

# Learning from largely suboptimal teachers and the role of compliance

Louis Faury

Advisors : Mahdi Khoramshahi & Andrew Sutcliffe

Semester Project at LASA

April 30, 2017

# Plan

## 1 Motivations

## 2 Background

- Reinforcement learning
- Imitation learning
- Transfer learning

## 3 Settings & Results

- Markov Decision Process
- Compliance-based learning
- Method comparison

## 4 Future work

## Plan

- 1 Motivations
- 2 Background
- 3 Settings & Results
- 4 Future work

## ■ An example :

- ▶ A human teacher is showing the robot to reach for an object
- ▶ The teacher is not a robotic expert, and guides the robot along a trajectory near the edge of the robot's workspace, or very close to some obstacles
- ▶ Should the robot discard the demonstration ?
- ▶ There is still some valuable information in the demonstration (pose of the object, general direction of motion, ..)

- ▶ How to take the teacher's demonstration into account?
  - ▶ Exactly reproduce the teacher's actions
  - ▶ Use demonstration data to build a representation of the environment's dynamics
  - ▶ **Use the teacher demonstration as an exploration baseline**
- ▶ Child learning to dance : first follow its dance teacher moves, before trying out new ones once he feels he has exploited the teacher's recommendation
  - ⇒ notion of **compliance** w.r.t the teacher.

## ■ Goal :

- ▶ Introduce a theoretical framework for compliance-based learning
- ▶ Grasp ideas and intuition about how such an approach can
  - ▶ Overcome a mentor's (large) sub-optimality.
  - ▶ Speed up the learning
  - ▶ Generalize to *transfer learning*

in a **reinforcement learning framework**.

## ■ Method :

- ▶ Create a simple but generic Markov Decision Process
- ▶ Solve it using classical RL method
- ▶ Implement compliant-based learning methods

# Plan

## 1 Motivations

## 2 Background

- Reinforcement learning
- Imitation learning
- Transfer learning

## 3 Settings & Results

## 4 Future work

## ■ RL :

- ▶ Framework in which an agent (or a learner) learns its actions from interaction with its environment
- ▶ The environment generates scalar values called rewards, that the agent is seeking to maximize over time.

Under a Markovian assumption for the dynamics and reward system, the reinforcement learning problem can be formulated as a *Markov Decision Process* :

$$(\mathcal{S}, \mathcal{A}(\mathcal{S}), \mathcal{P}_{ss'}^a, \mathcal{R}_{ss'}^a) \quad (1)$$

where :

$$\begin{aligned} \mathcal{P}_{ss'}^a &= \underbrace{\mathbb{P}(s_{t+1} = s' \mid s_t = s, a_t = a)}_{\text{dynamics}} & \mathcal{S} : \text{state space} \\ \mathcal{R}_{ss'}^a &= \underbrace{\mathbb{E} [r_t \mid s_{t+1} = s', s_t = s, a_t = a]}_{\text{immediate reward}} & \mathcal{A}(\mathcal{S}) : \text{action space} \end{aligned} \quad (2)$$



## ■ RL :

- ▶ Define state value and action value function under a policy (probabilistic decision rule)  $\pi : \mathcal{S} \rightarrow \mathcal{A}$  :

$$\begin{aligned} V^\pi(s) &= \mathbb{E}_\pi \left[ \sum_i \gamma^i r_{t+i+1} \mid s_t = s \right] \\ Q^\pi(s, a) &= \mathbb{E}_\pi \left[ \sum_i \gamma^i r_{t+i+1} \mid s_t = s, a_t = a \right] \end{aligned} \tag{3}$$

- ▶ All algorithm computing optimal policies rely on various mix of a *Generalized Policy Iteration*<sup>1</sup> :
  1. Evaluate the current policy (DP,...)
  2. Improve the current policy (greedization)
  3. Repeat

---

<sup>1</sup>Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning : An Introduction*. MIT Press, 1998. Print.

