Reinforcement Learning-based Fast Charging Control Strategy for Li-ion Batteries

TBSI RL-Course

Saehong Park, Andrea Pozzi, Michael Whitmeyer, Hector Perez
Wontae Joe, Davide Raimondo, Scott Moura

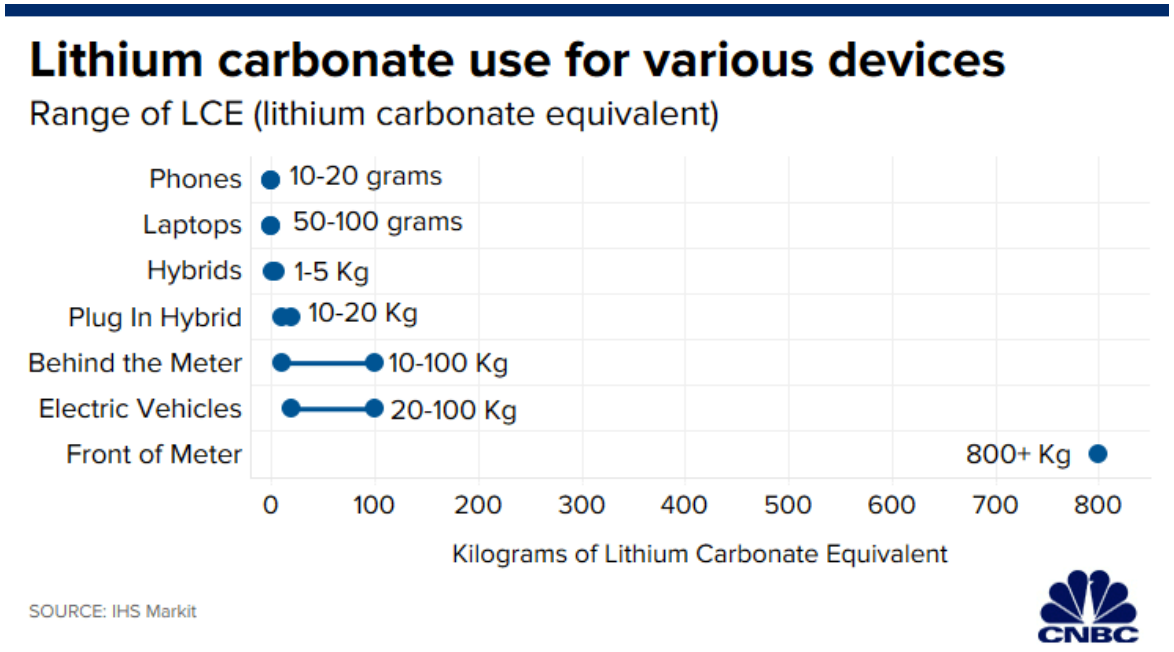
July 16, 2020

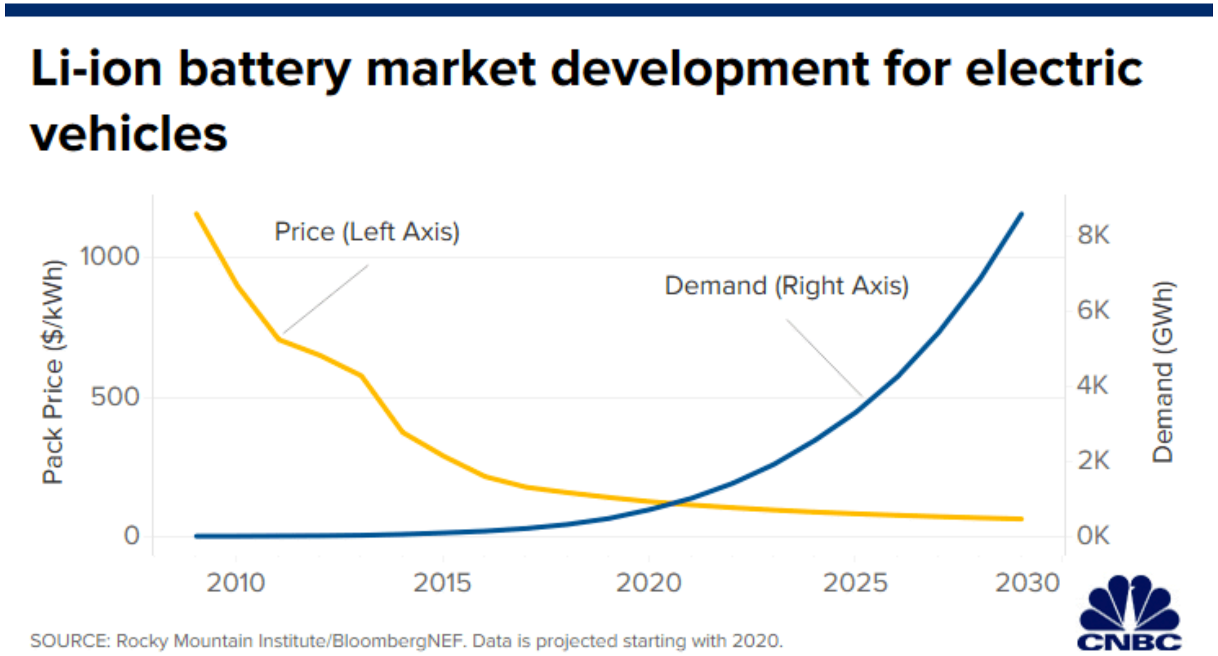
Agenda

- **Battery Overview and Literature Review**
- Reinforcement Learning
- Battery Model
- Simulation Results

