Laplace

LABORATOIRE PLASMA ET CONVERSION D'ÉNERGIE

Reinforcement Learning for Distributed Energy Resource Management

SAGIP, CADO, CSE
Bordeaux, France
Valentin BURGAUD, Grégoire LE GOFF, Pauline KERGUS, Maurice FADEL

June 11, 2026

cnrs TOULOUSE INP N7 UNIVERSITÉ TOULOUSE III PAUL SABATIER

Table of contents

Introduction

Introduction

Context :

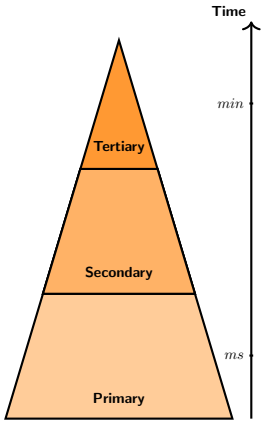
- Growing RE sources are **intermittent** and require **storage** integration [1];

Challenges :

- **Optimizing** storage use depends on accurate **profile forecasting** ;
- Networks must ensure **robustness** and **resilience** against disturbances of varying time scales ;

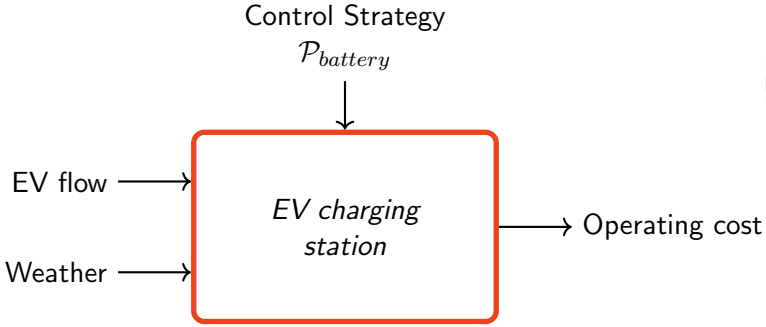
Aim :

- Implement a **real-time control strategy using AI** to achieve **near-optimal** problem solutions.



Laplace

Introduction



Functional diagram of the system: an electric vehicle charging station.

Laplace

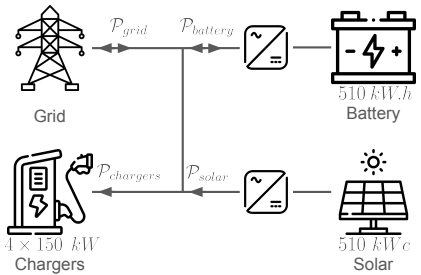
Introduction

	OC	MPC	Rule-based	RL-based
Offline	x			
Online		x	x	x

- **Objective :** Compare five online control strategies (MPC, rule-based, and RL-based methods) against an offline optimal control solution computed under ideal assumptions [2, 3].

Methodology

System Description



Power flow architecture for a given system sizing.

- 510 $kW.h$ battery with a ± 375 kW discharge rate ;
- 4 years of weather[1] and traffic[2] data from the A84 highway near Saint James, France, both refined to a 1-minute sampling period ($T_s = 1$ min) ;
- EV dataset including consumption, charge rates, etc.

[1] https://re.jrc.ec.europa.eu/pvg_tools/fr/

[2] https://www.cerema.fr/fr/mots-cles/donnees-trafic

Laplace

Optimal Control with Linear Programming

Offline optimal control solved using linear programming :

$$\min_{\mathcal{P}_{battery}(t)} \sum_{t=0}^{T} \mathcal{P}_{grid}(t) \times c(t) \quad \text{s.t.}$$

$$\begin{cases} \mathcal{P}_{grid}(t) = \mathcal{P}_{chargers}(t) - \mathcal{P}_{battery}(t) - \mathcal{P}_{solar}(t) \\ SOC(t+1) = SOC(t) - \dfrac{\mathcal{P}_{battery}(t+1) \times T_s}{E_{bat}} \\ SOC(t_0) = SOC_{init} \\ SOC(t_{d+1}) = SOC(t_d) \\ \mathcal{P}_{battery}^{min} \leq \mathcal{P}_{battery}(t) \leq \mathcal{P}_{battery}^{max} \\ 0.2 \leq SOC(t) \leq 1 \end{cases} \tag{1}$$

Laplace

Model Predictive Control

$$\min_{\mathcal{P}_{battery}(t)} \sum_{t=\tau}^{T} \mathcal{P}_{grid}(t) \times c(t) \quad \text{s.t.} \quad (1)$$

- Online optimization ;
- Based on linear programming ;
- Receding horizon formulation ;
- Identical constraints to OC.

