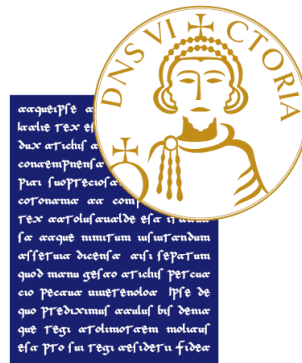


UNIVERSITY OF SANNIO

DEPARTMENT OF ENGINEERING

MASTER'S DEGREE IN

Electronics Engineering for Automation and Sensing



# Deep Reinforcement Learning for Adaptive Bidirectional Electric Vehicle Charging Management (Vehicle-to-Grid)

**Supervisor:**

Prof. Carmela Bernardo

**Co-Supervisor:**

Dr. Antonio Pepiciello

**Candidate:**

Angelo Caravella

Student ID 389000016

ACADEMIC YEAR 2024–2025

# Contents

<b>List of Acronyms</b>	<b>4</b>
<b>1 Introduction</b>	<b>7</b>
1.0.1 Background and Relevance of Electric Vehicles and Vehicle-to-Grid . . . . .	7
1.0.2 Challenges in EV Integration into the Electricity Grid and the Role of Artificial Intelligence . . . . .	8
1.0.3 Objectives and Contributions of the Thesis . . . . .	9
1.0.4 Research Methodology . . . . .	9
1.0.5 Thesis Structure . . . . .	11
<b>2 State of the Art in Optimal V2G Management</b>	<b>12</b>
2.1 The V2G Imperative: A Cornerstone of Europe’s Green Transition . .	12
2.2 The Optimizer’s Trilemma: Navigating a Stochastic World . . . . .	14
2.3 The Economics of V2G: Navigating Energy Markets . . . . .	14
2.3.1 Sources for Energy Price Data . . . . .	14
2.3.2 Buying vs. Selling: The Retail-Wholesale Spread . . . . .	15
2.4 The Electric Vehicle and the V2G Scenario . . . . .	16
2.4.1 Key Characteristics of a Grid-Interactive EV . . . . .	16
2.4.2 Deconstructing the V2G Scenario: A Conceptual Overview . .	16
2.5 A New Paradigm for Control: Reinforcement Learning . . . . .	17
2.5.1 The Language of Learning: Markov Decision Processes (MDPs)	18
2.5.2 Judging the Future: Value Functions and Actor-Critic Architectures . . . . .	18
2.6 Reward Engineering: Shaping Agent Behavior . . . . .	18
2.6.1 Potential-Based Reward Shaping (PBRS) . . . . .	19
2.6.2 Dynamic and Adaptive Rewards . . . . .	19
2.6.3 Curriculum Learning . . . . .	19
2.7 The Rise of Deep Reinforcement Learning for V2G Control . . . . .	20
2.7.1 Off-Policy Methods: Data-Efficient Learning from Experience	20
2.7.2 On-Policy Methods: Stability through Cautious Updates . . .	21
2.7.3 Gradient-Free Methods: An Alternative Path . . . . .	22
2.8 The Model-Based Benchmark: Model Predictive Control (MPC) . . .	22
2.8.1 Implicit MPC: Online Optimization . . . . .	23
2.8.2 Explicit MPC: Offline Pre-computation . . . . .	23
2.9 A Comparative Perspective on Control Methodologies . . . . .	24

2.10	A Primer on Lithium-Ion Battery Chemistries and Degradation . . .	25
2.10.1	Fundamental Concepts and Degradation Mechanisms . . . . .	26
2.10.2	Key Automotive Chemistries . . . . .	27
2.10.3	Voltage Profiles and the Challenge of SoC Estimation . . . . .	27
2.10.4	Comparative Analysis and Safety Considerations . . . . .	28
2.10.5	Battery Pack Architecture . . . . .	28
<b>3</b>	<b>An Enhanced V2G Simulation Framework for Robust Control</b>	<b>30</b>
3.1	Core Simulator Architecture . . . . .	30
3.2	Core Physical Models . . . . .	31
3.2.1	EV Model and Charging/Discharging Dynamics . . . . .	31
3.2.2	Battery Degradation Model . . . . .	31
3.2.3	EV Behavior and Grid Models . . . . .	32
3.3	A Unified Experimentation and Evaluation Workflow . . . . .	32
3.3.1	Orchestration via <code>run_experiments.py</code> . . . . .	32
3.3.2	Dual-Mode Training: Specialists and Generalists . . . . .	33
3.3.3	Reproducible Benchmarking and Evaluation . . . . .	33
3.4	Evaluation Metrics . . . . .	33
3.5	Simulator Implementation Details . . . . .	34
3.6	Reinforcement Learning Formulation . . . . .	35
3.6.1	State Space ( $S$ ) . . . . .	36
3.6.2	Action Space ( $A$ ) . . . . .	36
3.6.3	Reward Function . . . . .	36
3.7	Reinforcement Learning Algorithms . . . . .	37
3.7.1	A History-Based Adaptive Reward for Profit Maximization . .	39
3.8	Model Predictive Control (MPC) . . . . .	41
3.8.1	System Model . . . . .	41
3.8.2	Optimization Problem . . . . .	41
3.9	Offline Optimization with Gurobi . . . . .	41
3.9.1	Decision Variables . . . . .	41
3.9.2	Objective Function (Example: Profit Maximization) . . . . .	42
3.9.3	Main Constraints . . . . .	42
3.10	Online MPC Formulation (PuLP Implementation) . . . . .	42
3.10.1	Mathematical Formulation . . . . .	42
3.11	Conceptual Comparison: PuLP MPC vs. Gurobi Offline Optimizer .	44
3.11.1	Core Philosophy: Controller vs. Judge . . . . .	44
3.11.2	Objective Function: Operational Profit vs. Energy Arbitrage .	44
3.11.3	Handling of User Satisfaction: Hard vs. Soft Constraints . . .	45

## Abstract in italian

L'adozione crescente dei **Veicoli Elettrici (EV)** in concomitanza con la sempre maggiore penetrazione di **Fonti di Energia Rinnovabile (RES)** intermittenti, presenta sfide significative alla **stabilità** e all'**efficienza della rete elettrica**. La tecnologia **Vehicle-to-Grid (V2G)** emerge come soluzione fondamentale, trasformando gli EV da carichi passivi a **risorse energetiche flessibili** capaci di fornire vari **servizi di rete**. Questa tesi affronta il complesso **problema di ottimizzazione multi-obiettivo** della gestione intelligente di carica e scarica degli EV, che intrinsecamente implica un equilibrio tra **benefici economici**, **esigenze di mobilità dell'utente**, **preservazione della salute della batteria** e **stabilità della rete** in condizioni stocastiche.

Di fronte alla complessa sfida di ottimizzare la ricarica dei veicoli elettrici (EV) in scenari Vehicle-to-Grid (V2G), un approccio che si limita a un singolo modello di controllo, come il Deep Q-Networks (DQN), risulterebbe inadeguato. La natura del problema, caratterizzata da molteplici obiettivi contrastanti (benefici economici, esigenze dell'utente, salute della batteria, stabilità della rete) e da una profonda incertezza; richiede un'analisi comparativa e rigorosa di un'ampia gamma di strategie di controllo. Per questo motivo, la ricerca si concentra sulla valutazione di un portafoglio diversificato di algoritmi, che include numerosi modelli di Deep Reinforcement Learning (DRL), approcci euristici e il Model Predictive Control (MPC). Questo metodo consente di mappare in modo completo il panorama delle soluzioni, identificando i punti di forza e di debolezza di ciascun approccio in relazione alle diverse sfaccettature del problema V2G.

In conclusione questa lavoro di tesi non si focalizza su un singolo modello, ma adotta un approccio comparativo su larga scala perché:

**Non esiste una soluzione unica:** La complessità del problema V2G rende improbabile che un solo algoritmo sia ottimale in tutte le condizioni.

**Si ricercano i compromessi:** L'obiettivo è comprendere i trade-off tra l'efficienza dei dati, la stabilità dell'addestramento, la robustezza all'incertezza e la complessità computazionale delle diverse famiglie di algoritmi.

**La validazione è più rigorosa:** Confrontare i modelli di DRL non solo tra loro ma anche con benchmark consolidati come le euristiche e l'MPC fornisce una misura molto più credibile del loro reale valore aggiunto.

# Abstract

The growing adoption of **Electric Vehicles (EVs)**, combined with the increasing penetration of intermittent **Renewable Energy Sources (RES)**, presents significant challenges to the **stability** and **efficiency** of the power grid<sup>1</sup>. **Vehicle-to-Grid (V2G)** technology emerges as a key solution, transforming EVs from passive loads into **flexible energy resources** capable of providing various **grid services**. This thesis addresses the complex **multi-objective optimization problem** of smart EV charging and discharging, which requires balancing **economic benefits**, **user mobility needs**, **battery health preservation**, and **grid stability** under stochastic conditions.

Given the complexity of optimizing EV charging in V2G scenarios, relying on a single control model is insufficient. The nature of the problem, characterized by multiple conflicting objectives (economic benefits, user needs, battery health, grid stability) and profound uncertainty, demands a rigorous comparative analysis of a wide range of control strategies.

For this reason, the research focuses on evaluating a diverse portfolio of algorithms, including numerous Deep Reinforcement Learning (DRL) models, heuristic approaches, and Model Predictive Control (MPC). This method allows for a complete mapping of the solution landscape, identifying the strengths and weaknesses of each approach in relation to the different facets of the V2G problem.

In short, this thesis adopts a **broad-spectrum comparative approach** for several reasons:

**No Single Solution:** The complexity of the V2G problem makes it unlikely that a single algorithm can be optimal in all conditions.

**Understanding Trade-offs:** The goal is to understand the trade-offs between data efficiency, training stability, robustness to uncertainty, and the computational complexity of different algorithm families.

**Rigorous Validation:** Comparing DRL models not only against each other but also against established benchmarks like heuristics and MPC provides a more credible measure of their true value.

## List of Acronyms

Acronym	Description
---------	-------------

<b>Artificial Intelligence &amp; Control</b>	
--	--

A2C	Advantage Actor-Critic
-----	------------------------

AC	Actor-Critic
----	--------------

AI	Artificial Intelligence
----	-------------------------

AL-SAC	Augmented Lagrangian Soft Actor-Critic
--------	--

ARS	Augmented Random Search
-----	-------------------------

CL	Curriculum Learning
----	---------------------

---

<sup>1</sup>32.

<b>Acronym</b>	<b>Description</b>
CMDP	Constrained Markov Decision Process
DDPG	Deep Deterministic Policy Gradient
DQN	Deep Q-Networks
DRL	Deep Reinforcement Learning
LQR	Linear Quadratic Regulator
LSTM	Long Short-Term Memory
MARL	Multi-Agent Reinforcement Learning
MDP	Markov Decision Process
MILP	Mixed-Integer Linear Program
MPC	Model Predictive Control
NN	Neural Network
PER	Prioritized Experience Replay
PPO	Proximal Policy Optimization
RL	Reinforcement Learning
SAC	Soft Actor-Critic
TD3	Twin-Delayed Deep Deterministic Policy Gradient
TQC	Truncated Quantile Critics
TRPO	Trust Region Policy Optimization
<b>Electric Vehicles &amp; Charging</b>	
AFAP	As Fast As Possible (Heuristic)
ALAP	As Late As Possible (Heuristic)
CAFA	Charge As Fast As Possible
CALA	Charge As Late As Possible
CPO	Charge Point Operator
EV	Electric Vehicle
G2V	Grid-to-Vehicle
SCP	Scheduled Charging Power
SoC	State of Charge
SoH	State of Health
V2B	Vehicle-to-Building
V2G	Vehicle-to-Grid
V2H	Vehicle-to-Home
V2M	Vehicle-to-Microgrid
V2V	Vehicle-to-Vehicle
VPP	Virtual Power Plant
<b>Power Grid &amp; Energy Markets</b>	
ACE	Area Control Error
ARR	Area Regulation Requirement
DER	Distributed Energy Resources
DR	Demand Response
RES	Renewable Energy Sources
<b>Metrics &amp; Technical Parameters</b>	
DC	Constant Current (charging phase)
CV	Constant Voltage (charging phase)
DoD	Depth of Discharge

<b>Acronym</b>	<b>Description</b>
MSE	Mean Square Error
OU	Ornstein-Uhlenbeck (stochastic process)
RMSE	Root Mean Square Error

# Chapter 1

## Introduction

The shift toward electric mobility is a pivotal element in global strategies for decarbonizing transport. However, the large-scale integration of electric vehicles into existing power grids presents a complex set of challenges and opportunities, which this thesis aims to investigate.

