POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Enhancing Energy Flexibility in Electric Vehicle Charging Stations using Reinforcement Learning

Laurea Magistrale in Automation and Control Engineering - Ingegneria dell'Automazione

Author: Antonio Alejandro Aslan Suarez

Advisor: Prof. Fredy Ruiz Palacios

Co-advisor: Cesar Diaz Londono

Academic year: 2024-2025

1. Purpose of the study

As an effort to minimize the impact of global warming, electricity generation is increasingly shifting from fossil fuel power plants to renewable energy sources (RES) such as wind, solar and hydropower. While these sources provide a cleaner and more sustainable solution, they suffer from a main drawback intrinsic to their nature. Which is that power generated by RES cannot be scheduled. This could cause a mismatch between supply and demand on the power grid which, if not dealt with, can lead to grid instability and voltage fluctuations. Thus, the concept of flexibility is introduced as the capability of the grid to maintain this delicate balance between power generation and consumption. On one hand, grid flexibility can be enhanced on the supply side, as thermal, nuclear and hydroelectric plants can adjust their power outputs in cases of demand fluctuations or sudden drops in RES power output. On the other hand, flexibility can also be provided on the demand side by using variable loads or power storage methods such as batteries or hydroelectric plants.

To reduce carbon emissions, countries and cities around the world are offering various incentives to encourage consumers to switch to electric vehicles (EVs). Thus, the demand for charging stations is increasing. Charging stations can tweak their power consumption to smooth demand curves and, as such, provide flexibility. A single aggregator that controls several Direct Current Fast Charging (DCFC) stations which have a reserved power capacity of between 50 and 350kW could aid significantly in maintaining grid balance.

Over the years, several Reinforcement Learning (RL) [3] and optimization-based strategies have been proposed for controlling aggregators to minimize the operation cost of charging stations, but only Model Predictive Control (MPC) has had successful results for cost minimization and flexibility maximization [2], as no simulator currently exists to train RL agents in the flexibility maximization case. For this reason, the purpose of this thesis has been to formulate the flexibility maximization problem in the RL framework and compare it to the current MPC solution. For this task, a simulator in which to

train the RL agents should be created.

2. State of the art

The economic MPC (eMPC) is an optimization-based strategy that generates a power profile which minimizes the operation cost of every charger i in an aggregator by solving a constrained optimization problem (COP) at every time step k [1]. To obtain said trajectory, the eMPC chooses a input trajectory u that minimizes the cost function (1) while satisfying the charging dynamics model from (2), meaning that the state of charge (SoC) of the EVs are taken as the states of the system. Finally, to ensure that the EVs are fully charged at dispatch time d_j, constrain (6) is added to the COP.

$$\min_{u_{i,k}} \quad \Delta t \sum_{k=0}^{H_p-1} \left(c_k \sum_{i=1}^{I} u_{i,k} \right) \tag{1}$$

$$x_{i,k+1} \begin{cases} SoC_{j,a_j} & \text{if } \xi_{i,k} = 1 \text{ and } k = a_j, \\ x_{i,k} + P_{i,k}\,\Delta t & \text{if } \xi_{i,k} = 1 \text{ and } a_j < k < d_j, \\ 0 & \text{if } \xi_{i,k} = 0 \text{ or } k = d_j \end{cases} \tag{2}$$

The eMPC strategy can be extended to accommodate flexibility, resulting in the Optimal Control with minimum Cost and Maximum Flexibility (OCCF) strategy. First, a mathematical representation of flexibility in both directions shall be constructed as in (3). Here, the measure of the upward flexibility (U^F) represents the capacity of the charger to draw more power from the grid while downward flexibility (L^F) considers the capacity of the charger to lower its consumption.

$$F_k = \sum_{i=1}^{I} F_{i,k} = \sum_{i=1}^{I} \left(U^F_{i,k} + L^F_{i,k} \right) \tag{3}$$

Flexibility can be added to the cost function of the MPC with a negative sign (4) as it shall be maximized, flexibility is rewarded by a fraction of the electricity price known as the flexibility multiplier or π_F. As the maximum power of the charging station is bounded, flexibility can be added to the problem with (9) and the rest of the constraints just bound the variables to their maximum and minimum values.

