Reinforcement Learning Portfolio Optimization of Electric Vehicle Virtual Power Plants

Master Thesis

Author: Tobias Richter (Student ID: 558305)
Supervisor: Univ.-Prof. Dr. Wolfgang Ketter
Co-Supervisor: Karsten Schroer, Philipp Artur Kienscherf

Department of Information Systems for Sustainable Society
Faculty of Management, Economics and Social Sciences
University of Cologne

February 18, 2019

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne die Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden. Ich versichere, dass die eingereichte elektronische Fassung der eingereichten Druckfassung vollständig entspricht.

Die Strafbarkeit einer falschen eidesstattlichen Versicherung ist mir bekannt, namentlich die Strafandrohung gemäß § 156 StGB bis zu drei Jahren Freiheitsstrafe oder Geldstrafe bei vorsätzlicher Begehung der Tat bzw. gemäß § 161 Abs. 1 StGB bis zu einem Jahr Freiheitsstrafe oder Geldstrafe bei fahrlässiger Begehung.

Tobias Richter

Köln, den 01.05.2019

Contents

List of Figures

List of Tables

List of Abbreviations

ANN	Artificial Neural Network
DSO	Distribution System Operator
DP	Dynamic Programming
EV	Electric Vehicle
GP	Genetic Programming
GPI	Generalized Policy Iteration
GCRM	German Control Reserve Market
MAW	Mean Asymmetric Weighted Objective Function
MC	Monte Carlo
MDP	Markov Decision Process
PDF	Probability Density Function
RL	Reinforcement Learning
TD	Temporal-Difference
TSO	Transmission System Operator
V2G	Vehicle-to-Grid
VPP	Virtual Power Plant

Summary of Notation

Capital letters are used for random variables, whereas lower case letters are used for the values of random variables and for scalar functions. Quantities that are required to be real-valued vectors are written in bold and in lower case (even if random variables).

$\doteq$	equality relationship that is true by definition
$\approx$	approximately equal
$\mathbb{E}[X]$	expectation of a random variable X, i.e., $\mathbb{E}[X] \doteq \sum_x p(x)x$
$\mathbb{R}$	set of real numbers
$\leftarrow$	assignment
ε	probability of taking a random action in an ε-greedy policy
α	step-size parameter
γ	discount-rate parameter
λ	decay-rate parameter for eligibility traces
s, s'	states
a	an action
r	a reward
$\mathcal{S}$	set of all nonterminal states
$\mathcal{A}$	set of all available actions
$\mathcal{R}$	set of all possible rewards, a finite subset of $\mathbb{R}$
$\subset$	subset of; e.g., $\mathcal{R} \subset \mathbb{R}$
$\in$	is an element of; e.g., $s \in \mathcal{S}$, $r \in \mathcal{R}$
t	discrete time step
$T, T(t)$	final time step of an episode, or of the episode including time step t
A_t	action at time t
S_t	state at time t, typically due, stochastically, to S_{t-1} and A_{t-1}
R_t	reward at time t, typically due, stochastically, to S_{t-1} and A_{t-1}
π	policy (decision-making rule)
$\pi(s)$	action taken in state s under *deterministic* policy π
$\pi(a\|s)$	probability of taking action a in state s under *stochastic* policy π
G_t	return following time t
$p(s', r \mid s, a)$	probability of transition to state s' with reward r, from state s and action a
$p(s' \mid s, a)$	probability of transition to state s', from state s taking action a
$v_\pi(s)$	value of state s under policy π (expected return)

$v_*(s)$	value of state s under the optimal policy
$q_\pi(s,a)$	value of taking action a in state s under policy π
$q_*(s,a)$	value of taking action a in state s under the optimal policy
V, V_t	array estimates of state-value function v_π or v_*
Q, Q_t	array estimates of action-value function q_π or q_*
d	dimensionality—the number of components of $\mathbf{w}$
$\mathbf{w}$	d-vector of weights underlying an approximate value function
$\hat{v}(s, \mathbf{w})$	approximate value of state s given weight vector $\mathbf{w}$
$\mu(s)$	on-policy distribution over states
$\overline{\mathrm{VE}}$	mean square value error

1 Introduction

1.1 Research Motivation

- (Lopes et al., 2011)

1.2 Research Question

1.3 Relevance

2 Theoretical Background

2.1 Electricity Markets

On electricity markets actors participate in auctions to match the supply of electricity generation and the demand for electricity consumption. Participants place asks (sale offers) and bids (purchase orders). The price is determined by an auction mechanism, which can take different forms depending on the type of market. Germany, like many other countries, has a liberalized energy system in which the generation and distribution of electricity are decoupled. Multiple electricity markets exist in a liberalized energy system. They differ in the auction design and in their reaction time between the order contract and the delivery of electricity. Day-ahead markets and spot markets have a reaction time between a day and several hours, whereas in operating reserve markets the reaction time ranges from minutes to seconds. The auction mechanism design is essential for electricity markets (Kambil & van Heck, 1998). Electricity markets work according to the merit order principle in which resources are considered in an ascending order of the energy price until the capacity demand is met. The clearing price is determined by the energy price, at the point where supply meets demand. Payment models differ in the markets: In contrast to day-ahead markets, where a uniform pricing schema is applied, in secondary reserve markets and intraday markets bidders get compensated by the price they bid (pay-as-bid principle).

Carsharing fleets can offer the capacity of their EV batteries on multiple markets at the same time to make use of the different market properties. On operating reserve markets are prices usually more volatile and consequently more attractive for VPPs (Tomić & Kempton, 2007). Operating reserve markets also bear a higher risk for the fleet: Commitments have to be made one week in advance when customer demands are still of uncertain. In order to not face penalties for unfulfilled commitments only a conservative amount of capacity can be offered to the market. On the other hand, spot markets allow participants to continuously trade electricity products up to fifteen minutes prior to delivery. At this point, it is possible to predict if an EV is likely going to be rented out with a high accuracy. This creates the possibility to trade the remaining available capacity with a low risk at the spot market. In the following, we will explain the market design of balancing markets and spot markets in more detail, since they are the markets we included in our research.

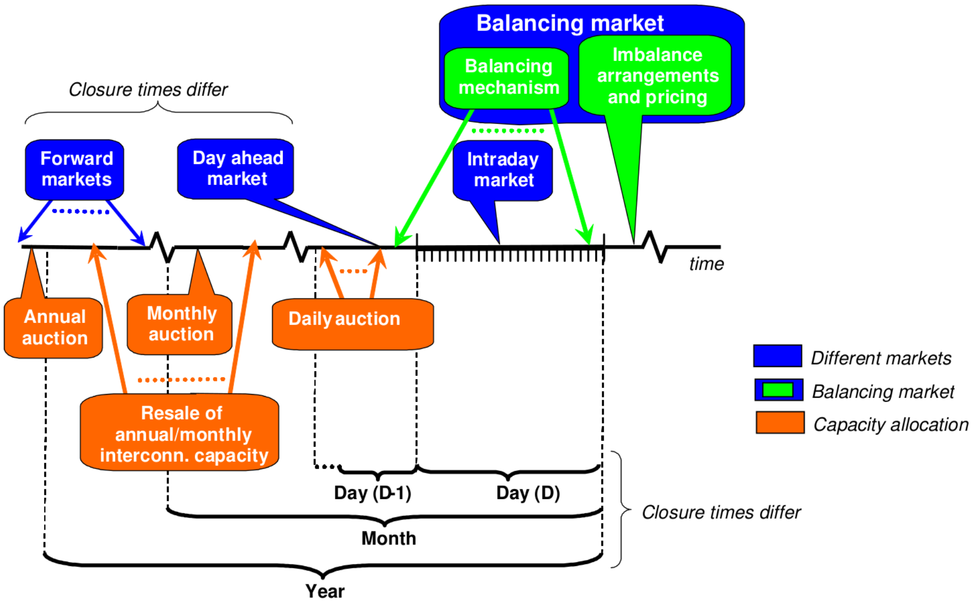

Figure 1: **Adapt own figure!** Interaction between electricity markets in relation to capacity allocation