Li-ion Battery in the World

- **Li-ion Batteries are everywhere**

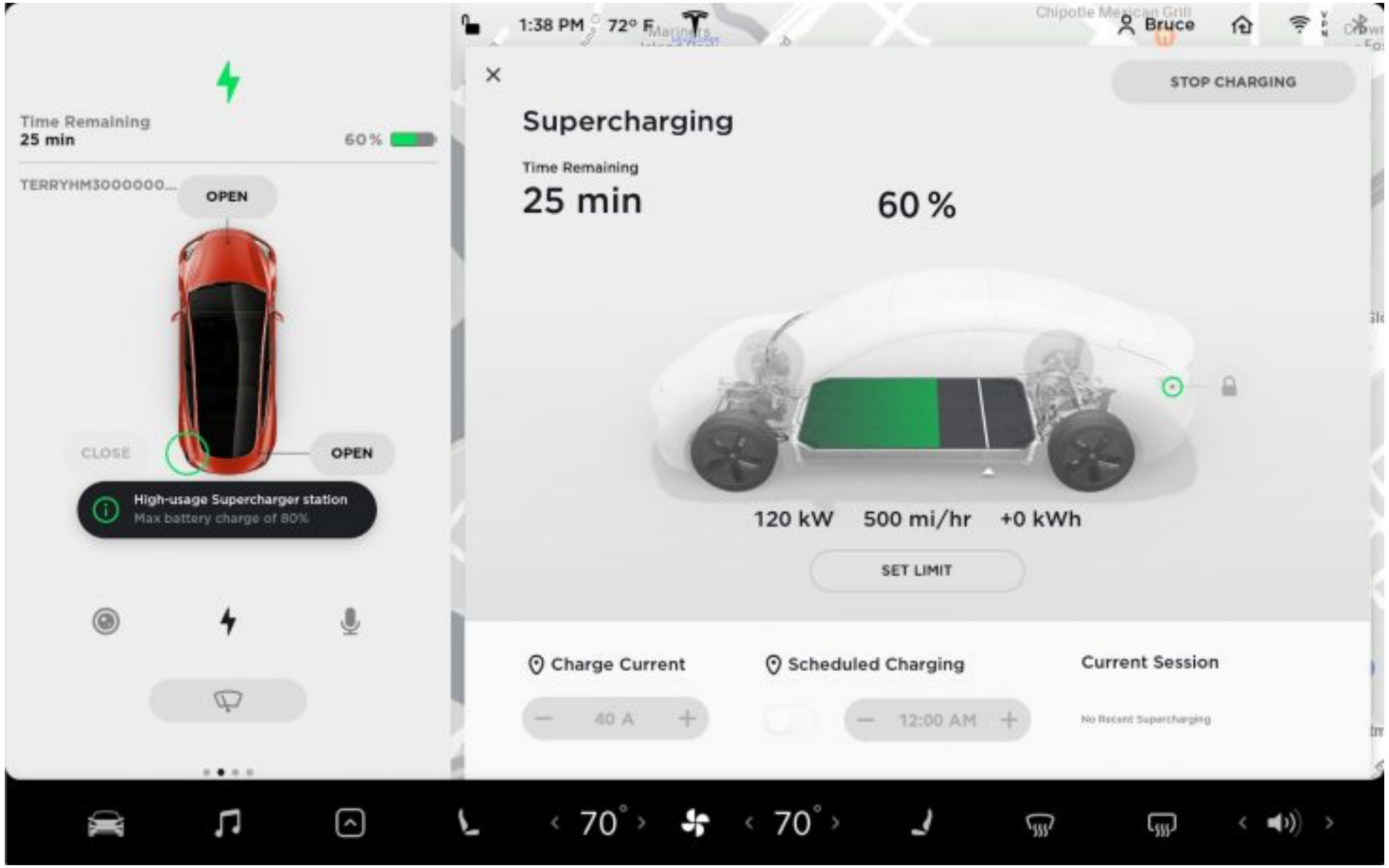


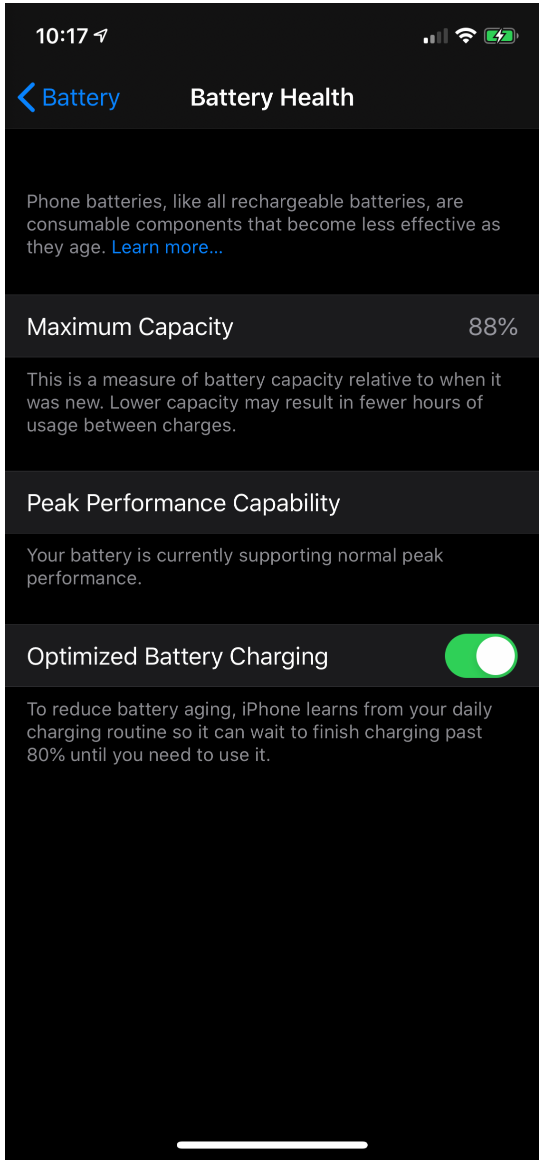


156$/kWh in 2019

Battery Charging

- **Tesla SuperCharger | iPhone Battery Charging**





Lithium Ion Batteries

Lithium-ion batteries require a Battery Management System (BMS) in order to work properly.

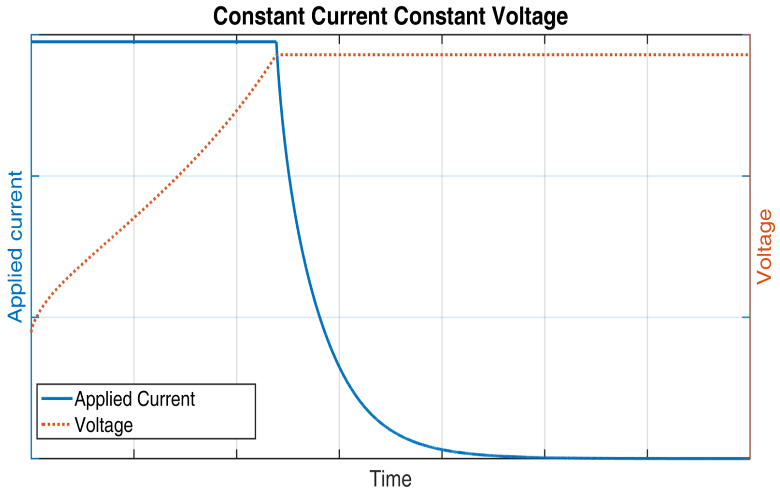
The BMS provides suitable charging procedures by finding the optimal trade-off between the following requirements:

- Fast Charging
- Safety

Standard Charging Methods

The mostly used charging protocol is the Constant-Current Constant Voltage (CC-CV).

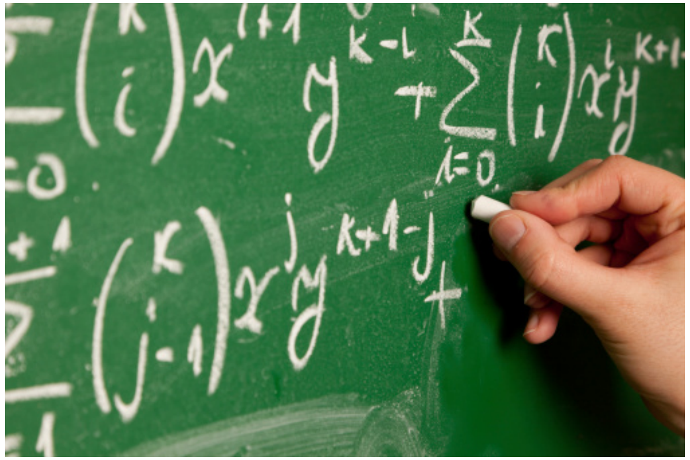
CC-CV is a **simple control procedure** which results in **reasonable performance.**



LIMITING FACTOR: CC-CV **does not consider temperature constraints**, whose satisfaction is crucial for guaranteeing battery safe operations.

Model-based Optimal Charging

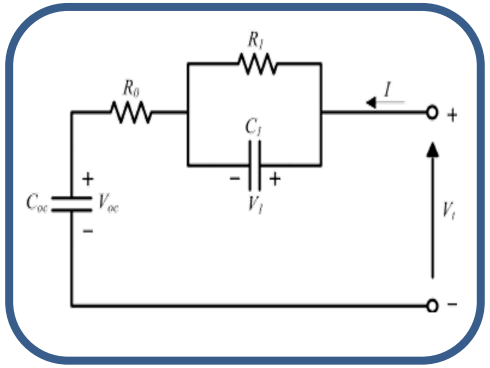
Advanced Battery Management Systems (ABMS) rely on ***mathematical models*** in order to achieve high performance.

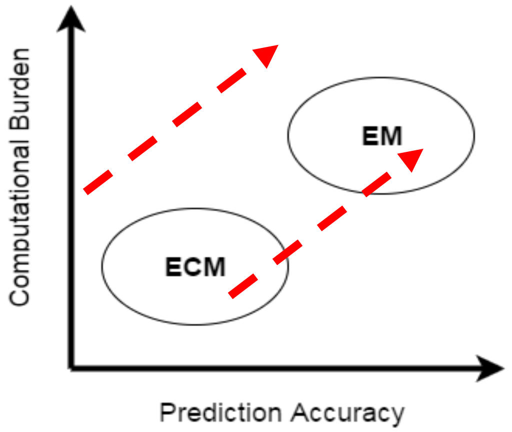


Model-based Optimal Charging

The model choice is fundamental during the advanced BMS design phase.