Laplace

Model Predictive Control

$$\min_{\mathcal{P}_{battery}(t)} \sum_{t=\tau}^{T} \mathcal{P}_{grid}(t) \times c(t) \quad \text{s.t.} \quad (1)$$

- Online optimization ;
- Based on linear programming ;
- Receding horizon formulation ;
- Identical constraints to OC.

$$\hat{\mathcal{P}}_{\text{chargers}}(t) = \sum_{i=1}^{4} w_i \, \mathcal{P}_{\text{chargers}}\,(t - i \times 24 \times 60)$$

$$P_7(t) = \sum_{i=1}^{4} w_i \, \mathcal{P}_{\text{solar}}\,(t - i \times 24 \times 60) \quad (2)$$

$$\hat{\mathcal{P}}^{\tau}_{\text{solar}}(t) = \max\left(0, P_7(t) + [\mathcal{P}_{\text{solar}}(\tau) - P_7(\tau)]\right)$$

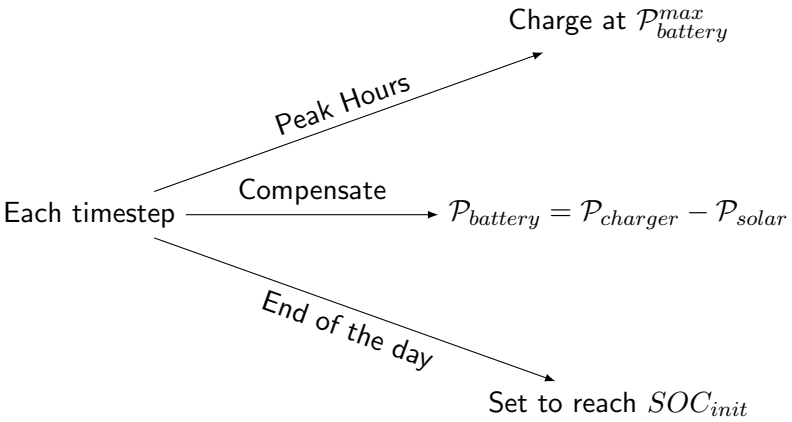
Mean-based prediction system using previous days.

	RMS [kW]	ΔE [kWh]	RMS [%]	ΔE [%]
$\hat{\mathcal{P}}_{\text{chargers}}$	137.15	2601.1	30.55	25.74
$\hat{\mathcal{P}}_{\text{solar}}$ 9 AM	111.99	871.69	30.39	72.92
$\hat{\mathcal{P}}_{\text{solar}}$ 3 PM	61.66	327.96	21.29	67.33

Worst-day prediction performance.

Laplace

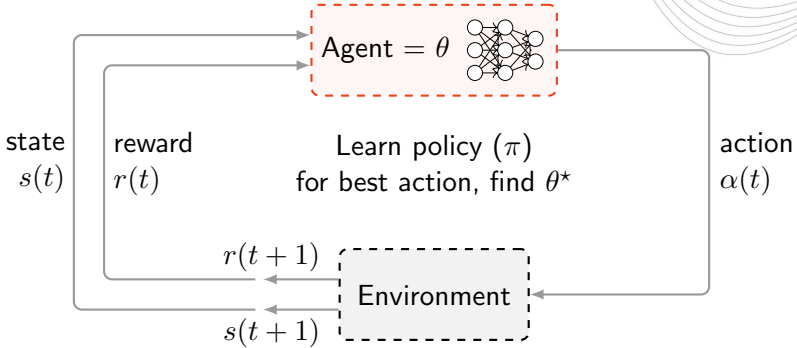
Rule-Based



Simplified enhanced rule-based control Algorithm.

Laplace

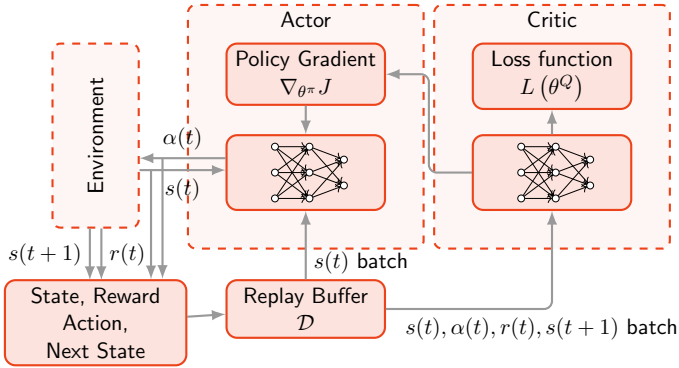
Deep Reinforcement Learning



The agent-environment interaction during training in RL.

Laplace

Deep Reinforcement Learning



Actor-Critic reinforcement learning framework used by DDPG, SAC, TD3.

