Learning from suboptimal teachers

The role of compliance in the exploration-exploitation tradeoff

Final Presentation - Semester Project

Louis Faury

Under the supervision of :
Mahdi Khoramshahi and Andrew Sutcliffe

June 19, 2017

Outline

Introduction

Introduction

- Mapping state to action : *policy*
- Crucial problem in many robotics applications..

Introduction

- Mapping state to action : *policy*
- Crucial problem in many robotics applications..

  ..but hard to design by hand !

Introduction

- Mapping state to action : *policy*
- Crucial problem in many robotics applications..

  ..but hard to design by hand !

Super Mario Bros

Introduction

- Mapping state to action : *policy*
- Crucial problem in many robotics applications..

..but hard to design by hand !

Super Mario Bros

Nao (SoftBank Robotics)

REINFORCEMENT LEARNING[1]

- Formulated for Markov Decision Process (MDP) :

$$(\mathcal{S}, \mathcal{A}, \mathcal{P}^a_{ss'}, \mathcal{R}^a_{ss'})$$

with :

$\mathcal{S}$	state space
$\mathcal{A}$	action space
$\mathcal{P}^a_{ss'} = \mathbb{P}(s_{t+1} = s' \mid s_t = s, a_t = a)$	Markovian dynamics
$\mathcal{R}^a_{ss'} = \mathbb{E}(r_t \mid s_t = s, a_t = a, s_{t+1} = s')$	Markovian reward

[1] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning : An Introduction*

REINFORCEMENT LEARNING[1]

- Formulated for Markov Decision Process (MDP) :

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

with :

$\mathcal{S}$	state space
$\mathcal{A}$	action space
$\mathcal{P}_{ss'}^a = \mathbb{P}(s_{t+1} = s' \mid s_t = s, a_t = a)$	Markovian dynamics
$\mathcal{R}_{ss'}^a = \mathbb{E}(r_t \mid s_t = s, a_t = a, s_{t+1} = s')$	Markovian reward

- Objective : find the policy

$$\pi : \quad \mathcal{S} \to \mathcal{A}(s)$$
$$s \to a$$

that maximizes the accumulated reward

(1) Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning : An Introduction*

- **REINFORCEMENT LEARNING**

REINFORCEMENT LEARNING

- Model-based solving : dynamic programing (*value iteration algorithm*)

evaluate and improve the state-value function (static)

$$V_\pi(s) = \mathbb{E}_\pi(R_t = \sum_i r_{t+i+1} \,|\, s_t = s)$$

REINFORCEMENT LEARNING

- Model-based solving : dynamic programing (*value iteration algorithm*)

evaluate and improve the state-value function (static)

$$V_\pi(s) = \mathbb{E}_\pi(R_t = \sum_i r_{t+i+1} \,|\, s_t = s)$$

- Model-free solving : temporal difference (among others)

  evaluate and improve the action-value function (try-out)

$$Q_\pi(s,a) = \mathbb{E}_\pi(R_t = \sum_i r_{t+i+1} \,|\, s_t = s, a_t = a)$$

REINFORCEMENT LEARNING

- Model-based solving : dynamic programing (*value iteration algorithm*)

evaluate and improve the state-value function (static)

$$V_\pi(s) = \mathbb{E}_\pi(R_t = \sum_i r_{t+i+1} \,|\, s_t = s)$$

- Model-free solving : temporal difference (among others)

  evaluate and improve the action-value function (try-out)

$$Q_\pi(s,a) = \mathbb{E}_\pi(R_t = \sum_i r_{t+i+1} \,|\, s_t = s, a_t = a)$$

Bootstrap, explore and backup (tabular RL)!

$$Q_\pi(s,a) \sim r_t + Q_\pi(s',a')$$

- **REINFORCEMENT LEARNING**

REINFORCEMENT LEARNING

$$Q_\pi(s,a) \sim r_t + Q_\pi(s',a')$$

Who is *a'* ? Can we do more than 1 step backup ?

REINFORCEMENT LEARNING

$$Q_\pi(s,a) \sim r_t + Q_\pi(s',a')$$

Who is *a'* ? Can we do more than 1 step backup ?