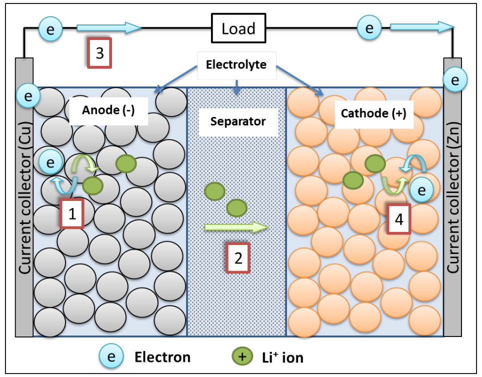
- **Equivalent circuit models (ECM)**





- **Electrochemical models (EM)**

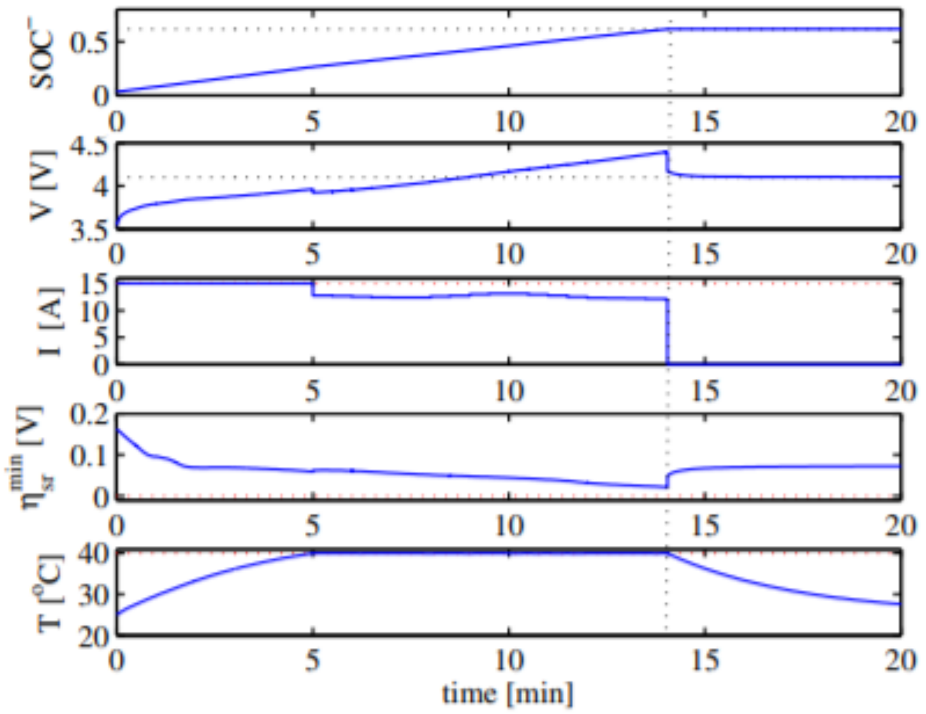
SPMeT	DFN model



Model-based Optimal Charging

The work of **Klein et al. 2011**:

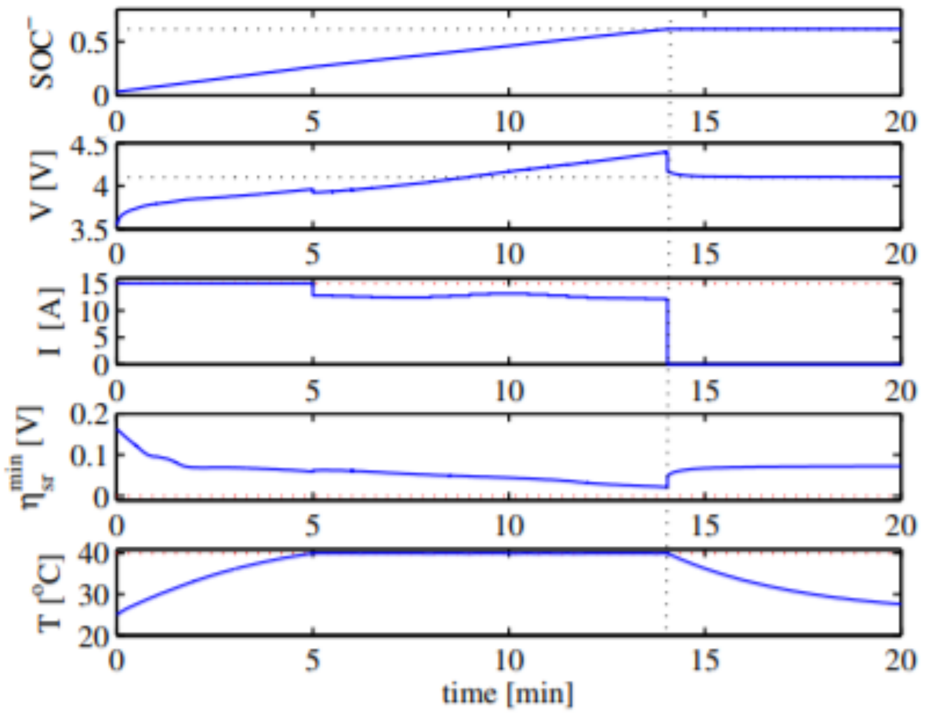
- bounds on temperature and current
- bounds on the side reaction overpotential in order to avoid lithium-ion plating



Model-based Optimal Charging

The work of **Klein et al. 2011**:

- bounds on temperature and current
- bounds on side reaction overpotential in order to avoid lithium ion plating

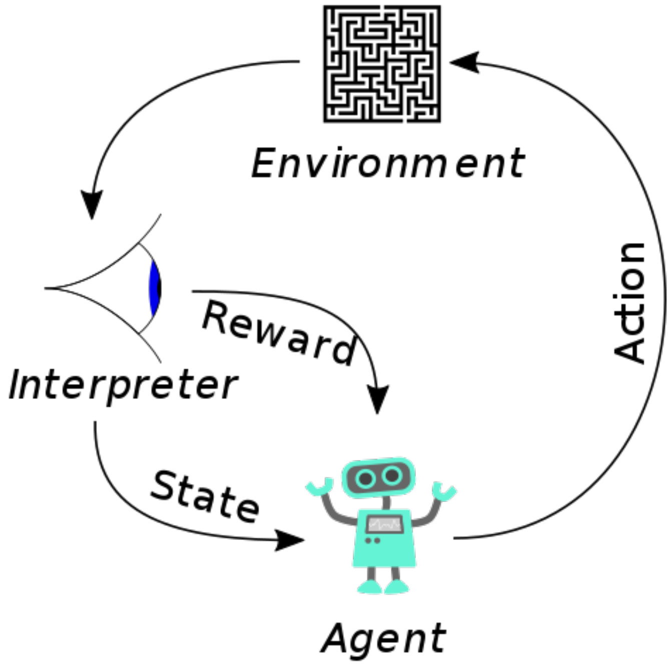


The **overpotential constraint** allows to remove the conservative voltage constraint but requires state estimation because it is **not measurable**.

Model-free Optimal Charging

Solution: exploitation of **model-free** control strategies which are able to provide fast and safe charging while relying on the available measurements.

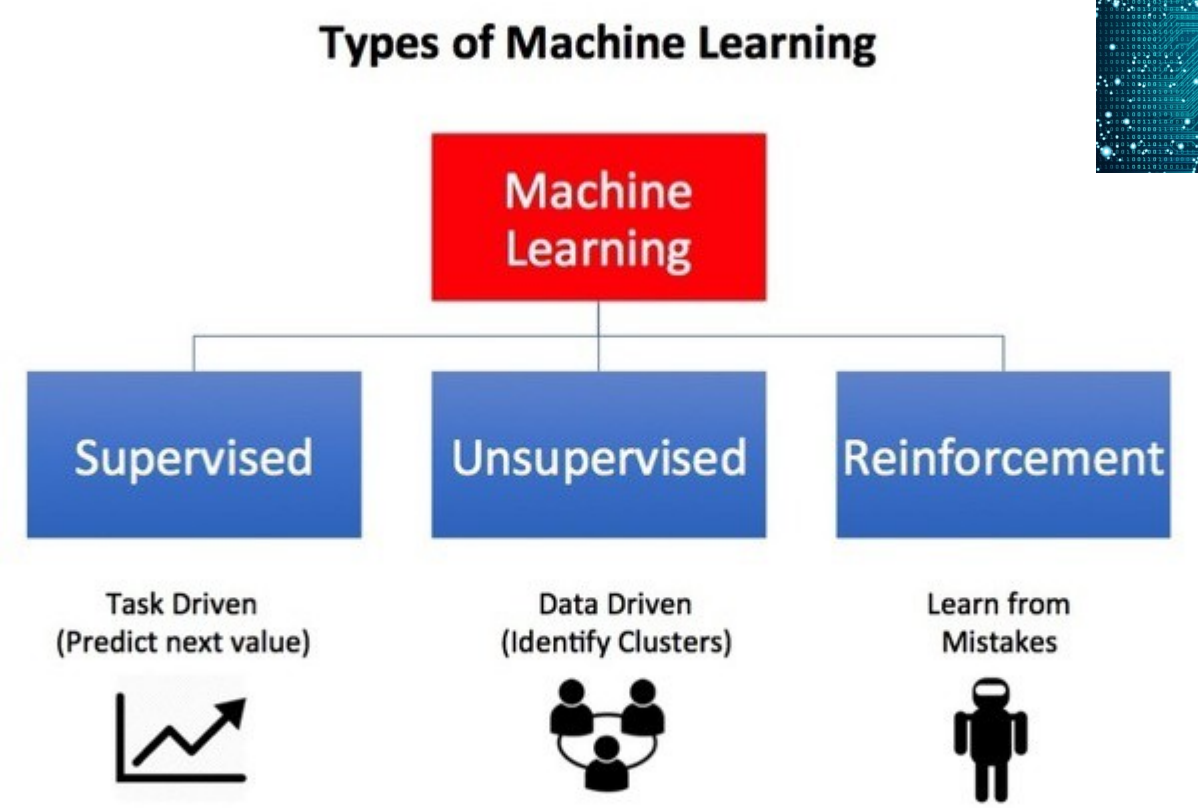
For the **first time** we propose the use of **reinforcement learning (RL)** algorithms for battery cahrging applications.