2.1.1 Balancing Market

The balancing market is a tool to balance frequency deviations in the power grid. It offers auctions for primary control reserve, secondary control reserve as well as tertiary control reserve (minute reserve), which primarily differ in the required ramp-up times of the participants. As depicted in Figure 1, the balancing market can be seen as the last link in a chain of electricity markets (van der Veen & Hakvoort, 2016). In this study, we will look at the German Control Reserve Market (GCRM), one of the largest frequency regulation markets in the world. However, the presented concepts can be easily transferred to other balancing markets in unbundled energy systems, since the market design is similar (Brandt et al., 2017). Transmission Systems Operators (TSO) procure their required control reserve via tender auctions at the GCRM. The market conducts daily auctions for the three types of control reserve that were previously mentioned. This thesis focuses on the secondary operating reserve auction, in which participants must be able to supply or absorb a minimum of 1MW of power over a 4 hour interval with a reaction time of 30 seconds.[1] Since EV batteries can absorb energy almost instantly, when they are connected to a charging station, they are suitable to provide such balancing services. Operating reserve providers have to be qualified by the TSO to participate in the market and are able to reliably provide the committed capacity. Although EV fleets are currently not qualified by the GCRM to be used as operating reserve, they could theoretically handle the minimum capacity requirements. Around 220 EVs would need to simultaneously charge at standard 4.6kW charging stations to provide 1MW of downward regulating capacity.

Up until 28$^{\text{th}}$ July 2018, auctions were held weekly, with two different segments each week (peak hours/non-peak hours). Afterwards, the auction mechanism changed to *daily* auctions of six four-hour segments of positive and negative control reserve.[2] The shorter auction cycles facilitate the integration of renewable energy generators into the secondary control reserve market, as they are dependent on accurate (short-term) capacity forecasts.

Positive control reserve is supplied energy to the grid, which is required when the grid frequency falls below 50Hz. It can be provided by increasing electricity generation or by reducing load. On the contrary, negative control reserve is required when the grid frequency rises above 50Hz and can be provided by adding load or reducing electricity generation. Since we do not consider V2G in this thesis, the EV fleets in our model are only able provide *negative control reserve*, which we will refer to as *control reserve* until the end of the thesis. Market participants submit bids in following form to the market: (P, p_c, p_w), where P is the amount of electrical power that can be supplied on demand in kW, p_c is the capacity price for keeping the power available in €/MW and p_w is the working price for delivered energy in €/MWh. The TSO determines the target quantity of energy to acquire per timeslot, it usually acquires much higher regulation capacity to minimize risks and activates the capacity on demand. The TSO accepts the bids based on the capacity price in a merit order. Providers, whose bids were accepted, instantly get compensated for the provided capacity: $R_c = p_c \times P$. At the time regulation capacity is needed (usually a day or a week later), the TSO activates the capacity according to a merit order of the ascending *working prices* p_w. Hence, providers are also compensated according to the actual power they supplied: $R_w = p_w \times P_a$, where P_a is the supplied power at the timeslot. Since, provider get paid according to their submitted price p_w, this type of auction is a *pay-as-bid* auction.

[1] See `https://regelleistung.net/`, accessed on 15$^{\text{th}}$ February 2019, for further information on the market design and historical data.

[2] `https://www.bundesnetzagentur.de/SharedDocs/Pressemitteilungen/DE/2017/28062017_Regelenergie.html`, accessed 18$^{\text{th}}$ February, 2019

2.1.2 Spot Market

2.2 Reinforcement Learning

The following chapter will give an overview of the most important Reinforcement Learning (RL) concepts and will introduce the corresponding mathematical formulations. If not noted otherwise, the notation, equations, and insights are adopted from (Sutton & Barto, 2018), the de-facto reference book of RL research.

RL is an agent-based machine learning algorithm in which the agent learns to perform an optimal set of actions through interaction with its environment. The agents objective is to maximize the rewards it receives based on the actions it takes. Immediate rewards have to be weighted against long-term cumulative returns that also on its future actions. The RL problem is formalized as Markov Decision Processes (MDPs) which will be introduced in Chapter 2.2.1. A critical task of RL agents is to continuously estimate the value of the environments state. Values indicate the long-term desirability of a state, that is the total amount of reward the agent can expect to accumulate over the future, following a learned set of actions, called the policy. Policies and values are covered in Chapter 2.2.2, whereas the core mathematical foundations for evaluating policies and updating value functions are introduced in Chapter 2.2.3. When the model of the environment is fully known, the learning problem is reduced to a planning problem (Chapter 2.2.4) in which optimal policies can be computed with iterative approaches. Model-free RL approaches can be applied when rewards and state transitions are unknown, and the agent's behavior has to be learned from experience (Chapter 2.2.5). The last two chapter cover methods that solve the RL problem more efficiently, tackle new challenges and are widely used in practice and research.

2.2.1 Markov Decision Processes

Markov Decision Processes (MDPs) are a classical formulation of sequential decision making and an idealized mathematical formulation of the RL problem. MDPs allow to derive exact theoretical statements about the learning problem and possible solutions. Figure 2 depicts the *agent-environment interaction.*

In RL the agent and the environment continuously interact with each other. The agent takes actions that influence the environment, which in return presents rewards to the agent. The agent's goal is to maximize rewards over time, trough an optimal choice of actions. In each discrete timestep $t = 0, 1, 2, ..., T$ the RL

[3] **Figure 3.1** from "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andew G. Barto is licencsed under CC BY-NC-ND 2.0 (https://creativecommons.org/licenses/by-nc-nd/2.0/)

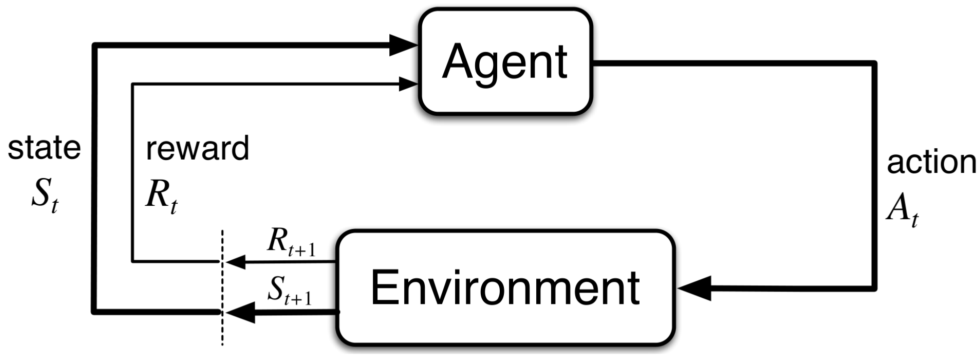

Figure 2: The agent-environment interaction in a Markov decision process (Sutton & Barto, 2018) [3]

agent interacts with the environment, which is perceived by the agent as a representation, called *state*, $S_t \in \mathcal{S}$. Based on the state, the agents selects an *action*, $A_t \in \mathcal{A}$, and receives a numerical *reward* signal, $R_{t+1} \in \mathcal{R} \subset \mathbb{R}$, in the next timestep. Actions influence immediate rewards and successive states, and consequently also influence future rewards. The agent has to continuously make a trade-off between immediate rewards and delayed rewards to achieve its long-term goal.

The *dynamics* of a MDP are defined by the probability that a state $s' \in \mathcal{S}$ and a reward $r \in \mathcal{R}$ occurs, given the preceding state $s \in \mathcal{S}$ and action $a \in \mathcal{A}$. In *finite* MDPs, the random variables $\mathcal{R}_t$ and S_t have well-defined probability density functions (PDF), which are solely dependent on the previous state and action. Consequently, it is possible to define ($\doteq$) the *dynamics* of the MDP as following:

$$p(s', r|s, a) \doteq \Pr\{S_t = s', R_t = r|S_{t-1} = s, A_t = a\}, \tag{1}$$

for all $s', s \in \mathcal{S}$, $r \in \mathcal{R}$ and $a \in \mathcal{A}$. Note that each possible value of the state $\mathcal{S}_t$ depends only on the immediately preceding state $\mathcal{S}_{t-1}$. When a state includes all information of *all* previous states, the state possesses the so-called *Markov property*. If not noted otherwise, the Markov property is assumed throughout the whole chapter. The dynamics function allows computing the *state-transition probabilities*, another important characteristic of an MDP, as following:

$$p(s'|s, a) \doteq \Pr\{S_t = s'|S_{t-1} = s, A_t = a\} = \sum_{r \in \mathcal{R}} p(s', r|s, a), \tag{2}$$

for $s', s \in \mathcal{S}$, $r \in \mathcal{R}$ and $a \in \mathcal{A}$.

