectiveness of the DDPG algorithm, 100 replays were generated, each representing a 12-hour charging period in a single day between 6 am and 6 pm. The outcomes of applying the DDPG and the benchmark algorithms to these replays

were analyzed. The results demonstrated the DDPG algorithm's superior performance in several metrics, especially over the CAFAP benchmark. Notably, in the case study involving 10 EV chargers, the DDPG algorithm significantly reduced the energy tracking error by 34% in kWh per replay compared to the CAFAP algorithm. Moreover, the DDPG reduced the power tracker surplus metric, indicating that the DDPG algorithm charged EVs while exceeding 78.2% less on the power setpoints than CAFAP. This reduction is crucial for CPOs because it decreases costs and better complies with transformer power limits.

Nevertheless, the DDPG algorithm's strategy highlights a trade-off, achieving significant reductions in power tracking surplus and energy tracking error but potentially at the cost of user satisfaction. Specifically, average user satisfaction dropped by 11.2% when switching from CAFAP to DDPG, with a decrease from 99.8% to 88.6% in 100 generated replays. Despite this decrease, such a level of user satisfaction may still be considered acceptable within the context of this problem. However, this finding points to areas for further improvement in balancing minimizing the PST error with user satisfaction in future works of this study.

Additionally, while the CAFAP algorithm outperformed the DDPG in terms of user satisfaction metric, it is important to address the circumstances under which CAFAP might fail. In particular, when EV demand exceeds the charging capacity, CAFAP fails, as indicated by the power tracker surplus metric. Considering that the power capacity can not be exceeded in practice, the user satisfaction obtained by CAFAP would decrease significantly. Thus, CAFAP's strategy to charge EVs as fast as possible may not suffice in such scenarios, necessitating a smarter scheduling approach like that of the DDPG algorithm. Furthermore, it is important to note that the user satisfaction metric indicates the SoC of EVs at departure; hence, another user satisfaction metric taking the charging prices into account might drastically change the perspective of DDPG's and CAFAP's performance in terms of user satisfaction. This comparison unfolds considering that the power setpoints are set according to the energy amounts contracted in the day-ahead market. If consumption exceeds these setpoints, the CPO is responsible for acquiring additional energy from the intraday market, which is usually more expensive than the day-ahead market. As a result, the CAFAP algorithm tends to charge EVs with more costly energy, which can lead to decreased user satisfaction due to higher prices.

It is worth mentioning that the DDPG algorithm failed to outperform the Optimal benchmark algorithm in any of the comparison metrics. However, it is important to note that the Optimal benchmark algorithm requires information about EVs' arrival and departure times and their SoC levels; thus, the results of the Optimal algorithm give theoretical optimal results. Additionally, the required information is not practically feasible for the CPO to obtain. On the other hand, once trained, the DDPG algorithm optimizes EV charging schedules much faster than the Optimal benchmark algorithm.

Moreover, the DDPG algorithm was tested for scalability with varying numbers of chargers, including 3, 20, and 50. The same hyperparameter settings were used as in the initial case study, which was conducted for 10 chargers. The scability tests conducted using the DDPG algorithm showed that it was able to find policies that led to the convergence of mean rewards in tests with 3 and 20 chargers. However, the mean rewards did not converge when 50 chargers were employed. This indicates that the DDPG algorithm was unable to capture the patterns due to the high complexity of the environment with 50 chargers using the same hyperparameter set. It is possible to scale the DDPG algorithm with 50 chargers by adjusting the hyperparameter set. This can be achieved by increasing the size of the DNNs to capture more complex patterns. However, this process requires more computational time, especially when tuning the hyperparameters for varied training sessions.

During the next testing phase, the DDPG algorithm was implemented to solve 100 generated replays. This approach did not yield satisfactory results for scenarios with 3 and 50 chargers. However, the 20-charger scenario produced similar results to the case study with 10 chargers, indicating that the proposed algorithm effectively scales up to 20 chargers without changing any hyperparameters. The DDPG algorithm can also be scaled down or up by adjusting the utilized hyperparameter sets, but this comes at the cost of increased computational burden, as mentioned.

5

Conclusion and Recommendations

Four research questions were determined at the beginning of this study alongside a research objective. In this chapter, firstly the determined research questions are answered in detail in Section 5.1.1. Consecutively the main research question related to the research objective of this thesis is explained and answered in detail in Section 5.1.2. The chapter ends with the recommendations in Section 5.2, which gives directions for the future work of this study.

5.1 Conclusion

5.1.1 Answers to the Research Questions

1. What are the key characteristics and constraints of the model-free online EV charging problem in the context of a workplace parking lot?