	ON POLICY	OFF POLICY
TD(0)	Sampled from the exploratory policy (SARSA)	Sampled from the current greedy policy (QLearning)
TD(λ)	SARSA with eligibility traces	QLearning with eligibility traces

LEARNING FROM DEMONSTRATION[1,2]

(1) Aude G. Billard, Sylvain Calinon, and Rüdiger Dillmann. *"Learning from Humans"*

(2) Argall, Brenna D., et al. *"A survey of robot learning from demonstration."*

LEARNING FROM DEMONSTRATION[1,2]

- Help learning a policy by providing *good* examples of a task

[1] Aude G. Billard, Sylvain Calinon, and Rüdiger Dillmann. *"Learning from Humans"*
[2] Argall, Brenna D., et al. *"A survey of robot learning from demonstration."*

LEARNING FROM DEMONSTRATION[1,2]

- Help learning a policy by providing *good* examples of a task

- Infer the policy from demonstrations (statistical learning) ..
    - not robust to changes in the environment !

(1) Aude G. Billard, Sylvain Calinon, and Rüdiger Dillmann. *"Learning from Humans"*
(2) Argall, Brenna D., et al. *"A survey of robot learning from demonstration."*

LEARNING FROM DEMONSTRATION[1,2]

- Help learning a policy by providing *good* examples of a task

- Infer the policy from demonstrations (statistical learning) ..

  - not robust to changes in the environment !

- Use reinforcement learning !

  - demonstrations counterbalance the greediness of RL
  - speed up the learning

(1) Aude G. Billard, Sylvain Calinon, and Rüdiger Dillmann. *"Learning from Humans"*

(2) Argall, Brenna D., et al. *"A survey of robot learning from demonstration."*

LEARNING FROM DEMONSTRATION

- How to use demonstrations in a reinforcement learning context ?

(1) S. Ross, G. Gordon and A. Bagnell. *"A reduction of imitation learning and structured prediction to no-regret online learning"*

(2) B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, *"Maximum entropy inverse reinforcement learning"*

(3) B. Piot, M. Geist, and O. Pietquin, *"Bridging the gap between imitation learning and inverse reinforcement learning"*

LEARNING FROM DEMONSTRATION

- How to use demonstrations in a reinforcement learning context ?
  - Bootstrap RL
  - Involve a teacher's policy in a policy mixture[(1)]
  - .. or equivalently let the teacher take control along the learning

[(1)] S. Ross, G. Gordon and A. Bagnell. *"A reduction of imitation learning and structured prediction to no-regret online learning"*

[(2)] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, *"Maximum entropy inverse reinforcement learning"*

[(3)] B. Piot, M. Geist, and O. Pietquin, *"Bridging the gap between imitation learning and inverse reinforcement learning"*

LEARNING FROM DEMONSTRATION

- How to use demonstrations in a reinforcement learning context ?
  - Bootstrap RL
  - Involve a teacher's policy in a policy mixture[(1)]
  - .. or equivalently let the teacher take control along the learning
  - Inverse Reinforcement Learning : learn the reward function[(2,3)]

(1) S. Ross, G. Gordon and A. Bagnell. *"A reduction of imitation learning and structured prediction to no-regret online learning"*

(2) B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, *"Maximum entropy inverse reinforcement learning"*

(3) B. Piot, M. Geist, and O. Pietquin, *"Bridging the gap between imitation learning and inverse reinforcement learning"*

LEARNING FROM DEMONSTRATION

- How to use demonstrations in a reinforcement learning context ?
    - Bootstrap RL
    - Involve a teacher's policy in a policy mixture[1]
    - .. or equivalently let the teacher take control along the learning
    - Inverse Reinforcement Learning : learn the reward function[2,3]
    - Reward similarity to the teacher

[1] S. Ross, G. Gordon and A. Bagnell. *"A reduction of imitation learning and structured prediction to no-regret online learning"*

[2] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, *"Maximum entropy inverse reinforcement learning"*

[3] B. Piot, M. Geist, and O. Pietquin, *"Bridging the gap between imitation learning and inverse reinforcement learning"*

LEARNING FROM DEMONSTRATION

- How to use demonstrations in a reinforcement learning context ?
  - Bootstrap RL
  - Involve a teacher's policy in a policy mixture[(1)]
  - .. or equivalently let the teacher take control along the learning
  - Inverse Reinforcement Learning : learn the reward function[(2,3)]
  - Reward similarity to the teacher
  - ..