The use of a *reward signal* R_t to formalize the agent's goal is a unique characteristic of RL. Each timestep the agent receives the rewards as a scalar value $\mathcal{R}_t \in \mathbb{R}$. The sole purpose of the RL agent is to maximize the long-term cumulative reward (as opposed to the immediate reward). The long-term cumulative reward can also be expressed as the *expected return* G_t:

$$\begin{aligned} G_t &\doteq R_{t+1} + \gamma R_{t+2} + \gamma R_{t+3} + \cdots \\ &= \sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \\ &= R_{t+1} + \gamma G_{t+1}, \end{aligned} \tag{3}$$

where γ, $0 \leq \gamma \leq 1$, is the *discount rate* parameter. The discount rate determines how "myopic" the agent is. If γ approaches 0, the agent is more concerned with maximizing immediate rewards. On the contrary, when $\gamma = 1$, the agent takes future rewards strongly into account, the agent is "farsighted".

2.2.2 Policies and Value Functions

An essential task of almost every RL agent is estimating *value functions.* These functions describe how "good" it is to be in a given state, or how "good" it is to perform an action in a given state. More formally, they take a state s or a state-action pair s, a as input and give the expected return G_t as output. The expected return is dependent on the actions the agent will take in the future. Consequently, value functions are formulated with respect to a *policy* π. A policy is a mapping of states to actions; it describes the probability that an agent performs a certain action, based on the current state. More formally, the policy is defined as $\pi(a|s) \doteq \Pr\{A_t = a | S_t = s\}$, a PDF of all $a \in \mathcal{A}$ for each $s \in \mathcal{S}$. RL approaches mainly differ in how the policy is updated, based on the agent's interaction with the environment.

In RL, value functions of states and value functions of state-action pairs are used. The *state-value function of policy* π is denoted as $v_\pi(s)$ and is defined as the expected return when starting in s and following policy π:

$$v_\pi(s) \doteq \mathbb{E}_\pi[G_t | S_t = s], \text{ for all } s \in \mathcal{S} \tag{4}$$

The *action-value function of policy* π is denoted as $q_\pi(s, a)$ and is defined as the expected return when starting in s, taking action a and following policy π afterwards:

$$q_\pi(s, a) \doteq \mathbb{E}_\pi[G_t | S_t = s, A_t = a], \text{ for all } a \in \mathcal{A}, s \in \mathcal{S} \tag{5}$$

The *optimal policy* π_* has a greater (or equal) expected return than all other policies. The *optimal* state-value function and *optimal* action-value function are defined as follows:

$$v_*(s) \doteq \max_a v_\pi(s), \text{ for all } s \in \mathcal{S} \tag{6}$$

$$q_*(s,a) \doteq \max_a q_\pi(s), \text{ for all } s \in \mathcal{S}, a \in \mathcal{A} \tag{7}$$

The *optimal* action-value function describes the expected return when taking action a in state s following the optimal policy π_* afterwards. Estimating q_* to obtain an optimal policy is a substantial part of RL and has been known as *Q-learning* (Watkins & Dayan, 1992), which is described in Chapter 2.2.5.

2.2.3 Bellman Equations

A central characteristic of value functions is the recursive relationship between the values. Similar to Equation (3), current values are related to expected values of successive states. This relationship is heavily used in RL and has been formulated as *Bellman equations* (Bellman, 1957). The Bellman equation for $v_\pi(s)$ is defined as follows:

$$\begin{aligned} v_\pi(s) &\doteq \mathbb{E}_\pi[G_t|S_t = s] \\ &= \mathbb{E}_\pi[R_{t+1} + \gamma G_{t+1}|S_t = s] \\ &= \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)\Big[r + \gamma v_\pi(s')\Big], \end{aligned} \tag{8}$$

where $a \in \mathcal{A},\ s, s' \in \mathcal{S},\ r \in \mathcal{R}$. In other words, the value of a state equals the immediate reward plus the expected value of all possible successor states, weighted by their probability of occurring. $v_\pi(s)$ is the only solution to its Bellman equation. The Bellman equation of the optimal value function v_* is called the *Bellman optimality equation*:

$$\begin{aligned} v_*(s) &= \max_{a \in \mathcal{A}(s)} q_{\pi_*}(s,a) \\ &= \max_a \mathbb{E}_{\pi_*}[R_{t+1} + \gamma G_{t+1}|S_t = s, A_t = a] \\ &= \max_a \mathbb{E}_{\pi_*}[R_{t+1} + \gamma v_*(S_{t+1})|S_t = s, A_t = a] \\ &= \max_a \sum_{s',r} p(s',r|s,a)\Big[r + \gamma v_*(s')\Big] \end{aligned} \tag{9}$$

where $a \in \mathcal{A},\ s, s' \in \mathcal{S},\ r \in \mathcal{R}$. In other words, the value of a state under an optimal policy equals the expected return for the best action from that state. Note that the Bellman optimality equation does not refer to a specific policy, it has a unique solution independent from one. It can be seen as an equation system, which can be solved when the dynamics of the environment p are known. Similar Bellman equations to Equations (8) and (9) can also be formed for $q_\pi(s,a)$ and $q_*(s,a)$. Bellman equations form the basis for computing and approximating value functions and were an important milestone in RL history. Most RL methods are *approximately* solving the Bellman optimality equation, by using experienced state transitions instead of expected transition probabilities. The most common methods will be explored in the following chapters.

2.2.4 Dynamic Programming

Dynamic Programming (DP) is a method to compute optimal policies, the primary goal of every RL method. DP makes use of value functions to facilitate the search for good policies. Once an optimal value function, (i.e., one that satisfies the Bellman optimality equation) is found, optimal policies can be easily obtained. Despite the limited utility of DP in real-world settings, it provides the theoretical foundation for all other RL methods. In fact, all of the RL methods try to achieve the same goal, but without the assumption of a perfect model of the environment and less computational effort. Because DP assumes full knowledge of the environment, it is known as *planning*, in which optimal solutions are *computed*. In *control* problems (Chapter 2.2.5), optimal solutions are *learned* from an unknown environment.

The two most popular DP algorithms that compute optimal policies are called *policy iteration* and *value iteration*. These methods perform "sweeps" through the whole state set and update the estimated value of each state via an *expected update* operation. In policy iteration, a value function for a given policy v_π needs to be computed first, a step called *policy evaluation*. A sequence of approximated value functions $\{v_k\}$ are updated using the Bellman equation for v_π (Eq. 8) until convergence to v_π is achieved. After computing the value function for a given policy, it is possible to modify the policy and see if the value $v_\pi(s)$ for a given state increases (*policy improvement*). A way of doing this, is evaluating the action-value function $q_\pi(s,a)$ by *greedily* taking the best short-term action $a \in A$ at a given timestep. Alternating between these two steps monotonically improves the policies and the value functions until they converge to the optimum. This algorithm is called *policy iteration*:

$$\pi_0 \xrightarrow{\text{E}} v_{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} v_{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \dots \xrightarrow{\text{I}} \pi_* \xrightarrow{\text{E}} v_*, \tag{10}$$

where $\xrightarrow{\text{E}}$ denotes a policy evaluation step, $\xrightarrow{\text{I}}$ denotes a policy improvement step. π_* and v_* are the optimal policy and optimal value function, respectively. Note that in each iteration of the policy iteration algorithm, a policy evaluation has to be performed, which requires multiple sweeps through the state space. In *value iteration*, the policy evaluation step is stopped after one sweep. In this case, the two previous steps can be combined into one single update step:

$$\begin{aligned} v_{k+1}(s) &\doteq \max_a \mathbb{E}[R_{t+1} + \gamma v_*(S_{t+1}) | S_t = s, A_t = a] \\ &= \max_a \sum_{s',r} p(s', r | s, a) \Big[r + \gamma v_k(s') \Big], \end{aligned} \tag{11}$$

where $a \in \mathcal{A}$, $s, s' \in \mathcal{S}$, $r \in \mathcal{R}$. It can be shown, that for any given v_0, the sequence v_k converges to the optimal value function v_*. In value iteration, the Bellman optimality equation (9) is simply turned into an update rule. Both of the algorithms can be effectively used to compute optimal values and value function in finite MDPs with a perfect model of the environment.