Agenda

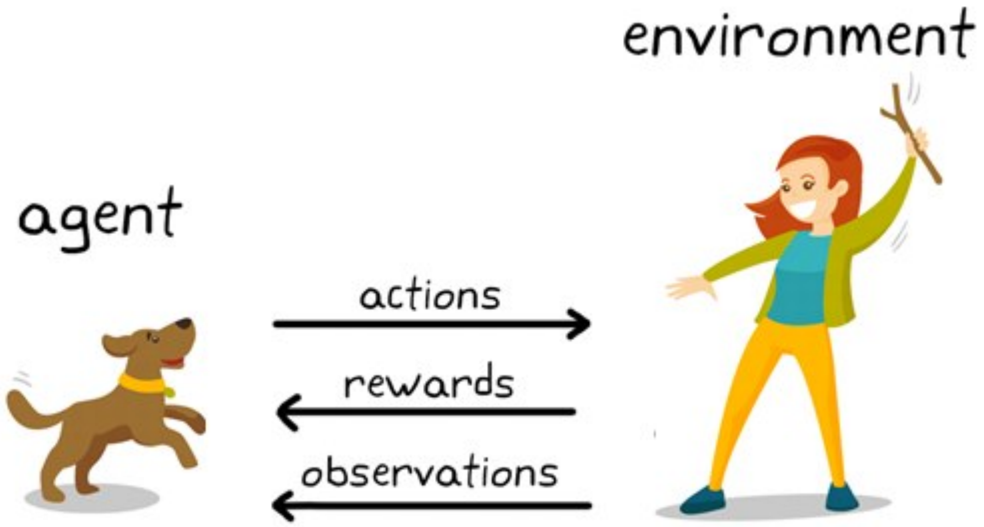
- Battery Overview and Literature Review
- **Reinforcement Learning**
- Battery Model
- Simulation Results

Reinforcement Learning Framework



Reinforcement Learning Framework

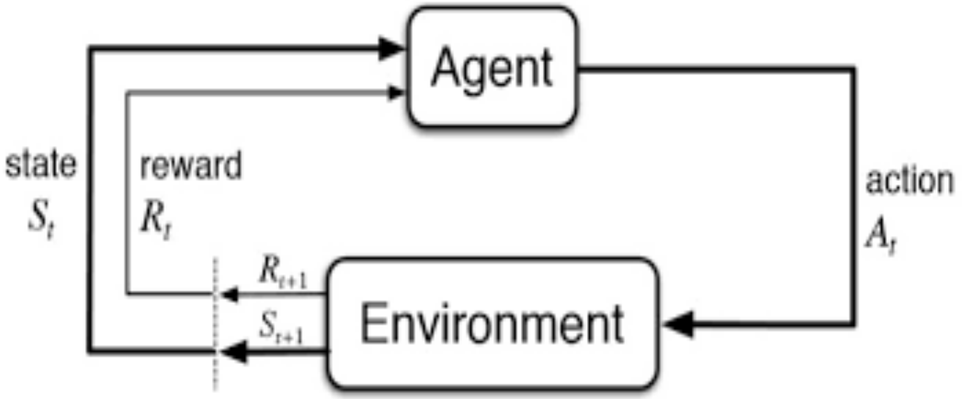
Definition: **reinforcement learning** (**RL**) is an area of **machine learning** concerned with how **agents** ought to take **actions** in an **environment** in order to maximize some notion of cumulative **reward**.



Reinforcement Learning Framework

Consider a Markov Decision Process (MDP):

- S: set of possible states
- A: set of possible actions
- *R*: reward distribution
- *P*: transition probability
- *γ*: discount factor



The agent selects the action according to the **policy** $\boldsymbol{\pi^*: S \rightarrow A}$ which **maximizes the long term expected return (**a.k.a. **state value function)**

$$R_t = \sum_{k=0}^{\infty} \gamma^k r(s_{t+k}, a_{t+k})$$

$$V^{\pi}(s_t) \doteq \mathbb{E}_{r_{i>t},\, s_{i>t} \sim E,\, a_{i \geq t} \sim \pi}\left[R_t \mid s_t\right]$$

State-Action Value Function

The state-action value function corresponds to the long-term expected return when action a_t is taken in state s_t and then the policy $\boldsymbol{\pi}$ is followed henceforth:

$$\boxed{Q^{\pi}(s_t, a_t) \doteq \mathbb{E}_{r_{i>t},\, s_{i>t} \sim E,\, a_{i>t} \sim \pi}\Big[R_t \mid s_t,\, a_t\Big]}$$

The state-action value function can also be expressed by the following recursive relationship also known as **Bellman equation**:

$$Q^{\pi}(s_t, a_t) = \mathbb{E}_{r_{i>t},\, s_{i>t} \sim E}\Big[r(s_t, a_t) + \gamma \mathbb{E}_{a_{t+1} \sim \pi}\left[Q^{\pi}(s_{t+1}, a_{t+1})\right]\Big]$$

Optimal Value Functions and Optimal Policy

By definition the optimal policy is given as:

$$\pi^* = \arg\max_{\pi} V^{\pi}(\boldsymbol{s}_t)$$

If one considers the Q-function:

$$\pi^* = \arg\max_{\boldsymbol{a}_t \in \mathcal{A}} Q^*(\boldsymbol{s}_t, \boldsymbol{a}_t)$$

Q-learning

where the following equation holds:

$$V^*(\boldsymbol{s}_t) = \max_{\boldsymbol{a}_t \in \mathcal{A}} Q^*(\boldsymbol{s}_t, \boldsymbol{a}_t)$$

Different RL algorithms

The main RL algorithms can be divided in two main groups:

- **Tabular methods**: the value functions are expressed using tables whose entrances are states and actions. These approaches are suitable for small and discrete actions and states spaces (curse of dimensionality).

