Aprendizado por Reforço: conceitos, aplicações e desafios

ANNA HELENA REALI COSTA
UNIVERSIDADE DE SÃO PAULO
anna.reali@usp.br

Parte 3

RL Algorithms

- **Value learning**:

Find **Q(s,a)**

Then chose a:

$$a = \arg\max_a Q(s,a)$$

- **Policy learning**

Find $\boldsymbol{\pi(a|s)}$

Then sample a:

$$a \sim \pi(a|s)$$

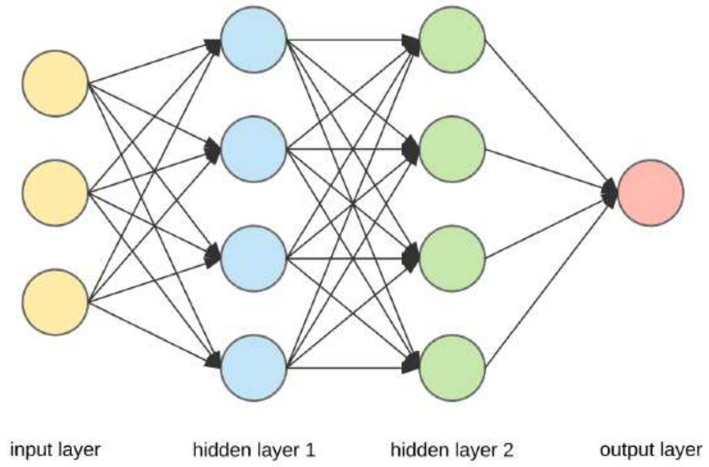
Q representations

Q(s,a)

	a_1	a_2	…..	a_n	$\pi(s)$
s_1					a_i
s_2					a_j
					a_k
…					
s_m					

enumerated states

State features and Action a

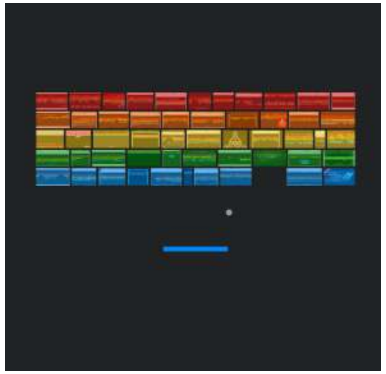


Q(s,a)

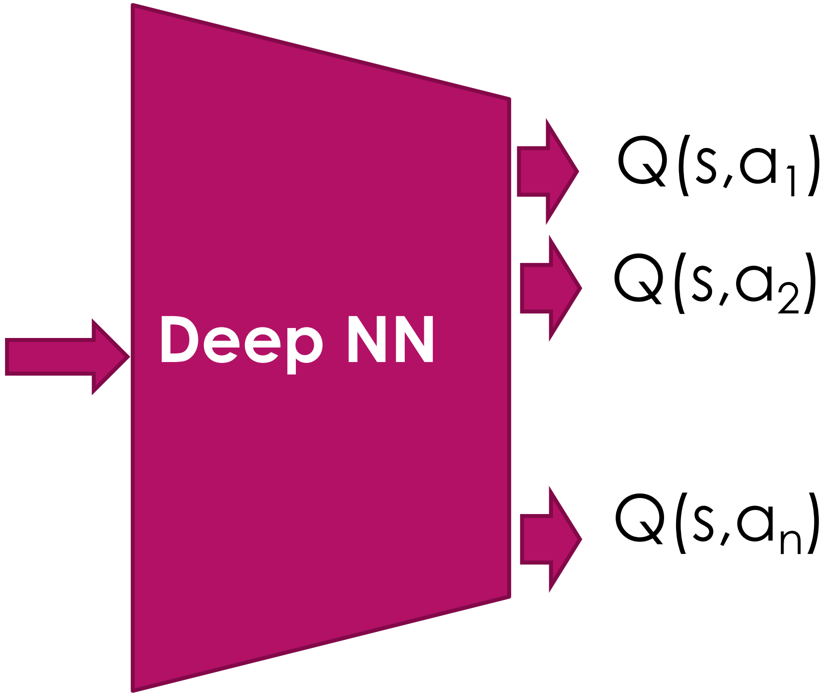
factored states

Deep Q Networks (DQN)

DNN to model Q-functions



State
(discrete or
continuous)