## ■ Solving RL : Two baseline methods :

- ▶ Model-based ( $\mathcal{P}_{ss'}^a$  and  $\mathcal{R}_{ss'}^a$  are known) : dynamic programming (value iteration algorithm, ...)
- ▶ Model-free : **exploitation vs exploration** paradigm for computing the optimal policy's Q-values :

$$\{Q(s, a)\}_{s \in \mathcal{S}, a \in \mathcal{A}(s)} \quad (4)$$

- ▶ Bootstrap from initial value
- ▶ Update in direction of the sampled expected return

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha \mathbb{E} [R_t | s, a] \quad (5)$$

- ▶ Many different variations : SARSA, Q-learning, R-learning, eligibility traces, ...

## ■ Imitation learning :

- ▶ For long and complex tasks : common machine learning algorithm are usually very slow to converge
- ▶ Accelerate learning via prior knowledge of the environment or task : provide a **demonstration** of the task
- ▶ Framework of *learning from demonstration* (LfD)<sup>2</sup>

—→ Ex. : robotic arm grabbing a cup  
: maze solver

---

<sup>2</sup>Aude G. Billard, Sylvain Calinon, and Rüdiger Dillmann. “Learning from Humans”. *Springer Handbook of Robotics*. Ed. Bruno Siciliano and Oussama Khatib. Cham: Springer International Publishing, 2016. 1995–2014. Web.

■ **Transfer learning** : speeding a learning process thanks to another learning experience.

- ▶ Provide the learner with a mentor that is another learner
- ▶ In *homogeneous settings*<sup>3</sup>
- ▶ Study how convergence is affected

and eventually generalize to

- ▶ multiple teacher
- ▶ inhomogeneous settings

---

<sup>3</sup>Bob Price and Craig Boutilier. "Accelerating reinforcement learning through implicit imitation". *Journal of Artificial Intelligence Research* 19 (2003): 569–629. [Print.](#) 

# Plan

## 1 Motivations

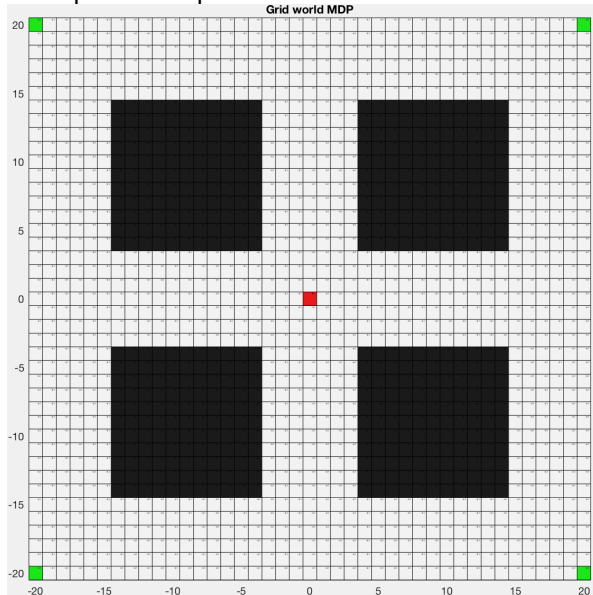
## 2 Background

## 3 Settings & Results

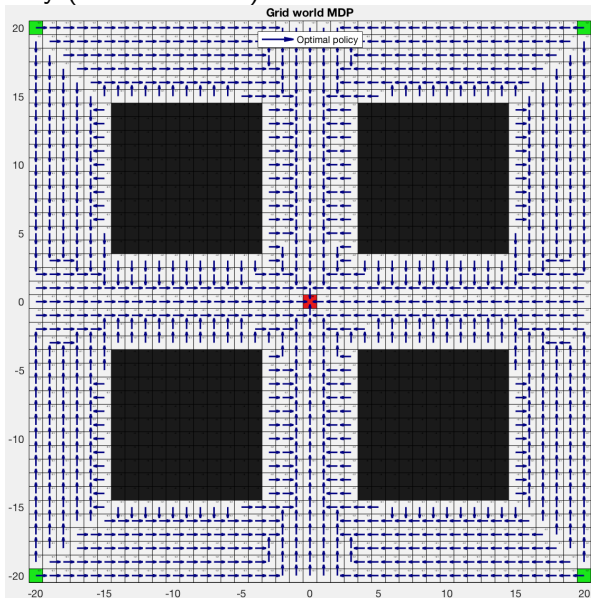
- Markov Decision Process
- Compliance-based learning
- Method comparison