$$\min_{u_{i,k}} \quad \Delta t \sum_{k=0}^{H_p-1} \left(c_k \sum_{i=1}^{I} (u_{i,k} - \pi_F F_{i,k}) \right) \tag{4}$$

$$\text{s.t.} \quad \text{Dynamic Equation (2)} \tag{5}$$

$$x_{i,d_j} = x_{i,\max} \tag{6}$$

$$x_{i,a_j} = SoC_j \tag{7}$$

$$x_{i,\max} = C_j \tag{8}$$

$$L^F_{i,k} \le u_{i,k} \le \xi_{i,k}(P_{\max} - U^F_{i,k}) \tag{9}$$

$$0 \le u_{i,k} \le P_{\max} \tag{10}$$

$$0 \le x_{i,k} \le x_{i,\max} \tag{11}$$

$$0 \le U^F_{i,k} \le P_{\max} \tag{12}$$

$$0 \le L^F_{i,k} \le P_{\max} \tag{13}$$

3. Methodology

RL agents derive their decision-making strategy (also known as policy) by interacting with their environment in a process called training. During this phase, the agent selects an action based on the current state of the environment, which then returns the next state and reward back to the agent, a process which is repeated until the termination of the episode. During each interaction, the agent is learning and improving its policy based on the feedback received by the agent, as the goal of any agent should be to maximize this reward.

Agent selection

Most agents nowadays derive a policy that directly maps a state to the action that leads to the highest reward, which are known as policy gradient methods [4]. These methods use a neural network (NN) to estimate a action-value for each state and action pair, which represents the reward obtained by following the action plus the set of future rewards that can be achieved once the future state is reached. Then, the policy is set to choose the action with the highest action-value estimation. The NN is trained using the experience acquired during training, but during this phase, random action selection is introduced to allow the agent to explore other state and action pairs that have not been yet explore and thus their value is low or non-existing.

For the purpose of this thesis, the selected agents were the Deep Deterministic Policy Gradient (DDPG) and its successor (TD3), Proximal Policy Optimization (PPO) and Trust-Region Policy Optimization (TRPO). They were chosen as they allow for continuous action-space without discretization.

Simulator Environment

For the environment in which the agents will be trained on, the simulator EV2Gym [3] has been selected as it is specifically tailored towards training RL agents. This open-source simulator uses experimental data to training episodes which are as close as possible to the real-world. Also, due to its modular design in which EVs, charging stations and transformers are simulated as different entities, modifying the code to accommodate for flexibility becomes relatively easy.

As the simulator was designed for situations in which agents follow a charging trajectory (set-point tracking), some modifications shall be done to the code for agents to achieve cost minimization and flexibility maximization. Flexibility can be easily added to the code to charger i at time step k using the definitions $U^F_{i,k} = P_{\max} - P_{i,k}$ and $L^F_{i,k} = P_{i,k}$. However, these equations are only applicable under conditions in which the charger can actually change its power profile, that is, when the following three conditions are satisfied:

- EV is connected to charging station.
- The EV connected is not fully charged.
- The time to charge at maximum power is lower than the departure time. This condition ensures that the EV can change its power profile as it is not required to charge at maximum power.

Using this set of equations and conditions, the flexibility of every charging station controlled by the aggregator can be calculated at each time step. To give the agent feedback regarding the flexibility that is being achieved and other parameters regarding future electricity prices and EV arrivals, the state vector shall be changed. This step can be done by just changing the function that generates the vector by including:

- Current timestep, scaled from 0 to 1.
- Total power consumed in the last time step.
- Total flexibility provided in the last time step.
- Future Electricity prices during the Prediction Horizon.
- For each charging station: SoC of the EV connected and time until departure.

The final major change with respect to the simulator is the reward function $r(k)$, which plays a huge role in the agent's final behavior. As this function is being maximized, one can take (4) and invert the signs, thus the final reward function is shown in (14).

$$r(k) = -c_k \sum_{i=1}^{K} P_{i,k} + \pi \cdot c_k \sum_{i=1}^{K} F_{i,k} + L^{USR}(k) \tag{14}$$

To ensure that EVs are fully charged before leaving the station, the MPC strategy can use a hard constraint (6). However, as RL methods cannot impose hard constraints, a term is added to the reward function L^{USR} (15) which penalizes the agent when an EV leaves the station and it is not fully charged. This penalization is dependent on the difference between the SoC at dispatch and the maximum SoC of the EV, known as User Satisfaction (US). If not used, the agent will maximize the reward by providing upwards flexibility and not charging the EVs.