The problem is formulated in Chapter 3, Section 3.1 for a workplace parking lot. One of the key characteristics is the predictable nature of EVs' arrival and departure times, which align with fixed working hours. Typically, EVs arrive after 6 am and depart before 6 pm. This predictability leads to the training and testing of the DDPG algorithm using actual arrival, departure times, and SoC data. Additionally, the transformer power limit was recognized as a key characteristic. In the context of the formulated problem, the transformer power limit was used as a metric and was determined differently than a traditional approach. It was determined by taking the reduced and intermittent energy demand during the charging hours into consideration. Unlike traditional power limit calculations that calculate a power limit by taking the maximum current and voltage levels of the chargers into consideration, this approach adjusts the transformer power limit better to suit the specific usage patterns of a workplace environment. This helps to optimize both energy usage and infrastructure efficiency.

2. What are the key factors influencing the performance of the Deep Deterministic Policy Gradient (DDPG) algorithm in optimizing power setpoint tracking (PST) for EV smart charging?

The performance of the proposed DDPG algorithm is affected by several key factors. These can be categorized by environmental and algorithmic related factors. Environmental factors such as the configuration of charging infrastructure, EV models, used data, variability in EV arrival and departure times, and fluctuations in electricity prices play crucial roles. Secondly, the performance of the DDPG algorithm is heavily dependent on the design of the state and action spaces, reward function, and appropriate tuning of hyperparameters. These steps are crucial in determining the performance outcome of the algorithm. Therefore, careful consideration and selection of these design elements are fundamental to finding a promising policy for the DDPG algorithm.

3. How do RL-based smart charging methods improve upon or differ from mathematical optimization methods used for smart charging in managing the energy demands and grid interactions of EVs?

The proposed DDPG algorithm and a MINLP formulation of the PST problem have shown distinct outcomes in their application to the problem. The MINLP approach offers a theoretically optimal solution, incorporating detailed parameters such as arrival and departure times and the SoC of the EVs. In contrast, the DDPG algorithm, once trained, demonstrated its effectiveness in addressing the PST minimization problem and outperformed an uncontrolled charging benchmark algorithm, CAFAP. Notably, the DDPG algorithm quickly allocated power to 10 chargers in about 10 seconds, compared to 35 seconds for the MINLP solution. This speed advantage extended to larger scenarios as well; for 50 chargers, the DDPG algorithm took about 20 seconds to allocate power, while the MINLP required approximately 300 seconds. These results highlight the DDPG algorithm's capability for less computational burden during operation and faster decision-making and show that when the complexity of the problem increases, the computational burden of MINLP exponentially increases. It is worth noting that while the DDPG algorithm is faster, it did not reach the highest possible performance in all metrics compared to the theoretical optimal solution provided by the MINLP formulation.

4. How does the applied RL algorithm scale with the varying number of EV chargers?

In Chapter 4 Section 4.1.4, the scalability of the DDPG algorithm was examined in detail. The findings indicated that a sub-optimal policy was found across different scales, due to the convergence of mean rewards in all scales except for the extreme 50 chargers case. Additionally, some challenges were revealed during a detailed analysis of the performance during 100 replay tests. It was found that the smallest and the largest models underperformed, indicating scalability issues at these extremes. Interestingly, it was observed that doubling the number of EV chargers from 10 to 20 did not negatively impact the performance, implying that the algorithm can scale up to twice its initial size without modifications to the algorithm's parameters, except for adjusting the size of the state and action space. This suggests that robust scalability was achieved for moderate increases in scale, although further experiments in the hyperparameters, state space and reward function are needed for larger scales with better results to increase user satisfaction while decreasing the PST error.

5.1.2 Research Objective

How to effectively optimize the charging schedules of EVs to meet the CPO's contracted power setpoints in a workplace setting using RL algorithms?

This study began with a literature review to investigate the optimization of EV charging using various methods, with the main focus on RL algorithms. As such, the dynamics of the Dutch electricity market were also researched to ensure that the proposed solution is applicable in the real world for all stakeholders. Thus, the problem is formulated from the point of view of CPOs. The problem was then framed in the context of a workplace to emphasize the RL algorithm's capability to identify EV usage patterns. The limitations of RL algorithms were identified after the literature review, particularly with respect to discrete state and action spaces, which are inherent in RL algorithms like Q-learning and DQN. Therefore, the DDPG algorithm, which offers continuous state and action spaces, was selected to avoid these limitations. A V2G simulator called EV2Gym was used to run simulations and test the algorithm. The DDPG algorithm was incorporated into the EV2Gym simulation environment using the stable baselines library. A methodology was then formulated to test the proposed approach fairly. The algorithm was trained with real-world open-source data for tuning, and then it was applied to 100 generated random replays that the algorithm had not been trained on before. The algorithm's performance was evaluated using several comparison metrics with benchmark algorithms such as CAFAP and theoretical optimal provided by a MINLP formulation of the PST problem. It was found that the trained DDPG algorithm performed better than the CAFAP algorithm in terms of minimizing the PST error objective, even at the expense of user satisfaction. However, the decrease in user satisfaction was found to be acceptable, at only 11%, while the PST error decreased by 34%. This study shows that by disaggregating the contracted energy capacity of a CPO, the proposed DDPG algorithm can significantly reduce CPO costs and improve compliance with grid constraints at CPO's charging stations.