## 4 Future work

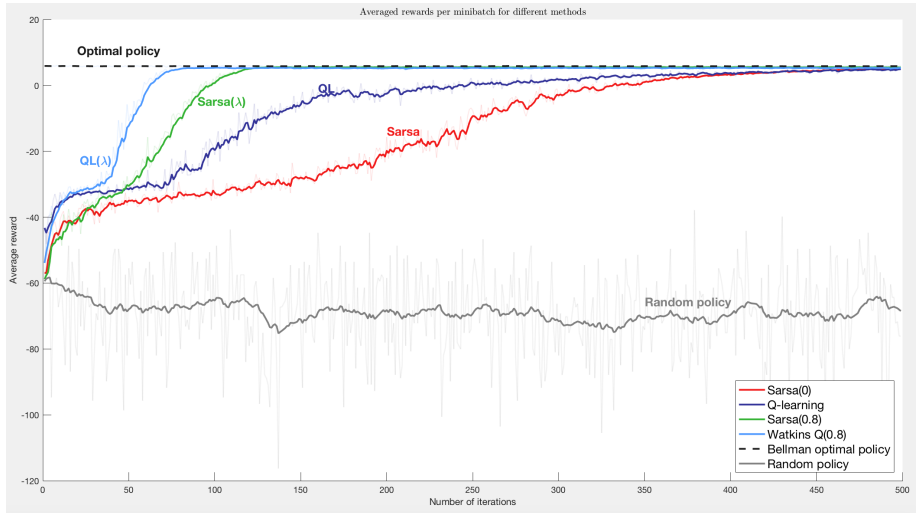
## ■ Generic and simple state space :