### 1.0.1 Background and Relevance of Electric Vehicles and Vehicle-to-Grid

The surging Electric Vehicle (EV) market is accelerating a major reconfiguration of modern mobility, promising to lower carbon emissions and foster greater energy efficiency<sup>1</sup>. This evolution is more than a technological trend: it underpins environmental sustainability by reducing dependence on fossil resources, alleviating the impacts of climate change through diminished greenhouse gas emissions, and improving air quality in densely populated areas. However, embedding millions of EVs into existing power systems is a non-trivial task. It can intensify peak demand, place additional stress on transmission and distribution networks, and trigger side effects such as voltage irregularities or higher line losses<sup>2</sup>.

It is in this context that the **Vehicle-to-Grid (V2G)** concept emerges as a strategic solution. Through bidirectional power exchange, V2G redefines EVs: no longer passive electrical loads, but mobile and flexible energy assets, able to deliver a spectrum of services to the power system<sup>3</sup>. This potential becomes even more compelling when one considers that, on average, EVs remain parked and unused for nearly 96% of the day, offering an ample time window to actively engage with the grid<sup>4</sup>. A further distinctive benefit lies in the rapid responsiveness of EV batteries, which makes them especially suitable for ancillary services demanding quick interventions, such as frequency regulation<sup>5</sup>. Alongside V2G, other schemes of bidirectional power flow have been proposed, each with its own scope:

1. **Vehicle-to-Home (V2H)**, where an EV sustains household demand during

---

<sup>1</sup>23.

<sup>2</sup>23, 27.

<sup>3</sup>2.

<sup>4</sup>7.

<sup>5</sup>2.



outages or periods of elevated prices, strengthening domestic energy resilience;

2. **Vehicle-to-Building (V2B)**, extending this logic to commercial or industrial facilities, enabling EVs to support load management and improve consumption efficiency; and
3. **Vehicle-to-Vehicle (V2V)**, which allows direct power transfer among EVs, a valuable feature for emergency charging or shared resources.

Taken together, these modalities highlight the versatility of EV batteries as distributed energy units, reinforcing both energy resilience and the transition toward a more sustainable energy ecosystem.

### 1.0.2 Challenges in EV Integration into the Electricity Grid and the Role of Artificial Intelligence

Modern electricity systems are increasingly shaped by the penetration of intermittent **Renewable Energy Sources (RESs)** such as wind and solar. Their variability causes pronounced swings in power generation and persistent mismatches between supply and demand, which in turn fuels price volatility and complicates dispatch strategies. This continuously puts the stability and economic efficiency of the grid under strain. Managing these fluctuations, while making rapid operational choices to balance the system and minimize costs, has proven difficult for conventional control frameworks<sup>6</sup>.

The parallel rise of EV adoption and RES deployment has produced an environment marked by both uncertainty and complexity. In such conditions, traditional approaches are increasingly inadequate, prompting a growing reliance on methods rooted in artificial intelligence—and particularly in **Reinforcement Learning (RL)**. This shift alters the very nature of the grid: from a relatively predictable and centralized infrastructure to one that is decentralized, stochastic, and highly dynamic. Rule-based or deterministic controllers, designed for a past paradigm, are ill-suited to cope with this degree of volatility. The result is a pressing need for more adaptive and intelligent decision-making mechanisms. This transformation extends beyond the simple challenge of absorbing extra load or integrating new generators: it signals a genuine paradigm change towards a *smart grid*<sup>7</sup>, where adaptive, real-time, and autonomous operation is no longer optional but vital to preserve efficiency, resilience, and reliability. From this perspective, RL is not just an optimization tool, but an enabling technology for a more cognitive and robust energy infrastructure—one capable of navigating the uncertainties of a decarbonized, electrified future.

Against this backdrop, **Deep Reinforcement Learning (DRL)** has gained attention as an especially powerful approach. Its capacity to derive near-optimal strategies in dynamic and uncertain environments—without requiring a precise model of the system or flawless forecasts—makes DRL particularly well-suited for EV integration and advanced grid management<sup>8</sup>.

---

<sup>6</sup>23, 20.

<sup>7</sup>10.

<sup>8</sup>23, 31.

### 1.0.3 Objectives and Contributions of the Thesis

This thesis confronts the complex multi-objective optimization problem at the heart of Vehicle-to-Grid (V2G) systems. The overarching objective is to move beyond a purely theoretical analysis by actively developing, testing, and enhancing a high-fidelity simulation architecture. This platform serves as a digital twin to rigorously evaluate and compare advanced control strategies, balancing economic benefits, user mobility needs, battery health, and grid stability under realistic stochastic conditions.

More than a simple review of existing literature, this work focuses on the practical implementation and validation of a V2G simulation framework in Python. This tool is leveraged to demonstrate and explore novel perspectives for training intelligent agents. The main contributions are:

- **Enhancement of a V2G Simulation Architecture:** A key contribution is the systematic testing, validation, and enhancement of the **EV2Gym** simulation framework. This work solidifies its role as a robust and flexible platform for benchmarking control algorithms, ensuring that the models for battery physics, user behavior, and grid dynamics are coherent and realistic for advanced research.
- **Exploration of Novel Reinforcement Learning Perspectives:** The validated simulation environment is used to investigate and implement advanced training methodologies for RL agents. A key focus is placed on techniques like **adaptive reward shaping**, where the reward function dynamically evolves during training to guide the agent towards a more holistic and robust control policy, overcoming the limitations of static reward definitions.
- **Practical Implementation of Advanced Control Paradigms:** The thesis demonstrates the practical transition from a theoretical, offline optimal controller to a realistic, online controller. Specifically, it details the implementation of an **offline MPC using Gurobi**, which acts as a "judge" with perfect foresight, and contrasts it with an **online MPC formulated in PuLP**, designed to operate as a real-time "controller" with limited future information, highlighting the trade-offs and challenges of real-world deployment.

### 1.0.4 Research Methodology

The research conducted for this thesis was guided by a central question: how can the economic profit of Vehicle-to-Grid (V2G) operations be maximized without imposing undue costs in terms of battery degradation or creating instability by overloading local grid infrastructure? To address this, a systematic research methodology was adopted.

The initial phase consisted of an extensive literature review, primarily leveraging Google Scholar as a search engine for academic papers. This review focused on identifying the state-of-the-art in V2G control strategies, with a particular emphasis on identifying the common trade-offs between profitability and physical constraints.

Following the literature review, this work transitioned to an empirical phase centered on the simulation framework detailed in Chapter 3. The development and testing of various control agents were not performed in a vacuum. The significance of the obtained results was continuously evaluated through a multi-faceted validation process, which included:

- **Comparative Analysis:** Benchmarking the performance of novel agents against the results reported by other researchers in the field.
- **Heuristic Evaluation:** Assessing the outcomes based on the experience and intuition developed over the course of the study, ensuring that the results were not only quantitatively strong but also qualitatively sound.
- **Academic Grounding:** Constantly relating the empirical results back to the foundational knowledge acquired during the literature review to understand the underlying reasons for an agent’s success or failure.

This iterative process of exploration, implementation, and critical evaluation forms the methodological backbone of the contributions presented in this thesis.

### 1.0.5 Thesis Structure

The remainder of this thesis is organized as follows:

- **Chapter 2: Overview of Optimal Management of EV Charging and Discharging** provides foundational knowledge on V2G technology, the complex multi-objective nature of EV charging optimization, and presents a comprehensive review of state-of-the-art research approaches.
- **Chapter 3: The V2G Simulation Framework: A Digital Twin for V2G Research** details the architecture and core models of the simulation environment. This chapter describes the enhancements made to the framework, establishing it as the central experimental platform for implementing and evaluating the control agents analyzed in this work.
- **Chapter 4: Experimental Campaign and Results Analysis** This chapter presents the results of the comparative analysis between the different control strategies (DRL, MPC, heuristics). It analyzes the performance of novel training techniques and discusses the implications of the findings.
- **Bibliography** lists all cited references.

## Chapter 2

# State of the Art in Optimal V2G Management

### 2.1 The V2G Imperative: A Cornerstone of Europe's Green Transition

Society finds itself at a critical juncture, grappling with the twin revolutions of decarbonizing transport and reshaping our energy systems. This is not merely an ambition but a legally binding mandate, enshrined in frameworks like the **European Green Deal** and its ambitious "**Fit for 55**" package<sup>1</sup>. These policies impose a rapid phase-out of internal combustion engines and demand a massive scale-up of renewable energy sources, as detailed in the revised Renewable Energy Directive (RED III). At the very nexus of this challenge lies the proliferation of Electric Vehicles (EVs).

Initially viewed with apprehension—a looming threat of massive, synchronized loads poised to destabilize fragile distribution networks—that perception is now obsolete. Today, EVs must be seen not as a problem, but as a foundational pillar of the solution. This change in perspective is embodied in the concept of **Vehicle-to-Grid (V2G)**. V2G acts as the critical enabling technology that can transform millions of EVs from passive consumers into active, distributed, and intelligent assets for the grid. The key lies hidden in plain sight: private vehicles remain parked and connected for an astonishing 96% of their existence<sup>2</sup>, representing a potential of terawatt-hours of mobile storage waiting to be harnessed.

The true power of V2G emerges not from the individual, but from the collective. A single EV's contribution is a whisper, but a coordinated fleet, managed by an aggregator, becomes a roar—a **Virtual Power Plant (VPP)**. This collective entity, with the lightning-fast response of battery inverters, can deliver a spectrum of critical services. This capability is the linchpin for stabilizing a grid increasingly reliant on the fluctuating whims of wind and sun, making the high renewable penetration targets of the EU feasible. The services unlocked by this capability are foundational to building the smart, resilient grid of tomorrow:

---

<sup>1</sup>1.

<sup>2</sup>7.

- **Frequency Regulation:** The grid's heartbeat. V2G fleets can inject or absorb power in seconds, instantly counteracting supply-demand imbalances to maintain the stable 50/60 Hz frequency, preventing cascading failures and blackouts<sup>3</sup>.
- **Demand Response and Peak Shaving:** By intelligently shifting charging to off-peak hours and discharging during peak demand, V2G flattens the load curve. This reduces our reliance on expensive and polluting "peaker" plants and can defer trillions in grid infrastructure upgrades<sup>4</sup>.
- **Renewable Energy Integration:** Perhaps its most profound impact, V2G fleets can act as a giant, distributed sponge, absorbing surplus solar and wind energy that would otherwise be curtailed, and releasing it when the sun sets or the wind dies down. This directly supports the integration goals of RED III and mitigates intermittency<sup>5</sup>.

This vision is no longer a distant prospect; it is actively being codified into European law and technical standards. The landmark **Alternative Fuels Infrastructure Regulation (AFIR, EU 2023/1804)** now mandates that new public charging infrastructure must support smart and bidirectional charging. This legal requirement is given its technical teeth by specific standards, as a delegated regulation specifies that from 2027, charging points must comply with **ISO 15118-20**—a standard that explicitly defines the communication protocols for bidirectional power transfer. This regulatory push is complemented by large-scale pilot projects like '**SCALE**' and '**V2G Balearic Islands**', which are testing the technology's technical and economic viability on an industrial scale.

However, while the regulatory foundation is being laid, significant barriers to widespread adoption remain, creating a complex landscape that technology and policy must navigate together. Key challenges include:

- **Market and Economic Hurdles:** A clear, pan-European framework for remunerating EV owners for grid services is still absent. Critical issues like the "**double taxation**" of electricity—taxed both on charging and discharging—create significant economic disincentives that must be resolved.
- **Regulatory and Grid Access Rules:** The role of EV fleets as a flexibility resource is not yet uniformly recognized in electricity markets. Standardized procedures for grid connection, aggregator certification, and secure data exchange are still under development, hindering market access.
- **Technical and Consumer Barriers:** On the consumer side, concerns about accelerated **battery degradation** and its impact on vehicle warranties remain a primary obstacle. Furthermore, the reality is that not all EVs or chargers are currently equipped with the necessary hardware and software to support V2G.

---

<sup>3</sup>2, 26.

<sup>4</sup>23.

<sup>5</sup>12, 35.

The central challenge, which is the focus of this thesis, is therefore not simply about enabling V2G, but about orchestrating it *intelligently*. This requires a control strategy sophisticated enough to operate within this nascent regulatory framework, navigate its economic uncertainties, and overcome technical constraints to unlock the immense potential of EVs as a cornerstone of a sustainable energy future.

## 2.2 The Optimizer’s Trilemma: Navigating a Stochastic World

While the potential is immense, orchestrating this symphony of distributed assets presents a formidable challenge. An aggregator’s primary driver is economic viability, but pursuing profit in isolation is a recipe for failure. Optimal V2G management is a delicate balancing act, a genuine multi-objective optimization problem often framed as the "V2G trilemma": the simultaneous pursuit of **economic profitability**, the preservation of **battery longevity**, and the guarantee of **user convenience**.

This is not a simple trade-off but a dynamic problem steeped in **stochasticity** and **uncertainty** from multiple sources:

- **Market Volatility:** Electricity prices can fluctuate wildly based on unpredictable supply and demand.
- **Renewable Intermittency:** The output of solar and wind generation is inherently variable.
- **Human Behavior:** EV owners’ arrival times, departure times, and energy needs are not deterministic; a driver might need to leave unexpectedly, a non-negotiable constraint that any intelligent system must respect.