[(1)] S. Ross, G. Gordon and A. Bagnell. *"A reduction of imitation learning and structured prediction to no-regret online learning"*

[(2)] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, *"Maximum entropy inverse reinforcement learning"*

[(3)] B. Piot, M. Geist, and O. Pietquin, *"Bridging the gap between imitation learning and inverse reinforcement learning"*

LEARNING FROM DEMONSTRATION

LEARNING FROM DEMONSTRATION

- Suppose a *near-optimal* teachers

.. or at least *good* demonstration in the dataset

LEARNING FROM DEMONSTRATION

- Suppose a *near-optimal* teachers

  .. or at least *good* demonstration in the dataset

- What does it mean to be suboptimal ? *Largely* suboptimal ?

LEARNING FROM DEMONSTRATION

- Suppose a *near-optimal* teachers

  .. or at least *good* demonstration in the dataset

- What does it mean to be suboptimal ? *Largely* suboptimal ?

  - Poor transfer to the robot abilities

LEARNING FROM DEMONSTRATION

- Suppose a *near-optimal* teachers

  .. or at least *good* demonstration in the dataset

- What does it mean to be suboptimal ? *Largely* suboptimal ?

  - Poor transfer to the robot abilities
  - Does not maximize a numerical criterion evaluating the fitness of a behavior

LEARNING FROM DEMONSTRATION

- Suppose a *near-optimal* teachers

  .. or at least *good* demonstration in the dataset

- What does it mean to be suboptimal ? *Largely* suboptimal ?

  - Poor transfer to the robot abilities
  - Does not maximize a numerical criterion evaluating the fitness of a behavior
  - Obvious downside - possible danger !

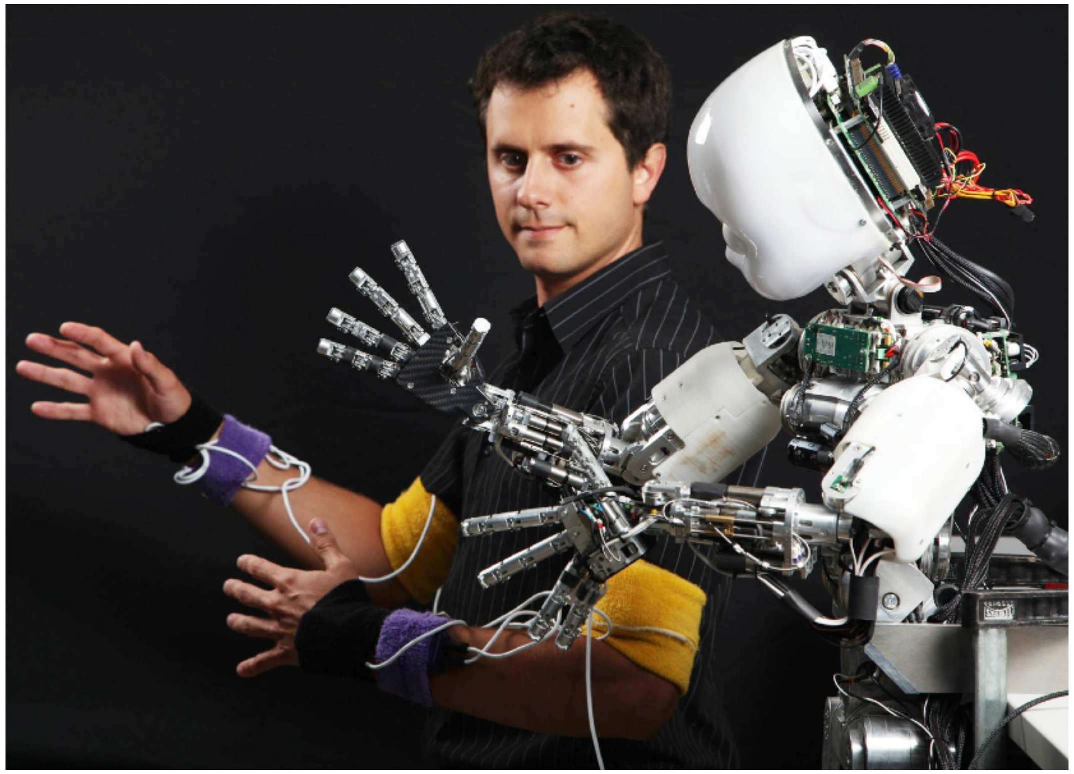
LEARNING FROM DEMONSTRATION

- Suppose a *near-optimal* teachers

  .. or at least *good* demonstration in the dataset

- What does it mean to be suboptimal ? *Largely* suboptimal ?