5.2 Recommendations

The recommendations for the future work of this study are highlighted as bullet points in this section below:

- In the case study with 10 chargers, the results showed that using the DDPG algorithm reduced PST error but decreased user satisfaction. To improve this, further tests can be conducted by including a charging priority variable or user satisfaction in the state and reward functions.
- It took approximately 3 hours to complete one training session for 10 EV chargers. Optimizing the DDPG algorithm will decrease this computational burden, making experimentation less effortful.
- Electricity prices can be implemented to the user satisfaction metric to highlight the significance of charging prices and how they affect EV users' satisfaction in practice. Incorporating this will present the advantages of the DDPG algorithm over CAFAP drastically.
- Automating the training of the DDPG algorithm can involve gradually adjusting hyperparameters for various state and reward functions. This enables scanning a larger space of possible hyperparameter sets to identify an optimal policy that minimizes the PST error while ensuring EV users remain satisfied.
- A new approach can be introduced to increase DDPG's learning performance by sampling mini-batches according to a priority strategy.
- The open source data used in the DDPG algorithm was sourced from the Netherlands. The algorithm can be tested in various settings globally to evaluate its performance and robustness.
- Using continuous state and action spaces provides a more accurate representation of the EV charging process. However, this approach also increases the complexity of the RL algorithm required to learn the process. Alternatively, other RL algorithms that use discrete state and action spaces, such as DQN, can be used to compare the performance with DDPG.
- The simulation's resolution can be enhanced from 15 minutes to 1 minute for more precise calculation of metrics such as the cost of electricity. This will help in accurately determining the total cost of the PST error for CPOs.
- Implementing V2G scenarios to the formulated case study can be interesting. However, it is worth mentioning that the problem will get more complex.

References

[1] IEA. *Net Zero by 2050*. License: CC BY 4.0. 2021. URL: https://www.iea.org/reports/net-zero-by-2050.

[2] IEA. *Global EV Outlook 2023*. License: CC BY 4.0. 2023. URL: https://www.iea.org/reports/global-ev-outlook-2023.

[3] Jason Eden. *EU Approves 2035 Ban on Internal Combustion Engines*. Accessed on: 25.10.2023. 2023. URL: https://www.energyintel.com/00000187-28b5-df45-a9df-7cf51aac0000.

[4] Bill Visnic. *Europe steps back from 2035 ICE ban*. 2023. URL: https://www.sae.org/news/2023/03/european-ice-ban.

[5] Omid Sadeghian, Arman Oshnoei, Behnam Mohammadi-ivatloo, Vahid Vahidinasab, and Amjad Anvari-Moghaddam. "A comprehensive review on electric vehicles smart charging: Solutions, strategies, technologies, and challenges". In: *Journal of Energy Storage* 54 (Oct. 2022), p. 105241. ISSN: 2352-152X. DOI: 10.1016/j.est.2022.105241.

[6] Kang Miao Tan, Vigna K. Ramachandaramurthy, and Jia Ying Yong. "Integration of electric vehicles in smart grid: A review on vehicle to grid technologies and optimization techniques". In: *Renewable and Sustainable Energy Reviews* 53 (Jan. 2016), pp. 720–732. ISSN: 1364-0321. DOI: 10.1016/j.rser.2015.09.012.

[7] Khizir Mahmud, M. J. Hossain, and Jayashri Ravishankar. "Peak-Load Management in Commercial Systems With Electric Vehicles". In: *IEEE Systems Journal* 13.2 (June 2019), pp. 1872–1882. ISSN: 1937-9234. DOI: 10.1109/JSYST.2018.2850887.

[8] Sungwoo Bae and Alexis Kwasinski. "Spatial and Temporal Model of Electric Vehicle Charging Demand". en. In: *IEEE Transactions on Smart Grid* 3.1 (Mar. 2012), pp. 394–403. ISSN: 1949-3053, 1949-3061. DOI: 10.1109/TSG.2011.2159278.

[9] Omid Sadeghian, Arman Oshnoei, Behnam Mohammadi-ivatloo, Vahid Vahidinasab, and Amjad Anvari-Moghaddam. "A comprehensive review on electric vehicles smart charging: Solutions, strategies, technologies, and challenges". In: *Journal of Energy Storage* 54 (Oct. 2022), p. 105241. ISSN: 2352-152X. DOI: 10.1016/j.est.2022.105241.