🙁 Cannot handle continuous action spaces

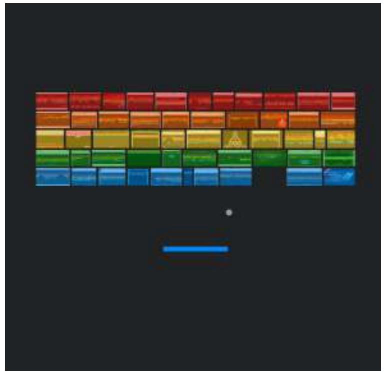
🙁 Cannot learn stochastic policies

$$\mathcal{L} = E\left[\left\| \underbrace{\left(r + \gamma \max_{a'} Q(s', a')\right)}_{\text{target}} - \underbrace{Q(s, a)}_{\text{predicted}} \right\|^2\right]$$

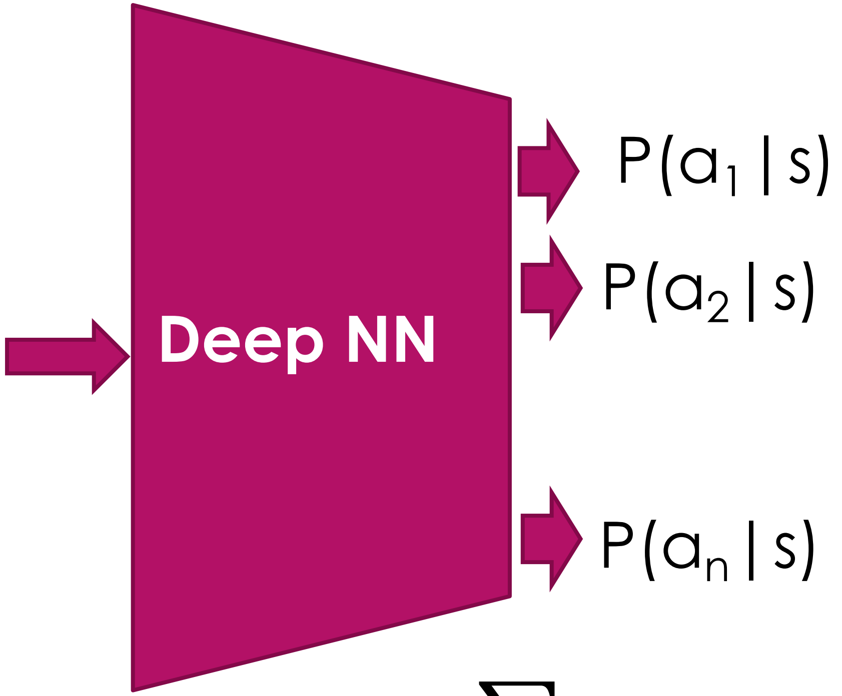
Policy gradient in RL

- **Policy gradient** methods are a type of **reinforcement learning** techniques that rely upon optimizing parametrized **policies** with respect to the expected return (long-term cumulative reward) by **gradient** descent.

PG and stochastic policies



State (discrete or continuous)