Laplace

Environment Configuration

Action, reward and state definition :

- Action $\alpha(t) = \mathcal{P}_{battery}(t)$;

The RL agent is trained to approximate optimal control trajectories.

Laplace

Environment Configuration

Action, reward and state definition :

- Action $\alpha(t) = \mathcal{P}_{battery}(t)$;
- Reward

$$r(t) = p_{\text{soc}}(t) + p_{\text{soc\_final}}(t) + \left(e^{\frac{-\varepsilon(t)^2}{2\sigma^2}} - 1 \right)$$

$$\varepsilon(t) = \left(\alpha(t) - \mathcal{P}^*_{battery}(t) \right) / \mathcal{P}^{max}_{battery}$$

The RL agent is trained to approximate optimal control trajectories.

Laplace

Environment Configuration

Action, reward and state definition :

- Action $\alpha(t) = \mathcal{P}_{battery}(t)$;
- Reward

$$r(t) = p_{\text{soc}}(t) + p_{\text{soc\_final}}(t) + \left(e^{\frac{-\varepsilon(t)^2}{2\sigma^2}} - 1 \right)$$

$$\varepsilon(t) = \left(\alpha(t) - \mathcal{P}^*_{battery}(t) \right) / \mathcal{P}^{max}_{battery}$$

- State $s(t) =$ `[instant, time, forecasts]`

  → `[soc_norm, solar_p_norm, load_p_norm, is_vehicle queue_size_norm, price_buy_norm, time_remaining_norm soc_target, hour_sin, hour_cos, future_vehicles_norm future_price_norm, solar_forecast_hourly (24)]`

The RL agent is trained to approximate optimal control trajectories.

Laplace

Data

Dataset :

The RL agents were trained using three years of historical data, with three complete passes performed over each year, resulting in a total of $2 \times 60 \times 365 \times 3 \times 3 \approx 4.73 \times 10^6$ training samples.

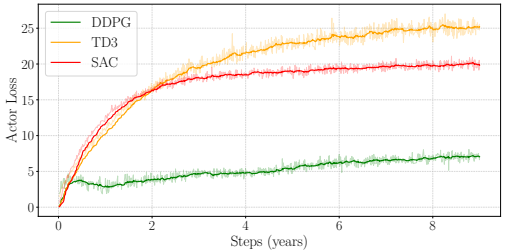
Laplace

Episode Reward During Training

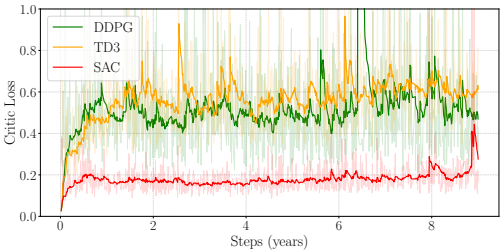
Evolution of average episode reward for 3 RL algorithms.

Laplace

Stability Across RL Agents



Evolution of actor loss.

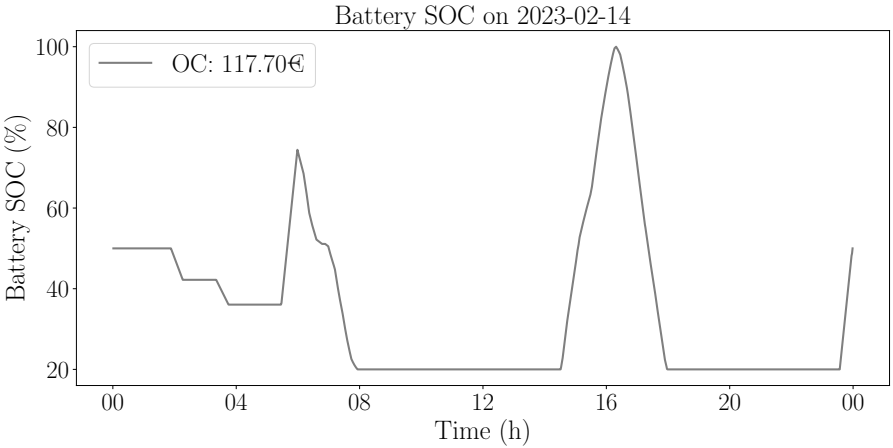


Evolution of critic loss.

Actor and critic losses typically converge to non-zero values due to stochastic noise in the targets and gradients.