$$L^{USR}(x) = \begin{cases} 100x - 80 & \text{if } x < 0.80 \\ 0 & \text{if } 0.80 \leq x < 0.85 \\ 13.33x - 11.33 & \text{if } x \geq 0.95 \end{cases} \tag{15}$$

Agent Training

Once the environment has been properly modified, 10 random evaluation episodes are generated on which the evaluation metrics from section 4 are obtained. First, the four RL agents are trained using different flexibility multipliers for 1 million iterations each. During training, an evaluation function is called every 2500 iterations to test the agent's performance in deterministic mode (meaning that no exploration is performed) for 5 random episodes. After obtain-

ing the mean reward of the evaluation episodes, the script saves the current agent's weights if the mean reward is the highest achieved so far. Once training has stopped, the weights of the best performing agent are loaded and the agent is evaluated on the initial 10 evaluation episodes, thus obtaining the final metrics. On the other hand, as MPC does not need training, the metrics can be obtained by just running the evaluation episodes with their corresponding flexibility multipliers.

4. Results

As the objective of the thesis is to develop an effective charging strategy, one of the main metrics to consider is user satisfaction. From Figure 1 one can see that the best results are obtained for values of $\pi \in [0.1, 0.5]$ as they achieve a good balance between US and profit margin. This is because when flexibility is introduced, the agent is incentivized to wait and charge at the instant when electricity prices are lower, and in the meantime it is rewarded as it is providing upwards flexibility. For $\pi > 0.5$, the agent maximizes the profits by just charging over 85% in order not to be penalized by (15), as such, those values of π should no longer be considered.

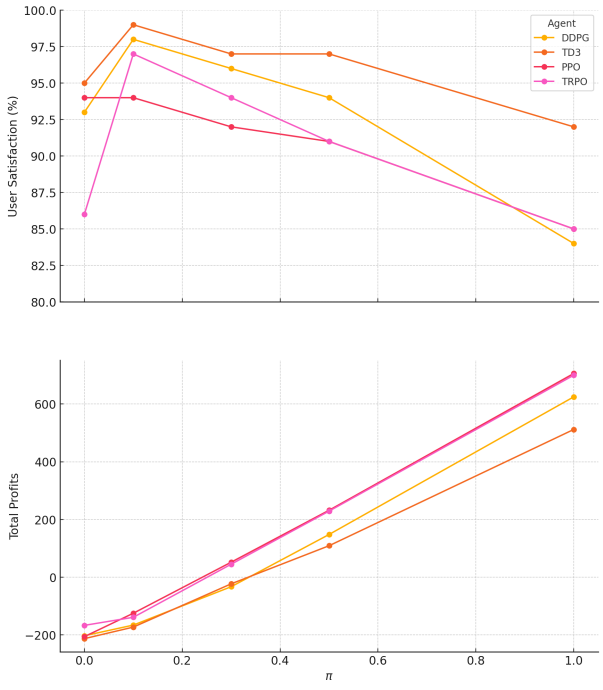

Figure 1: US and profits obtained for RL agents

Out of the four RL agents selected, only DDPG and TD3 will be considered for the final analysis against MPC, as PPO and TRPO US metrics are lower. Also, when looking at the power profile of said agents from Figure 2 one can see that the discarded agents follow a binary or bang-bang control strategy, which is not recommended for battery health.

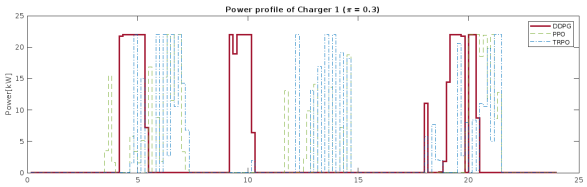

Figure 2: Power profile of RL agents for $\pi = 0.3$

RL vs. MPC

The metrics for the final RL agents along with the MPC are shown in Figure 3 . Overall, MPC consistently outperformed RL methods across all performance metrics. Although when considering $\pi = 0$, the profits from the MPC are slightly lower than the ones obtained by RL methods. This result can be attributed to the fact that while MPC charged all EVs to 100% (as it is set by a hard constraint), RL agents only arrived at 93% and 95% respectively, thus achieving slightly higher profits.