2.2.5 Temporal-Difference Learning

The previous chapter dealt with solving a *planning* problem, that is computing an optimal solution (i.e., an optimal policy π_*) of an MDP when a perfect model of the environment is known. In the following chapters, we will look at *model-free* prediction and *model-free* control. As opposed to planning, model-free methods learn from experience and require no prior knowledge of the environment. Remarkably, these methods can still achieve optimal behavior.

The *TD prediction problem* is concerned with estimating state-values v_π using past experiences of following a given policy π. TD methods update an estimate V of v_π in every timestep. At time $t+1$ they immediately perform an update operation on $V(S_t)$. Because of the step-by-step nature of TD learning, it is categorized as *online learning*. Also note that TD methods perform update operations on value estimates based on other learned estimates, a procedure called *bootstrapping*. In simple TD prediction, the value estimates V are updated as following:

$$V(S_t) \leftarrow V(S_t) + \alpha \big[R_{t+1} + \gamma V(S_{t+1}) - V(S_t) \big], \tag{12}$$

where α is a constant *step-size* parameter and γ is the *discount rate*. Here, the update of the state-value is performed using the observed reward R_{t+1} and the estimated value $V(S_{t+1})$.

When a model is not available, it is useful to estimate *action-values*, instead of *state-values*. If the environment is completely known, it is possible for the agent to look one step ahead and select the best action. Without that knowledge, the value of each action in a given state needs to be estimated. The latter constitutes a problem, since not every *state-action* pair will be visited when the agent follows a deterministic policy. A deterministic policy $\pi(a|s)$ returns exactly one action given the current state, hence the agent will only observe returns for one of the actions. In order to evaluate the value function for all *state-action* pairs q_π, continuous *exploration* needs to be ensured. In other words, the agent has to explore state-action pairs which are seemingly disadvantageous given the current policy. This dilemma is also known as the *exploration-exploitation* trade-off. One way to achieve exploration is using *stochastic* policies for the action selection. Stochastic policies have a non-zero probability of selecting each action in each state. A typical stochastic policy is the ϵ-*greedy policy*, which selects the action with the highest estimated value, except for a probability ϵ, it selects an action at random.

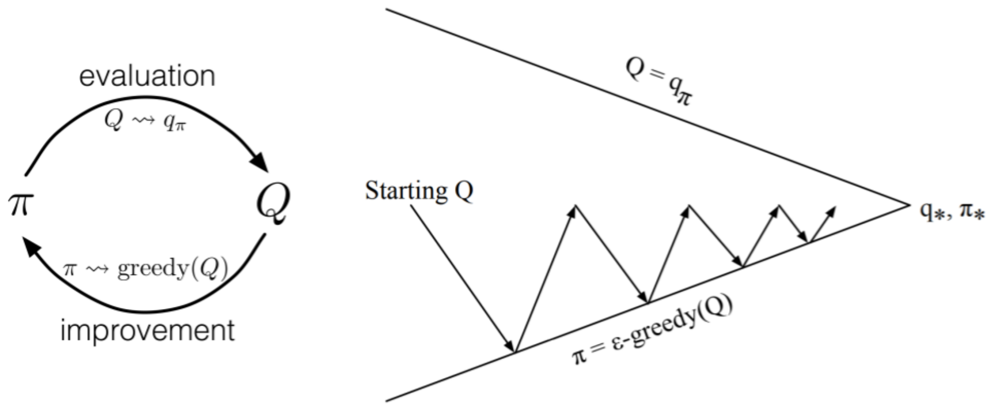

Figure 3: On-policy control with Sarsa (Sutton & Barto, 2018). [4]

There are two approaches to make use of stochastic policies to ensure all actions are chosen infinitely often. On-policy methods improve the (stochastic) decision policy, by continually estimating q_π in regard to π, while simultaneously driving π towards q_π, e.g., with a ϵ-greedy action selection. Figure 3 depicts this learning process. Off-policy methods improve the deterministic decision policy, by using a second stochastic policy to generate behavior. The first policy is becoming the optimal policy by evaluating the exploratory behavior of the second policy. Off-policy approaches are considered more powerful than on-policy approaches and have a variety of additional use cases. On the other side, they often have a higher variance and take more time to converge to an optimum.

A popular on-policy TD control method is Sarsa, developed by Rummery and Niranjan (1994). In the prediction step, the action-value function $q_\pi(s, a)$ of all actions and states has to be estimated for the current policy π. The estimation can be done similar to TD prediction of state values (Eq. 12). Instead of considering state transitions, state-action transitions are considered in this case. The update

[4] The in-text figure of **Chapter 5.3** from "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andew G. Barto is licencsed under CC BY-NC-ND 2.0 (https://creativecommons.org/licenses/by-nc-nd/2.0/)

rule is constructed as follows:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\big[R_{t+1} + \gamma Q(S_{t+1}, A_{t+1}) - Q(S_t, A_t)\big] \tag{13}$$

After every transition from a state S_t, an update operation using the events $(S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1})$ is performed. This quintuple also constituted the name Sarsa. The on-policy control step of the algorithm is straightforward, and uses a ϵ-greedy policy improvement, as described in the previous paragraph. It has been shown that Sarsa converges to the optimal policy π_* under the assumption of infinite visits to all state-action pairs.

A breakthrough in RL has been achieved when Watkins and Dayan (1992) developed the *off-policy* TD control algorithm, called Q-learning. The update rule is defined as follows:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha\big[R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)\big] \tag{14}$$

Here, the estimated action-values Q are updated towards the highest estimated action-value of the next time step. In this way, Q directly approximates the optimal action-value function q_*, independently of the policy the agent follows. Due to this simplification, Q-learning is a widely used model-free method, and its convergence can be proved easily.

This chapter covered the most important RL methods. They work online, learn from experience, and can be easily applied to real-world problems with low computational effort. Moreover, the mathematical complexity of the presented approaches is limited, and they can be easily implemented into computer programs. Temporal-Difference learning is a *tabular* method, in which Q-values are stored and updated in a lookup table. If the state and action spaces are continuous or the number of states and actions is very large, a table representation is computational infeasible and the speed of convergence is drastically reduced. In this case, a *function approximator* can replace the lookup table. The next chapter will briefly cover function approximation, as well as other advancements in RL.

2.2.6 Approximation Methods

Up to this point, only tabular RL methods have been covered, which form the theoretical foundation of RL in general. But in many real-world use cases, the state space is enormous and it is improbable to find an optimal value function with tabular methods. Not only is it a problem to store such a large table in the memory, but also would it take an almost infinite amount of time to fill every entry with meaningful results. Contrarily, *function approximation* tries to find a function that approximates the optimal value function as closely as possible, with limited computational resources. The experience with a small subset of visited states is generalized to approximate values of the whole state set. Function approximation has been widely studied in supervised machine learning: Gradient methods, as well as linear and non-linear models have shown good results for RL.