Laplace

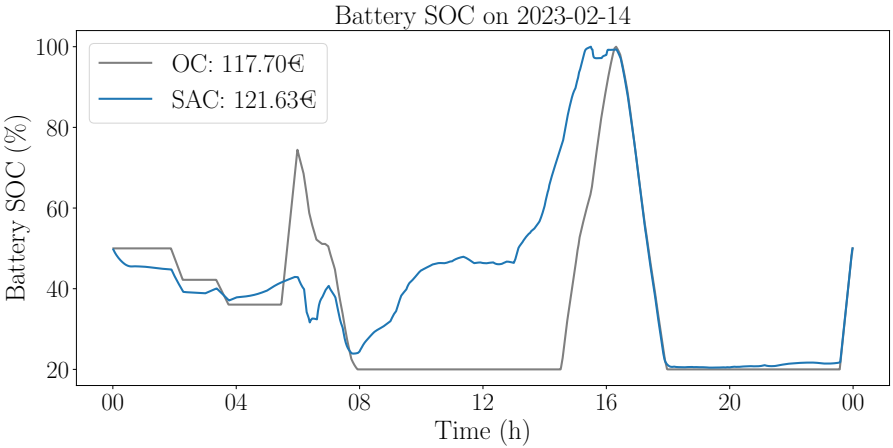
Analysis of Results



Evolution of the SOC over a randomly selected day for different strategies.

Laplace

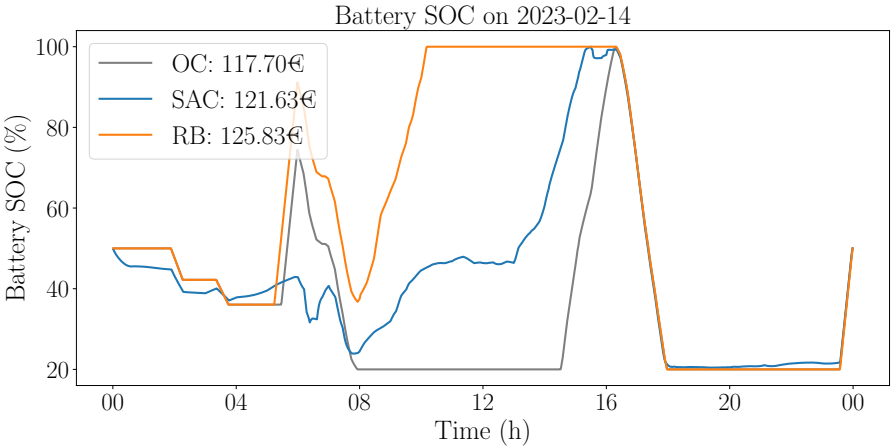
Analysis of Results



Evolution of the SOC over a randomly selected day for different strategies.

Laplace

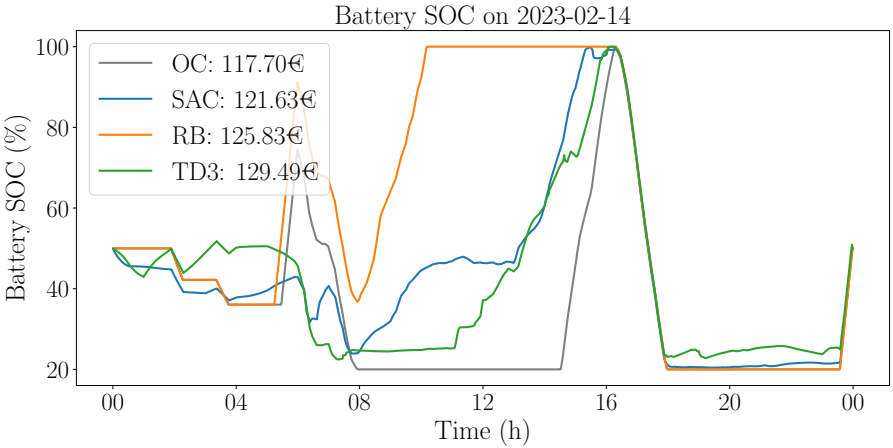
Analysis of Results



Evolution of the SOC over a randomly selected day for different strategies.

Laplace

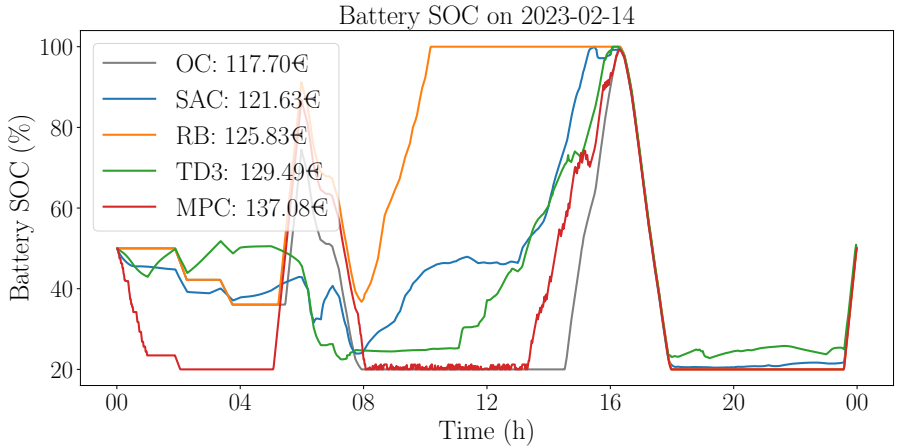
Analysis of Results



Evolution of the SOC over a randomly selected day for different strategies.