- **Approximate Dynamic Programming (ADP)**: the value functions are represented via approximators (e.g., neural networks in deep reinforcement learning). In particular:

    - Deep Q-learning: discrete set of actions

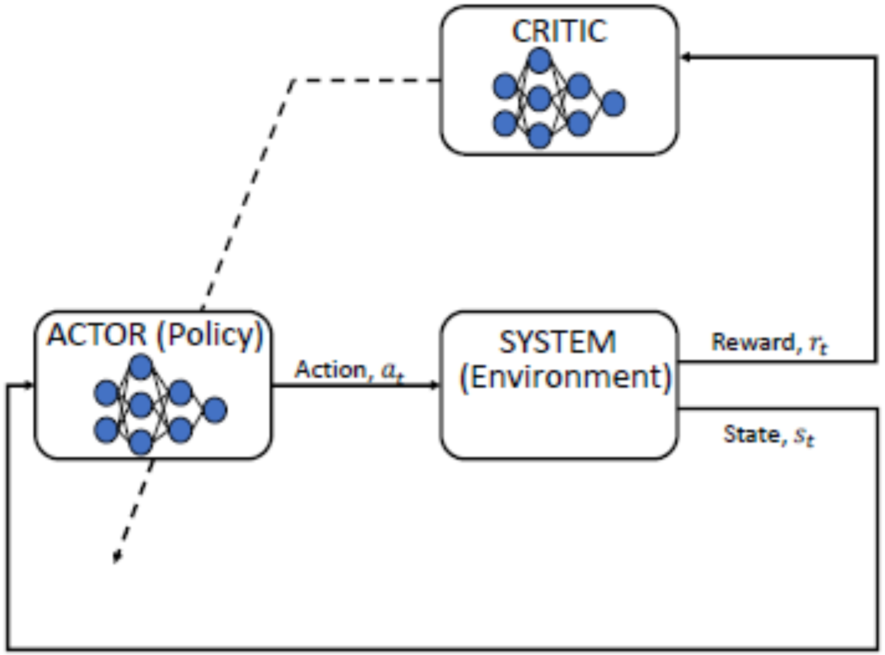
    - Deep Deterministic Policy Gradient: **continuous set of actions**

Deep Deterministic Policy Gradient: actor-critic

The DDPG algorithm is based on the **actor-critic** paradigm.

Actor-critic methods learn approximations to both policy and value functions:

- **actor** is a reference to the learned policy
- **critic** refers to the learned value function



Deep Deterministic Policy Gradient: algorithm

Algorithm 1 DDPG algorithm

Randomly initialize critic network $Q(s,a|\theta^Q)$ and actor $\mu(s|\theta^\mu)$ with weights θ^Q and θ^μ.
Initialize target network Q' and μ' with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$
Initialize replay buffer R
for episode = 1, M **do**
  Initialize a random process $\mathcal{N}$ for action exploration
  Receive initial observation state s_1
  for t = 1, T **do**
    Select action $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$ according to the current policy and exploration noise
    Execute action a_t and observe reward r_t and observe new state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in R
    Sample a random minibatch of N transitions (s_i, a_i, r_i, s_{i+1}) from R
    Set $y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$
    Update critic by minimizing the loss: $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^Q))^2$
    Update the actor policy using the sampled policy gradient:

$$\nabla_{\theta^\mu} J \approx \frac{1}{N}\sum_i \nabla_a Q(s,a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)|_{s_i}$$

    Update the target networks:

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}$$

$$\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$$

  end for
end for

Lillicrap et al. 2016.

Deep Deterministic Policy Gradient: exploration

The exploration is performed by **adding a noise** to the action computed by the actor.

$$a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$$

During the testing phase of the strategy the exploration noise is removed.

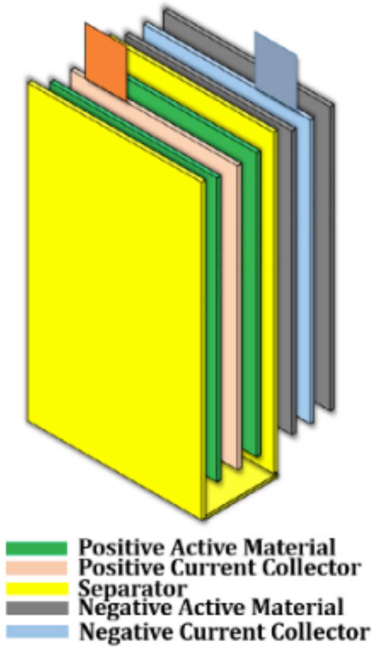
$\boxed{\text{GREEDY POLICY}}$

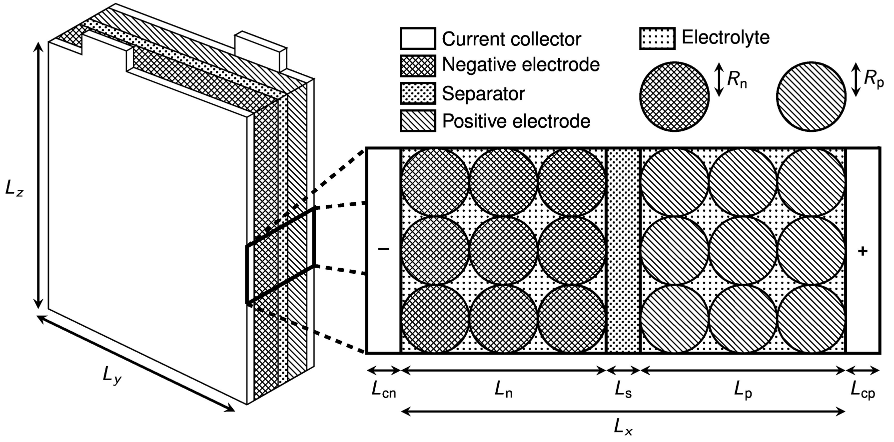
Agenda

- Battery Overview and Literature Review
- Reinforcement Learning
- **Battery Model**
- Simulation Results

Li-ion Battery

- **Battery Modeling**





Sources:
http://www.maths.ox.ac.uk/node/34037
Goutam, Shovon, et al. "Three-dimensional electro-thermal model of Li-ion pouch cell: Analysis and comparison of cell design factors and model assumptions." *Applied thermal engineering* 126 (2017): 796-808.

Electrochemical Model

- **Single Particle Model w/ Electrolyte and Thermal (SPMeT)**
  - Reduced-Order Model

Governing Equations

1. Solid-phase dynamics (PDE)
2. Electrolyte-phase dynamics (PDE)
3. Thermal dynamics (ODE)
4. Voltage output

$$\frac{\partial c_s^{\pm}}{\partial t}(r,t) = \frac{1}{r^2}\frac{\partial}{\partial r}\left[D_s^{\pm} r^2 \frac{\partial c_s^{\pm}}{\partial r}(r,t)\right]$$

$$\varepsilon_e^j \frac{\partial c_e^j}{\partial t}(x,t) = \frac{\partial}{\partial x}\left[D_e^{eff}\left(c_e^j\right)\frac{\partial c_e^j}{\partial x}(x,t) + \frac{1-t_c^0}{F} i_e^j(x,t)\right] \qquad j \in \{-, sep, +\}$$

$$\frac{dT_{cell}}{dt}(t) = \frac{\dot{Q}(t)}{mC_p} - \frac{T_{cell}(t) - T_{\infty}}{mC_p R_{th}}$$

$$V_T(t) = \frac{RT_{cell}(t)}{\alpha F}\sinh\left(\frac{I(t)}{2a^+AL^+i_0^+(t)}\right) - \frac{RT_{cell}(t)}{\alpha F}\sinh\left(\frac{-I(t)}{2a^-AL^-i_0^-(t)}\right)$$
$$+U^+\left(c_{ss}^+(t)\right) - U^-\left(c_{ss}^-(t)\right) + \cdots$$

Electrochemical Model-based Controls

- **Optimal Control Problem**
  - Based on the physical information, we can **design** an optimal controller for **Fast-Charging.**
  - Fast-charging problem is **"Constrained Minimum-Time Optimal Control Problem"**