  - Poor transfer to the robot abilities
  - Does not maximize a numerical criterion evaluating the fitness of a behavior
  - Obvious downside - possible danger !
  - Ex : **Grasping objects**



Credits : Sylvain Calinon

Motivations

- **Find a way to learn from suboptimal teachers**

(1) J. Choi and K.-E. Kim, *“Hierarchical bayesian inverse reinforcement learning”*

(2) V. Chu and A. L. Thomaz, *“Analyzing differences between teachers when learning object affordances via guided exploration”*

Motivations

- **Find a way to learn from suboptimal teachers**
  - Extract useful information from any demonstration
  - Extend to non-expert teachers !
  - Enlarge human-robot interaction possibilities

[1] J. Choi and K.-E. Kim, *"Hierarchical bayesian inverse reinforcement learning"*

[2] V. Chu and A. L. Thomaz, *"Analyzing differences between teachers when learning object affordances via guided exploration"*

Motivations

- **Find a way to learn from suboptimal teachers**
  - Extract useful information from any demonstration
  - Extend to non-expert teachers !
  - Enlarge human-robot interaction possibilities

- Some approaches in the literature :

(1) J. Choi and K.-E. Kim, *"Hierarchical bayesian inverse reinforcement learning"*

(2) V. Chu and A. L. Thomaz, *"Analyzing differences between teachers when learning object affordances via guided exploration"*

Motivations

- **Find a way to learn from suboptimal teachers**
  - Extract useful information from any demonstration
  - Extend to non-expert teachers !
  - Enlarge human-robot interaction possibilities

- Some approaches in the literature :
  - Bayesian Inverse Reinforcement Learning[(1)]

[(1)] J. Choi and K.-E. Kim, *"Hierarchical bayesian inverse reinforcement learning"*

[(2)] V. Chu and A. L. Thomaz, *"Analyzing differences between teachers when learning object affordances via guided exploration"*

Motivations

- **Find a way to learn from suboptimal teachers**
  - Extract useful information from any demonstration
  - Extend to non-expert teachers !
  - Enlarge human-robot interaction possibilities

- Some approaches in the literature :
  - Bayesian Inverse Reinforcement Learning[1]
  - Guided exploration[2]

[1] J. Choi and K.-E. Kim, *"Hierarchical bayesian inverse reinforcement learning"*

[2] V. Chu and A. L. Thomaz, *"Analyzing differences between teachers when learning object affordances via guided exploration"*

Motivations

- **Find a way to learn from suboptimal teachers**
  - Extract useful information from any demonstration
  - Extend to non-expert teachers !
  - Enlarge human-robot interaction possibilities

- Some approaches in the literature :
  - Bayesian Inverse Reinforcement Learning[(1)]
  - Guided exploration[(2)]
  - ..

(1) J. Choi and K.-E. Kim, *"Hierarchical bayesian inverse reinforcement learning"*

(2) V. Chu and A. L. Thomaz, *"Analyzing differences between teachers when learning object affordances via guided exploration"*

Motivations

- A (universal) learning from demonstration approach for humans :

Motivations

- A (universal) learning from demonstration approach for humans :
  - Try to reproduce the teacher's moves as closely as possible

Motivations

- A (universal) learning from demonstration approach for humans :
    - Try to reproduce the teacher's moves as closely as possible
    - Gather some intuition and knowledge about the task

Motivations

- A (universal) learning from demonstration approach for humans :
  - Try to reproduce the teacher's moves as closely as possible
  - Gather some intuition and knowledge about the task
  - Evaluate the teacher's actions under that acquired knowledge

Motivations

- A (universal) learning from demonstration approach for humans :
  - Try to reproduce the teacher's moves as closely as possible
  - Gather some intuition and knowledge about the task
  - Evaluate the teacher's actions under that acquired knowledge
  - Follow own decisions