## ■ Optimal Policy (value iteration) :

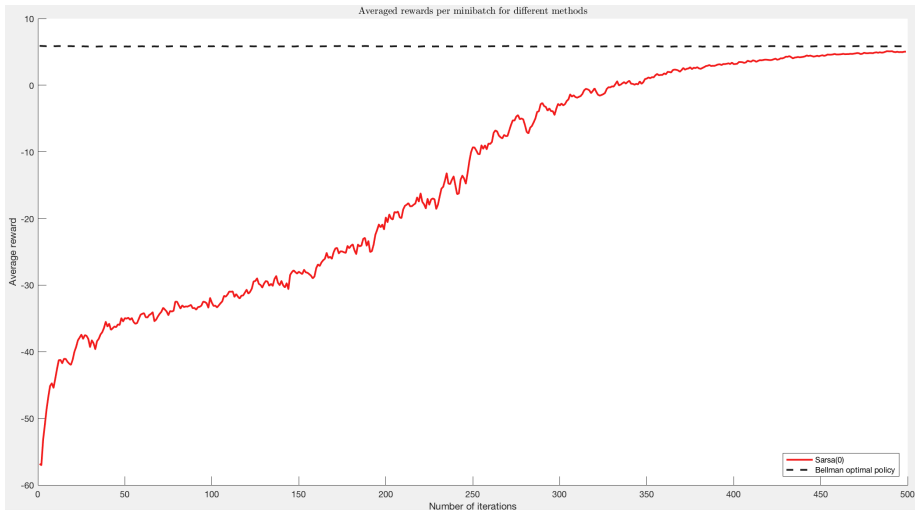


## ■ Learning the optimal policy

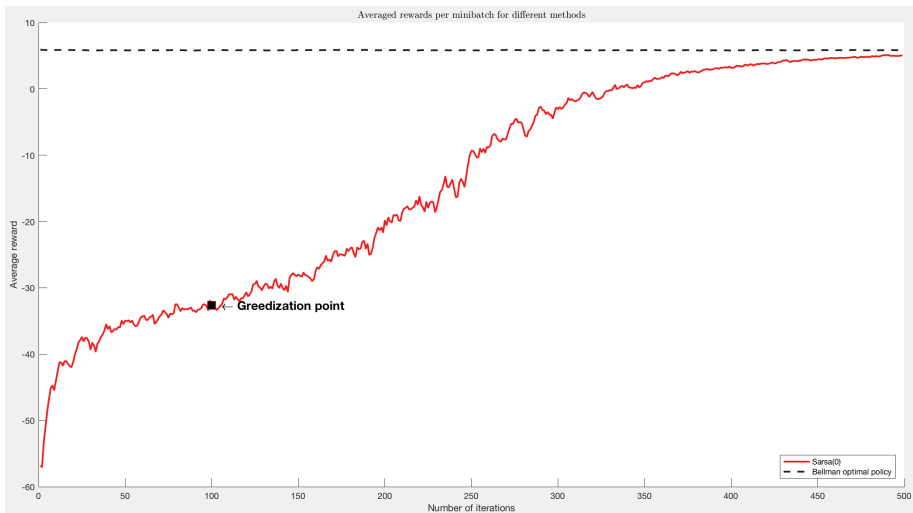




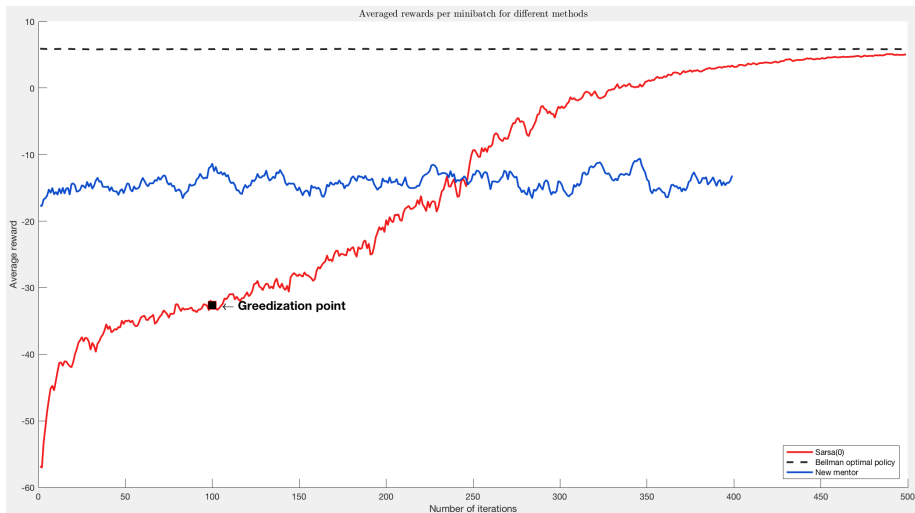
## ■ Generating a suboptimal mentor



## ■ Generating a suboptimal mentor



## ■ Generating a suboptimal mentor



## ■ Compliance learning

→ Intuitively :

- ▶ Follow the teacher
- ▶ Gain some knowledge about the environment and the task
- ▶ Take our own actions

→ The teacher should only influence our action selection:

- ▶ Global **compliance term** :  $p \in [0, 1]$
- ▶  $p$ -greedy action selection w.r.t the mentor's action  $a_m$  :  $\forall s \in \mathcal{S}$

$$\pi(s) = \begin{cases} a_m \text{ with probability } p \\ a \in \mathcal{A}(s) \text{ (Gibbs softmax)} \end{cases} \quad (6)$$

## ■ Vanishing compliance :

### ► Constantly decreasing compliance :

$$\left| \begin{array}{l} p_0 \in [0, 1] \\ p_{t+1} = \beta p_t, \quad \beta < 1 \end{array} \right. \quad (7)$$

### ► Along with SARSA update :

$$Q(s, a) \leftarrow Q(s, a) + \alpha(r + \gamma Q(s', a') - Q(s, a)) \quad (8)$$

► Start with  $p_0 \simeq 1$  (high confidence) and slowly decide to take your own decisions.