Such a chaotic environment renders static, rule-based control systems obsolete. We need an approach that can learn, adapt, and make intelligent decisions in real-time under profound uncertainty. This is precisely the domain where Reinforcement Learning excels.

## 2.3 The Economics of V2G: Navigating Energy Markets

The economic viability of many Vehicle-to-Grid services, particularly energy arbitrage, hinges on the price differential between buying and selling electricity. Understanding the structure of energy markets and the composition of electricity prices is therefore essential for any realistic V2G analysis.

### 2.3.1 Sources for Energy Price Data

The most relevant prices for V2G optimization are those from the wholesale electricity markets, where energy is traded in bulk before reaching the end consumer. In Europe, the primary markets are:

- **Day-Ahead Market:** This is an auction-based market where prices and volumes are determined for every hour of the following day. This is the most common signal used for planning charging and discharging schedules.
- **Intraday Market:** This market operates closer to real-time, allowing participants to adjust their positions based on updated forecasts (e.g., for renewable generation).

Access to this data is crucial for both simulation and real-world deployment. Key public sources for European market data include:

- **ENTSO-E Transparency Platform:** A mandatory, open-access platform that provides a vast repository of pan-European electricity market data, including harmonized day-ahead prices for numerous countries. It is a primary source for academic research and is accessible via a web portal and a free RESTful API.
- **National Transmission System Operators (TSOs):** Many national TSOs (e.g., Terna in Italy, RTE in France) publish detailed market data for their respective regions.
- **Power Exchanges:** Exchanges like **EPEX SPOT** are where the actual trading occurs. While they are the direct source of price data, comprehensive and real-time access is typically part of a commercial subscription service.

### 2.3.2 Buying vs. Selling: The Retail-Wholesale Spread

A critical distinction for V2G economics is the difference between the price an EV owner pays to charge their vehicle versus the price an aggregator receives for selling that energy back to the grid.

- **Selling Price (V2G Revenue):** When an EV provides energy to the grid, the revenue is based on the **wholesale price** (e.g., the day-ahead spot price). This price reflects the pure cost of energy at a specific time.
- **Buying Price (Charging Cost):** The price paid by an end consumer to charge their EV is the **retail price**. This price is significantly higher than the wholesale price because it includes numerous additional components:
  - The wholesale energy cost itself.
  - **Grid Fees:** Charges for using the transmission and distribution networks.
  - **Taxes and Levies:** National or regional taxes applied to electricity consumption.
  - **Supplier Margin:** The profit margin for the retail energy provider.



This significant gap between the retail buying price and the wholesale selling price is often referred to as the "retail-wholesale spread." It is this spread that creates the primary opportunity for profitable energy arbitrage. A V2G agent aims to "buy low" (charge the EV when the retail price is lowest, often corresponding to low wholesale prices overnight) and "sell high" (discharge the EV to the grid when the wholesale price is highest, typically during evening peak demand), capturing a portion of this spread as profit. Any successful control strategy must be acutely aware of this quantitative and qualitative difference to make economically rational decisions.

## 2.4 The Electric Vehicle and the V2G Scenario

Before delving into the control algorithms, it is useful to establish a clear understanding of the two core components of the problem: the electric vehicle itself, viewed as a controllable asset, and the environment or "scenario" in which it operates. The interaction between these two components defines the boundaries and objectives of any V2G optimization task.

### 2.4.1 Key Characteristics of a Grid-Interactive EV

From the perspective of the power grid, an electric vehicle is more than just a mode of transport; it is a sophisticated, mobile energy storage device. For the purposes of V2G, an EV can be abstracted by a few key characteristics:

- **The Battery:** This is the heart of the EV as a grid asset. It is defined by its **energy capacity** (measured in kWh), which determines how much energy it can store, and its **power limits** (in kW), which dictate the maximum rate at which it can charge or discharge.
- **The On-Board Charger (OBC):** For AC charging (the most common type), the OBC converts alternating current from the grid to direct current to charge the battery. Its power rating is often the bottleneck for charging and V2G power, even if the battery itself could handle more.
- **Communication Interface:** To participate in smart charging or V2G, the EV must be able to communicate with the charging station. This is governed by standards like **ISO 15118**, which enables the exchange of information required for bidirectional power flow and advanced control.

When these characteristics are combined with the vehicle's availability—its arrival and departure times from a charging station—the EV transforms from a simple load into a dispatchable grid resource, capable of both consuming and providing power within a specific set of constraints.

### 2.4.2 Deconstructing the V2G Scenario: A Conceptual Overview

The environment in which an EV operates is just as important as the vehicle's own characteristics. In this research, these environments are defined by a set of con-

figuration files (e.g., ‘BusinessPST.yaml’, ‘V2GProfitMax.yaml’), each representing a distinct, plausible real-world scenario. While the technical details of these files are explored in Chapter 3, their conceptual purpose is to define the "rules of the game" for the control agent. The parameters within can be understood in three main categories:

- **Physical Infrastructure:** This group of parameters defines the physical world. It includes the number and type of charging stations, their maximum power, and the capacity of the local power transformer. For example, a scenario might represent a small office with 10 AC chargers or a large public fast-charging hub with a high-capacity grid connection.
- **Market and Environmental Context:** This defines the external conditions the agent must react to. It includes the time-series data for day-ahead electricity prices, which sets the economic incentive for V2G. It can also include data for other loads connected to the same transformer (an "inflexible base load") and local renewable generation, such as a solar panel array on the building’s roof. These elements create a dynamic environment where the net power flow must be managed.
- **User Demand and Behavior:** This is perhaps the most crucial and stochastic component. These parameters do not define a fixed schedule, but rather the statistical nature of the EV fleet. For instance, a configuration file might specify that EV arrival times follow a normal distribution centered around 9:00 AM and departures around 5:00 PM, with a certain average and standard deviation for the amount of energy each driver needs. This simulates a typical office parking scenario (‘BusinessPST.yaml’). In contrast, a ‘PublicPST.yaml’ scenario might use different distributions to represent a public charging station with shorter, more random parking durations throughout the day.

By creating different combinations of these elements, the configuration files allow for the construction of a rich and diverse set of testbeds. Scenarios like ‘V2GProfitMax.yaml’ are likely configured to emphasize price volatility to test profit-seeking behavior, while others might be configured with tight transformer limits to test the agent’s ability to avoid overloads. Understanding these conceptual roles is key to interpreting the agent’s performance across different environments.

## 2.5 A New Paradigm for Control: Reinforcement Learning

To tackle the V2G challenge, this work turns to Reinforcement Learning (RL), a domain of machine learning focused on how an intelligent agent can learn to make optimal decisions through trial and error. Unlike traditional methods that require a perfect model of the world, RL learns directly from interaction, making it exceptionally robust.

### 2.5.1 The Language of Learning: Markov Decision Processes (MDPs)

The mathematical bedrock of RL is the **Markov Decision Process (MDP)**, formally defined by the tuple  $(S, A, p, R, \gamma)$ . In the V2G context, these elements represent:

- $S$ : The state (a snapshot of the world: battery levels, electricity price, time).
- $A$ : The action (the decision: the charging/discharging rate for each EV).
- $p(s', r|s, a)$ : The environment's response (the probability of transitioning to a new state  $s'$  and receiving reward  $r$ ).
- $R$ : The reward (the feedback signal: profit generated, penalty for user dissatisfaction).
- $\gamma$ : The discount factor, which balances immediate versus future rewards.

This framework rests on the **Markov Property**, which posits that the future is independent of the past given the present, allowing the agent to make decisions based solely on the current state.

### 2.5.2 Judging the Future: Value Functions and Actor-Critic Architectures

The agent's goal is to learn a **policy**,  $\pi(a|s)$ , which is a strategy for choosing actions. To achieve this, it learns **value functions** that estimate the long-term value of being in a certain state ( $v_\pi(s)$ ) or taking a specific action in a state ( $q_\pi(s, a)$ ). The **Actor-Critic** architecture offers an elegant way to learn this policy. It maintains two distinct neural networks:

- **The Critic**: It learns the value function. Its job is to judge the actor's decisions.
- **The Actor**: It represents the policy itself. Its job is to select actions, using the critic's feedback to refine its strategy over time.

This architecture is particularly potent for V2G because it can directly learn a policy over a continuous action space, allowing for fine-grained control of power. The agent's entire behavior, however, is shaped by the reward signal it receives. The complex art of designing this signal to align the agent's goals with our multifaceted objectives is a critical discipline in itself, known as reward engineering.

## 2.6 Reward Engineering: Shaping Agent Behavior

The reward function is arguably the most critical component in a Reinforcement Learning system. It is the sole mechanism through which a designer can communicate the desired behavior to the agent. A poorly designed reward can lead to

unintended and suboptimal behaviors, even if the agent successfully maximizes it. The entire field of reward engineering has become a cornerstone of modern RL, as it is fundamental to applying these algorithms effectively in complex, real-world applications<sup>6</sup>. The V2G problem, with its multiple competing objectives—maximizing profit, ensuring user satisfaction, preserving battery health, and maintaining grid stability—makes reward design particularly challenging. This work explores several sophisticated techniques to guide the learning process effectively.

### 2.6.1 Potential-Based Reward Shaping (PBRS)

Potential-Based Reward Shaping (PBRS) is perhaps the most theoretically sound method for reward augmentation<sup>7</sup>. It involves adding a shaping term,  $F(s, s')$ , to the environment’s intrinsic reward,  $R(s, a, s')$ . The new, shaped reward  $R'$  is thus defined as:

$$R'(s, a, s') = R(s, a, s') + F(s, s')$$

The shaping term itself is defined as the difference between a potential function,  $\Phi$ , evaluated at the new state and the old state:

$$F(s, s') = \gamma\Phi(s') - \Phi(s)$$

where  $\gamma$  is the discount factor. The key theoretical guarantee of PBRS is **policy invariance**: adding a potential-based shaping reward does not change the optimal policy of the underlying MDP. While the agent learns the same optimal behavior, the shaping term can provide dense, intermediate rewards that significantly accelerate the learning process by guiding the agent’s exploration.

### 2.6.2 Dynamic and Adaptive Rewards

Unlike PBRS, which provides a static reward bonus, a dynamic or adaptive reward function can evolve during training. This approach is particularly useful for complex problems where the relative importance of different objectives may change as the agent becomes more competent. For example, an agent could initially be rewarded simply for keeping an EV charged, but as training progresses, the reward function could adapt to introduce penalties for grid overloads or incentives for V2G services<sup>8</sup>. This allows the agent to master different facets of the problem sequentially.

### 2.6.3 Curriculum Learning

While technically a training paradigm rather than a reward modification technique, Curriculum Learning (CL) is highly relevant to reward engineering. CL involves training the agent on a sequence of tasks that gradually increase in difficulty. In the V2G context, an agent might first be trained in a simple scenario with only a few EVs and stable prices. Once it masters this, it is moved to a more complex

---

<sup>6</sup>11.

<sup>7</sup>22.

<sup>8</sup>33.

environment with more EVs, volatile prices, and hard constraints. This structured learning process prevents the agent from being overwhelmed by the full complexity of the problem from the start and can lead to more robust and generalizable policies.

## 2.7 The Rise of Deep Reinforcement Learning for V2G Control

Fusing RL with the representational power of deep neural networks gives rise to **Deep Reinforcement Learning (DRL)**, which currently represents the forefront of V2G control. The evolution of DRL algorithms for V2G has produced a sophisticated and robust toolkit, primarily branching into two main families: off-policy and on-policy methods, each with its own philosophy and trade-offs.

### 2.7.1 Off-Policy Methods: Data-Efficient Learning from Experience

Off-policy algorithms are distinguished by their ability to learn the optimal policy from data generated by a different, often more exploratory, policy. This decoupling allows them to reuse past experiences stored in a *replay buffer*, making them highly sample-efficient and well-suited for complex problems where real-world interaction is costly.

**Deep Deterministic Policy Gradient (DDPG)** A seminal algorithm that extended the success of Deep Q-Networks (DQN) to continuous action spaces, DDPG was a foundational breakthrough for control problems like V2G<sup>9</sup>. As an Actor-Critic method, it learns a deterministic policy (the Actor) that maps states to specific actions, guided by a Q-value function (the Critic). However, its practical application is often hampered by training instability and a crippling vulnerability to **overestimation bias**, where the Critic systematically overestimates Q-values. This error propagates through the learning process, leading the Actor to converge on suboptimal policies<sup>10</sup>.