The approximated value of a state s is denoted as the parameterized functional form $\hat{v}(s, \mathbf{w}) \approx v_\pi(s)$, given a weight vector $\mathbf{w} \in \mathbb{R}^d$. Function approximation methods are approximating v_π by learning (i.e., adjusting) the weight vector $\mathbf{w}$ from the experience of following the policy π. By assumption, the dimensionality d of $\mathbf{w}$ is much lower than the number of states, which is the reason for the desired generalization effect: Adjusting one weight affects the values of many states. However, optimizing an estimate for one state negatively affects the accuracy of the estimates for other states. This effect motivates the specification of a state distribution $\mu(s)$, which represents the importance of the prediction error for each state. In on-policy prediction, $\mu(s)$ is often selected to be proportion of time spend in each state s. The prediction error of a state is defined as the squared difference between the predicted (i.e., approximated) value $\hat{v}(s, \mathbf{w})$ and the true value $v_\pi(s)$. Consequently, the objective function of the supervised learning problem can be defined as the *Mean Squared Value Error* $\overline{\mathrm{VE}}$, which weights the prediction error with the state distribution $\mu(s)$:

$$\overline{\mathrm{VE}}(\mathbf{w}) \doteq \sum_{s \in \mathcal{S}} \mu(s) \Big[v_\pi(s) - \hat{v}(s, \mathbf{w}) \Big]^2, \text{ where } \mathbf{w} \in \mathbb{R}^d \tag{15}$$

Minimizing $\overline{\mathrm{VE}}$ in respect to $\hat{v}$ will yield a value function, which facilitates finding a better policy — the primary goal of RL. Remember that $\hat{v}$ can take any form of a linear or non-linear function of the state s.

In practice, deep artificial neural networks networks (ANNs) have shown great success as function approximators, which coined the term *Deep Reinforcement Learning* (Mnih et al., 2015; Silver et al., 2016). A simple feedforward ANN can be found in Figure 4. ANNs have the advantage that they can theoretically approximate any continuous function by adjusting the connection weights of the network $\mathbf{w} \in \mathbb{R}^{d \times d}$ (Cybenko, 1989). Advancements in the field of *Deep Learning* facilitated remarkable performance improvements in RL applications. Despite that, the RL theory is mostly limited to tabular and linear approximation methods. Refer to Bengio (2009) for a comprehensive review of deep learning methods.

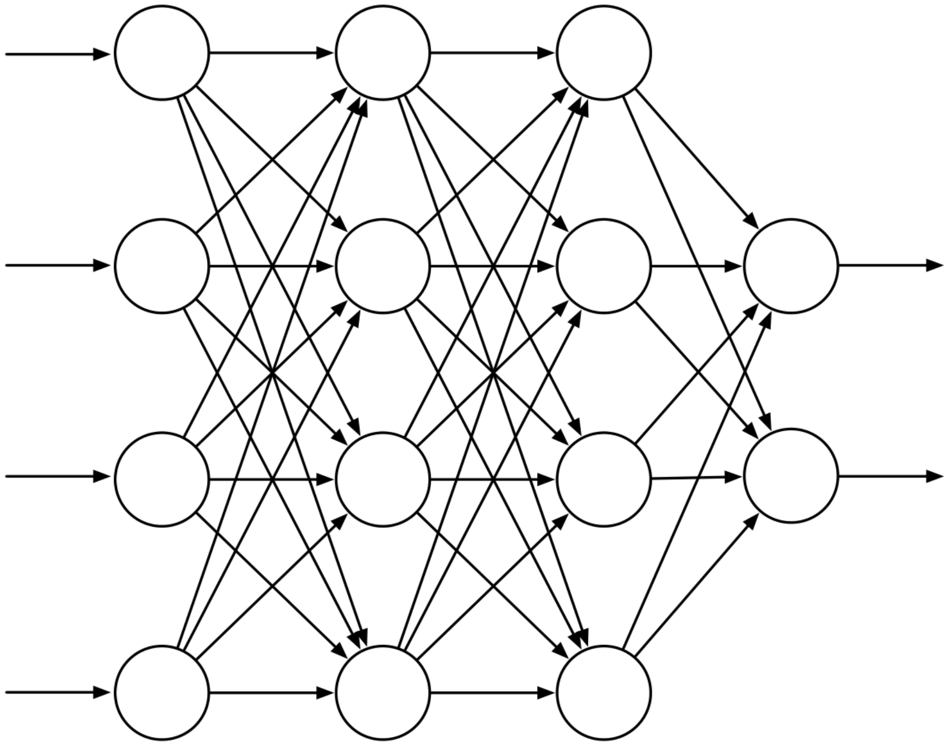

Figure 4: A sample ANN consisting of four input nodes, two fully connected hidden layers and two output nodes (Sutton & Barto, 2018). [5]

2.2.7 Further Topics

The previous chapters provided a detailed overview of the most important concepts and mathematical foundations in RL. In the research there are many more topics that were not covered here. *Eligibility traces* offer a way to more general learning and faster convergence rates. Almost any TD method can be extended to use eligibility traces, a popular methods is called Watkins's Q(λ) (Watkins, 1989). *Fitted-Q Iteration* (Ernst et al., 2003) combined Q-learning and fitted value iteration with batch-mode RL. In batch-mode the whole dataset is available offline, contrary to online RL where the data is acquired by the agent's action in its the environment. *Actor-critic* methods (Sutton, 1984) directly learn a parameterized policy instead of action-values, which inherently allow continuous state spaces and learning appropriate levels of exploration. Simultaneously to learning the policy, they approximate a state-value function, which serves as a "critic" to the learned policy, the "actor". In the current theory most RL models are single-agent models. For certain real-world applications multi-agent RL algorithms are necessary to coordinate interaction between the agents. When multiple learning agents interact with a non-stationary environment, convergence and stability are

[5]**Figure 9.14** from "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andew G. Barto is licencsed under CC BY-NC-ND 2.0 (https://creativecommons.org/licenses/by-nc-nd/2.0/)

a serious problem. *W-learning* (Humphrys, 1996) is an multi-agent approach that aims to solve these difficulties.

3 Related Literature

3.1 Smart Charging and Balancing the Electric Grid with EV Fleets

The increasing penetration of EVs has a substantial effect on electricity consumption patterns. During charging periods, power flows and grid losses increase considerably and challenge the grid. Operators have to reinforce the grid to ensure that transformers and substations do not overload (Sioshansi, 2012; Lopes et al., 2011). Loading multiple EVs in the same neighborhood, or worse, whole EV fleets at once, stress the grid. In these cases, even brown- or blackouts can occur. (Kim et al., 2012). Despite these challenges, it is possible to support the physical reinforcement by adopting smart charging strategies. In smart charging, EVs get charged when the grid is less congested to ensure grid stability. Smart charging reduces peaks in electricity demand, called *Peak Cutting*, and complement the grid in times of low demand, called *Valley Filling*. Smart charging has been researched thoroughly in the IS literature, in the following we will outline some of the most important contributions.

Valogianni et al. (2014) found that using intelligent agents to schedule EV charging substantially reshapes the energy demand and reduces peak demand without violating individual household preferences. Moreover, they showed that the proposed smart charging behavior reduces average energy prices and thus benefit households economically. In another study, Kara et al. (2015) investigated the effect of smart charging on public charging stations in California. Controlling for arrival and departure times, the authors presented beneficial results for the distribution system operator (DSO) and the owners of EVs. Their approach resulted in a price reduction in energy bills and a peak load reduction. An extension of the smart charging concept is Vehicle-to-Grid (V2G). When equipped with V2G devices, EVs can discharge their batteries back into the grid. Existing research has focused on this technology in respect to grid stabilization effects and arbitrage possibilities. For instance, Schill (2011) showed that the usage of EVs can decrease average consumer electricity prices. Excess EV battery capacity can be used to charge in off-peak hours and discharge in peak hours, when the prices are higher. These arbitrage possibilities reverse welfare effects of generators and increase the overall welfare and consumer surplus. Tomić and Kempton (2007) found that the arbitrage opportunities are especially prominent when a high variability in electricity prices on the target electricity market exists. The authors stated that short intervals between the contract of sale and the physical delivery of electricity increase arbitrage benefits. Consequently, ancillary service markets, like frequency control and operating reserve markets, are attractive for smart charging.