## ■ Constantly decreasing compliance :

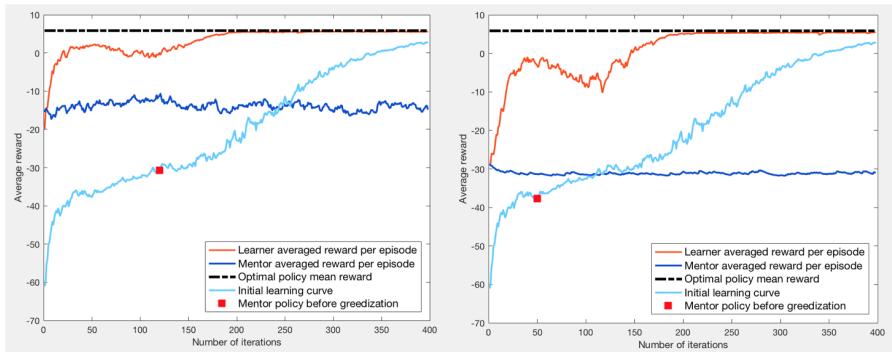


Figure: Average reward for two naive compliance learners

<span style="color: green; font-size: 2em;">+</span> Easy to implement Fast convergence (200 vs 500)	<span style="color: red; font-size: 2em;">-</span> Undershoot <b>Precise tuning</b>
--	---

## ■ Learning the compliance term

$\forall s \in \mathcal{S}$ , define  $p(s)$  - compliance term that impact the action selection :

$$\pi(s) = \begin{cases} a_m \text{ with probability } p(s) \\ a \in \mathcal{A}(s) \setminus a_m \text{ with probability } 1-p(s) \end{cases} \quad (9)$$

**Goal** : learn  $p(s)$ ,  $\forall s \in \mathcal{S} \rightarrow$  measure how right the teacher seems to be

► **Implicit  $\beta$ -compliance :**

- $\forall s \in \mathcal{S}$ , provide  $p(s)$  with a Beta prior ( $\alpha \gg \beta$ )

$$p(s) \sim B(\alpha(s), \beta(s)) \quad (10)$$

- Given a  $(s, a, r, s', a')$  5-tuple, compute the critic TD value :

$$\delta_t = r + \gamma Q(s', a') - Q(s, a_m) \quad \rightarrow \text{how good is the teacher ?} \quad (11)$$

- Compute posterior distribution over  $p(s)$  :

$$\begin{aligned} \alpha_t(s) &\leftarrow \alpha_t(s) + \mathbb{1}_{a=a_m} \delta_t \varepsilon_t \\ \beta_t(s) &\leftarrow \beta_t(s) + \mathbb{1}_{a \neq a_m} \delta_t \varepsilon_t \end{aligned} \quad (12)$$



## ► Implicit $\beta$ -compliance :

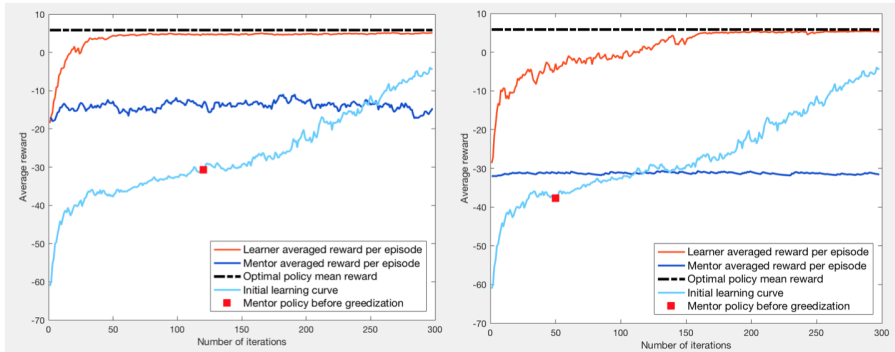
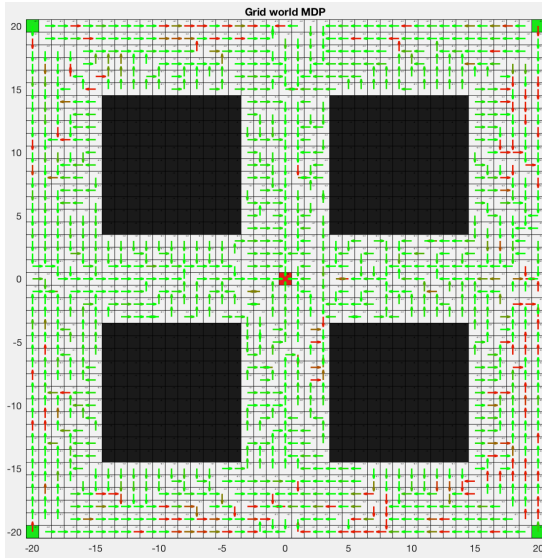