Laplace

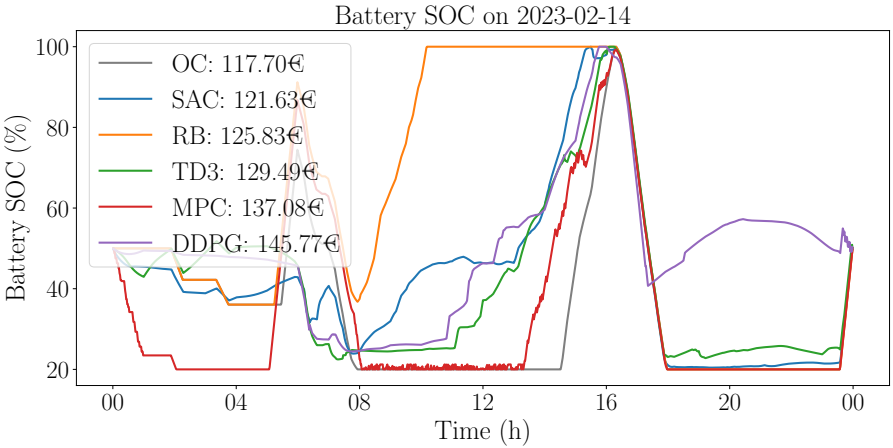
Analysis of Results



Evolution of the SOC over a randomly selected day for different strategies.

Laplace

Analysis of Results



Evolution of the SOC over a randomly selected day for different strategies.

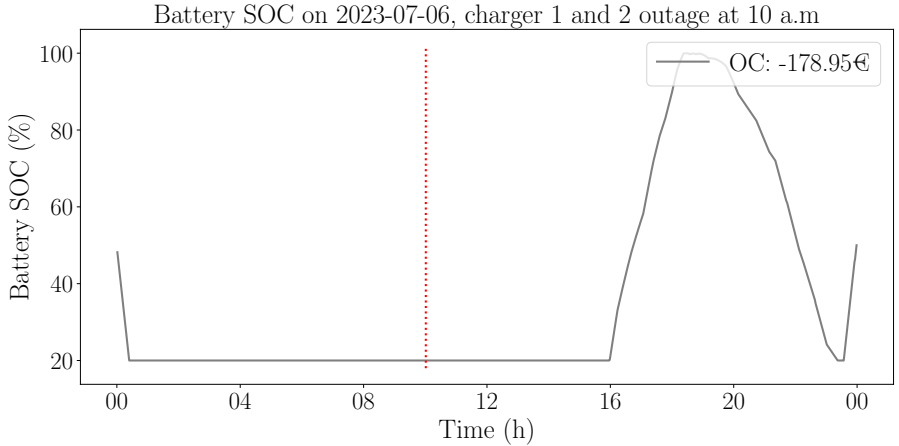
Laplace

Analysis of Results

	OC	SAC	RB	TD3	DDPG	MPC
Cost (€)	146.45	164.32	165.87	173.27	183.26	187.25
Error (%)	0	12.19	13.26	18.32	25.11	27.86
Time (s)	2.5e-4	2.2e-3	6.6e-6	2.1e-3	2.7e-3	1.37e-1

Simulation results: cost, error, and computation time for the considered EMS.

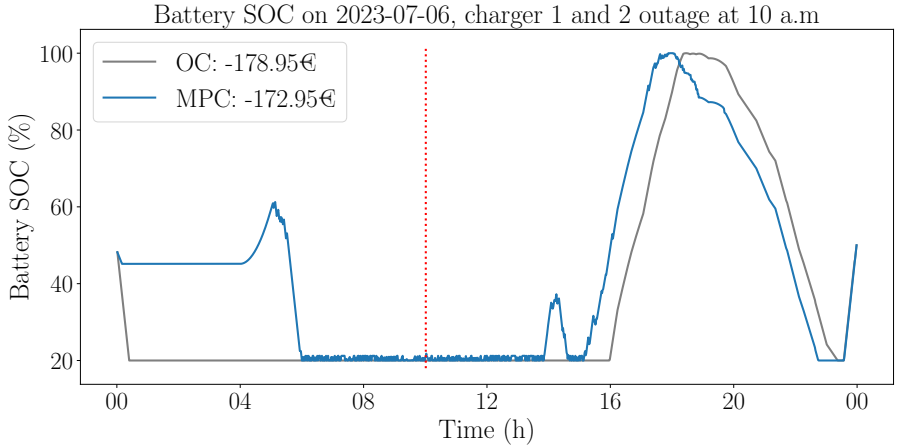
Perspectives



Evolution of the SOC over a randomly selected day under a fault condition for different strategies.