**Twin Delayed DDPG (TD3)** TD3 was engineered specifically to counteract the instabilities of DDPG<sup>11</sup>. It introduces three crucial innovations:

1. **Clipped Double Q-Learning:** It learns two independent Critic networks and uses the minimum of their Q-value predictions to calculate the target value. This conservative approach effectively mitigates overestimation bias.
2. **Delayed Policy Updates:** The Actor and target networks are updated less frequently than the Critic. This allows the Critic’s value estimate to stabilize before the policy is modified, leading to smoother and more reliable training.

---

<sup>9</sup>15.

<sup>10</sup>23, 2.

<sup>11</sup>9.

3. **Target Policy Smoothing:** A small amount of clipped noise is added to the target action, which helps to regularize the learning process and prevent the policy from exploiting narrow peaks in the value function.

These additions make TD3 a much more robust and reliable baseline for V2G tasks<sup>12</sup>.

**Soft Actor-Critic (SAC)** SAC stands at the cutting edge of continuous control, offering superior sample efficiency and stability<sup>13</sup>. Its core innovation is the **maximum entropy framework**. Here, the agent's objective is not just to maximize cumulative reward, but to do so while acting as randomly (stochastically) as possible. This entropy bonus encourages broad exploration, preventing premature convergence to a narrow, suboptimal policy, and improves robustness by teaching the agent to "keep its options open"<sup>14</sup>.

**Truncated Quantile Critics (TQC)** TQC tackles overestimation bias from a distributional perspective, presenting a more fundamental solution than TD3<sup>15</sup>. Instead of learning a single expected return (a Q-value), it learns the entire *distribution of returns* using quantile regression with multiple Critic networks. Its key mechanism is to "truncate" (discard) the top-k most optimistic quantile estimates when forming the target distribution, thereby systematically removing the primary source of overestimation bias.

**Enhancement: Prioritized Experience Replay (PER)** This is not a standalone algorithm but a crucial modification for off-policy methods. Rather than sampling uniformly from the replay buffer, PER samples transitions with a probability proportional to their "importance," typically measured by the magnitude of their TD error. This focuses the learning process on "surprising" or informative experiences, significantly accelerating convergence<sup>16</sup>.

## 2.7.2 On-Policy Methods: Stability through Cautious Updates

On-policy methods learn from data generated exclusively by the current policy. This means that after each policy update, all previously collected data must be discarded. While this makes them inherently less sample-efficient, their updates are often more stable and less prone to divergence.

**Advantage Actor-Critic (A2C/A3C)** A2C is a foundational on-policy Actor-Critic algorithm. Its practical and powerful extension, **Asynchronous Advantage Actor-Critic (A3C)**, employs parallel workers to interact with multiple copies of the environment. These workers update a global set of parameters asynchronously,

---

<sup>12</sup>16, 34.

<sup>13</sup>haarnoja2019soft.

<sup>14</sup>17.

<sup>15</sup>14.

<sup>16</sup>28.

which decorrelates the data stream and provides a powerful stabilizing effect on the learning process<sup>17</sup>.

**Trust Region Policy Optimization (TRPO)** TRPO was the first algorithm to rigorously formalize the idea of controlling the policy update size to guarantee stable, monotonic improvements<sup>18</sup>. It maximizes a surrogate objective function subject to a constraint on the "behavioral change" of the policy, measured by the Kullback-Leibler (KL) divergence. This creates a "trust region" around the old policy, preventing catastrophic updates that could destroy performance. Its implementation, however, is complex as it requires second-order optimization.

**Proximal Policy Optimization (PPO)** PPO achieves the stability benefits of TRPO using only first-order optimization, making it far simpler to implement and more broadly applicable<sup>19</sup>. Instead of a hard constraint, PPO modifies the objective function with a **clipping** mechanism that disincentivizes policy updates resulting in a large probability ratio between the new and old policies. This creates a "soft" trust region and has become a default choice for many on-policy applications due to its robustness and ease of use.

### 2.7.3 Gradient-Free Methods: An Alternative Path

**Augmented Random Search (ARS)** ARS is an on-policy, gradient-free method that optimizes the policy by operating directly in the parameter space<sup>20</sup>. Instead of calculating gradients, it explores random directions around the current policy parameters and updates them based on the observed performance. While often much less sample-efficient than gradient-based methods for complex V2G problems, its simplicity can make it competitive in certain scenarios.

## 2.8 The Model-Based Benchmark: Model Predictive Control (MPC)

While DRL offers a powerful model-free approach, it must be benchmarked against its most robust model-based counterpart: **Model Predictive Control (MPC)**. MPC is not a recent invention; its roots trace back to the 1970s in the chemical engineering and process control industries. Pioneers in the field, such as Richalet et al. with their Model Algorithmic Control (MAC)<sup>21</sup> and Cutler and Ramaker with Dynamic Matrix Control (DMC)<sup>22</sup>, developed these techniques to manage complex industrial processes with slow dynamics. The core ideas have since been rigorously formalized and expanded for a vast range of applications by leading researchers like

---

<sup>17</sup>21.

<sup>18</sup>schulman2020trust.

<sup>19</sup>29.

<sup>20</sup>18.

<sup>21</sup>25.

<sup>22</sup>6.

Mayne, Rawlings, and Bemporad, whose work established firm theoretical foundations for stability and optimality<sup>23</sup>.

At its heart, MPC is an advanced control method that uses an explicit model of the system to predict its future evolution and compute an optimal control sequence over a finite prediction horizon,  $N$ . Its primary strength, especially relevant in the V2G context, is the ability to proactively handle complex dynamics and operational constraints<sup>24</sup>.

### 2.8.1 Implicit MPC: Online Optimization

The most common formulation of MPC is **Implicit MPC**, where a detailed optimization problem is solved online at each control step. For a linear time-invariant system, this problem is typically a Quadratic Program (QP).

The controller's objective is to find a sequence of future control inputs  $U = [u_{t|t}, \dots, u_{t+N-1|t}]$  that minimizes a cost function  $J$ , which penalizes deviations from a desired state and the control effort itself.

$$\min_U J(x_t, U) = \sum_{k=0}^{N-1} (x_{t+k|t}^T Q x_{t+k|t} + u_{t+k|t}^T R u_{t+k|t}) + x_{t+N|t}^T P x_{t+N|t} \quad (2.1)$$

where  $x_{t+k|t}$  is the predicted state at future time  $k$  based on information at current time  $t$ , and  $Q$ ,  $R$ , and  $P$  are weighting matrices.

This optimization is subject to critical constraints that define the system's valid operating envelope:

- **System Dynamics:** The model that predicts the next state.

$$x_{t+k+1|t} = A x_{t+k|t} + B u_{t+k|t} \quad (2.2)$$

- **State and Input Constraints:** Physical or operational limits.

$$x_{min} \leq x_{t+k|t} \leq x_{max} \quad (2.3)$$

$$u_{min} \leq u_{t+k|t} \leq u_{max} \quad (2.4)$$

At each time step  $t$ , this problem is solved to find the optimal control sequence  $U^*$ . Only the first action,  $u_{t|t}^*$ , is applied to the system. The entire process is then repeated at the next time step,  $t + 1$ , using new state measurements—a principle known as a *receding horizon*. This constant re-evaluation gives MPC its feedback mechanism and robustness to disturbances.

### 2.8.2 Explicit MPC: Offline Pre-computation

For systems with fast dynamics or limited online computational power, **Explicit MPC** offers an alternative. In this paradigm, the optimization problem is solved

---

<sup>23</sup>19.

<sup>24</sup>20.



offline for all possible states within the operating range using multi-parametric programming.

The result is a pre-computed, explicit control law,  $\pi(x_t)$ , which is a **piecewise affine function** of the state vector  $x_t$ . The state space is partitioned into a set of distinct polyhedral regions,  $\mathcal{X}_i$ , each with its own corresponding optimal control law.

$$u^*(x_t) = F_i x_t + g_i \quad \text{if } x_t \in \mathcal{X}_i \quad (2.5)$$

Here,  $F_i$  is the gain matrix and  $g_i$  is the offset for region  $i$ . The online operation is reduced to two simple steps:

1. Identify which region  $\mathcal{X}_i$  the current state  $x_t$  belongs to (a fast lookup procedure).
2. Apply the corresponding pre-computed affine control law.

This eliminates the need for a powerful online solver but comes at the cost of a potentially very high offline computation burden and significant memory requirements to store the lookup table of control laws.

## 2.9 A Comparative Perspective on Control Methodologies

While DRL represents the cutting edge, it is crucial to contextualize it within the broader landscape.

**Model Predictive Control (MPC)** is the most powerful model-based alternative<sup>25</sup>. Its primary strength is its ability to handle constraints. However, its performance is fundamentally shackled to the accuracy of its internal model and forecasts<sup>26</sup>. In the V2G domain, creating an accurate model is nearly impossible due to non-linear battery dynamics, market volatility, and human unpredictability. Furthermore, solving the large-scale Mixed-Integer Linear Program (MILP) required at each time step becomes computationally intractable for large fleets<sup>27</sup>.

Other methods, such as **meta-heuristic algorithms** (e.g., genetic algorithms), are typically used for offline scheduling and lack the real-time responsiveness required for dynamic V2G control<sup>28</sup>.

Ultimately, the singular advantage of DRL lies in its native ability to learn and internalize the complex, non-linear trade-offs of the multi-objective V2G problem directly from data. This makes it uniquely suited to navigating the uncertainties of the real world. While other methods have their place, DRL stands out as the most promising technology for deploying the truly intelligent, autonomous, and robust V2G management systems required to achieve the ambitious energy and climate goals of the European Union.

---

<sup>25</sup>3.

<sup>26</sup>8.

<sup>27</sup>30.

<sup>28</sup>13.

Table 2.1: Comparative Analysis: DRL vs. Model Predictive Control (MPC) for V2G

Aspect	Deep Reinforcement Learning (DRL)	Model Predictive Control (MPC)
<b>Paradigm</b>	Model-Free, learning-based. Learns optimal policy via trial-and-error.	Model-Based, optimization-based. Solves an optimization problem at each step.
<b>Strengths</b>	<ul style="list-style-type: none"> <li>• Highly robust to uncertainty and stochasticity.</li> <li>• No need for an explicit system model.</li> <li>• Can learn complex, non-linear control policies.</li> <li>• Fast inference time once trained.</li> </ul>	<ul style="list-style-type: none"> <li>• Explicitly handles hard constraints (safety guarantees).</li> <li>• Proactive and anticipatory if forecasts are accurate.</li> <li>• Well-established and understood.</li> </ul>
<b>Weaknesses</b>	<ul style="list-style-type: none"> <li>• Can be sample-inefficient during training.</li> <li>• Lacks hard safety guarantees (an active research area).</li> <li>• "Black box" nature can make policies hard to interpret.</li> </ul>	<ul style="list-style-type: none"> <li>• Performance is fundamentally tied to model and forecast accuracy.</li> <li>• Computationally expensive at each time step (curse of dimensionality).</li> <li>• Brittle to forecast errors and unmodeled dynamics.</li> </ul>
<b>V2G Suitability</b>	Excellent for dynamic, uncertain environments with complex trade-offs.	Good for problems with simple dynamics and reliable forecasts, but struggles with real-world V2G complexity.

## 2.10 A Primer on Lithium-Ion Battery Chemistries and Degradation

The effectiveness of any V2G strategy is, at its core, constrained by the physical characteristics of the vehicle’s battery. The choice of battery chemistry is not a minor detail; it dictates the operational envelope of the EV, influencing its energy density, power capabilities, lifespan, and, critically, its safety. A clear understanding

of these trade-offs is essential to the development of robust and realistic control algorithms. This section provides an overview of the primary degradation mechanisms, the most prevalent lithium-ion chemistries, and the core concepts governing their performance.

### 2.10.1 Fundamental Concepts and Degradation Mechanisms

Battery degradation is an irreversible process that reduces a battery’s capacity (energy fade) and increases its internal resistance (power fade). It can be broadly categorized into two types: calendar aging and cyclic aging<sup>29</sup>.

- **Calendar Aging:** This refers to degradation that occurs whenever the battery is at rest, even when not in use. The primary mechanism is the slow, continuous growth of the **Solid Electrolyte Interphase (SEI)** layer on the anode surface. The SEI is a necessary passivation layer that forms during the first few cycles, but its continued growth consumes active lithium ions and electrolyte, leading to irreversible capacity loss and increased impedance. The rate of SEI growth is strongly accelerated by two factors:
  - **High Temperature:** Higher temperatures increase the rate of chemical reactions, causing the SEI to grow faster.
  - **High State of Charge (SoC):** A high SoC corresponds to a low anode potential, which makes the anode more reactive with the electrolyte, thus promoting SEI growth<sup>30</sup>. Storing a battery at 100% SoC, especially in a hot environment, is one of the most significant contributors to calendar aging.
- **Cyclic Aging:** This degradation occurs as a direct result of charging and discharging the battery. Key mechanisms include:
  - **Mechanical Stress:** During intercalation and de-intercalation, the active materials in the electrodes expand and contract. Over many cycles, this repeated mechanical stress can cause micro-cracks in the electrode particles, leading to a loss of electrical contact and capacity. This effect is more pronounced with larger **Depths of Discharge (DoD)**.
  - **SEI Layer Instability:** The volume changes during cycling can also crack the protective SEI layer, exposing fresh anode material to the electrolyte. This triggers the formation of new SEI, consuming more lithium in the process.
  - **Lithium Plating:** Under conditions of high charging rates (high C-rate) and/or low temperatures, lithium ions may not have sufficient time to properly intercalate into the graphite anode. Instead, they deposit on the anode surface as metallic lithium. This is highly detrimental as it causes rapid, irreversible capacity loss and can form needle-like structures

---

<sup>29</sup>4.