[10] EDGAR/JRC. *Distribution of carbon dioxide emissions worldwide in 2022, by sector [Graph]*. In Statista. Retrieved November 19, 2023. Sept. 2023. URL: https://www-statista-com.tudelft.idm.oclc.org/statistics/1129656/global-share-of-co2-emissions-from-fossil-fuel-and-cement/.

[11] IEA. *Distribution of carbon dioxide emissions produced by the transportation sector worldwide in 2022, by sub sector [Graph]*. In Statista. Retrieved November 19, 2023. July 2023. URL: https://www-statista-com.tudelft.idm.oclc.org/statistics/1185535/transport-carbon-dioxide-emissions-breakdown/.

[12] Carla B. Robledo, Vincent Oldenbroek, Francesca Abbruzzese, and Ad J. M. van Wijk. "Integrating a hydrogen fuel cell electric vehicle with vehicle-to-grid technology, photovoltaic power and a residential building". In: *Applied Energy* 215 (Apr. 2018), pp. 615–629. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2018.02.038.

[13] International Energy Agency. *Global EV Data Explorer*. IEA, Paris. 2023. URL: https://www.iea.org/data-and-statistics/data-tools/global-ev-data-explorer.

[14] *Electric Vehicles - Netherlands*. Retrieved November 20, 2023. Sept. 2023. URL: https://www-statista-com.tudelft.idm.oclc.org/outlook/mmo/electric-vehicles/netherlands.

[15] Lance Noel, Gerardo Zarazua de Rubens, Johannes Kester, and Benjamin K. Sovacool. "Understanding the socio-technical nexus of Nordic electric vehicle (EV) barriers: A qualitative discussion of range, price, charging and knowledge". In: *Energy Policy* 138 (Mar. 2020), p. 111292. ISSN: 0301-4215. DOI: 10.1016/j.enpol.2020.111292.

[16] H. S. Das, M. M. Rahman, S. Li, and C. W. Tan. "Electric vehicles standards, charging infrastructure, and impact on grid integration: A technological review". In: *Renewable and Sustainable Energy Reviews* 120 (Mar. 2020), p. 109618. ISSN: 1364-0321. DOI: 10.1016/j.rser.2019.109618.

[17] Murat Yilmaz and Philip T. Krein. "Review of benefits and challenges of vehicle-to-grid technology". In: *2012 IEEE Energy Conversion Congress and Exposition (ECCE)*. Sept. 2012, pp. 3082–3089. DOI: 10.1109/ECCE.2012.6342356. URL: https://ieeexplore.ieee.org/abstract/document/6342356?casa_token=wTUQdoLf0QgAAAAA:XRtg4EFIRxg__ys1_GooYM31Mu5PcAwQIFl239mhgmrLQXcNEi9whwugzr1L2zmpRd9nIYnHS7be.

[18] Remco A. Verzijlbergh, Marinus O. W. Grond, Zofia Lukszo, Johannes G. Slootweg, and Marija D. Ilic. "Network Impacts and Cost Savings of Controlled EV Charging". In: *IEEE Transactions on Smart Grid* 3.3 (Sept. 2012), pp. 1203–1212. ISSN: 1949-3061. DOI: 10.1109/TSG.2012.2190307.

[19] M. Secchi, G. Barchi, D. Macii, and D. Petri. "Smart electric vehicles charging with centralised vehicle-to-grid capability for net-load variance minimisation under increasing EV and PV penetration levels". In: *Sustainable Energy, Grids and Networks* 35 (Sept. 2023), p. 101120. ISSN: 2352-4677. DOI: 10.1016/j.segan.2023.101120.

[20] Javier Gallardo-Lozano, M. Isabel Milanés-Montero, Miguel A. Guerrero-Martínez, and Enrique Romero-Cadaval. "Electric vehicle battery charger for smart grids". In: *Electric Power Systems Research* 90 (Sept. 2012), pp. 18–29. ISSN: 0378-7796. DOI: 10.1016/j.epsr.2012.03.015.

[21] Scott B. Peterson, Jay Apt, and J. F. Whitacre. "Lithium-ion battery cell degradation resulting from realistic vehicle and vehicle-to-grid utilization". In: *Journal of Power Sources* 195.8 (Apr. 2010), pp. 2385–2392. ISSN: 0378-7753. DOI: 10.1016/j.jpowsour.2009.10.010.

[22] TenneT. *Market Roles*. 2023. URL: https://www.tennet.eu/market-roles.

[23] Laurens De Vries. *01 Organization of the electricity sector*. Lecture Notes in SET3055. Sept. 2022.