Motivations

- A (universal) learning from demonstration approach for humans :
  - Try to reproduce the teacher's moves as closely as possible
  - Gather some intuition and knowledge about the task
  - Evaluate the teacher's actions under that acquired knowledge
  - Follow own decisions

➡ Shifting compliance (with respect to the teacher)

Motivations

- A (universal) learning from demonstration approach for humans :
  - Try to reproduce the teacher's moves as closely as possible
  - Gather some intuition and knowledge about the task
  - Evaluate the teacher's actions under that acquired knowledge
  - Follow own decisions

➡ Shifting compliance (with respect to the teacher)

GOAL : - define a compliance-based approach for learning from suboptimal teachers
- experimentally evaluate its performances

INTUITION

- Bias the action-selection (exploratory policy) towards the teacher's demonstrations

INTUITION

- Bias the action-selection (exploratory policy) towards the teacher's demonstrations

- Define :
  - $a_m(s)$ the mentor's action at state s
  - $p(s)$ the **compliance** at state s

$$\forall s \in \mathcal{S}, \quad \pi_p(s) = \begin{cases} a_m(s) \text{ with probability } p(s) \\ a \in \mathcal{A}(s) \text{ with probability } (1 - p(s)) \end{cases}$$

INTUITION

- Bias the action-selection (exploratory policy) towards the teacher's demonstrations

- Define : - $a_m(s)$ the mentor's action at state s
  - $p(s)$ the **compliance** at state s

$$\forall s \in \mathcal{S}, \quad \pi_p(s) = \begin{cases} a_m(s) \text{ with probability } p(s) \\ a \in \mathcal{A}(s) \text{ with probability } (1 - p(s)) \end{cases}$$

- **Make the compliance vary throughout the learning**

- **VANISHING-COMPLIANCE LEARNER**

VANISHING-COMPLIANCE LEARNER

- The compliance is initialized near 1 and slowly vanishes :

$$p_{k+1} = \gamma p_k, \quad \gamma < 1$$

VANISHING-COMPLIANCE LEARNER

- The compliance is initialized near 1 and slowly vanishes :

$$p_{k+1} = \gamma p_k, \quad \gamma < 1$$

- Extremely simple and easy to implement !

VANISHING-COMPLIANCE LEARNER

- The compliance is initialized near 1 and slowly vanishes :

$$p_{k+1} = \gamma p_k, \quad \gamma < 1$$

- Extremely simple and easy to implement !
- .. but rather naive

VANISHING-COMPLIANCE LEARNER

- The compliance is initialized near 1 and slowly vanishes :

$$p_{k+1} = \gamma p_k, \quad \gamma < 1$$

- Extremely simple and easy to implement !
- .. but rather naive
- .. and no teacher evaluation !

VANISHING-COMPLIANCE LEARNER

- The compliance is initialized near 1 and slowly vanishes :

$$p_{k+1} = \gamma p_k, \quad \gamma < 1$$

- Extremely simple and easy to implement !
- .. but rather naive
- .. and no teacher evaluation !
- also, tuning might be difficult

- Evaluate a teacher's recommandations and shift the bias accordingly

- Evaluate a teacher's recommandations and shift the bias accordingly

β-IMPLICIT COMPLIANCE LEARNER

- Evaluate a teacher's recommandations and shift the bias accordingly

$\boldsymbol{\beta}$-IMPLICIT COMPLIANCE LEARNER

- Provide a Beta prior distribution for the compliance (initial bias)

$$\forall s \in \mathcal{S}, \quad p(s) \sim \boldsymbol{\beta}(\alpha(s), \beta(s))$$