<sup>30</sup>vetter2022ageing.

called dendrites, which can pierce the separator and cause an internal short circuit, posing a severe safety risk<sup>31</sup>.

For V2G applications, which inherently involve frequent charge/discharge cycles, understanding and mitigating cyclic aging is paramount.

### 2.10.2 Key Automotive Chemistries

The EV market is dominated by a few key families of lithium-ion batteries, primarily distinguished by their cathode materials.

- **Lithium Nickel Manganese Cobalt Oxide (NMC):** A highly popular choice due to its balanced performance. By adjusting the ratio of Nickel, Manganese, and Cobalt, manufacturers can tailor the battery to prioritize either energy density (higher Nickel content, e.g., NMC811) or safety and longevity (higher Manganese/Cobalt content, e.g., NMC532).
- **Lithium Nickel Cobalt Aluminum Oxide (NCA):** Similar to NMC but uses Aluminum instead of Manganese. This chemistry, famously used by Tesla for many years, offers very high energy density, enabling longer ranges, but at the cost of slightly lower cycle life and safety margins compared to NMC.
- **Lithium Iron Phosphate (LFP):** This chemistry is rapidly gaining market share. It contains no cobalt, making it cheaper and more ethically sourced. LFP batteries offer exceptional cycle life and are considered the safest among common Li-ion types. Their main drawbacks are lower nominal voltage and lower energy density.
- **Lithium Titanate Oxide (LTO):** LTO batteries use a titanate anode. They are exceptional in terms of safety, cycle life ( $>10,000$  cycles), and low-temperature performance. However, their very low energy density and high cost make them a niche solution.

### 2.10.3 Voltage Profiles and the Challenge of SoC Estimation

The relationship between a battery's voltage and its SoC is a critical, non-linear function. The derivative of the cell voltage with respect to the DoD,  $\frac{dV_{cell}}{d(DoD)}$ , is a crucial parameter for the Battery Management System (BMS). A steep, consistent slope allows the BMS to accurately infer the SoC from a voltage measurement. Conversely, a flat slope ( $\frac{dV_{cell}}{d(DoD)} \approx 0$ ) makes this estimation extremely difficult, as a small voltage measurement error can translate into a massive SoC error.

As shown in Figure 2.1, LFP's flat voltage plateau makes it difficult for the BMS to determine the precise SoC in the central part of its operating range. This necessitates periodic full charges to 100% to recalibrate the system at a point where the voltage curve is steep again, an important operational constraint for V2G control strategies.

---

<sup>31</sup>4.

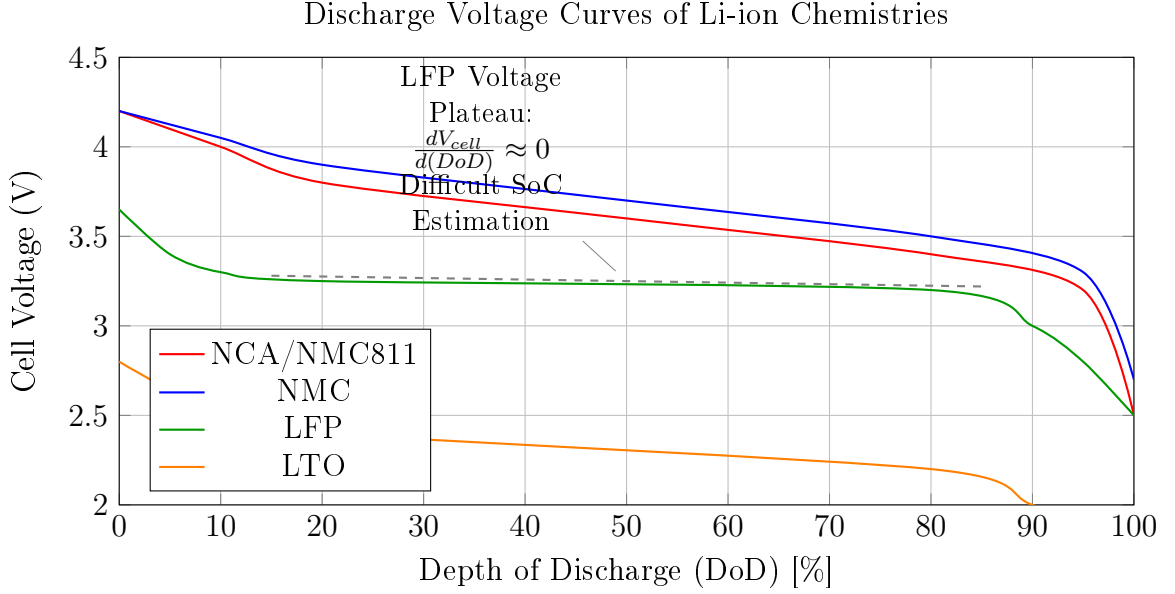


Figure 2.1: Typical discharge voltage curves for various lithium-ion chemistries. The flat profile of LFP makes accurate SoC estimation challenging based on voltage alone<sup>33</sup>.

#### 2.10.4 Comparative Analysis and Safety Considerations

The trade-offs between chemistries are summarized in Table 2.2. Safety is paramount, and the primary risk is thermal runaway. The risk is directly related to the stored energy density ( $\Delta E/\Delta m$ ). A higher energy density means more energy is packed into a smaller mass, which can be released violently if the cell's structure is compromised. Consequently, the critical temperature for initiating thermal runaway is generally lower for higher energy density chemistries. As energy density decreases, the thermal stability increases.

Table 2.2: Comparative analysis of key automotive battery chemistries<sup>34</sup>.

Metric	NCA	NMC	LFP
<b>Energy Density (Wh/kg)</b>	200 - 260 (Highest)	150 - 220 (High)	90 - 160 (Moderate)
<b>Cycle Life</b>	1000 - 2000	1000 - 2500	2000 - 5000+
<b>Safety</b>	Good	Very Good	Excellent
<b>Thermal Runaway Temp (°C)</b>	~150 - 180	~180 - 210	~220 - 270

#### 2.10.5 Battery Pack Architecture

Individual cells are assembled into modules and packs using a series-parallel configuration, denoted as **XsYp**. 'X' cells in series determine the pack voltage ( $V_{pack} = X \cdot V_{cell}$ ), which is typically 350-400V for modern EVs. 'Y' cells in parallel determine the pack capacity ( $C_{pack} = Y \cdot C_{cell}$ ). While the physical form factor (cylindrical,

prismatic, pouch) and BMS design are critical for engineering, our focus remains on the electrochemical degradation influenced by V2G control strategies.

## Chapter 3

# An Enhanced V2G Simulation Framework for Robust Control

Developing, validating, and benchmarking advanced control algorithms for Vehicle-to-Grid (V2G) systems is a complex endeavor. Real-world experimentation is often impractical due to prohibitive costs, logistical challenges, and risks to grid stability and vehicle hardware. To bridge the gap between theory and practice, a realistic, flexible, and standardized simulation environment is a scientific necessity. This thesis builds upon the foundation of **EV2Gym**, a state-of-the-art, open-source simulator designed for V2G smart charging research<sup>1</sup>. This work, however, extends the original framework significantly, transforming it into a high-fidelity **digital twin** engineered not just for single-scenario optimization, but for the development and rigorous evaluation of **robust, generalist control agents**.

This enhanced framework offers a two-pronged approach to experimentation: it allows for deep-dive analysis of agents specialized for a single environment, while also introducing a novel methodology for training and testing agents designed to generalize across a multitude of diverse, unpredictable scenarios. This chapter provides an in-depth tour of this extended architecture, its data-driven models, and its unique evaluation capabilities, establishing the methodological bedrock for the rest of this work.

### 3.1 Core Simulator Architecture

The framework is built on the modular architecture of EV2Gym, which mirrors the key entities of a real-world V2G system. Its foundation on the OpenAI Gym (now Gymnasium) API is a cornerstone, providing a standardized agent-environment interface defined by the familiar language of states, actions, and rewards<sup>2</sup>.

The architecture consists of several interacting components:

- **Charge Point Operator (CPO):** The central intelligence of the simulation, managing the charging infrastructure and serving as the primary interface for

---

<sup>1</sup>24.

<sup>2</sup>5.

the control algorithm (the DRL agent). The CPO aggregates system state information and dispatches control actions to individual chargers.

- **Chargers:** Digital representations of physical charging stations, configurable by type (AC/DC), maximum power, and efficiency. This allows for the simulation of heterogeneous charging infrastructures.
- **Power Transformers:** These components model the physical connection points to the grid, aggregating the electrical load from multiple chargers. Crucially, they enforce the physical power limits of the local distribution network and can model inflexible base loads (e.g., buildings) and local renewable generation (e.g., solar panels).
- **Electric Vehicles (EVs):** Dynamic and autonomous agents, each defined by its unique battery capacity, power limits, current and desired energy levels, and specific arrival and departure times.

The simulation process follows a reproducible three-phase structure: (1) **Initialization** from a comprehensive YAML configuration file, (2) a discrete-time **Simulation Loop** where the agent interacts with the environment, and (3) a final **Evaluation and Visualization** phase that generates standardized performance metrics.

## 3.2 Core Physical Models

The simulation’s fidelity is anchored in its detailed, empirically validated models, which are essential for developing control strategies robust enough for real-world application.

### 3.2.1 EV Model and Charging/Discharging Dynamics

The framework implements a realistic two-stage charging/discharging model that captures the non-linear behavior of lithium-ion batteries, simulating both the **constant current (CC)** and **constant voltage (CV)** phases. Each EV is defined by a rich parameter set: maximum capacity ( $E_{max}$ ), a minimum safety capacity ( $E_{min}$ ), separate power limits for charging and discharging ( $P_{ch}^{max}, P_{dis}^{max}$ ), and distinct efficiencies for each process ( $\eta_{ch}, \eta_{dis}$ ).

### 3.2.2 Battery Degradation Model

To address the critical issue of battery health in V2G operations, the simulator incorporates a semi-empirical battery degradation model. It quantifies capacity loss ( $Q_{lost}$ ) as the sum of two primary aging mechanisms<sup>3</sup>:

- **Calendar Aging ( $d_{cal}$ ):** Time-dependent capacity loss, influenced by the battery’s average State of Charge (SoC) and temperature.

---

<sup>3</sup>24.



- **Cyclic Aging ( $d_{cyc}$ ):** Wear resulting from charge/discharge cycles, dependent on energy throughput, depth-of-cycle, and C-rate.

This integrated model allows for the direct quantification of how different control strategies impact the battery’s long-term State of Health (SoH), enabling the training of agents that balance profitability with battery preservation.

### 3.2.3 EV Behavior and Grid Models

To ensure realism, the simulation is driven by authentic, open-source datasets. EV arrival/departure patterns and energy requirements are modeled using probability distributions derived from a large real-world dataset from **ElaadNL**. Grid conditions are similarly grounded in reality, using inflexible load data from the **Pecan Street** project and solar generation profiles from the **Renewables.ninja** platform<sup>4</sup>.

## 3.3 A Unified Experimentation and Evaluation Workflow

A key contribution of this thesis is the development of a unified and powerful experimentation workflow, orchestrated by the main script `run_experiments.py`. This script replaces the previous fragmented approach, providing a single, interactive interface to manage the entire lifecycle of training, benchmarking, and evaluation for V2G control agents. This workflow is designed to be both flexible for research and rigorous for evaluation, supporting the dual goals of developing specialized and generalized agents.

### 3.3.1 Orchestration via `run_experiments.py`

The `run_experiments.py` script acts as the central hub for all experimentation. It guides the user through an interactive command-line process, ensuring consistency and reproducibility. The key steps are:

1. **Algorithm Selection:** The user can select from a predefined list of algorithms to benchmark. This includes Deep Reinforcement Learning agents (e.g., SAC, DDPG+PER, TQC), classical optimization methods (Model Predictive Control), and rule-based heuristics (e.g., Charge As Fast As Possible).
2. **Scenario Selection:** The script automatically detects all available `.yaml` configuration files, allowing the user to choose one or more scenarios for the experiment. This choice determines the mode of operation (single-domain vs. multi-scenario).
3. **Reward Function Selection:** The framework’s flexibility is enhanced by allowing the user to dynamically select the reward function for the RL agents from the `reward.py` module.