[24] Fehmi Tanrisever, Kursad Derinkuyu, and Geert Jongen. "Organization and functioning of liberalized electricity markets: An overview of the Dutch market". In: *Renewable and Sustainable Energy Reviews* 51 (Nov. 2015), pp. 1363–1374. ISSN: 1364-0321. DOI: 10.1016/j.rser.2015.07.019.

[25] Laurens de Vries, Aad F. Correljé, Hamilcar P.A. Knops, and Reinier van der Veen. *Electricity Markets*. Lecture Notes in SET3055. 2019.

[26] J. M. Jørgensen, S. H. Sørensen, K. Behnke, and P. B. Eriksen. "EcoGrid EU — A prototype for European Smart Grids". In: *2011 IEEE Power and Energy Society General Meeting*. July 2011, pp. 1–7. DOI: 10.1109/PES.2011.6038981. URL: https://ieeexplore.ieee.org/document/6038981.

[27] Willett Kempton, Jasna Tomic, Steven Letendre, Alec Brooks, and Timothy Lipman. "Vehicle-to-Grid Power: Battery, Hybrid, and Fuel Cell Vehicles as Resources for Distributed Electric Power in California". In: (June 2001). URL: https://escholarship.org/uc/item/5cc9g0jp.

[28] A. Brooks. "Integration of electric drive vehicles with the power grid-a new application for vehicle batteries". In: *Seventeenth Annual Battery Conference on Applications and Advances. Proceedings of Conference (Cat. No.02TH8576)*. Jan. 2002, pp. 239–. DOI: 10.1109/BCAA.2002.986406. URL: https://ieeexplore.ieee.org/document/986406/authors#authors.

[29] Ksenia Poplavskaya and Laurens de Vries. "Chapter 5 - Aggregators today and tomorrow: from intermediaries to local orchestrators?" In: *Behind and Beyond the Meter*. Ed. by Fereidoon Sioshansi. Academic Press, Jan. 2020, pp. 105–135. ISBN: 978-0-12-819951-0. DOI: 10.1016/B978-0-12-819951-0.00005-0. URL: https://www.sciencedirect.com/science/article/pii/B9780128199510000050.

[30] Stavros Orfanoudakis, Cesar Diaz-Londono, Yunus E. Yılmaz, Peter Palensky, and Pedro P. Vergara. *EV2Gym: A Flexible V2G Simulator for EV Smart Charging Research and Benchmarking*. 2024. arXiv: 2404.01849.

[31] P.M. Pardalos and M.G.C. Resende. *Handbook of Applied Optimization*. Oxford University Press, 2002.

[32] Andu Dukpa and Boguslaw Butrylo. "MILP-Based Profit Maximization of Electric Vehicle Charging Station Based on Solar and EV Arrival Forecasts". en. In: *Energies* 15.1515 (Jan. 2022), p. 5760. ISSN: 1996-1073. DOI: 10.3390/en15155760.

[33] Edathil Srilakshmi and Shiv P. Singh. "Energy regulation of EV using MILP for optimal operation of incentive based prosumer microgrid with uncertainty modelling". In: *International Journal of Electrical Power & Energy Systems* 134 (Jan. 2022), p. 107353. ISSN: 0142-0615. DOI: 10.1016/j.ijepes.2021.107353.

[34] Dennis van der Meer, Gautham Ram Chandra Mouli, Germán Morales-España Mouli, Laura Ramirez Elizondo, and Pavol Bauer. "Energy Management System With PV Power Forecast to Optimally Charge EVs at the Workplace". In: *IEEE Transactions on Industrial Informatics* 14.1 (Jan. 2018), pp. 311–320. ISSN: 1941-0050. DOI: 10.1109/TII.2016.2634624.

[35] Christos S. Ioakimidis, Dimitrios Thomas, Pawel Rycerski, and Konstantinos N. Genikomsakis. "Peak shaving and valley filling of power consumption profile in non-residential buildings using an electric vehicle parking lot". In: *Energy* 148 (Apr. 2018), pp. 148–158. ISSN: 0360-5442. DOI: 10.1016/j.energy.2018.01.128.

[36] Shaolun Xu, Donghan Feng, Zheng Yan, Liang Zhang, Naihu Li, Lei Jing, and Jianhui Wang. "Ant-Based Swarm Algorithm for Charging Coordination of Electric Vehicles". en. In: *International Journal of Distributed Sensor Networks* 9.5 (May 2013), p. 268942. ISSN: 1550-1329. DOI: 10.1155/2013/268942.