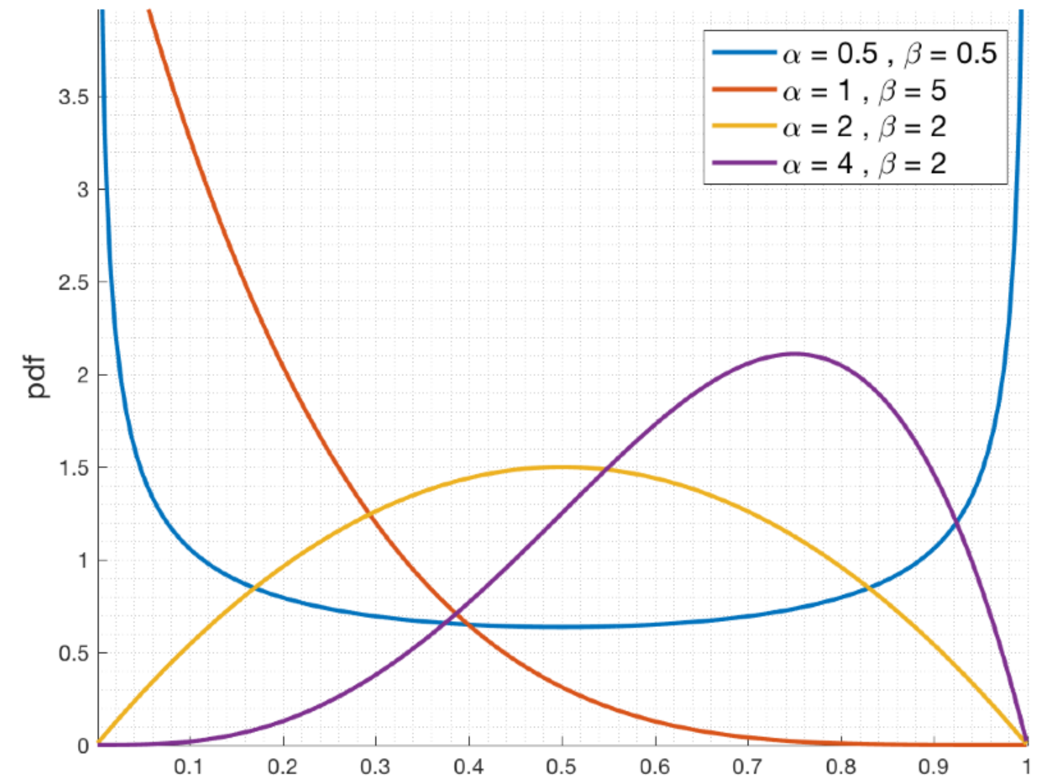


Figure : Beta p.d.f

- Evaluate a teacher's recommandations and shift the bias accordingly

$\boldsymbol{\beta}$-IMPLICIT COMPLIANCE LEARNER

- Provide a Beta prior distribution for the compliance (initial bias)

$$\forall s \in \mathcal{S}, \quad p(s) \sim \boldsymbol{\beta}(\alpha(s), \beta(s))$$

- Sample in the current policy (SARSA)