---

<sup>4</sup>24.

4. **Training and Benchmarking:** Based on the user’s selections, the script proceeds to the optional training phase and then to a comprehensive benchmark, saving all results in a timestamped directory.

### 3.3.2 Dual-Mode Training: Specialists and Generalists

The new workflow elegantly unifies the training of both "specialist" and "generalist" agents, a concept previously handled by separate scripts. The behavior is determined implicitly by the number of selected scenarios:

- **Single-Domain Specialization:** If the user selects a single scenario, the script trains an RL agent exclusively on that environment. This produces a specialist agent, optimized to extract maximum performance from a specific, known set of conditions (e.g., a particular charging station topology and price profile).
- **Multi-Scenario Generalization:** If multiple scenarios are selected, the script automatically utilizes the `MultiScenarioEnv` wrapper. This custom Gymnasium environment dynamically switches between the different selected configurations at the start of each training episode. This process forces the agent to learn a robust and generalizable policy that performs well across a wide range of conditions, preventing overfitting to any single scenario. To handle the technical challenge of varying observation and action space sizes across scenarios, a `CompatibilityWrapper` is used to pad and slice the state-action vectors, enabling a single neural network policy to control heterogeneous environments.

### 3.3.3 Reproducible Benchmarking and Evaluation

To ensure a fair and scientifically valid comparison, the `run_benchmark` function implements a rigorous evaluation protocol. For each scenario, it first generates a "replay" file containing the exact sequence of stochastic events (e.g., EV arrivals, energy demands). This exact same sequence is then used to evaluate every algorithm, eliminating randomness as a factor in performance differences. The script runs multiple simulations for statistical robustness, aggregates the mean results, and automatically generates a suite of comparative plots, including overall performance metrics and detailed battery degradation analyses.

## 3.4 Evaluation Metrics

To ensure a fair and comprehensive comparison, all algorithms are evaluated against an identical set of pre-generated scenarios through a "replay" mechanism. The **mean** and **standard deviation** of performance are calculated across multiple simulation runs. The key metrics include:

- **Total Profit (\$):** The net economic outcome, calculated as revenue from energy sales minus the cost of energy purchases.

$$\Pi_{\text{total}} = \sum_{t=0}^{T_{\text{sim}}} \sum_{i=1}^N (C_{\text{sell}}(t)P_{\text{dis},i}(t) - C_{\text{buy}}(t)P_{\text{ch},i}(t)) \Delta t$$

- **Tracking Error (RMSE, kW):** For grid-balancing scenarios, this measures the root-mean-square error between the fleet’s aggregated power and a target setpoint.

$$E_{\text{track}} = \sqrt{\frac{1}{T_{\text{sim}}} \sum_{t=0}^{T_{\text{sim}}-1} (P_{\text{setpoint}}(t) - P_{\text{total}}(t))^2}$$

- **User Satisfaction (Average):** The fraction of energy delivered compared to what was requested by the user, averaged across all EV sessions. A score of 1 indicates perfect service.

$$US_{\text{avg}} = \frac{1}{N_{\text{EVs}}} \sum_{k=1}^{N_{\text{EVs}}} \min \left( 1, \frac{E_k(t_k^{\text{dep}})}{E_k^{\text{des}}} \right)$$

- **Transformer Overload (kWh):** The total energy that exceeded the transformer’s rated power limit. An ideal controller should achieve a value of 0.

$$O_{\text{tr}} = \sum_{t=0}^{T_{\text{sim}}} \sum_{j=1}^{N_T} \max(0, P_j^{\text{tr}}(t) - P_j^{\text{tr,max}}) \cdot \Delta t$$

- **Battery Degradation (\$):** The estimated monetary cost of battery aging due to both cyclic and calendar effects.

$$D_{\text{batt}} = \sum_{k=1}^{N_{\text{EVs}}} (\text{CyclicCost}_k + \text{CalendarCost}_k)$$

### 3.5 Simulator Implementation Details

During the analysis and implementation of new metrics, fundamental details about the EV2Gym simulator’s architecture emerged, which warrant documentation. The configuration of Electric Vehicles (EVs) and the calculation of their degradation follow a specific logic dependent on a key parameter in the `.yaml` configuration files.

#### Vehicle Definition Modes

The simulator operates in two distinct modes, controlled by the boolean flag `heterogeneous_ev_spec`.

- **Heterogeneous Mode (True):** In this mode, the simulator ignores the default vehicle specifications in the `.yaml` file. Instead, it loads a list of vehicle

profiles from an external JSON file, specified by the `ev_specs_file` parameter (e.g., `ev_specs_v2g_enabled2024.json`). This allows for the creation of a realistic fleet with diverse battery capacities, charging powers, and efficiencies. For instance, the fleet may include a **Peugeot 208** with a 46.3 kWh battery and a 7.4 kW charge rate, alongside a **Volkswagen ID.4** with a 77 kWh battery and an 11 kW charge rate. A vehicle is randomly selected from this list for each new arrival event.

- **Homogeneous Mode (False):** In this mode, the external JSON file is ignored. All vehicles created in the simulation are identical, and their characteristics are defined exclusively by the `ev:` block within the `.yaml` configuration file. The `battery_capacity` parameter in this block becomes the single source of truth for the entire fleet.

### Empirical Calibration of the Degradation Model

A significant enhancement in this work is the move towards a more physically representative and flexible battery degradation model. While the underlying semi-empirical model for calendar and cyclic aging remains, the methodology for parameterizing it has been fundamentally improved, addressing previous inconsistencies.

This is achieved through the `Fit_battery.py` script, a new utility for empirical model calibration. The script implements the following workflow:

1. **Data Loading:** It loads time-series data from real-world battery aging experiments. The expected data includes measurements of capacity loss over time, along with contextual variables like state of charge (SoC), temperature, and energy throughput.
2. **Model Fitting:** Using the `curve_fit` function from the SciPy library, the script fits the parameters of the `Qlost_model` (which combines calendar and cyclic aging) to the empirical data. This optimization process finds the physical constants (e.g.,  $\epsilon_0, \zeta_0$ ) that best explain the observed degradation.
3. **Parameter Export:** The script outputs the calibrated parameters. These values can then be used directly in the simulator’s configuration, ensuring that the degradation model for a specific EV fleet is grounded in experimental evidence for that battery type.

This calibration workflow, integrated optionally into the main `run_experiments.py` script, elevates the simulation’s fidelity. It allows the framework to move beyond a single, fixed degradation model (previously calibrated for a 78 kWh battery) and enables the creation of high-fidelity digital twins for a wide variety of EV batteries, provided that the necessary experimental data is available.

## 3.6 Reinforcement Learning Formulation

The control problem is formalized as a Markov Decision Process (MDP), defined by the tuple  $(S, A, P, R, \gamma)$ .

### 3.6.1 State Space ( $S$ )

The state  $s_t \in S$  is a feature vector providing a snapshot of the environment at time  $t$ . A representative state, as defined in modules like `V2G_profit_max_loads.py`, includes:

$$s_t = [t, P_{\text{total}}(t-1), \mathbf{c}(t, H), \mathbf{L}_1(t, H), \mathbf{P}\mathbf{V}_1(t, H), \dots, \mathbf{s}_1^{\text{EV}}(t), \dots, \mathbf{s}_N^{\text{EV}}(t)]^T$$

where the components are:

- $t$ : The current time step.
- $P_{\text{total}}(t-1)$ : The aggregated power from the previous time step.
- $\mathbf{c}(t, H)$ : A vector of **predicted future** electricity prices over a horizon  $H$ .
- $\mathbf{L}_j(t, H), \mathbf{P}\mathbf{V}_j(t, H)$ : Forecasts for inflexible loads and solar generation.
- $\mathbf{s}_i^{\text{EV}}(t) = [\text{SoC}_i(t), t_i^{\text{dep}} - t]$ : Key information for each EV  $i$ , including its State of Charge and remaining time until departure.

### 3.6.2 Action Space ( $A$ )

The action  $a_t \in A$  is a continuous vector in  $\mathbb{R}^N$ , where  $N$  is the number of chargers. For each charger  $i$ , the command  $a_i(t) \in [-1, 1]$  is a normalized value that is translated into a power command:

- If  $a_i(t) > 0$ , the EV is charging:  $P_i(t) = a_i(t) \cdot P_{\text{charge},i}^{\text{max}}$ .
- If  $a_i(t) < 0$ , the EV is discharging (V2G):  $P_i(t) = a_i(t) \cdot P_{\text{discharge},i}^{\text{max}}$ .

### 3.6.3 Reward Function

The reward function  $R(t)$  encodes the objectives of the control agent. The framework allows for the selection of different reward functions from the `reward.py` module to suit various goals. Key examples include:

- **Profit Maximization with Penalties** (`ProfitMax_TrPenalty_UserIncentives`): This function creates a balance between economic gain and physical constraints.

$$R(t) = \underbrace{\text{Profit}(t)}_{\text{Economic Gain}} - \underbrace{\lambda_1 \cdot \text{Overload}(t)}_{\text{Grid Penalty}} - \underbrace{\lambda_2 \cdot \text{Unsatisfaction}(t)}_{\text{User Penalty}}$$

The agent is rewarded for profit but penalized for overloading transformers and for failing to meet the charging needs of departing drivers.

- **Squared Tracking Error** (`SquaredTrackingErrorReward`): Used for grid service applications where precision is paramount.

$$R(t) = - \left( P_{\text{setpoint}}(t) - \sum_{i=1}^N P_i(t) \right)^2$$

The reward is the negative squared error from the power setpoint, incentivizing the agent to minimize this error at all times.

By using this enhanced framework, this thesis moves beyond single-scenario optimization to develop and validate an intelligent V2G control agent that is not only high-performing but also robust, adaptable, and ready for the complexities of real-world deployment.

### 3.7 Reinforcement Learning Algorithms

This work benchmarks several state-of-the-art Deep Reinforcement Learning algorithms. The following sections provide a detailed mathematical description of the selected off-policy, actor-critic algorithms.

#### Soft Actor-Critic (SAC)

SAC is an off-policy actor-critic algorithm designed for continuous action spaces that optimizes a stochastic policy. Its core feature is entropy maximization, which encourages exploration and improves robustness. The agent aims to maximize not only the expected sum of rewards but also the entropy of its policy.

The objective function is:

$$J(\pi) = \sum_{t=0}^T \mathbb{E}_{(s_t, a_t) \sim \rho_\pi} [r(s_t, a_t) + \alpha \mathcal{H}(\pi(\cdot|s_t))]$$

where  $\mathcal{H}$  is the entropy of the policy  $\pi$  and  $\alpha$  is the temperature parameter, which controls the trade-off between reward and entropy.

SAC uses a soft Q-function, trained to minimize the soft Bellman residual:

$$L(\theta_Q) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim D} \left[ (Q(s_t, a_t) - (r_t + \gamma V_{\bar{\psi}}(s_{t+1})))^2 \right]$$

where  $D$  is the replay buffer and the soft state value function  $V$  is defined as:

$$V_{\text{soft}}(s_t) = \mathbb{E}_{a_t \sim \pi} [Q_{\text{soft}}(s_t, a_t) - \alpha \log \pi(a_t|s_t)]$$

To mitigate positive bias, SAC employs two Q-networks (Clipped Double-Q) and takes the minimum of the two target Q-values during the Bellman update.

#### Deep Deterministic Policy Gradient + PER (DDPG+PER)

DDPG is an off-policy algorithm that concurrently learns a deterministic policy  $\mu(s|\theta^\mu)$  and a Q-function  $Q(s, a|\theta^Q)$ . It is the deep-learning extension of the DPG algorithm for continuous action spaces.

- **Critic Update:** The critic is updated by minimizing the mean-squared Bellman error, similar to Q-learning. Target networks ( $Q'$  and  $\mu'$ ) are used to stabilize training.

$$L(\theta^Q) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim D} [(y_t - Q(s_t, a_t|\theta^Q))^2]$$

where the target  $y_t$  is given by:

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

- **Actor Update:** The actor is updated using the deterministic policy gradient theorem:

$$\nabla_{\theta^\mu} J \approx \mathbb{E}_{s_t \sim D} [\nabla_a Q(s, a|\theta^Q)|_{s=s_t, a=\mu(s_t)} \nabla_{\theta^\mu} \mu(s_t|\theta^\mu)]$$

- **Prioritized Experience Replay (PER):** This work enhances DDPG with PER. Instead of uniform sampling from the replay buffer  $D$ , PER samples transitions based on their TD-error, prioritizing those where the model has the most to learn. The probability of sampling transition  $i$  is:

$$P(i) = \frac{p_i^\beta}{\sum_k p_k^\beta}$$

where  $p_i = |\delta_i| + \epsilon$  is the priority based on the TD-error  $\delta_i$ , and  $\beta$  controls the degree of prioritization. To correct for the bias introduced by non-uniform sampling, PER uses importance-sampling (IS) weights.