Laplace

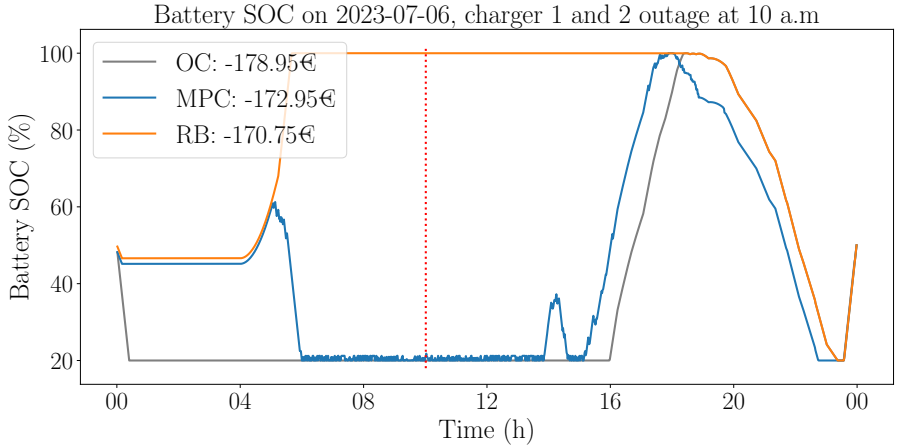
Perspectives



Evolution of the SOC over a randomly selected day under a fault condition for different strategies.

Laplace

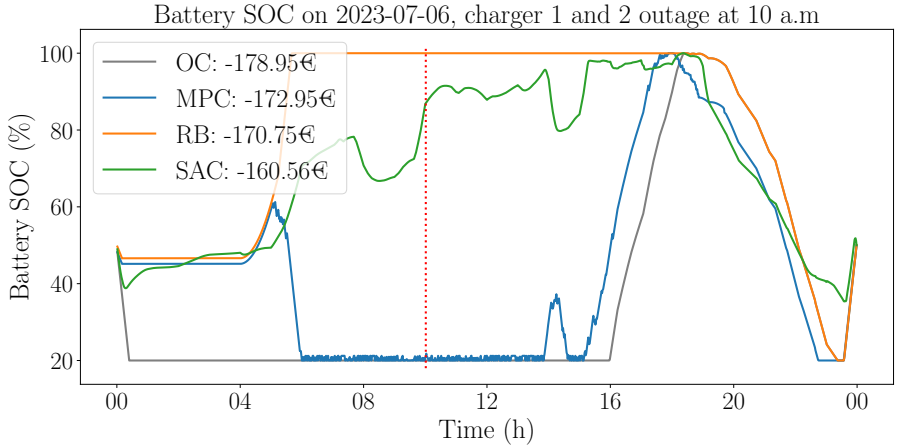
Perspectives



Evolution of the SOC over a randomly selected day under a fault condition for different strategies.

Laplace

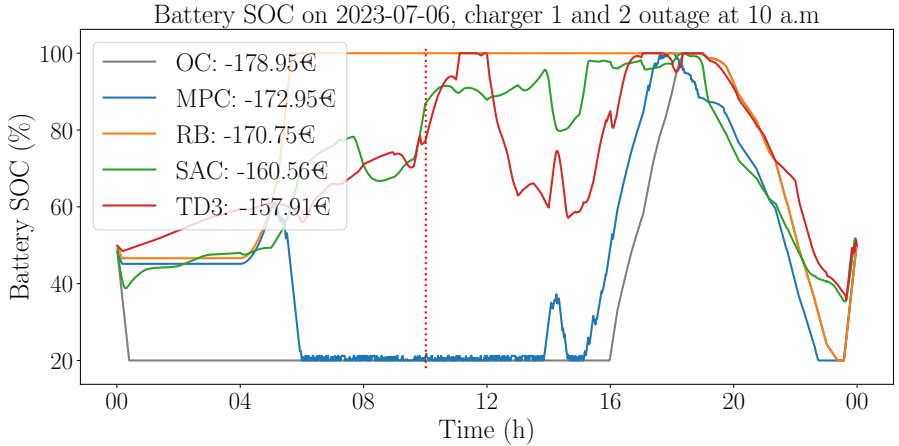
Perspectives



Evolution of the SOC over a randomly selected day under a fault condition for different strategies.