[37] G. Celli, E. Ghiani, F. Pilo, G. Pisano, and G. G. Soma. "Particle Swarm Optimization for minimizing the burden of electric vehicles in active distribution networks". In: *2012 IEEE Power and Energy Society General Meeting*. July 2012, pp. 1–7. DOI: 10.1109/PESGM.2012.6345458. URL: https://ieeexplore.ieee.org/document/6345458.

[38] Dawei Qiu, Yi Wang, Weiqi Hua, and Goran Strbac. "Reinforcement learning for electric vehicle applications in power systems:A critical review". In: *Renewable and Sustainable Energy Reviews* 173 (Mar. 2023), p. 113052. ISSN: 1364-0321. DOI: 10.1016/j.rser.2022.113052.

[39] Julianna Delua. *Supervised vs. Unsupervised Learning: What's the Difference?* en-US. Mar. 2021. URL: https://www.ibm.com/blog/supervised-vs-unsupervised-learning/.

[40] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning, Second Edition : An Introduction*. English. Vol. Second edition. Adaptive Computation and Machine Learning. Cambridge, Massachusetts: Bradford Books, 2018. ISBN: 978-0-262-03924-6.

[41] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. "Human-level control through deep reinforcement learning". en. In: *Nature* 518.75407540 (Feb. 2015), pp. 529–533. ISSN: 1476-4687. DOI: 10.1038/nature14236.

[42] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. "Deterministic Policy Gradient Algorithms". In: *Proceedings of the 31st International Conference on Machine Learning*. Ed. by Eric P. Xing and Tony Jebara. Vol. 32. Proceedings of Machine Learning Research 1. Bejing, China: PMLR, June 2014, pp. 387–395. URL: https://proceedings.mlr.press/v32/silver14.html.

[43] Qiang Xing, Zhong Chen, Ziqi Zhang, Ruisheng Wang, and Tian Zhang. "Modelling driving and charging behaviours of electric vehicles using a data-driven approach combined with behavioural economics theory". In: *Journal of Cleaner Production* 324 (Nov. 2021), p. 129243. ISSN: 0959-6526. DOI: 10.1016/j.jclepro.2021.129243.

[44] Fynn Liegmann, Alen Murtovi, Michael Kelker, and Jens Haubrock. "Analysis of user behaviour for modelling an electric vehicle loading profile generator". In: *PESS 2021; Power and Energy Student Summit*. Nov. 2021, pp. 1–5. URL: https://ieeexplore.ieee.org/document/9735248.

[45] Ruisheng Wang, Qiang Xing, Zhong Chen, Ziqi Zhang, and Bo Liu. "Modeling and Analysis of Electric Vehicle User Behavior Based on Full Data Chain Driven". en. In: *Sustainability* 14.1414 (Jan. 2022), p. 8600. ISSN: 2071-1050. DOI: 10.3390/su14148600.

[46] China Electric Power Research Institute, Dongxia Zhang, Xiaoqing Han, Taiyuan University of Technology, Chunyu Deng, and China Electric Power Research Institute. "Review on the research and practice of deep learning and reinforcement learning in smart grids". en. In: *CSEE Journal of Power and Energy Systems* 4.3 (Sept. 2018), pp. 362–370. ISSN: 20960042. DOI: 10.17775/CSEEJPES.2018.00520.

[47] Zhiqiang Wan, Hepeng Li, Haibo He, and Danil Prokhorov. "A Data-Driven Approach for Real-Time Residential EV Charging Management". In: *2018 IEEE Power & Energy Society General Meeting (PESGM)*. Aug. 2018, pp. 1–5. DOI: 10.1109/PESGM.2018.8585945.

[48] Xu Hao, Yue Chen, Hewu Wang, Han Wang, Yu Meng, and Qing Gu. "A V2G-oriented reinforcement learning framework and empirical study for heterogeneous electric vehicle charging management". In: *Sustainable Cities and Society* 89 (Feb. 2023), p. 104345. ISSN: 2210-6707. DOI: 10.1016/j.scs.2022.104345.

[49] Dawei Qiu, Yujian Ye, Dimitrios Papadaskalopoulos, and Goran Strbac. "A Deep Reinforcement Learning Method for Pricing Electric Vehicles With Discrete Charging Levels". In: *IEEE Transactions on Industry Applications* 56.5 (Sept. 2020), pp. 5901–5912. ISSN: 1939-9367. DOI: 10.1109/TIA.2020.2984614.

[50] Shuoyao Wang, Suzhi Bi, and Ying-Jun Angela Zhang. "A Reinforcement Learning Approach for EV Charging Station Dynamic Pricing and Scheduling Control". In: *2018 IEEE Power & Energy Society General Meeting (PESGM)*. Aug. 2018, pp. 1–5. DOI: 10.1109/PESGM.2018.8586075.