### Truncated Quantile Critics (TQC)

TQC enhances the stability of SAC by modeling the entire distribution of returns instead of just its mean. This is achieved through quantile regression and a novel truncation mechanism to combat Q-value overestimation.

- **Distributional Learning:** TQC employs a set of  $N$  critic networks,  $\{Q_{\phi_i}(s, a)\}_{i=1}^N$ , each trained to estimate a specific quantile  $\tau_i$  of the return distribution. The target quantiles are implicitly defined as  $\tau_i = \frac{i-0.5}{N}$ . The critics are trained by minimizing the quantile Huber loss,  $L_{QH}$ .
- **Distributional Target Calculation:** A distributional target is constructed for the Bellman update. First, an action is sampled from the target policy for the next state:  $\tilde{a}_{t+1} \sim \pi_{\theta'}(\cdot|s_{t+1})$ . Then, a set of  $N$  Q-value estimates for the next state is obtained from the  $N$  target critic networks:  $\{Q_{\phi'_j}(s_{t+1}, \tilde{a}_{t+1})\}_{j=1}^N$ .
- **Truncation:** This is the key idea of TQC. To combat overestimation, the algorithm discards the  $k$  largest Q-value estimates from the set of  $N$  target values. This truncation removes the most optimistic estimates, which are a primary source of bias, leading to more conservative and stable updates.
- **Critic Update:** The target value for updating the  $i$ -th critic is formed using the Bellman equation with the truncated set of next-state Q-values. The overall critic loss is the sum of the quantile losses across all critics:

$$L(\phi) = \sum_{i=1}^N \mathbb{E}_{(s,a,r,s') \sim D} [L_{QH}(r + \gamma Q_{\text{trunc}}(s', \tilde{a}') - Q_{\phi_i}(s, a))]$$

where  $Q_{\text{trunc}}$  represents the value derived from the truncated set of target quantiles.

### 3.7.1 A History-Based Adaptive Reward for Profit Maximization

To effectively steer the learning agent towards a policy that is both highly profitable and reliable, we have designed and implemented a novel, history-based adaptive reward function, named **FastProfitAdaptiveReward**. This function departs from traditional static-weight penalties and instead introduces a dynamic feedback mechanism where the severity of penalties is directly influenced by the agent’s recent performance. The core philosophy is to aggressively prioritize economic profit while using adaptive penalties as guardrails that become stricter only when the agent begins to consistently violate operational constraints.

The total reward at each timestep  $t$ ,  $R_t$ , is calculated as the net economic profit minus any active penalties for user dissatisfaction or transformer overload.

$$R_t = \Pi_t - P_t^{\text{sat}} - P_t^{\text{tr}} \quad (3.1)$$

#### Economic Profit

The foundation of the reward signal is the direct, instantaneous economic profit,  $\Pi_t$ . This component provides a clear and strong incentive for the agent to learn market dynamics, encouraging it to charge during low-price periods and discharge (V2G) during high-price periods.

$$\Pi_t = \sum_{i=1}^N \left( C_t^{\text{sell}} \cdot P_{i,t}^{\text{dis}} - C_t^{\text{buy}} \cdot P_{i,t}^{\text{ch}} \right) \Delta t \quad (3.2)$$

where  $N$  is the number of connected EVs,  $C_t^{\text{sell}}$  and  $C_t^{\text{buy}}$  are the electricity prices, and  $P_{i,t}^{\text{dis}}$  and  $P_{i,t}^{\text{ch}}$  are the discharging and charging powers for EV  $i$ .

#### Adaptive User Satisfaction Penalty

The penalty for failing to meet user charging demands,  $P_t^{\text{sat}}$ , is not a fixed value. Instead, it adapts based on the system’s recent history of performance. The environment maintains a short-term memory of the average user satisfaction over the last 100 timesteps. From this history, we calculate an average satisfaction score,  $\bar{S}_{\text{hist}}$ .

A *satisfaction severity multiplier*,  $\lambda_t^{\text{sat}}$ , is then calculated. This multiplier grows quadratically as the historical average satisfaction drops, meaning that if the system has been performing poorly, the consequences for a new failure become much more severe.

$$\lambda_t^{\text{sat}} = \lambda_{\text{base}}^{\text{sat}} \cdot (1 - \bar{S}_{\text{hist}})^2 \quad (3.3)$$

where  $\lambda_{\text{base}}^{\text{sat}}$  is a base scaling factor (e.g., 20.0). A penalty is only applied if any departing EV’s satisfaction,  $S_k$ , is below a critical threshold (e.g., 95%). The magnitude of the penalty is the product of the adaptive multiplier and the current satisfaction deficit.

$$P_t^{\text{sat}} = \lambda_t^{\text{sat}} \cdot (1 - \min(S_k)) \quad \forall k \in \text{EVs departing at } t \quad (3.4)$$



This creates a powerful feedback loop: a single, isolated failure in an otherwise well-performing system results in a mild penalty. However, persistent failures lead to a rapidly escalating penalty, forcing the agent to correct its behavior.

### Adaptive Transformer Overload Penalty

Similarly, the transformer overload penalty,  $P_t^{\text{tr}}$ , adapts based on the recent frequency of overloads. The environment tracks how often an overload has occurred in the last 100 timesteps, yielding an overload frequency,  $F_{\text{hist}}^{\text{tr}}$ .

This frequency is used to compute a linear *overload severity multiplier*,  $\lambda_t^{\text{tr}}$ . The more frequently overloads have happened, the higher the penalty for a new one.

$$\lambda_t^{\text{tr}} = \lambda_{\text{base}}^{\text{tr}} \cdot F_{\text{hist}}^{\text{tr}} \quad (3.5)$$

where  $\lambda_{\text{base}}^{\text{tr}}$  is a base scalar (e.g., 50.0). If the total power drawn,  $P_j^{\text{total}}(t)$ , exceeds the transformer’s limit,  $P_j^{\text{max}}$ , a penalty is applied. This penalty consists of a small, fixed base amount plus the adaptive component, which scales with the magnitude of the current overload.

$$P_t^{\text{tr}} = P_{\text{base}} + \lambda_t^{\text{tr}} \cdot \sum_{j=1}^{N_T} \max(0, P_j^{\text{total}}(t) - P_j^{\text{max}}) \quad (3.6)$$

This mechanism teaches the agent that while a rare, minor overload might be acceptable in pursuit of high profit, habitual overloading is an unsustainable and heavily penalized strategy.

### Rationale and Significance

This history-based adaptive reward function represents a significant advancement over static or purely state-based approaches. By making the penalty weights a function of the system’s recent performance history, we provide a more nuanced and stable learning signal. The agent is not punished excessively for isolated, exploratory actions that might lead to a minor constraint violation. Instead, it is strongly discouraged from developing policies that lead to chronic system failures.

The intuition is to mimic a more realistic management objective: maintain high performance on average, and react strongly only when performance trends begin to degrade. This method is also computationally efficient, avoiding complex state-dependent calculations in favor of simple updates to historical data queues. Ultimately, this reward structure guides the agent to discover policies that are not only profitable but also robust and reliable over time, striking a more intelligent balance between economic ambition and operational safety.

## 3.8 Model Predictive Control (MPC)

The MPC, implemented in `mpc.py` and `eMPC.py`, solves an optimization problem at every time step over a prediction horizon  $H$ .

### 3.8.1 System Model

The system is modeled in linear state-space form. The state  $\mathbf{x}_k \in \mathbb{R}^N$  is the vector of SoCs of all EVs at time  $k$ . The input  $\mathbf{u}_k \in \mathbb{R}^{2N}$  is the vector of charging and discharging powers.

$$\mathbf{x}_{k+1} = A_k \mathbf{x}_k + B_k \mathbf{u}_k$$

The matrices  $A_k$  (**A**mon) and  $B_k$  (**B**mon) are time-varying because they depend on which EVs are connected.  $A_k$  is typically a diagonal identity-like matrix modeling the persistence of EVs.  $B_k$  maps power to SoC change, including efficiencies and  $\Delta t$ .

### 3.8.2 Optimization Problem

At time  $t$ , the MPC solves:

$$\min_{\{\mathbf{u}_k\}_{k=t}^{t+H-1}} \sum_{k=t}^{t+H-1} \mathbf{f}_k^T \mathbf{u}_k$$

subject to:

$$\mathbf{x}_{k+1} = A_k \mathbf{x}_k + B_k \mathbf{u}_k, \quad \forall k \in [t, t+H-1] \quad (\text{Dynamics})$$

$$\mathbf{x}_k^{\min} \leq \mathbf{x}_k \leq \mathbf{x}_k^{\max} \quad (\text{SoC limits})$$

$$\mathbf{0} \leq \mathbf{u}_k^{\text{ch}} \leq \mathbf{u}_k^{\text{ch}, \max} \cdot \mathbf{z}_k \quad (\text{Charge limits})$$

$$\mathbf{0} \leq \mathbf{u}_k^{\text{dis}} \leq \mathbf{u}_k^{\text{dis}, \max} \cdot (1 - \mathbf{z}_k) \quad (\text{Discharge limits})$$

$$\sum_{i \in \text{CS}_j} (u_i^{\text{ch}} - u_i^{\text{dis}}) + L_j(k) - PV_j(k) \leq P_j^{\text{tr}, \max}(k) \quad (\text{Transformer limits})$$

where  $\mathbf{z}_k$  is a vector of binary variables to prevent simultaneous charge and discharge. The cost vector  $\mathbf{f}_k$  contains the energy prices. The code formulates this problem compactly as  $\mathbf{A}\mathbf{U} \leq \mathbf{b}\mathbf{U}$ , where  $\mathbf{U}$  is the vector of all actions over the horizon.

## 3.9 Offline Optimization with Gurobi

Gurobi is used to find the optimal offline (a posteriori) solution, providing a performance benchmark. The files `profit_max.py` and `tracking_error.py` define the optimization problem over the entire simulation horizon  $T_{\text{sim}}$ .

### 3.9.1 Decision Variables

- $E_{p,i,t}$ : Energy in the EV at port  $p$  of station  $i$  at time  $t$ .
- $I_{p,i,t}^{\text{ch}}, I_{p,i,t}^{\text{dis}}$ : Charging/discharging currents.

- $\omega_{p,i,t}^{\text{ch}}, \omega_{p,i,t}^{\text{dis}}$ : Binary variables for operating modes.

### 3.9.2 Objective Function (Example: Profit Maximization)

$$\max \sum_{t=0}^{T_{\text{sim}}} \sum_{i=1}^{N_{CS}} \sum_{p=1}^{N_p} (C_{\text{sell}}(t)P_{p,i,t}^{\text{dis}} - C_{\text{buy}}(t)P_{p,i,t}^{\text{ch}}) \Delta t - \lambda \sum_{k \in \text{EVs departed}} (E_k^{\text{des}} - E_k(t_k^{\text{dep}}))^2$$

where  $P = V \cdot I \cdot \eta$ .

### 3.9.3 Main Constraints

- **Energy Balance:**

$$E_{p,i,t} = E_{p,i,t-1} + (\eta_{\text{ch}} V_i I_{p,i,t}^{\text{ch}} - \frac{1}{\eta_{\text{dis}}} V_i I_{p,i,t}^{\text{dis}}) \Delta t$$

- **Activation of Current:**

$$I_{p,i,t}^{\text{ch}} \leq M \cdot \omega_{p,i,t}^{\text{ch}} \quad , \quad I_{p,i,t}^{\text{dis}} \leq M \cdot \omega_{p,i,t}^{\text{dis}}$$

- **Mutual Exclusion:**

$$\omega_{p,i,t}^{\text{ch}} + \omega_{p,i,t}^{\text{dis}} \leq 1$$

- **Current and SoC Limits:**

$$I^{\min} \leq I_{p,i,t} \leq I^{\max} \quad , \quad E^{\min} \leq E_{p,i,t} \leq E^{\max}$$

- **SoC at Departure:**

$$E_{p,i}(t^{\text{dep}}) \geq E_{p,i}^{\text{des}}$$

## 3.10 Online MPC Formulation (PuLP Implementation)

The Model Predictive Control (MPC) implemented with PuLP solves a profit maximization problem at each time step  $t$  over a finite prediction horizon  $H$ . This formulation is designed for online, real-time control, where decisions are made based on the current system state and future predictions.

### 3.10.1 Mathematical Formulation

At each time step  $t$ , the MPC controller solves the following optimization problem.