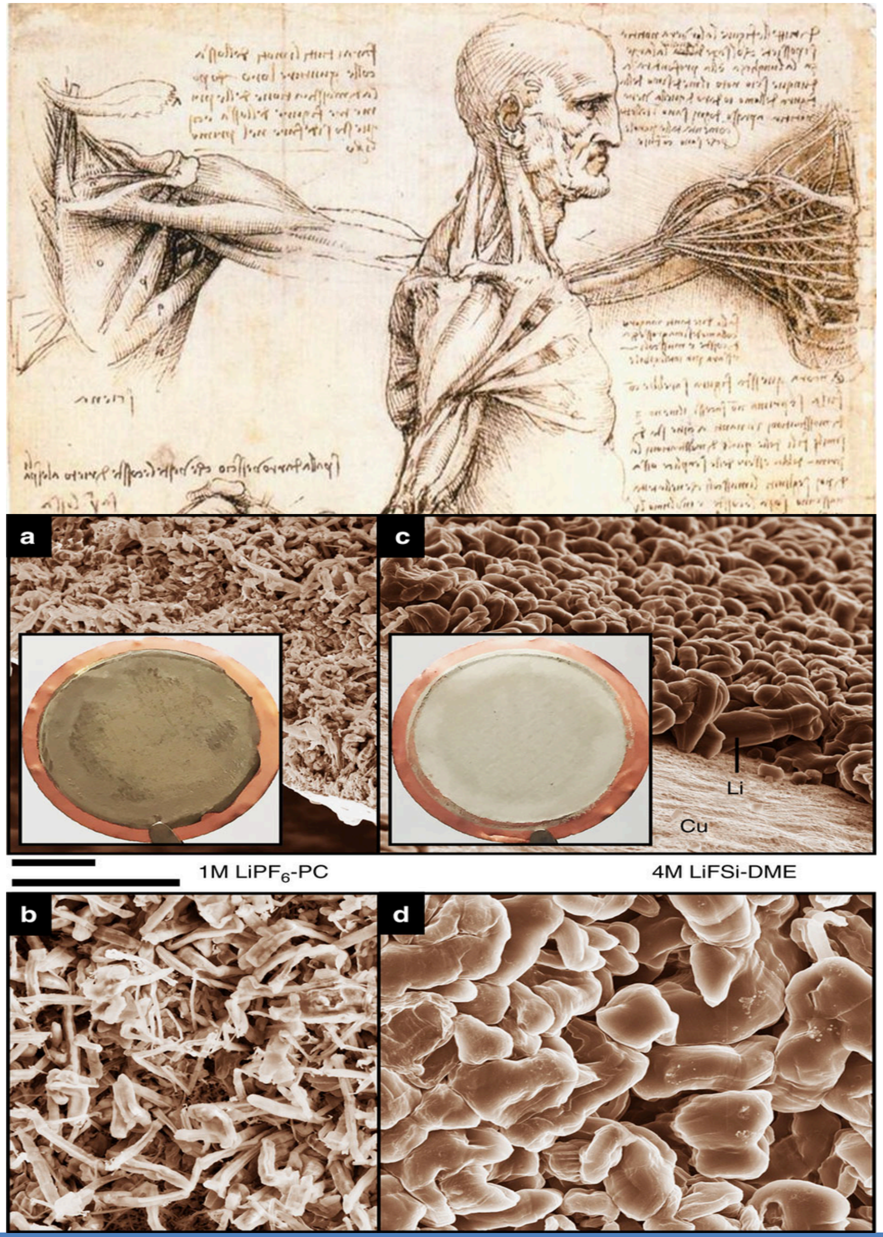
$$\min_{I(t),t_f} \sum_{t=t_0}^{t_f} 1$$

subject to

$$\begin{aligned}
&\text{battery dynamics in (17)-(22)} \\
&V_T(t_0) = V_0,\ T_{\text{cell}}(t_0) = T_0 \\
&SOC(t_f) = SOC_{\text{ref}},\ I(t) \in \left[I^{\min},\ I^{\max}\right] \\
&V_T(t) \le V_T^{\max},\ T_{\text{cell}}(t) \le T_{\text{cell}}^{\max}
\end{aligned}$$

Electrochemical Model

- **Challenges**
  - Electrochemical model is partially observable system
    - Limited measurements
    - Model complexity
  - Battery model changes over time
    - Aging
  - Discretizing PDEs results in large scale systems
    - Numerical challenges
  - Proving optimality of control is almost impossible
    - Curse of dimensionality



Goal

- Validate RL-framework for battery fast-charging problem

Research Questions

- Does RL learn “constrained optimal control” ?
- Does RL adapt its policy as the environment changes ?

Agenda

- Battery Overview and Literature Review
- Reinforcement Learning
- Battery Model
- **Simulation Results**

Fast Charging Problem

The fast charging problem is formulated as a **constrained optimization program**:

$$\min_{I(t),t_f} \sum_{t=t_0}^{t_f} 1$$

subject to

$$\begin{aligned}
&\text{battery dynamics} \\
&V_T(t_0) = V_0,\ T_{\text{cell}}(t_0) = T_0 \\
&SOC(t_f) = SOC_{\text{ref}},\ I(t) \in [I^{\min}, I^{\max}] \\
&V_T(t) \le V_T^{\max},\ T_{\text{cell}}(t) \le T_{\text{cell}}^{\max}
\end{aligned}$$

We consider a voltage constraint instead of the one on the side reaction overpotential since it is easier to check its violation in a realistic scenario.

Reward Design

The reward function is designed in order to achieve the required goal:

$$r_{t+1} = \boxed{r_{\text{fast}}} + \boxed{r_{\text{safety}}(\boldsymbol{s}_t, \boldsymbol{a}_t)}$$

with

$$r_{\text{fast}} = -0.1$$

$$r_{\text{safety}}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \boxed{r_{\text{volt}}(\boldsymbol{s}_t, \boldsymbol{a}_t)} + \boxed{r_{\text{temp}}(\boldsymbol{s}_t, \boldsymbol{a}_t)}$$

$$r_{\text{volt}}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \begin{cases} -100(V_{\text{T}}(t) - V_{\text{T}}^{\max}), & \text{if } V_{\text{T}}(t) \geq V_{\text{T}}^{\max} \\ 0, & \text{otherwise} \end{cases}$$

$$r_{\text{temp}}(\boldsymbol{s}_t, \boldsymbol{a}_t) = \begin{cases} -5(T_{\text{cell}}(t) - T_{\text{cell}}^{\max}), & \text{if } T_{\text{cell}}(t) \geq T_{\text{cell}}^{\max} \\ 0, & \text{otherwise} \end{cases}$$

Full and Reduced States

We perform two different simulations.

- Firstly, **all the states of the SPMeT (61)** are assumed to be measurable (solid phase concentration, electrolyte concentration and temperature).

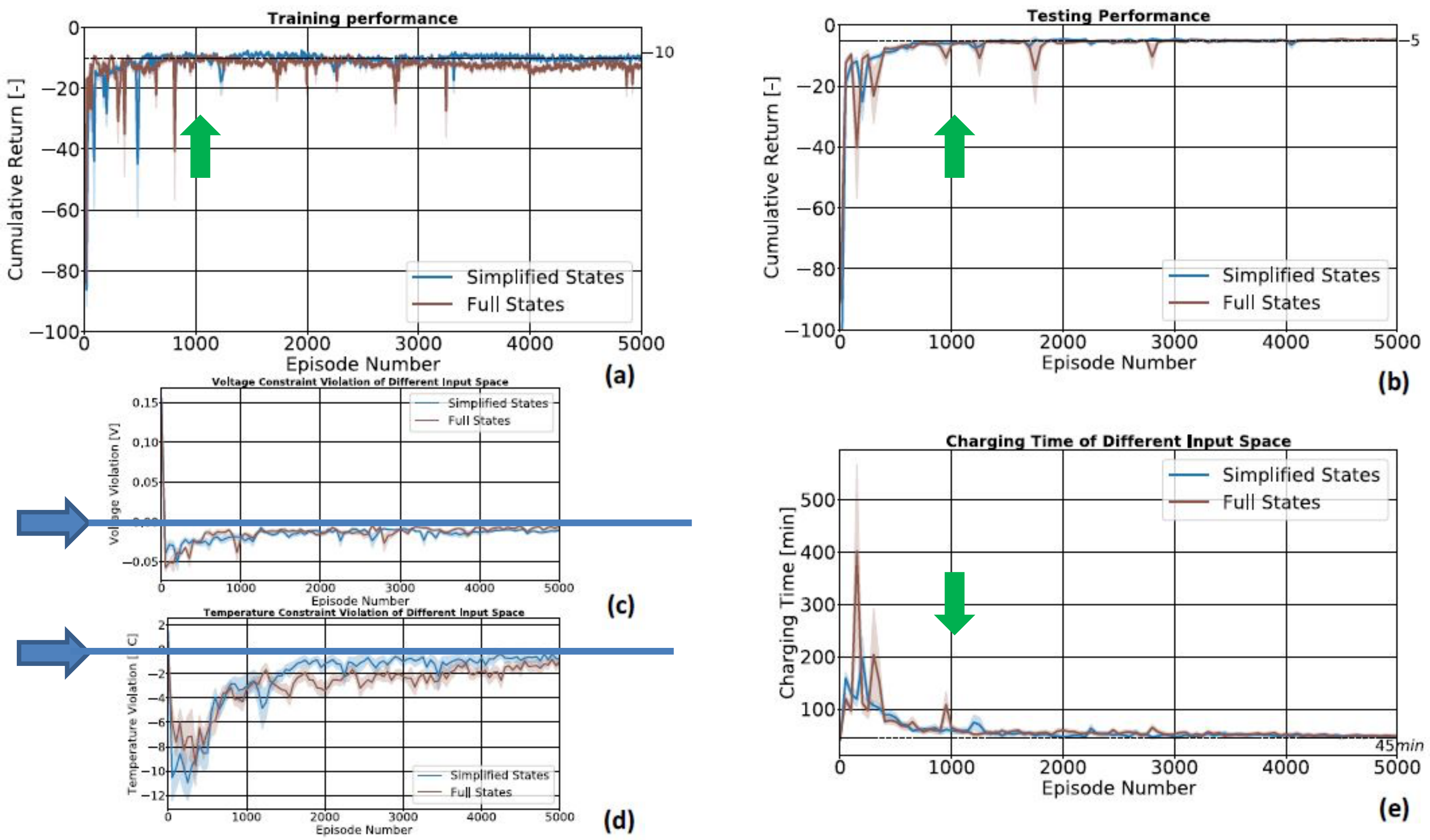
Issue: a suitable **model-based** state observer is required for applying this procedure in a realistic framework.