Figure: Average reward for two actor-critic compliance learners

- Faster convergence
- No undershoot + tuning is intuitive

## ► Implicit $\beta$ -compliance :



## ► Explicit compliance :

- Adds a hierarchical MDP :

$$\forall s \in \mathcal{S}, \mathcal{A}_c(s) = \{ 'listen', 'discard' \} \quad (13)$$

- Define exploration based on  $\{Q_c(s, l), Q_c(s, d)\}$  :

$$\forall s \in \mathcal{S}, \quad \pi_c(s) = \begin{cases} 'l' & \text{with probability } p(s) = \sigma \left( \frac{Q_c(s, l) - Q_c(s, d)}{\tau} \right) \\ 'd' & \text{with probability } 1 - p(s) \end{cases} \quad (14)$$

- Perform SARSA update
- Update :

$$\begin{cases} Q_c(s, l) \leftarrow \beta Q_c(s, l) + (1 - \beta) Q(s, a_m) \\ Q_c(s, d) \leftarrow \beta Q_c(s, d) + (1 - \beta) \max_{a \neq a_m} Q(s, a) \end{cases} \quad (15)$$

## ► Explicit compliance :

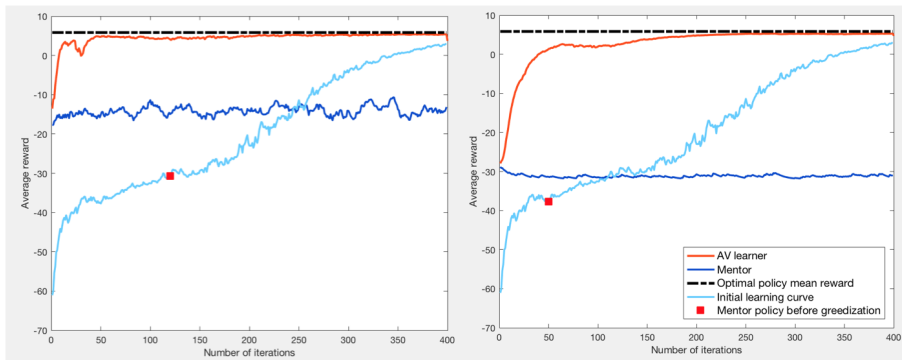
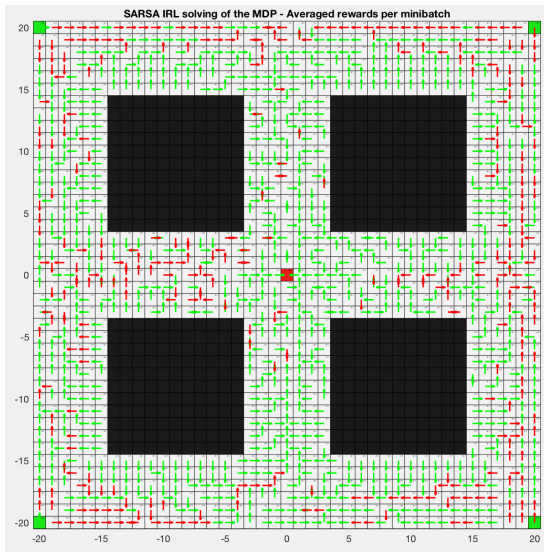
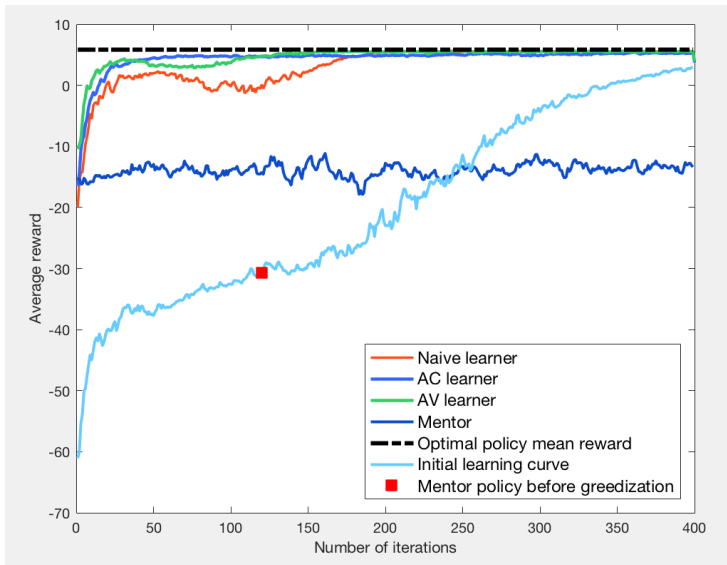