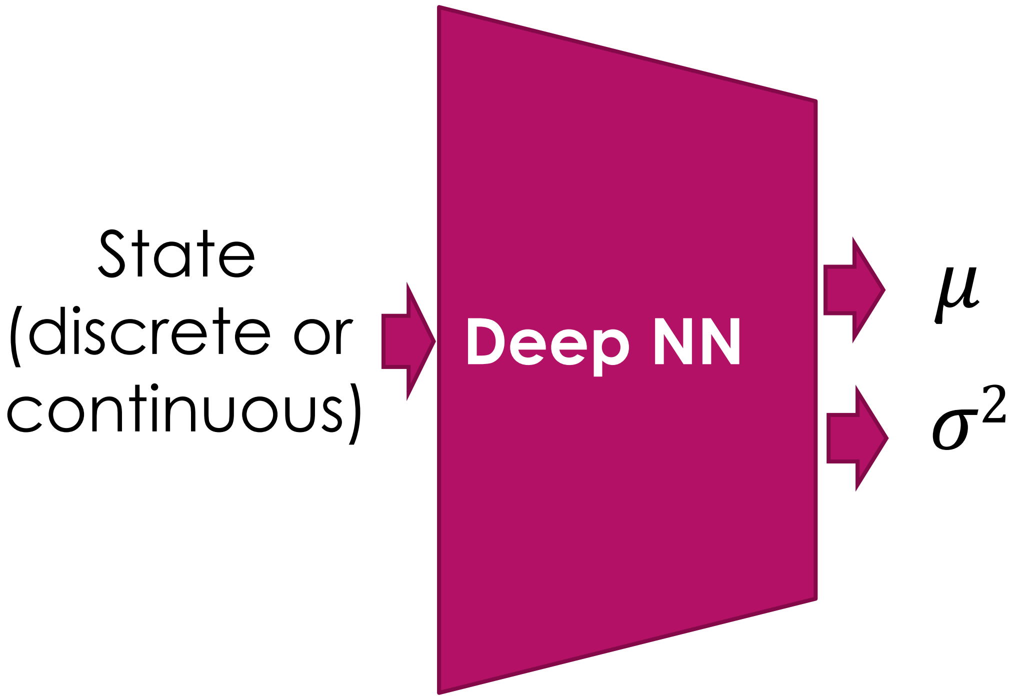
$$\sum_{a_i \in A} P(a_i|s) = 1$$

😃 Can learn stochastic policies

Policy gradient methods and DRL

- With PG methods we can handle continuous state and action spaces

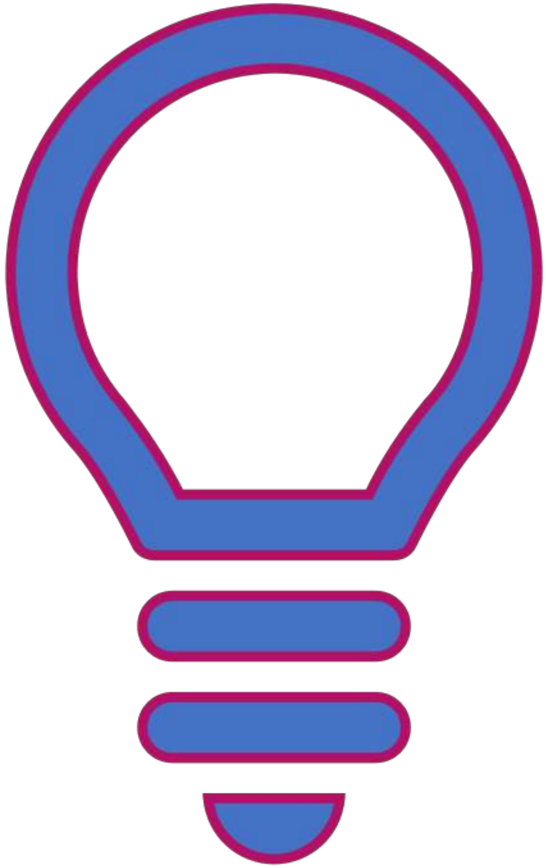
1. Run a policy for a while
2. Increase probability of actions that lead to high rewards
3. Decrease probability of actions that lead to low/no rewards