Laplace

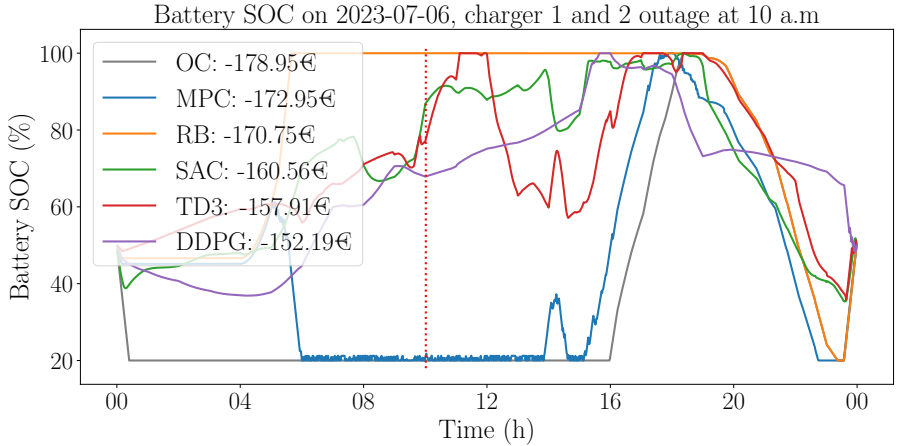
Perspectives



Evolution of the SOC over a randomly selected day under a fault condition for different strategies.

Laplace

Perspectives



Evolution of the SOC over a randomly selected day under a fault condition for different strategies.

Laplace

Conclusions

- High sensitivity to **reward/state** design impacting learning behavior.
  Normalize variables and validate formulations through systematic testing procedures.

Conclusions

- High sensitivity to **reward/state** design impacting learning behavior.
  Normalize variables and validate formulations through systematic testing procedures.
- Strong dependence on **data** quality and representativeness during training.
  Use large, diverse datasets with multiple training passes for robustness.

Conclusions

- High sensitivity to **reward/state** design impacting learning behavior.
  Normalize variables and validate formulations through systematic testing procedures.
- Strong dependence on **data** quality and representativeness during training.
  Use large, diverse datasets with multiple training passes for robustness.
- Potential **instability** during training and real-world deployment phases.
  Monitor convergence using gradient loss metrics.

Conclusions

- High sensitivity to **reward/state** design impacting learning behavior.
  Normalize variables and validate formulations through systematic testing procedures.
- Strong dependence on **data** quality and representativeness during training.
  Use large, diverse datasets with multiple training passes for robustness.
- Potential **instability** during training and real-world deployment phases.
  Monitor convergence using gradient loss metrics.
- Performance depends on **hyperparameters** and algorithm-specific tuning strategies.
  Apply automated hyperparameter optimization using dedicated solver frameworks.

Conclusions

- High sensitivity to **reward/state** design impacting learning behavior.
  Normalize variables and validate formulations through systematic testing procedures.
- Strong dependence on **data** quality and representativeness during training.
  Use large, diverse datasets with multiple training passes for robustness.
- Potential **instability** during training and real-world deployment phases.
  Monitor convergence using gradient loss metrics.
- Performance depends on **hyperparameters** and algorithm-specific tuning strategies.
  Apply automated hyperparameter optimization using dedicated solver frameworks.
- **Real-time** feasible, but deployment requires careful system-level integration.
  Use optimized implementations and hardware-aware design for efficient execution.

References

- F. Ahsan et al, "Data-driven next-generation smart grid towards sustainable energy evolution : techniques and technology review" in Protection and Control of Modern Power Systems, vol. 8, no. 3, pp. 1-42, Jul. 2023, doi: 10.1186/s41601-023-00319-5.
- V. Burgaud, G. Le Goff, P. Kergus, and M. Fadel, "Reinforcement Learning for Optimizing Battery Operation in Electric Vehicle Charging Microgrids," in Symposium de Génie Électrique, Toulouse, France, Aug. 2025, hal-05325798.
- V. Burgaud, G. Le Goff, P. Kergus, and M. Fadel, "Reinforcement learning vs. model-based control in electric vehicle charging microgrids," Control Engineering Practice, vol. 172, p. 106890, 2026, doi: 10.1016/j.conengprac.2026.106890.