Figure: Average reward for two action-value compliance learners

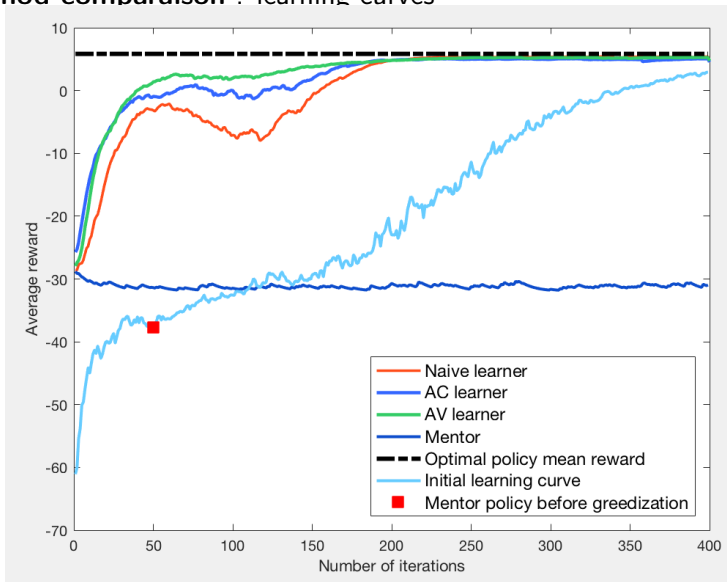
## ► Explicit compliance :



## ► Method comparison : learning curves



## ► Method comparison : learning curves



## ► Method comparison : f-fold metrics statistics

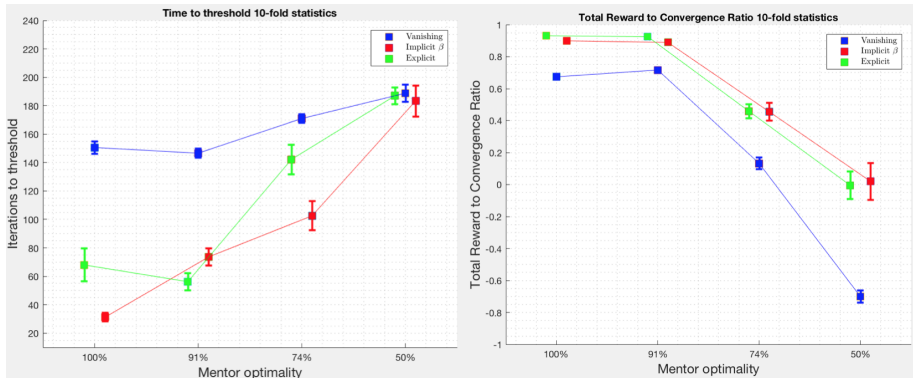


Figure: Metrics comparison for imitation learning methods



# Plan

- 1 Motivations
- 2 Background
- 3 Settings & Results
- 4 Future work

## ► Future Work

- Convergence and final result is too much impacted by the mentor : **off-policy** generalization
- Eligibility-trace formulation
- Generalize to several mentors
- Generalize to sparse recommendations