[51] Feiye Zhang, Qingyu Yang, and Dou An. "CDDPG: A Deep-Reinforcement-Learning-Based Approach for Electric Vehicle Charging Control". In: *IEEE Internet of Things Journal* 8.5 (Mar. 2021), pp. 3075–3087. ISSN: 2327-4662. DOI: 10.1109/JIOT.2020.3015204.

[52] Hang Li, Guojie Li, Tek Tjing Lie, Xingzhi Li, Keyou Wang, Bei Han, and Jin Xu. "Constrained large-scale real-time EV scheduling based on recurrent deep reinforcement learning". In: *International Journal of Electrical Power & Energy Systems* 144 (Jan. 2023), p. 108603. ISSN: 0142-0615. DOI: 10.1016/j.ijepes.2022.108603.

[53] Jaehyun Lee, Eunjung Lee, and Jinho Kim. "Electric Vehicle Charging and Discharging Algorithm Based on Reinforcement Learning with Data-Driven Approach in Dynamic Pricing Scheme". en. In: *Energies* 13.88 (Jan. 2020), p. 1950. ISSN: 1996-1073. DOI: 10.3390/en13081950.

[54] Nasrin Sadeghianpourhamami, Johannes Deleu, and Chris Develder. "Definition and Evaluation of Model-Free Coordination of Electrical Vehicle Charging With Reinforcement Learning". In: *IEEE Transactions on Smart Grid* 11.1 (Jan. 2020), pp. 203–214. ISSN: 1949-3061. DOI: 10.1109/TSG.2019.2920320.

[55] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. "OpenAI Gym". In: arXiv:1606.01540 (June 2016). arXiv:1606.01540 [cs]. DOI: 10.48550/arXiv.1606.01540. URL: http://arxiv.org/abs/1606.01540.

[56] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*. 2023. URL: https://www.gurobi.com.

[57] G. E. Uhlenbeck and L. S. Ornstein. "On the Theory of the Brownian Motion". In: *Phys. Rev.* 36 (5 Sept. 1930), pp. 823–841. DOI: 10.1103/PhysRev.36.823. URL: https://link.aps.org/doi/10.1103/PhysRev.36.823.

[58] Jakob Hollenstein, Sayantan Auddy, Matteo Saveriano, Erwan Renaudo, and Justus Piater. "Action Noise in Off-Policy Deep Reinforcement Learning: Impact on Exploration and Performance". In: arXiv:2206.03787 (June 2023). arXiv:2206.03787 [cs]. URL: http://arxiv.org/abs/2206.03787.

[59] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. "Continuous control with deep reinforcement learning". In: arXiv:1509.02971 (July 2019). arXiv:1509.02971 [cs, stat]. URL: http://arxiv.org/abs/1509.02971.

[60] *Open Datasets for Electric Mobility Research | Update April 2020*. 2024. URL: https://platform.elaad.io/analyses/ElaadNL_opendata.php (visited on 04/02/2024).

[61] ENTSO-E. *Central collection and publication of electricity generation, transportation and consumption data and information for the pan-European market*. 2024. URL: https://transparency.entsoe.eu/ (visited on 04/02/2024).

[62] Rijksdienst voor Ondernemend Nederland (RVO). *Statistics Electric Vehicles and Charging in The Netherlands up to and including January 2023*. Accessed: February 2024. 2023. URL: https://www.rvo.nl/sites/default/files/2023-02/Statistics-Electric-Vehicles-and-Charging-in-The-Netherlands-up-to-and-including-jan-2023.pdf.

[63] Rijksdienst voor Ondernemend Nederland (RVO). *Statistics Electric Vehicles and Charging in The Netherlands up to and including May 2023*. Accessed: February 2024. 2023. URL: https://www.rvo.nl/sites/default/files/2023-06/Statistics-Electric-Vehicles-and-Charging-in-The-Netherlands-up-to-and-including-May-2023.pdf.

[64] Rijksdienst voor Ondernemend Nederland (RVO). *Statistics Electric Vehicles and Charging in The Netherlands up to and including August 2023*. Accessed: February 2024. 2023. URL: https://www.rvo.nl/sites/default/files/2023-08/Statistics-Electric-Vehicles-and-Charging-in-The-Netherlands-up-to-and-including-May-2023.pdf.

[65] Rijksdienst voor Ondernemend Nederland (RVO). *Statistics Electric Vehicles and Charging in The Netherlands up to and including September 2023*. Accessed: February 2024. 2023. URL: https://www.rvo.nl/sites/default/files/2023-10/2023-09%20-%20Statistics%20Electric%20Vehicles%20and%20Charging%20in%20The%20Netherlands%20up%20to%20and%20including%20september%202023.pdf.