Peterson et al. (2010) investigated energy arbitrage profitability with V2G in the light of battery depreciation effects in the US. The results of their study indicate that large-scale use of EV batteries for grid storage does not yield enough monetary benefits to incentivize EV owners to participate in V2G activities. Considering battery depreciation cost, the authors arrived at an annual profit of only 6$ - 72$ per EV. Brandt et al. (2017) evaluated a business model for parking garage operators operating on the German frequency regulation market. When taking infrastructure costs and battery depreciation costs into account, they conclude that the proposed vehicle-grid integration is not profitable. Even with generous assumptions about EV adoption rates in Germany and altered auction mechanisms, the authors arrived at negative profits. Kahlen et al. (2017) used EV fleets to offer balancing services to the grid. Evaluating the impact of V2G in their model, the authors conclude that V2G would only be profitable if reserve power prices were twice as high. Considering the results from the studies mentioned above, our research does not include V2G, since only marginal profits are expected.

In order to maximize profits, it is essential for market participants to develop successful bidding strategies. Several authors have investigated bidding strategies to jointly participate in multiple markets (Mashhour & Moghaddas-Tafreshi, 2011; He et al., 2016). Mashhour and Moghaddas-Tafreshi (2011) used stationary battery storage to participate in the spinning reserve market and the day-ahead market at the same time. The authors developed a non-equilibrium model, which solves the presented mixed-integer program with Genetic Programming (GP). Contrarily, we use a model-free RL agent that learns an optimal policy (i.e., a trading strategy) from actions it takes in the environment (i.e., bidding on electricity markets). Using a model-free approach is especially beneficial for us, since additional unknown variables and constraints (i.e., customer mobility demand) complicate the formulation of a mathematical model.

He et al. (2016) conducted similar research to Mashhour and Moghaddas-Tafreshi (2011). The authors additionally incorporated battery life cycle in their profit maximization model, which proved to be a decisive factor. In contrast to the authors, we jointly participated in the secondary operating reserve and spot market with the *non-stationary* storage of EV batteries. Because shared EVs have to satisfy mobility demand, they have to be charged in any case, which allows us to safely exclude battery depreciation from our model. Further, we chose the intraday market over the day-ahead market, as it has the lowest reaction time among the spot markets, and thus potentially offers higher profits (Tomić & Kempton, 2007).

Previous studies often assume that car owners or households can directly trade on electricity markets. In reality, this is not possible due to the minimum capacity requirements of the markets, requirements that single EVs do not meet. For example, the German Control Reserve Market (GCRM) has a minimum trading capacity of 1MW to 5MW, depending on the specific market. In order to reach the minimum capacity, over 200 EVs would need to be connected to the grid via a standard 4.6kW charging station at the same time. Ketter et al. (2013) introduced the notion of electricity brokers, aggregators that act on behalf of a group of individuals or households to participate in electricity markets. Brandt et al. (2017) and Kahlen et al. (2014) successfully showed that electricity brokers can overcome the capacity issues by aggregating EV batteries. In addition to electricity brokers, we apply the concept of Virtual Power Plants (VPPs). VPPs are flexible portfolios of distributed energy resources, which are presented with a single load profile to the system operator, making them eligible for market participation and ancillary service provisioning (Pudjianto et al., 2007). Hence, VPPs allow providing regulation capacity to the market without knowing which exact sources provide the promised capacity until the delivery time (Kahlen et al., 2017). This concept is specially useful when dealing with EV fleets: VPPs enable carsharing providers to issue bids and asks based on an estimate of available fleet capacity, without knowing beforehand which exact EVs will provide the capacity at the time of delivery. Based on the battery charge and the availability of EVs, an intelligent agent decides in real-time which vehicles provide the capacity.

Centrally managed EV fleets make it possible for carsharing providers to use the presented concepts as a viable business extension. Free float carsharing is a popular concept that allows cars to be picked up and parked everywhere, and the customers are billed is by the minute. Free float carsharing offers flexibility to its users, saves resources, and reduces carbon emissions (Firnkorn & Müller, 2015). Most previous studies concerned with the usage of EVs for electricity trading, assumed that trips are fixed and known in advance, e.g., in Tomić and Kempton (2007). The free float concept adds uncertainty and nondeterministic behavior, which make predictions about future rentals a complex issue.

Kahlen et al. (2017) showed that it is possible to use free float carsharing fleets as VPPs to profitably offer balancing services to the grid. In their study, the authors compared cases from three different cities across Europe and the US. They used an event-based simulation, bootstrapped with real-world carsharing and secondary operating reserve market data from the respective cities. A central dilemma within their research was to decide whether an EV should be committed to a VPP or free for rent. Since rental profits are considerably higher than profits from electricity trading, it is crucial not to allocate an EV to a VPP when it could have been rented out otherwise. To deal with the asymmetric payoff, Kahlen et al. used stratified sampling in their classifier. This method gives rental misclassifications higher weights, reducing the likelihood of EVs to participate in VPP activities. The authors used a Random Forest regression model to predict the available balancing capacity on an aggregated fleet level. Only at the delivery time, the agent decides which individual EVs provide the regulation capacity. This heuristic is based on the likelihood that the vehicle is rented out and on its expected rental benefits.

In a similar study, the authors showed that carsharing companies can participate in day-ahead markets for arbitrage purposes (Kahlen et al., 2018). In the paper, the authors used a sinusoidal time-series model to predict the available trading capacity. Another central problem for carsharing providers is that committed trades, which can not be fulfilled, result in substantial penalties from the system operator or electricity exchange. In other words, fleet operators have to avoid buying any amount of electricity, which they can't be sure to charge with available EVs at the delivery time. To address this issue, the authors developed a mean asymmetric weighted (MAW) objective function. They used it for their time-series based prediction model, to penalize committing an EV to VPP when it would have been rented out otherwise. Because of the two issues mentioned above, Kahlen et al. (2018) could only make very conservative estimations and commitments of overall available trading capacity, resulting in a high amount of foregone profits. This effect is especially prominent when participating in the secondary operating reserve market, since commitments have to be made one week in advance when mobility demands are still uncertain. Kahlen et al. (2017) stated that in 42% to 80% of the cases, EVs are *not* committed to a VPP when it would have been profitable to do so.

This thesis proposes a solution in which the EV fleet participates in the balancing market and intraday market simultaneously. With this approach, we align the potentially higher profits on the balancing markets, with more accurate capacity predictions for intraday markets (Tomić & Kempton, 2007). This research followed Kahlen et al. (2017), who proposed to work on a combination of multiple markets in the future.

3.2 Reinforcement Learning in Smart Grids

Previous research shows that intelligent agents equipped with Reinforcement Learning (RL) methods can successfully take action in the smart grid. The following chapter outlines different research approaches of RL in the domain of smart grids. For a more thorough description, mathematical formulations and common issues, of RL refer to Chapter 2.2.

Reddy and Veloso (2011a, 2011b) used autonomous broker agents to buy and sell electricity from DER on a proposed *Tariff Market*. The agents use Markov Decision Processes (MDPs) and RL to learn pricing strategies to profitably participate in the Tariff Market. To control for a large number of possible states in the domain, the authors used *Q-Learning* with derived state space features. Based on descriptive statistics, they defined derived price and market participant features. By engaging with its environment, the agent learns an optional sequence of actions (policy) based on the state of the agent. Peters et al. (2013) built on that work and further enhanced the method by using function approximation. Function approximation allows to efficiently learn strategies over large state spaces, by deriving a function that describes the states instead of defining discrete states. By using this technique, the agent can adapt to arbitrary economic signals from its environment, resulting in better performance than previous approaches. Moreover, the authors applied feature selection and regularization methods to explore the agent's adaption to the environment. These methods are particularly beneficial in smart markets because market design, structures, and conditions might change in the future. Hence, intelligent agents should be able to adapt to it (Peters et al., 2013).