$$P(a\,|\,s) = N(\mu, \sigma^2)$$

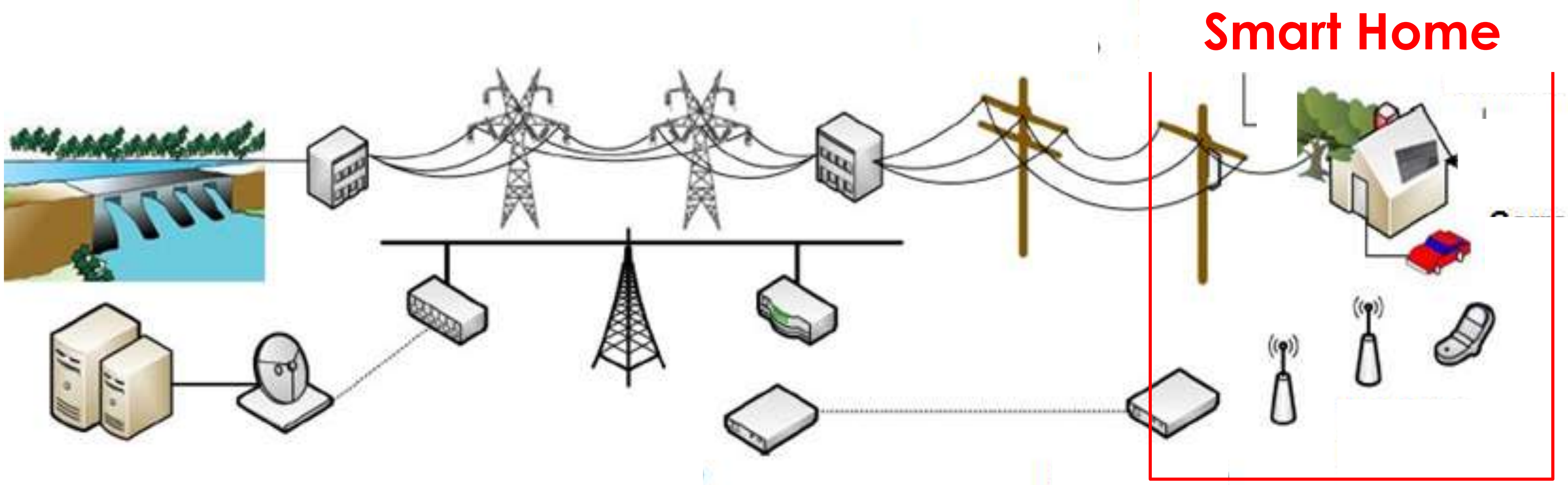
DQN x PG

	DQN	PG
Complex Q-function	Not OK	OK
Convergence Speed	Slow	Fast
Training Stability	More stable	Less stable
Stochastic Policies	Not OK	OK
Continuous Actions	Not OK	OK
Data Amount	Needs less data	Needs more data