A

Appendix A: Hyperparameters Tuning

This section explains the hyperparameter tuning process and the rationale behind the selected hyperparameter set. Table A.1 shows several hyperparameter sets utilized in the case study for 10 chargers. The table details each hyperparameter set used for each training and testing of the DDPG algorithm, with the first trained RL model serving as the starting point. The table starts with the 67th test since the problem was initially resolved utilizing a simple EV model before implementing different EV models registered in the Netherlands. Additionally, the power setpoint flexibility parameter was maintained at a constant value for the tests displayed in the table. However, power setpoint flexibility varied for other tests not included in this table.

Table A.1: Hyperparameter set alternatives

Test name	P. Setpoint Flexibility (%)	Minibatch M	Replay Buffer $\mathcal{R}$	Discount Factor γ	Soft Update τ	Noise $\mathcal{N}$	Actor N Network	Critic N Network	E. Tracking Error (kWh)	E. Tracking Error Std	Mean Reward Convergence
DDPG67	5	64	1000000	0.99	0.001	0.1	128	64	113.93	19.20	-1250
DDPG91	5	64	1000000	0.99	0.001	0.1	128 - 64	128 - 64	125.33	21.11	-1000
DDPG92	5	64	1000000	0.99	0.001	0.1	64 - 32	64 - 32	126.91	20.89	-1100
DDPG93	5	64	1000000	0.99	0.001	0.2	128	64	110.79	22.24	-1100
DDPG94	5	64	1000000	0.99	0.001	0.05	128	64	129.02	21.62	-1000
DDPG95	5	64	1000000	0.99	0.002	0.1	128	64	127.27	22.33	-1200
DDPG96	5	64	1000000	0.99	0.0005	0.1	128	64	116.94	20.03	-950
DDPG98	5	64	50000	0.99	0.001	0.1	128	64	117.68	19.44	-1200
DDPG99	5	64	100000	0.99	0.001	0.1	128	64	130.64	20.31	-900
DDPG100	5	64	1000000	0.99	0.001	0.3	128	64	126.92	21.62	-1000
DDPG101	5	64	1000000	0.99	0.001	0.4	128	64	139.94	21.36	-1000
DDPG102	5	64	1000000	0.99	0.0005	0.2	128	64	97.62	18.65	-1000
DDPG103	5	128	1000000	0.99	0.001	0.2	256	128	120.12	19.73	-1000
DDPG104	5	128	1000000	0.99	0.001	0.2	256 - 128	256 - 128	129.72	22.53	-800
DDPG107	5	64	1000000	0.85	0.0005	0.2	128	64	107.40	18.27	-1200
DDPG108	5	64	1000000	0.995	0.0005	0.2	128	64	131.04	19.62	-850

After the RL agent is trained for 25,000 episodes, its mean reward's convergence is checked and following that its performance is evaluated in the testing phase by utilizing the agent for 100 randomly generated replays. The energy tracking error metric was the main comparison metric for tuning the hyperparameters after the convergence of the mean reward because it directly shows the PST error in the kWh unit. As it can be seen from Table A.1, although the mean rewards of each model converge, the outcomes obtained in the testing phase are very different in terms of PST error minimization. This result highlights the importance of the testing phase of the algorithm.

The first hyperparameter set in Table A.1, DDPG67 was considered to be the first promising model. The reward function test was also done by using the hyperparameter set of DDPG67. However, consecutively, better results were obtained in terms of PST error minimization in the testing phase.

During the training of the RL agent, several hyperparameters are tuned to achieve optimal results. One such hyperparameter is the minibatch size (M), which was found to be effective when set to half the size of the actor network. This helps the agent to identify patterns and discover optimal policies.

The replay buffer ($\mathcal{R}$) size is another important hyperparameter that was experimented with. Smaller sizes than 10^6 were tested, but the results (DDPG98-DDPG99) did not show any significant improvement despite both of the model's convergence.

The discount factor (γ) was also varied to see its impact on the agent's performance. Increasing the discount factor did not yield better results while decreasing it showed some promise but still fell short of the best results achieved. This suggests that the earlier actions taken by the RL agent resulted in better policies.

After several experiments, it was found that it is crucial to find the right balance between the action noise ($\mathcal{N}$) that increases exploration and the soft update (τ) that varies the updating speed of networks. In fact, the selected hyperparameter set, painted in green in Table A.1, is found by balancing the noise and soft update by updating DDPG67.

It appears that the set of DNN architectures did not yield satisfactory outcomes, except for the selected architecture. This particular architecture involved using 128 neurons for both the main and target networks of the actor and 64 neurons for both the main and target networks of the critic. Considering that both the larger and smaller DNN architectures did not yield promising results, this suggests that the selected DNN sizes are appropriate for the size and complexities of the formulated PST problem for 10 chargers.