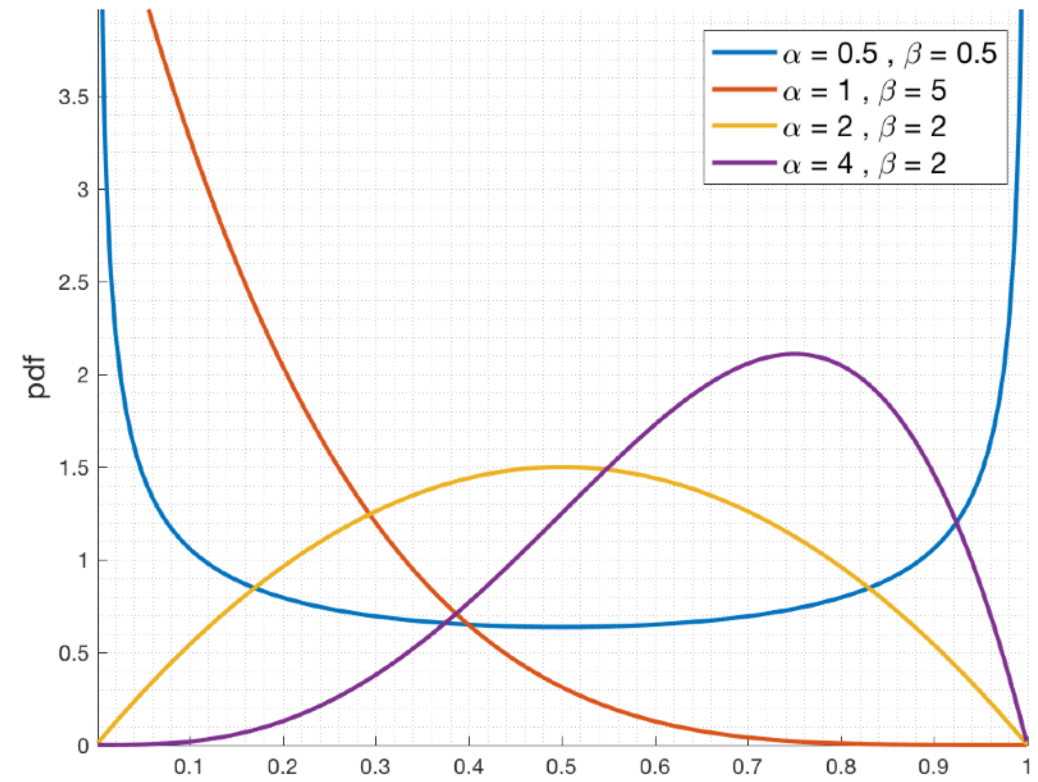


Figure : Beta p.d.f

- Evaluate a teacher's recommandations and shift the bias accordingly

$\boldsymbol{\beta}$-IMPLICIT COMPLIANCE LEARNER

- Provide a Beta prior distribution for the compliance (initial bias)

$$\forall s \in \mathcal{S}, \quad p(s) \sim \boldsymbol{\beta}(\alpha(s), \beta(s))$$

- Sample in the current policy (SARSA)

- Compute a TD(0) critic

$$\delta_t = r + \gamma Q(s', a') - Q(s, a_m)$$

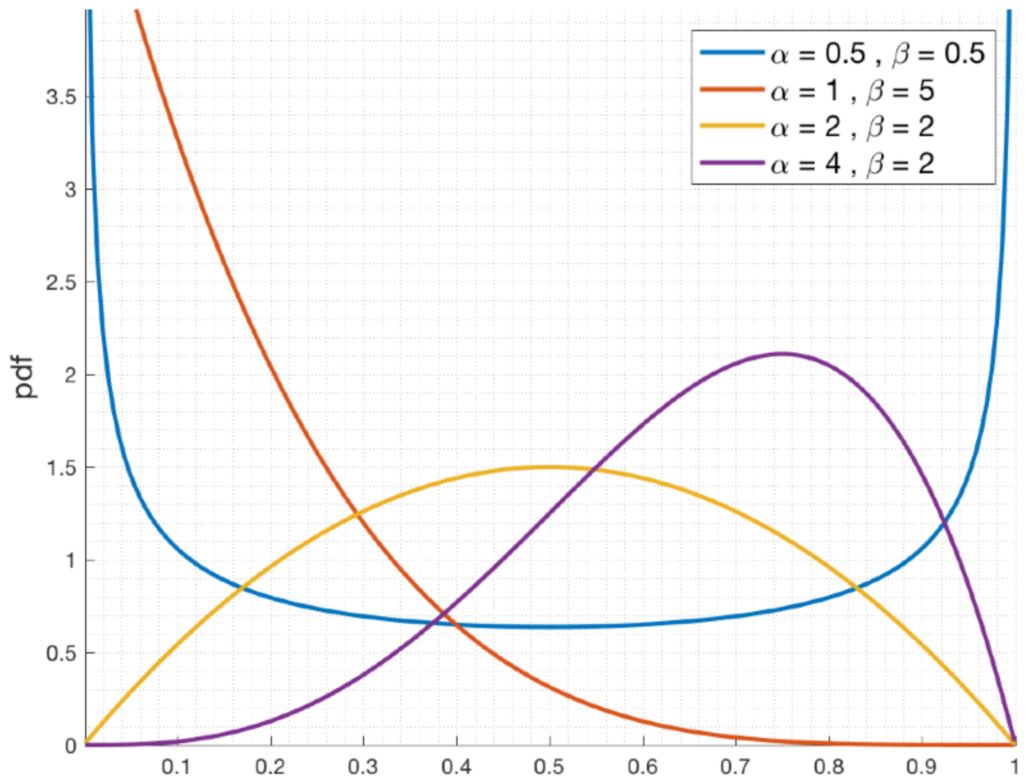


Figure : Beta p.d.f

- Evaluate a teacher's recommandations and shift the bias accordingly

$\boldsymbol{\beta}$-IMPLICIT COMPLIANCE LEARNER

- Provide a Beta prior distribution for the compliance (initial bias)

$$\forall s \in \mathcal{S}, \quad p(s) \sim \boldsymbol{\beta}(\alpha(s), \beta(s))$$