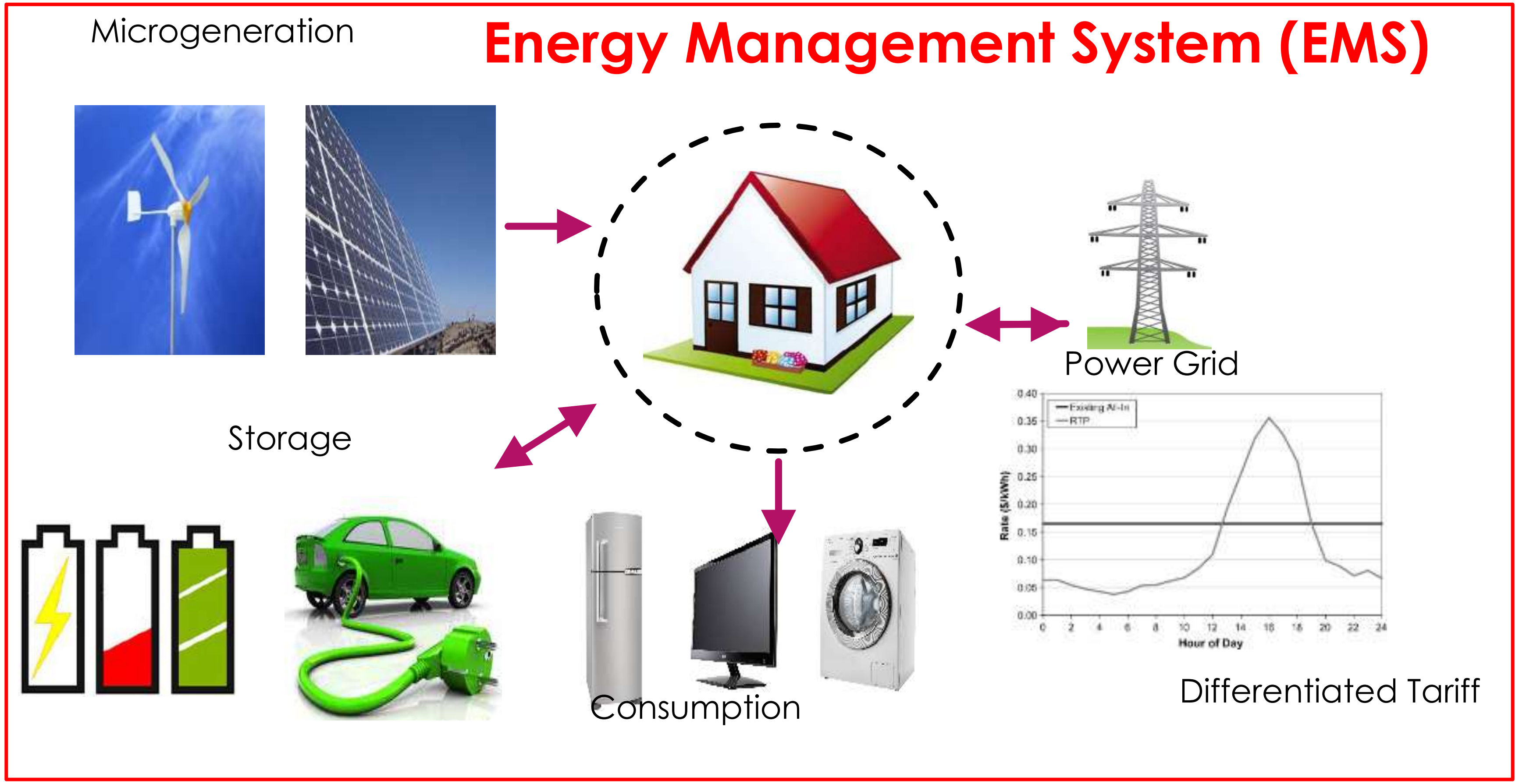
Some Applications

Energy Management with Batch-RL



Berlink, H; Reali Costa, AH. Batch reinforcement learning for smart home energy management, IJCAI 2015.

Energy Management with Batch-RL



Berlink, H; Reali Costa, AH. Batch reinforcement learning for smart home energy management, IJCAI 2015.

Energy Management with Batch-RL

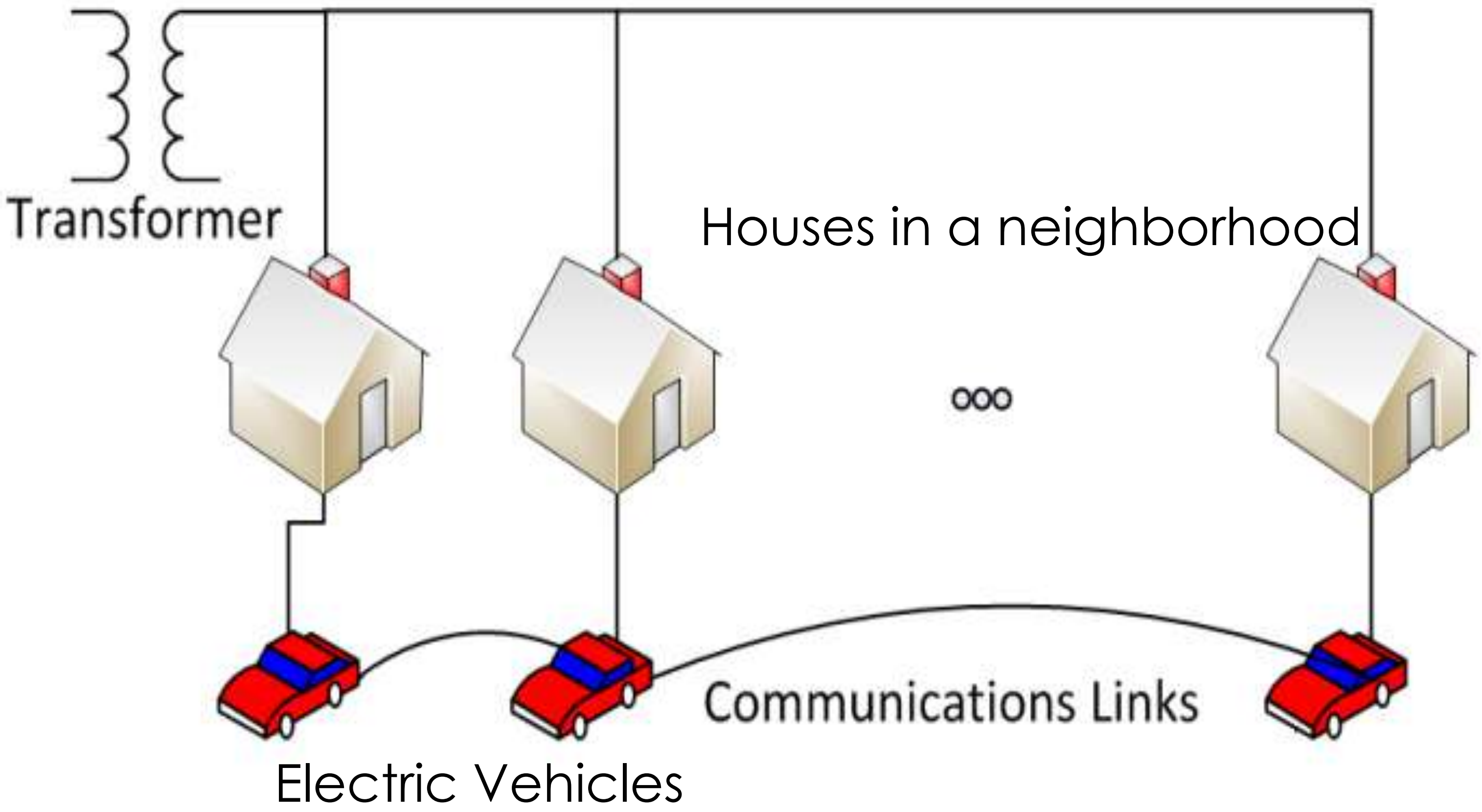
% Increase of the Financial Profit (compared to the Naive-greedy Policy):