This result are not surprising considering that MPC is solving a minimization problem and taking full advantage of the model's dynamics at each time step. In those cases where the full model is available, it is expected that using a model-free method such as RL will yield worst results.

5. Conclusions

Throughout this thesis, the framework to apply RL methods for the EV charging problem has been developed. And thanks to this, it has been proven that a wide variety of RL methods can be used to manage an aggregator to minimize cost while maximizing flexibility. However, the final behavior of the agent will not depend as much on its architecture, but on the reward function on which the agent is trained. This is based on the fact that none of the agents chose to charge the EVs to 100% as the reward for doing so is too weak, and when considering higher profits from flexibilities, this

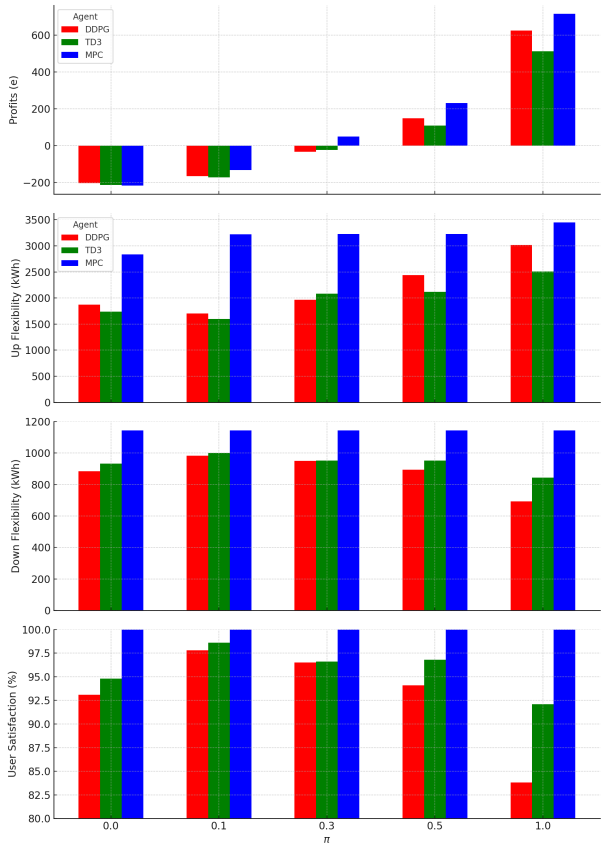

Figure 3: Metrics of DDPG, TD3 and MPC across different values of π.

metric drops even further, suggesting that the penalization for not charging shall be scaled with respect to the flexibility multiplier.

When comparing the results of the final RL agents with respect to MPC, it can be concluded that MPC outperforms RL methods when minimizing cost and maximizing flexibility. This is not surprising as when dealing with a simpler problem from which one can develop a plant model, using a method that solves a minimization problem at each iteration will always deliver best results. However, the results delivered by RL are still acceptable and, as such, these methods shall be considered for cases forecasted data is not accurate or available.

To continue the development of RL solutions for flexibility maximization, first it shall be proven that using RL in more complex and real scenarios could result in better performance. This means that agents should be trained without future knowledge of parking times and-or electricity prices so that they develop policies that could predict said values based on past inputs and current time of the day (also depending on the day of the week). In this case, MPC can also be applied but it would require building a NARMA predictor to estimate said parameters. This is a would be a sufficiently complex scenario to truly justify the use of RL.

References

[1] Cesar Diaz, Andrea Mazza, Fredy Ruiz, Diego Patino, and Gianfranco Chicco. Understanding Model Predictive Control for Electric Vehicle Charging Dispatch. In *2018 53rd International Universities Power Engineering Conference (UPEC)*, pages 1–6, Glasgow, September 2018. IEEE.

[2] Cesar Diaz-Londono, Luigi Colangelo, Fredy Ruiz, Diego Patino, Carlo Novara, and Gianfranco Chicco. Optimal Strategy to Exploit the Flexibility of an Electric Vehicle Charging Station. *Energies*, 12(20):3834, October 2019.

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

[4] Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: an introduction*. Adaptive computation and machine learning series. The MIT Press, Cambridge, Massachusetts, second edition edition, 2018.