Solution: we drop the assumption of availability of all the states and we considered only **2 states**

- SOC and temperature.

The **results are surprisingly similar** to the ones obtained by considering the whole states vector.

Results of the Learning Process



Validation of the Optimal Strategy

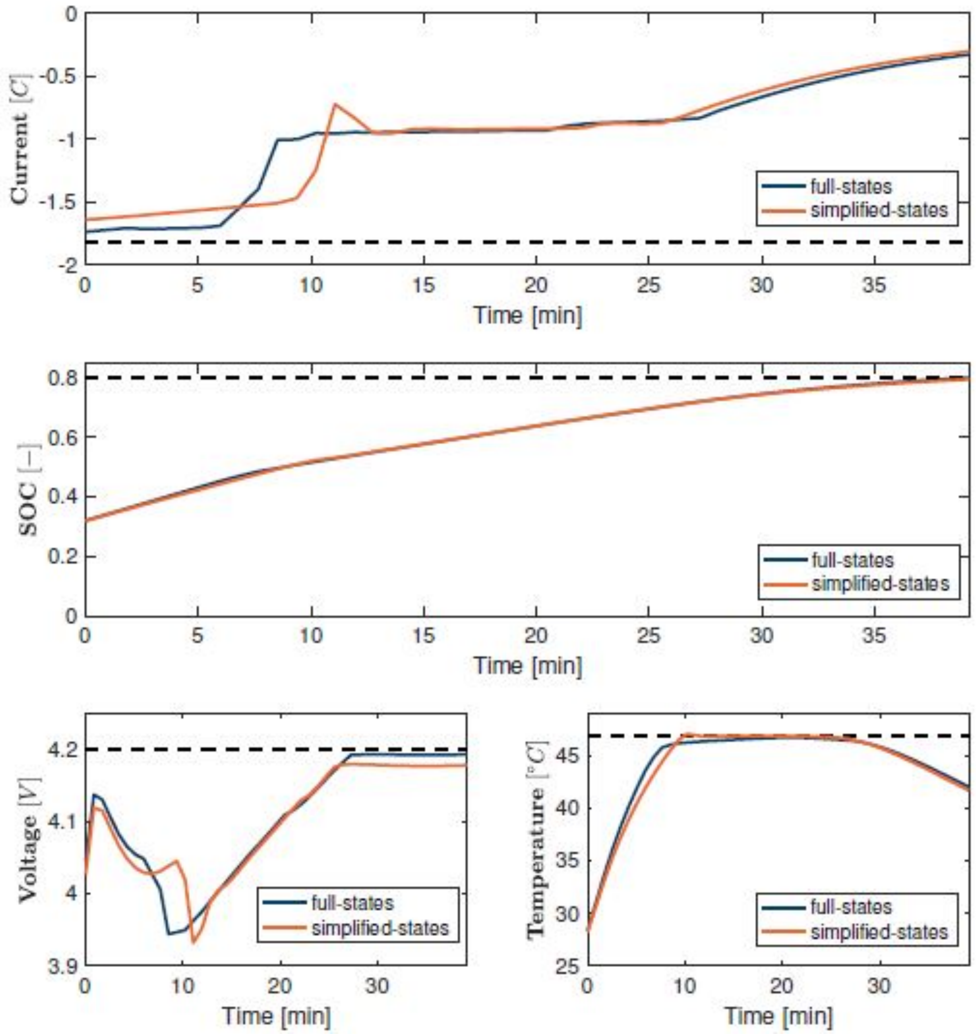
Initial condition of $3.6\,V$ and $27°C$ ($SOC = 0.3$).

The **charging time** is $40\ min$ for both the approaches (full and reduced states).

The obtained reward is also similar:

- -5.38 reduced states
- -4.69 full states

The constraints ($V_{\max} = 4.2\,V$ and $T_{\max} = 47°C$) are not violated.



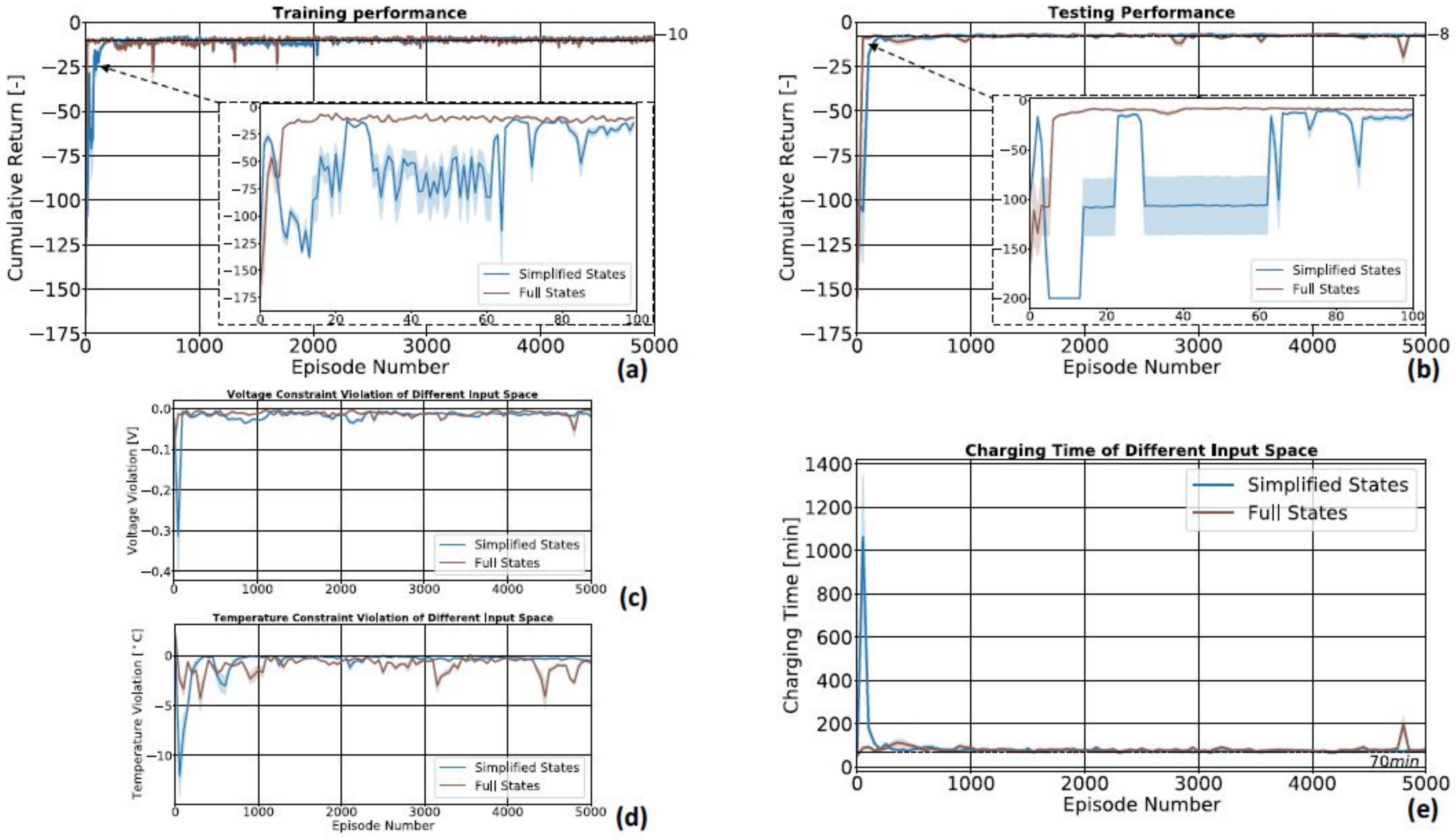
Online Adaptation to Environment Changes

Consider the Possibility of a **variation in the environment** parameters (e.g. ageing in Lithium-Ion batteries).