Brasil (TOU): 20.78%

USA (RTP): 14.51%

Berlink, H; Reali Costa, AH. Batch reinforcement learning for smart home energy management, IJCAI 2015.

Electric Vehicle Charge with Distributed MCRL



- Battery charge for daily journey
- No transformer overload

Silva, FL; Nishida, CEH; Roijers, DM; Costa, AHR. *Coordination of Electric Vehicle Charging through Multiagent Reinforcement Learning, IEEE Trans. Smart Grid 2019.*

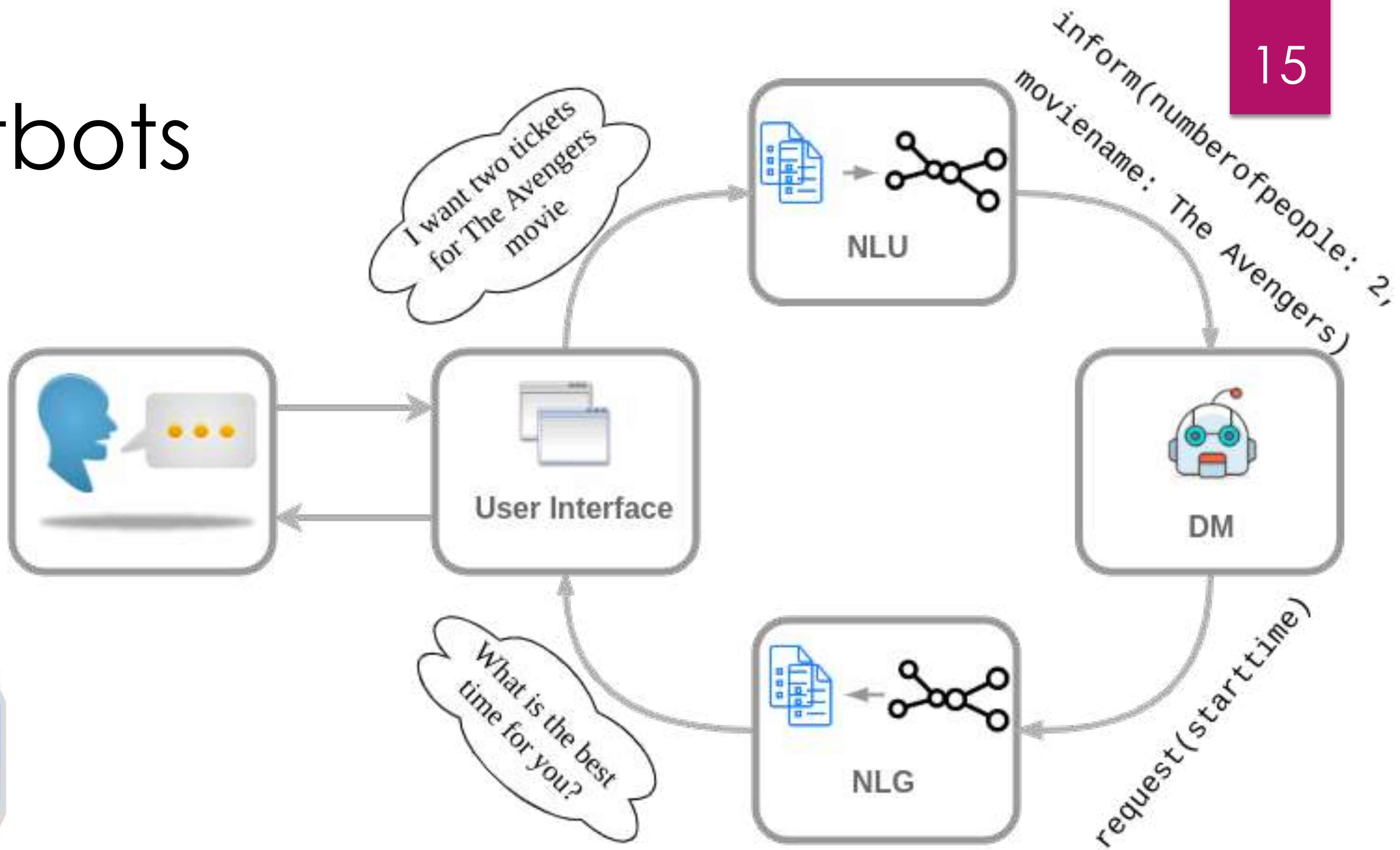
Multi Agent Selfish-Collaborative (MASCO)