### Objective Function: Net Operational Profit

The objective is to maximize the total net operational profit over the control horizon  $H$ . This provides a comprehensive economic model that goes beyond simple energy arbitrage.

$$\max_{P^{\text{ch}}, P^{\text{dis}}, z} \sum_{k=t}^{t+H-1} \sum_{i \in \text{CS}} (\text{Revenues}_{i,k} - \text{Costs}_{i,k}) \quad (3.7)$$

The revenue and cost components are defined for each station  $i$  at time step  $k$  as:

- **Revenues** consist of:

- Grid Sales Revenue (V2G):  $c_k^{\text{sell}} \cdot P_{i,k}^{\text{dis}} \cdot \Delta t$
- User Charging Revenue:  $c^{\text{user}} \cdot P_{i,k}^{\text{ch}} \cdot \Delta t$

- **Costs** consist of:

- Grid Purchase Cost:  $c_k^{\text{buy}} \cdot P_{i,k}^{\text{ch}} \cdot \Delta t$
- Battery Degradation Cost:  $c^{\text{deg}} \cdot (P_{i,k}^{\text{ch}} + P_{i,k}^{\text{dis}}) \cdot \Delta t$

where  $c_k^{\text{sell}}$  and  $c_k^{\text{buy}}$  are the time-varying electricity prices,  $c^{\text{user}}$  is the fixed price for the end-user,  $c^{\text{deg}}$  is the estimated cost of battery degradation per kWh cycled, and  $\Delta t$  is the time step duration.

### System Constraints

The optimization is subject to the following constraints for each station  $i$  and time step  $k \in [t, t + H - 1]$ .

**Energy Balance Dynamics.** The state of energy of the EV battery evolves according to:

$$E_{i,k} = E_{i,k-1} + \left( \eta^{\text{ch}} P_{i,k}^{\text{ch}} - \frac{1}{\eta^{\text{dis}}} P_{i,k}^{\text{dis}} \right) \cdot \Delta t \quad (3.8)$$

where the initial state  $E_{i,t-1}$  is the currently measured energy level of the EV.

**Power Limits and Mutual Exclusion.** Charging and discharging powers are bounded by the EV's capabilities and controlled by a binary variable  $z_{i,k}$  to prevent simultaneous operation.

$$0 \leq P_{i,k}^{\text{ch}} \leq P_i^{\text{ch}, \text{max}} \cdot z_{i,k} \quad (3.9)$$

$$0 \leq P_{i,k}^{\text{dis}} \leq P_i^{\text{dis}, \text{max}} \cdot (1 - z_{i,k}) \quad (3.10)$$

**State of Energy (SoE) Limits.** The battery energy level must remain within its physical operational window.

$$E_i^{\text{min}} \leq E_{i,k} \leq E_i^{\text{max}} \quad (3.11)$$

**User Satisfaction (Hard Constraint).** The desired energy level must be met at the time of departure. This is modeled as a hard constraint, reflecting a non-negotiable service requirement.

$$E_{i,k_{\text{dep}}} \geq E_i^{\text{des}} \quad (3.12)$$

where  $k_{\text{dep}}$  is the predicted departure step of the EV within the horizon.

**Transformer Power Limit.** The total net power drawn from (or injected into) the grid by all charging stations must not exceed the transformer's maximum capacity.

$$\sum_{i \in \text{CS}} (P_{i,k}^{\text{ch}} - P_{i,k}^{\text{dis}}) \leq P^{\text{tr,max}} \quad (3.13)$$

### 3.11 Conceptual Comparison: PuLP MPC vs. Gurobi Offline Optimizer

While both the PuLP MPC and the Gurobi offline optimizer are used to solve the EV charging problem, they operate on fundamentally different principles and serve distinct purposes. This section provides a discursive comparison of their core concepts.

#### 3.11.1 Core Philosophy: Controller vs. Judge

The most significant difference lies in their philosophy. The **PuLP MPC** is designed as a **controller**. It operates online, making decisions in real-time with incomplete information about the future (e.g., EV arrivals, price fluctuations beyond the prediction horizon). Its goal is to find a practical and robust strategy for the immediate future.

Conversely, the **Gurobi formulation** acts as a **judge**. It is an offline tool that solves the problem over the entire simulation period with perfect hindsight (a-posteriori). Its purpose is not to control the system in real-time, but to establish a theoretical performance benchmark—the "perfect score"—against which the performance of a practical controller like the MPC can be measured.

#### 3.11.2 Objective Function: Operational Profit vs. Energy Arbitrage

The objectives, while both related to profit, reflect their different roles. The PuLP MPC maximizes a detailed **Net Operational Profit**, incorporating a realistic business model that includes revenue from end-users and operational costs like battery degradation. This makes its decisions economically grounded from a business perspective.

The Gurobi optimizer, on the other hand, typically maximizes profit from a simpler **energy arbitrage** model, focusing on the difference between buying and selling

electricity. While it includes a penalty for not meeting user demand, it does not explicitly account for the same level of operational economic detail as the MPC.

### 3.11.3 Handling of User Satisfaction: Hard vs. Soft Constraints

This distinction is critical from an operational standpoint. The PuLP MPC treats user satisfaction as a **Hard Constraint**. The EV *must* reach its desired energy level by its departure time. If the model determines this is impossible, the optimization problem becomes infeasible, signaling a failure to meet a mandatory service level agreement.

The Gurobi formulation treats user satisfaction primarily as a **Soft Constraint** via a penalty term in its objective function. This allows the optimizer to make a trade-off: it can choose to not fully charge a vehicle if the economic benefit of doing so (e.g., selling a large amount of energy to the grid at a high price) outweighs the penalty for customer dissatisfaction. This is useful for theoretical analysis but less practical for guaranteeing service.

# Bibliography

- [1] *'Fit for 55': delivering the EU's 2030 Climate Target on the way to climate neutrality*. Tech. rep. COM(2021) 550 final. Brussels: European Commission, 2021.
- [2] Farag Alfaverh, Mouloud Denaï, and Yifei Sun. “Optimal vehicle-to-grid control for supplementary frequency regulation using deep reinforcement learning”. In: *Applied Energy* 325 (2022), p. 119881. DOI: 10.1016/j.apenergy.2022.119881.
- [3] Mohamad Alsabbagh and Wan-Chi Siu. “Reinforcement learning for vehicle-to-grid: a review”. In: *Journal of Energy Storage* 53 (2022), p. 105149. DOI: 10.1016/j.est.2022.105149.
- [4] Christoph R Birkel et al. “Degradation diagnostics for lithium ion cells”. In: *Journal of Power Sources* 341 (2017), pp. 373–386.
- [5] Greg Brockman et al. “OpenAI Gym”. In: *arXiv preprint arXiv:1606.01540* (2016). DOI: 10.48550/arXiv.1606.01540.
- [6] C. R. Cutler and B. L. Ramaker. “Dynamic Matrix Control - A Computer Control Algorithm”. In: *Proceedings of the 1980 Joint Automatic Control Conference*. Paper No. WP5-B. San Francisco, CA: American Society of Mechanical Engineers (ASME), 1980.
- [7] Albin Evertsson and Anton Nylander. “Investigating Vehicle-to-grid from a User-centric Perspective”. MA thesis. Chalmers University of Technology, 2024.
- [8] Gabriele Faggio. “Design and Testing of Online and Offline Optimization Algorithms for Vehicle-to-Grid (V2G) Industrial Applications”. MA thesis. Politecnico di Milano, 2023.
- [9] Scott Fujimoto, Herke van Hoof, and David Meger. “Addressing function approximation error in actor-critic methods”. In: *International conference on machine learning*. PMLR. 2018, pp. 1587–1596.
- [10] Ghaith Al-HMOUD and Hamed Al-Raweshidy. “A Review of Smart Grid Evolution and Reinforcement Learning: Applications, Challenges and Future Directions”. In: *Energies* 18.7 (2024), p. 1837. DOI: 10.3390/en18071837.
- [11] Sinan Ibrahim et al. “Comprehensive Overview of Reward Engineering and Shaping in Advancing Reinforcement Learning Applications”. In: *arXiv preprint arXiv:2408.10215* (2024).

- [12] Sheroz Ullah Khan et al. “A Review of Bidirectional Charging Grid Support Applications and Control”. In: *Energies* 17.6 (2024), p. 1320. DOI: 10.3390/en17061320.
- [13] Vinay Kumar, Surender Singh, and Dinesh Kumar. “Integration of electric vehicle into smart grid: a meta heuristic algorithm for energy management between V2G and G2V”. In: *Frontiers in Energy Research* 12 (2024), p. 1357863. DOI: 10.3389/fenrg.2024.1357863.
- [14] Anton Kuznetsov et al. “Controlling overestimation bias with truncated quantile critics”. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 5528–5538.
- [15] Timothy P Lillicrap et al. “Continuous control with deep reinforcement learning”. In: *arXiv preprint arXiv:1509.02971* (2015).
- [16] Zhaomiao Liu et al. “Optimal scheduling for charging and discharging of electric vehicles based on deep reinforcement learning”. In: *Frontiers in Energy Research* 11 (2023), p. 1273820. DOI: 10.3389/fenrg.2023.1273820.
- [17] J Logeshwaran, Chao Fan, and Swan Htet Naung. “A comparative study of deep reinforcement learning algorithms for energy management in commercial buildings”. In: *Energy and Buildings* 254 (2022), p. 111589.
- [18] Horia Mania, Aurelia Guy, and Benjamin Recht. “Simple random search of static linear policies is competitive for reinforcement learning”. In: *Advances in neural information processing systems*. Vol. 31. 2018.
- [19] D. Q. Mayne et al. “Constrained model predictive control: Stability and optimality”. In: *Automatica* 36.6 (2000), pp. 789–814.
- [20] Carlos A Minchala- Ávila, Pa úl Ar évalo, and Diego Ochoa-Correa. “A Systematic Review of Model Predictive Control for Robust and Efficient Energy Management in Electric Vehicle Integration and V2G Applications”. In: *Modelling* 6.1 (2025), p. 20. DOI: 10.3390/modelling6010020.
- [21] Volodymyr Mnih et al. “Asynchronous methods for deep reinforcement learning”. In: *International conference on machine learning*. PMLR. 2016, pp. 1928–1937.
- [22] Andrew Y Ng, Daishi Harada, and Stuart Russell. “Policy invariance under reward transformations: Theory and application to reward shaping”. In: *ICML* 99 (1999), pp. 278–287.
- [23] Stylianos Orfanoudakis et al. “A Deep Reinforcement Learning-Based Charging Strategy for Electric Vehicles with V2G Services in a Volatile Market”. In: *arXiv preprint arXiv:2209.09772* (2022). DOI: 10.48550/arXiv.2209.09772.
- [24] Stylianos Orfanoudakis et al. “EV2Gym: A Flexible V2G Simulator for EV Smart Charging Research and Benchmarking”. In: *arXiv preprint arXiv:2404.01849* (2024). DOI: 10.48550/arXiv.2404.01849.
- [25] J. Richalet et al. “Model predictive heuristic control: Applications to industrial processes”. In: *Automatica* 14 (1978), pp. 413–428.



- [26] Sadeghi. “Cost and Power Loss Aware Coalitions under Uncertainty in Trans-active Energy Systems”. In: *Université d’Ottawa / University of Ottawa* (2021).
- [27] Gabriel Antonio Salvatti et al. “Electric Vehicles Energy Management with V2G/G2V Multifactor Optimization of Smart Grids”. In: *Energies* 13.5 (2020), p. 1191. DOI: 10.3390/en13051191.
- [28] Tom Schaul et al. “Prioritized experience replay”. In: *arXiv preprint arXiv:1511.05952* (2015). DOI: 10.48550/arXiv.1511.05952.
- [29] John Schulman et al. “Proximal policy optimization algorithms”. In: *arXiv preprint arXiv:1707.06347* (2017).
- [30] Johannes Schwenk et al. “A computationally efficient MPC for residential EV charging with V2G and 3-phase power flow”. In: *Energies* 15.16 (2022), p. 5923. DOI: 10.3390/en15165923.
- [31] Mohamed M Shibl, Loay S Ismail, and Ahmed M Massoud. “Electric vehicles charging management using deep reinforcement learning considering vehicle-to-grid operation and battery degradation”. In: *Energy Reports* 10 (2023), pp. 494–509. DOI: 10.1016/j.egyrs.2023.07.039.
- [32] Ahmad Tavakoli et al. “Impacts of grid integration of solar PV and electric vehicle on grid stability, power quality and energy economics: a review”. In: *IET Energy Systems Integration* 2.3 (2020), pp. 233–245. DOI: 10.1049/iet-esi.2019.0047.
- [33] Zhong Wan et al. “A dynamic reward-based deep reinforcement learning for V2G control with EV battery degradation”. In: *Applied Energy* 309 (2022), p. 118462. DOI: 10.1016/j.apenergy.2021.118462.
- [34] Siyuan Wang, Shuo Wang, and Bo Liu. “Multi-objective optimal scheduling of charging stations with solar-storage-diesel generator system”. In: *Frontiers in Energy Research* 10 (2022), p. 1042882. DOI: 10.3389/fenrg.2022.1042882.
- [35] Hongbin Xie. “Deep reinforcement learning-based control strategies for electric vehicle charging and V2G services”. In: *Advances in Applied Energy* (2021). DOI: <https://doi.org/10.1016/j.adapen.2025.100214>.