Vandael et al. (2015) facilitated learned EV fleet charging behavior to optimally purchase electricity on the day-ahead market. Similarly to Kahlen et al. (2018), the problem is framed from the viewpoint of an aggregator that tries to define a cost-effective day-ahead charging plan in the absence of knowing EV charging parameters, such as departure time. A crucial point of the study is weighting low charging prices against costs that have to be paid when an excessive or insufficient amount of electricity is bought from the market (imbalance costs). Contrarily, Kahlen et al. (2018) did not consider imbalance cost in their model and avoid them by sacrificing customer mobility in order to balance the market (i.e., not showing the EV available for rent, when it is providing balancing capacity). Vandael et al. (2015) used a *fitted Q Iteration* to control for continuous variables in their state and action space. In order to achieve fast convergence, they additionally optimized the *temperature step* parameter of the Boltzmann exploration probability.

Dusparic et al. (2013) proposed a multi-agent approach for residential demand response. The authors investigated a setting in which 9 EVs were connected to the same transformer. The RL agents learned to charge at minimal costs, without overloading the transformer. Dusparic et al. (2013) utilized *W-Learning* to simultaneously learn multiple policies (i.e., objectives such as ensuring minimum battery charged or ensuring charging at low costs). Taylor et al. (2014) extended this research by employing Transfer Learning and *Distributed W-Learning* to achieve communication between the learning processes of the agents in a multi-objective, multi-agent setting. Dauer et al. (2013) proposed a market-based EV fleet charging solution. The authors introduced a double-auction call market where agents trade the available transformer capacity, complying with the minimum required State of Charge (SoC). The participating EV agents autonomously learn their bidding strategy with standard *Q-Learning* and discrete state and action spaces.

Di Giorgio et al. (2013) presented a multi-agent solution to minimize charging costs of EVs, a solution that requires neither prior knowledge of electricity prices nor future price predictions. Similar to Dauer et al. (2013), the authors employed standard *Q-Learning* and the ϵ-greedy approach for action selection. Vaya et al. (2014) also proposed a multi-agent approach, in which the individual EVs are agents that actively place bids in the spot market. Again, the agents use *Q-Learning*, with an ϵ-greedy policy to learn their optimal bidding strategy. The latter relies on the agents willingness-to-pay which depends on the urgency to charge. State variables, such as SoC, time of departure and price development on the market, determine the urgency to charge. The authors compared this approach with a centralized aggregator-based approach that they developed in another paper (Vaya & Andersson, 2015). Compared to the centralized approach, in which the aggregator manages charging and places bids for the whole fleet, the multi-agent approach causes slightly higher costs but solves scalability and privacy problems.

Shi and Wong (2011) consider a V2G control problem, while assuming real-time pricing. The authors proposed an online learning algorithm which they modeled as a discrete-time MDP and solved through *Q-Learning*. The algorithm controls the V2G actions of the EV and can react to real-time price signals of the market. In this single-agent approach, the action space compromises only charging, discharging and regulation actions. The limited action spaces makes it relatively easy to learn an optimal policy. Chis et al. (2016) looked at reducing the costs of charging for a single EV using known day-ahead prices and predicted next-day prices. A Bayesian ANN was employed for prediction and *fitted Q-Learning* was used to learn daily charging levels. In their research, the authors used function approximation and batch reinforcement learning, an offline, model-free learning method. Ko et al. (2018) proposed a centralized controller for managing V2G activities in multiple microgrids. The proposed method considers mobility and electricity demands of microgrids, as well as SoC of the EVs. The authors formulated a MDP with discrete state and action spaces and use standard *Q-Learning* with ϵ-greedy policy to derive an optimal charging policy. The approach takes microgrid autonomy and electricity prices into special consideration.

It should be noted that advanced RL methods and techniques are not the only solutions for problems in the smart grid, often basic algorithms and heuristics provide satisfactory results (Vázquez-Canteli & Nagy, 2019). Despite that, our paper considers RL as an optimal fit for the design of our proposed intelligent agent. Given the ability to learn user behavior (e.g., mobility demand) and the flexibility to adapt to the environment (e.g., electricity prices), RL methods are a promising way of solving complex challenges in smart grids.

4 Empirical Setting

4.1 Carsharing Fleets of Electric Vehicles

4.2 Electricity Markets Data

4.2.1 Secondary Operating Reserve Market

5 Model: FleetRL

5.1 Information Assumptions

5.2 Mobility Demand & Clearing Price Prediction

5.3 Reinforcement Learning Approach

5.4 Bidding Strategy

5.5 Dispatch Heuristic / Algorithm

6 Evaluation

6.1 Event-based Simulation

6.2 Benchmark: Ad-hoc Strategies

6.3 FleetRL

6.4 Sensitivity Analysis: Prediction Accuracy

6.5 Sensitivity Analysis: Infrastructure Changes

6.6 Sensitivity Analysis: Bidding Strategy

7 Discussion

7.1 Generalizability

7.2 Future Electricity Landscape

7.3 Limitations

8 Conclusion

8.1 Contribution

8.2 Future Research

References

Bellman, R. E. (1957). *Dynamic Programming.* Courier Dover Publications.

Bengio, Y. (2009). Learning deep architectures for ai. *Foundations and Trends® in Machine Learning*. Retrieved from https://doi.org/10.1561/2200000006 doi: 10.1561/2200000006

Brandt, T., Wagner, S., & Neumann, D. (2017). Evaluating a Business Model for Vehicle-Grid Integration: Evidence From Germany. *Transportation Research Part D: Transport and Environment*, 488-504. Retrieved from https://doi.org/10.1016/j.trd.2016.11.017 doi: 10.1016/j.trd.2016.11.017

Chis, A., Lunden, J., & Koivunen, V. (2016). Reinforcement Learning-Based Plug-In Electric Vehicle Charging With Forecasted Price. *IEEE Transactions on Vehicular Technology*. Retrieved from https://doi.org/10.1109/tvt.2016.2603536 doi: 10.1109/tvt.2016.2603536

Cybenko, G. (1989). Approximation By Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems*. Retrieved from https://doi.org/10.1007/bf02551274 doi: 10.1007/bf02551274

Dauer, D., Flath, C. M., Strohle, P., & Weinhardt, C. (2013). Market-Based EV Charging Coordination. In *Ieee/wic/acm international joint conferences on web intelligence (wi) and intelligent agent technologies (iat).* Retrieved from https://doi.org/10.1109/wi-iat.2013.97 doi: 10.1109/wi-iat.2013.97

Di Giorgio, A., Liberati, F., & Pietrabissa, A. (2013). On-board stochastic control of Electric Vehicle recharging. In *52nd ieee conference on decision and control.* Retrieved from https://doi.org/10.1109/cdc.2013.6760789 doi: 10.1109/cdc.2013.6760789

Dusparic, I., Harris, C., Marinescu, A., Cahill, V., & Clarke, S. (2013). Multi-agent residential demand response based on load forecasting. In *IEEE Conference on Technologies for Sustainability (SusTech).* Retrieved from https://doi.org/10.1109/sustech.2013.6617303 doi: 10.1109/sustech.2013.6617303

Ernst, D., Geurts, P., & Wehenkel, L. (2003). Iteratively extending time horizon reinforcement learning. In *European conference on machine learning.*

Firnkorn, J., & Müller, M. (2015). Free-Floating Electric Carsharing-Fleets in Smart Cities: the Dawning of a Post-Private Car Era in Urban Environments? *Environmental Science & Policy*, 30-40. Retrieved from https://doi.org/10.1016/j.envsci.2014.09.005 doi: 10.1016/j.envsci.2014.09.005

He, G., Chen, Q., Kang, C., Pinson, P., & Xia, Q. (2016). Optimal Bidding Strategy of Battery Storage in Power Markets Considering Performance-Based Regulation and Battery Cycle Life. *IEEE Transactions on Smart Grid*, 2359-2367. Retrieved from `https://doi.org/10.1109/tsg.2015.2424314` doi: 10.1109/tsg.2015.2424314

Humphrys, M. (1996). Action Selection Methods Using Reinforcement Learning. *From Animals to Animats*.

Kahlen, M., Ketter, W., & Gupta, A. (2017). Fleetpower: Creating Virtual Power Plants in Sustainable Smart Electricity Markets.

Kahlen, M., Ketter, W., & van Dalen, J. (2014). Balancing With Electric Vehicles: a Profitable Business Model.

Kahlen, M., Ketter, W., & van Dalen, J. (2018). Electric Vehicle Virtual Power Plant Dilemma: Grid Balancing Versus Customer Mobility.