- Average energy costs and number of overloads per day
  - ACP: Always Charging when Plugged
  - MASCO: Minimizes costs while avoiding overloads

Danish Tariff	costs	overloads
ACP	0.781 ± 0.003	8.40 ± 0.21
MASCO	$\mathbf{0.633 \pm 0.010}$	$\mathbf{3.76 \pm 0.67}$
Brazilian Tariff	**costs**	**overloads**
ACP	4.07 ± 0.01	8.40 ± 0.21
MASCO	$\mathbf{2.90 \pm 0.07}$	$\mathbf{1.08 \pm 0.58}$

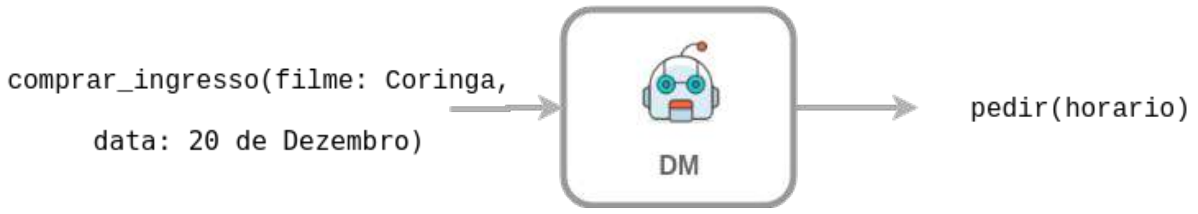
Silva, FL; Nishida, CEH; Roijers, DM; Costa, AHR. *Coordination of Electric Vehicle Charging through Multiagent Reinforcement Learning, IEEE Trans. Smart Grid 2019.*