How does the proposed approach perform?

We consider an increase in the **film resistance** ($R_{f,p}$ and $R_{f,n}$) and in the **heat generation** ($\dot{Q}$).

Results of the Learning Process – Online Adaptation



Validation of the Optimal Strategy – Online Adaptation

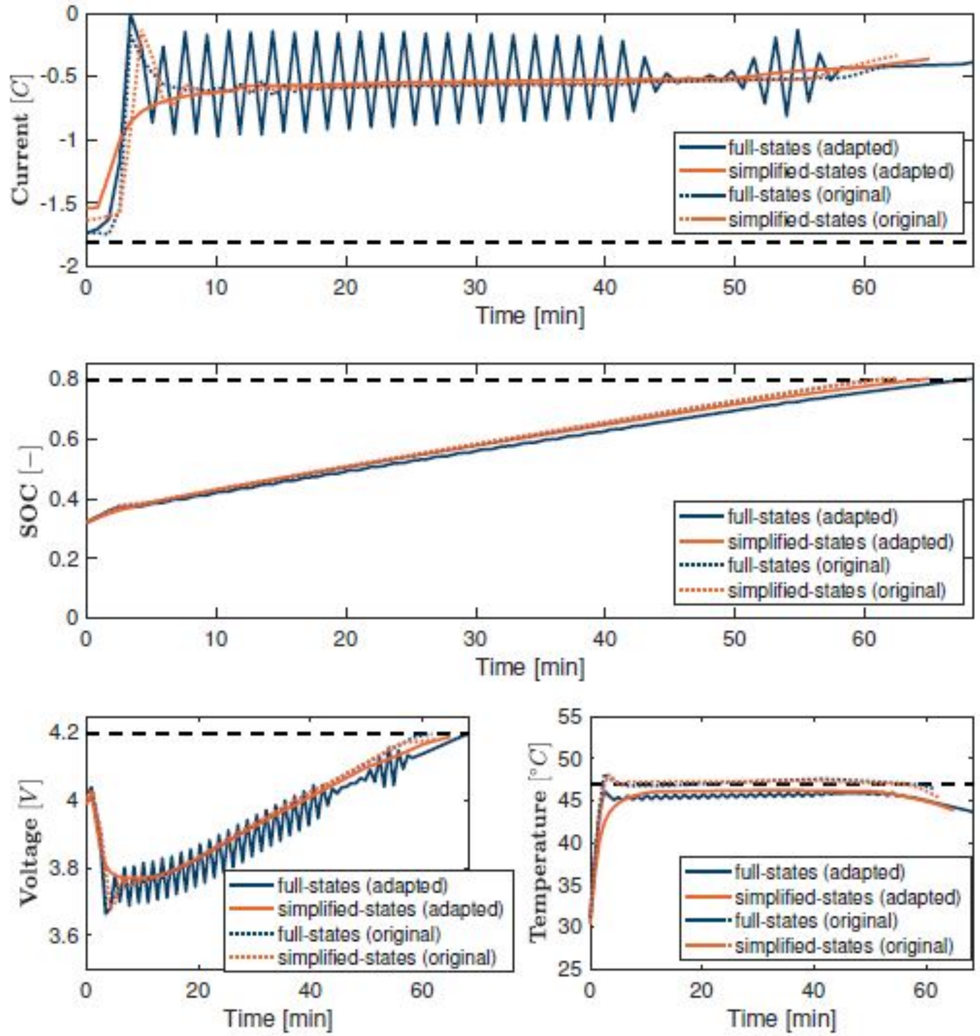
Initial condition of $3.6\,V$ and $27°C$ $(SOC = 0.3)$.

The **charging time** is $66\,min$ for the reduced states approach and $68\,min$ for the full one.

The obtained reward is also similar:

- -7.79 reduced states
- -8.19 full states

The constraints ($V_{\max} = 4.2\,V$ and $T_{\max} = 47°C$) are not violated.



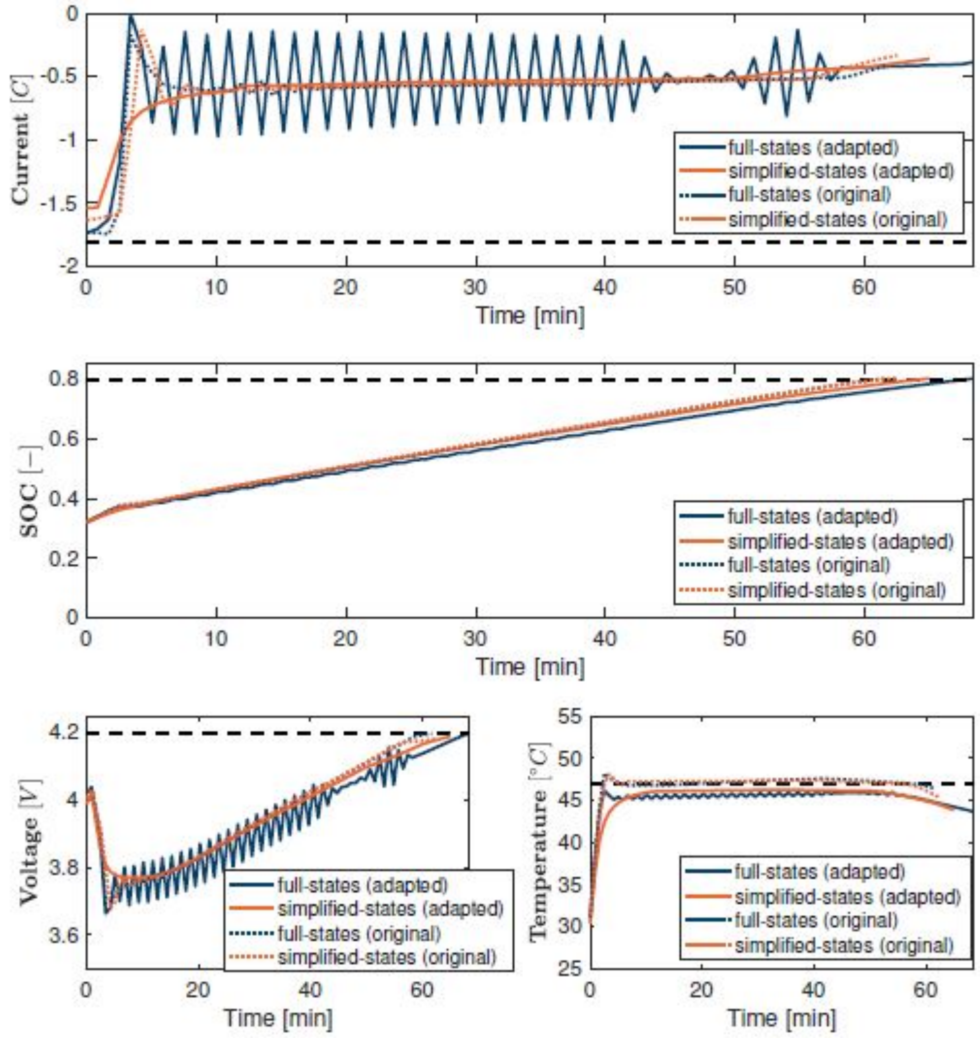
Validation of the Optimal Strategy – Online Adaptation

With the original policy without ageing adaptation the constraints are slightly violated.

This implies faster charging but also lower reward:

- -81.78 reduced states
- -82.16 full states

Finally, oscillations in the applied input current can be reduced with a **regularization term**.



Conclusion & Future Work

- **Validation of RL framework for Fast-charging**
- **Design of Full-states vs Reduced-states feedback controller**
- **Experimental validation**
- **Full-order model (P2D) model with electrochemical constraints.**

Thank you very much for your attention!!

Suggestions, questions and advices are welcomed!