Kambil, A., & van Heck, E. (1998). Reengineering the Dutch Flower Auctions: a Framework for Analyzing Exchange Organizations. *Information Systems Research*, 1-19. Retrieved from `https://doi.org/10.1287/isre.9.1.1` doi: 10.1287/isre.9.1.1

Kara, E. C., Macdonald, J. S., Black, D., Bérges, M., Hug, G., & Kiliccote, S. (2015). Estimating the Benefits of Electric Vehicle Smart Charging At Non-Residential Locations: a Data-Driven Approach. *Applied Energy*, 515-525. Retrieved from `https://doi.org/10.1016/j.apenergy.2015.05.072` doi: 10.1016/j.apenergy.2015.05.072

Ketter, W., Collins, J., & Reddy, P. (2013). Power Tac: a Competitive Economic Simulation of the Smart Grid. *Energy Economics*, 262-270. Retrieved from `https://doi.org/10.1016/j.eneco.2013.04.015` doi: 10.1016/j.eneco.2013.04.015

Kim, E. L., Tabors, R. D., Stoddard, R. B., & Allmendinger, T. E. (2012). Carbitrage: Utility Integration of Electric Vehicles and the Smart Grid. *The Electricity Journal*, 16-23. Retrieved from `https://doi.org/10.1016/j.tej.2012.02.002` doi: 10.1016/j.tej.2012.02.002

Ko, H., Pack, S., & Leung, V. C. M. (2018). Mobility-Aware Vehicle-To-Grid Control Algorithm in Microgrids. *IEEE Transactions on Intelligent Transportation Systems*. Retrieved from `https://doi.org/10.1109/tits.2018.2816935` doi: 10.1109/tits.2018.2816935

Lopes, J. A. P., Soares, F. J., & Almeida, P. M. R. (2011). Integration of Electric Vehicles in the Electric Power System. *Proceedings of the IEEE*, 168-183. Retrieved from https://doi.org/10.1109/jproc.2010.2066250 doi: 10.1109/jproc.2010.2066250

Mashhour, E., & Moghaddas-Tafreshi, S. M. (2011). Bidding Strategy of Virtual Power Plant for Participating in Energy and Spinning Reserve Markets-Part II: Numerical Analysis. *IEEE Transactions on Power Systems*, 957-964. Retrieved from https://doi.org/10.1109/tpwrs.2010.2070883 doi: 10.1109/tpwrs.2010.2070883

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., ... Hassabis, D. (2015). Human-Level Control Through Deep Reinforcement Learning. *Nature*. Retrieved from https://doi.org/10.1038/nature14236 doi: 10.1038/nature14236

Peters, M., Ketter, W., Saar-Tsechansky, M., & Collins, J. (2013). A reinforcement learning approach to autonomous decision-making in smart electricity markets. *Machine learning*, 5–39.

Peterson, S. B., Whitacre, J., & Apt, J. (2010). The Economics of Using Plug-In Hybrid Electric Vehicle Battery Packs for Grid Storage. *Journal of Power Sources*, 2377-2384. Retrieved from https://doi.org/10.1016/j.jpowsour.2009.09.070 doi: 10.1016/j.jpowsour.2009.09.070

Pudjianto, D., Ramsay, C., & Strbac, G. (2007). Virtual Power Plant and System Integration of Distributed Energy Resources. *IET Renewable Power Generation*, 10. Retrieved from https://doi.org/10.1049/iet-rpg:20060023 doi: 10.1049/iet-rpg:20060023

Reddy, P. P., & Veloso, M. M. (2011a). Learned Behaviors of Multiple Autonomous Agents in Smart Grid Markets. In *Aaai.*

Reddy, P. P., & Veloso, M. M. (2011b). Strategy learning for autonomous agents in smart grid markets. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence.*

Rummery, G. A., & Niranjan, M. (1994). *On-line Q-learning using connectionist systems.* University of Cambridge, Department of Engineering Cambridge, England.

Schill, W.-P. (2011). Electric Vehicles in Imperfect Electricity Markets: the Case of Germany. *Energy Policy*, 6178-6189. Retrieved from https://doi.org/10.1016/j.enpol.2011.07.018 doi: 10.1016/j.enpol.2011.07.018

Shi, W., & Wong, V. W. (2011). Real-time vehicle-to-grid control algorithm under price uncertainty. In *Ieee international conference on smart grid communications (smartgridcomm).* Retrieved from `https://doi.org/10.1109/smartgridcomm.2011.6102330` doi: 10.1109/smartgridcomm.2011.6102330

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., ... Hassabis, D. (2016). Mastering the Game of Go With Deep Neural Networks and Tree Search. *Nature*. Retrieved from `https://doi.org/10.1038/nature16961` doi: 10.1038/nature16961

Sioshansi, R. (2012). The Impacts of Electricity Tariffs on Plug-In Hybrid Electric Vehicle Charging, Costs, and Emissions. *Operations Research*, 506-516. Retrieved from `https://doi.org/10.1287/opre.1120.1038` doi: 10.1287/opre.1120.1038

Sutton, R. S. (1984). Temporal Credit Assignment in Reinforcement Learning. *University of Massechusetts, Amherst*.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction.* MIT press.

Taylor, A., Dusparic, I., Galvan-Lopez, E., Clarke, S., & Cahill, V. (2014). Accelerating Learning in multi-objective systems through Transfer Learning. In *International joint conference on neural networks (ijcnn).* Retrieved from `https://doi.org/10.1109/ijcnn.2014.6889438` doi: 10.1109/ijcnn.2014.6889438

Tomić, J., & Kempton, W. (2007). Using Fleets of Electric-Drive Vehicles for Grid Support. *Journal of Power Sources*, 459-468. Retrieved from `https://doi.org/10.1016/j.jpowsour.2007.03.010` doi: 10.1016/j.jpowsour.2007.03.010

Valogianni, K., Ketter, W., Collins, J., & Zhdanov, D. (2014). Effective Management of Electric Vehicle Storage Using Smart Charging. In *Aaai* (pp. 472–478).

Vandael, S., Claessens, B., Ernst, D., Holvoet, T., & Deconinck, G. (2015). Reinforcement Learning of Heuristic EV Fleet Charging in a Day-Ahead Electricity Market. *IEEE Transactions on Smart Grid*, 1795-1805. Retrieved from `https://doi.org/10.1109/tsg.2015.2393059` doi: 10.1109/tsg.2015.2393059

van der Veen, R. A., & Hakvoort, R. A. (2016). The Electricity Balancing Market: Exploring the Design Challenge. *Utilities Policy*. Retrieved from `https://doi.org/https://doi.org/10.1016/j.jup.2016.10.008` doi: https://doi.org/10.1016/j.jup.2016.10.008

Vaya, M. G., & Andersson, G. (2015). Optimal Bidding Strategy of a Plug-In Electric Vehicle Aggregator in Day-Ahead Electricity Markets Under Uncertainty. *IEEE Transactions on Power Systems*. Retrieved from https://doi.org/10.1109/tpwrs.2014.2363159 doi: 10.1109/tpwrs.2014.2363159

Vaya, M. G., Rosello, L. B., & Andersson, G. (2014). Optimal bidding of plug-in electric vehicles in a market-based control setup. In *Power systems computation conference.* Retrieved from https://doi.org/10.1109/pscc.2014.7038108 doi: 10.1109/pscc.2014.7038108

Vázquez-Canteli, J. R., & Nagy, Z. (2019). Reinforcement Learning for Demand Response: a Review of Algorithms and Modeling Techniques. *Applied Energy*, 1072-1089. Retrieved from https://doi.org/10.1016/j.apenergy.2018.11.002 doi: 10.1016/j.apenergy.2018.11.002

Watkins, C. J. C. H. (1989). Learning From Delayed Rewards. *King's College, Cambridge*.

Watkins, C. J. C. H., & Dayan, P. (1992). Q-Learning. *Machine Learning*. Retrieved from https://doi.org/10.1007/BF00992698 doi: 10.1007/BF00992698