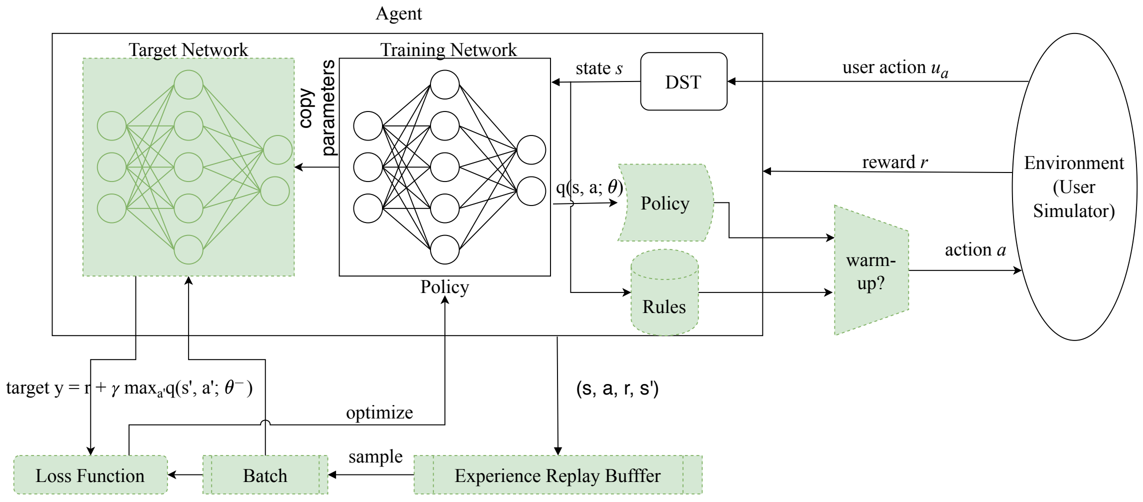
Chatbots

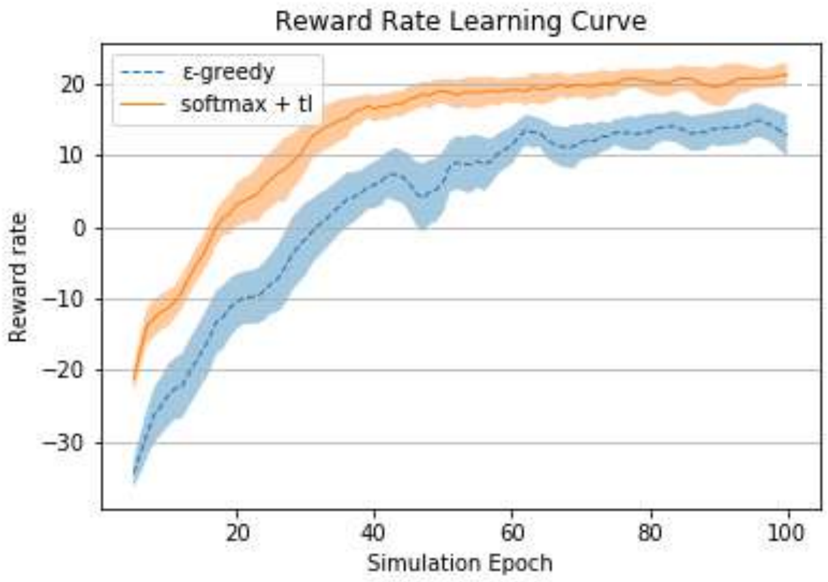


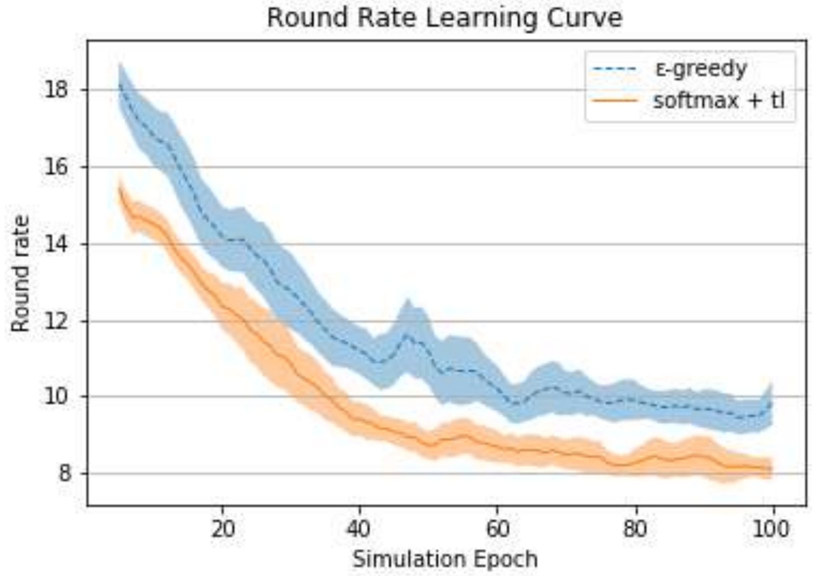
Itaú

B. Nishimoto and A.H. Reali Costa. DEEP-DIAL20 at AAAI 2020.


comprar_ingresso(filme: Coringa,
data: 20 de Dezembro)
DM
pedir(horario)


Agent
Target Network
Training Network
copy parameters
Policy
state s
DST
user action u_a
reward r
Environment (User Simulator)
q(s, a; θ)
Policy
Rules
warm-up?
action a
target y = r + γ max_a' q(s', a'; θ⁻)
optimize
(s, a, r, s')
sample
Loss Function
Batch
Experience Replay Bufffer


Reward Rate Learning Curve
ε-greedy
softmax + tl
Reward rate
Simulation Epoch


Round Rate Learning Curve
ε-greedy
softmax + tl
Round rate
Simulation Epoch