Laplace

End of the presentation

UNIVERSITÉ PAUL SABATIER
Bât. 3R3, 118 route de Narbonne
31062 Toulouse Cedex 9

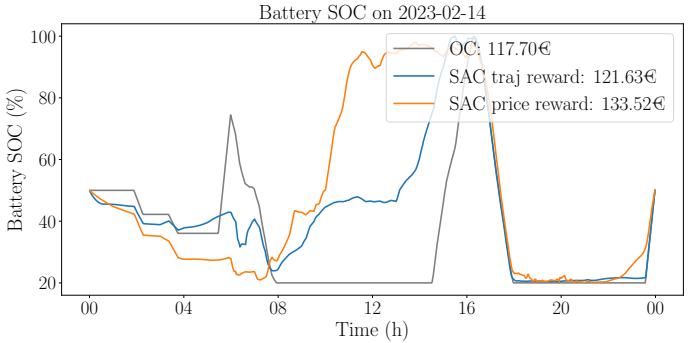
+33 (0)5 61 55 67 97
sec-ups@laplace.univ-tlse.fr

ENSEEIHT
2 Rue Camichel,
31071 Toulouse Cedex 7

+33 (0)5 34 32 23 91
sec-n7@laplace.univ-tlse.fr

Appendix

Hyperparameter optimization



Comparison between the trajectory based-reward and a purely price-driven reward.

$$r(t) = p_{\text{soc}}(t) + p_{\text{soc\_final}}(t) + \mathcal{P}_{battery}(t) \cdot c(t)$$

Laplace

Appendix

DRL Equations

$$y(t) = r(t) + \gamma Q\Big(s(t+1), \pi(s(t+1)|\theta^{\pi'}) \,|\, \theta^{Q'}\Big)$$

$$L(\theta^Q) = \mathbb{E}_{\mathcal{D}}\Big[\big(Q(s(t), \alpha(t)|\theta^Q) - y(t)\big)^2\Big]$$

$$\nabla_{\theta^\pi} J \approx \mathop{\mathbb{E}}_{s(t)\sim\mathcal{D}}\left[\nabla_{\alpha(t)} Q\left(s(t), \alpha(t)|\theta^Q\right) \cdot \nabla_{\theta^\pi} \pi\left(s(t)|\theta^\pi\right)\right]$$

Interpretation

- $L(\theta^Q)$: TD error between Q-estimation and target $y(t)$;
- $y(t)$: TD target combining reward and discounted future return;
- $\nabla_{\theta^\pi} J$: gradient used to improve the policy;
- $\nabla_\alpha Q$: critic feedback used to update the actor.

Appendix

Hyperparameter Optimization

Hyperparameters :

DDPG hyperparameters were automatically tuned [2] using a hyperparameter optimization framework (Optuna)[3], leading to a 17% performance improvement and mitigating sub-optimal convergence.

Parameter	Value
Discount Factor (γ)	0.99
Learning Rate (η)	1×10^{-4}
Target Update Speed (τ)	5×10^{-4}
Batch Size (B)	256
Hidden Layer Width 1 ($l1_{width}$)	256
Hidden Layer Width 2 ($l2_{width}$)	256
Hidden Layer Width 3 ($l3_{width}$)	256
Activation Function	ReLU
Training Frequency	1
Gradient Steps	1
Replay Buffer Size	3 years
Total Time steps	9 years
OU Noise σ	0.15
OU Noise θ	0.15

Table: Hyperparameters for the DDPG agent.

[3] T. Akiba et al., "Optuna: A Next-generation Hyperparameter Optimization Framework," KDD 2019.

Laplace

Appendix

SAC Hyperparameter

Parameter	Value
Discount Factor (γ)	0.99
Learning Rate (η)	1×10^{-4}
Target Update Speed (τ)	1×10^{-4}
Batch Size (B)	256
Hidden Layer Width 1 ($l1_{width}$)	256
Hidden Layer Width 2 ($l2_{width}$)	256
Hidden Layer Width 3 ($l3_{width}$)	256
Activation Function	ReLU
Training Frequency	1
Gradient Steps	2
Replay Buffer Size	3 years
Total Time steps	9 years
Entropy Regularization	
Entropy Coefficient (α)	auto
Target Entropy	auto

Table: Hyperparameters for the SAC agent.

Appendix

TD3 Hyperparameter

Parameter	Value
Discount Factor (γ)	0.99
Learning Rate (η)	1×10^{-4}
Target Update Speed (τ)	1×10^{-4}
Batch Size (B)	256
Hidden Layer Width 1 ($l1_{width}$)	256
Hidden Layer Width 2 ($l2_{width}$)	256
Hidden Layer Width 3 ($l3_{width}$)	256
Activation Function	ReLU
Training Frequency	1
Gradient Steps	2
Replay Buffer Size	3 years
Total Time steps	9 years
Policy Smoothing / Noise	
Action Noise (Gaussian)	0.1
Target Policy Noise σ	0.2
Target Noise Clip	0.4
Policy Delay	2

Table: Hyperparameters for the TD3 agent.

Laplace