- Sample in the current policy (SARSA)

- Compute a TD(0) critic

$$\delta_t = r + \gamma Q(s', a') - Q(s, a_m)$$

- Update the p.d.f parameters accordingly

$$\alpha_t(s) \leftarrow \alpha_t(s) + \mathbb{1}_{a=a_m} \delta_t \varepsilon_t$$
$$\beta_t(s) \leftarrow \beta_t(s) + \mathbb{1}_{a \neq a_m} \delta_t \varepsilon_t$$

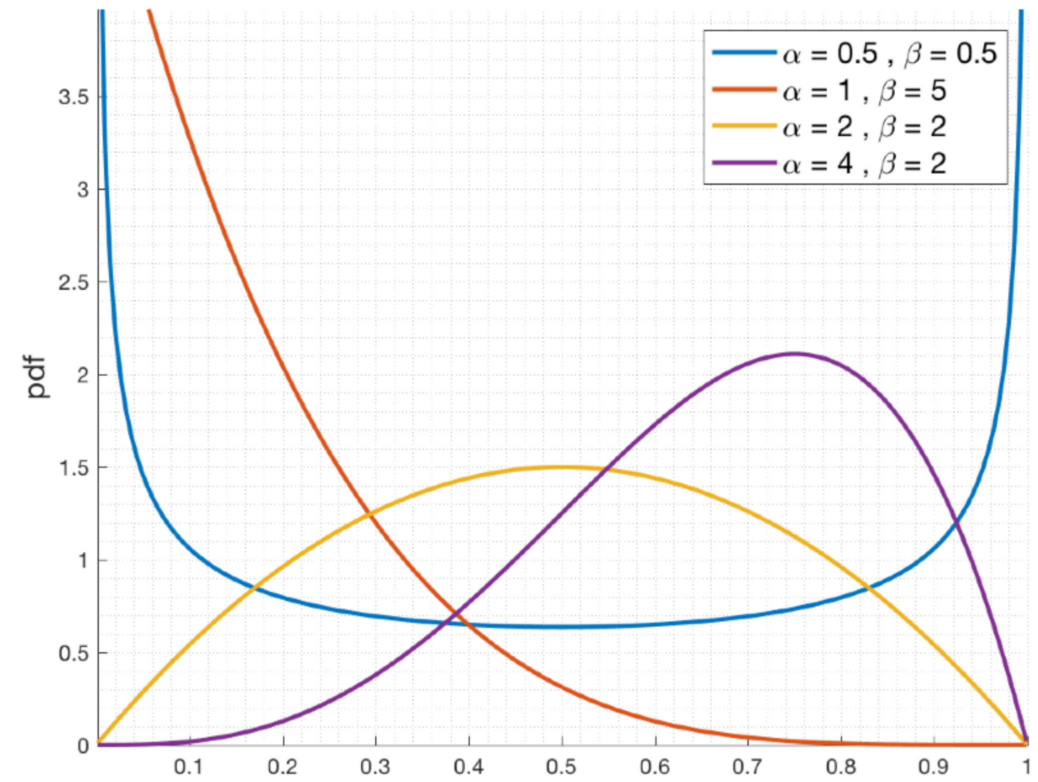


Figure : Beta p.d.f

- Evaluate a teacher's recommandations and shift the bias accordingly

$\boldsymbol{\beta}$-IMPLICIT COMPLIANCE LEARNER

- Provide a Beta prior distribution for the compliance (initial bias)

$$\forall s \in \mathcal{S}, \quad p(s) \sim \boldsymbol{\beta}(\alpha(s), \beta(s))$$

- Sample in the current policy (SARSA)

- Compute a TD(0) critic

$$\delta_t = r + \gamma Q(s', a') - Q(s, a_m)$$

- Update the p.d.f parameters accordingly

$$\alpha_t(s) \leftarrow \alpha_t(s) + \mathbb{1}_{a=a_m}\delta_t\varepsilon_t$$
$$\beta_t(s) \leftarrow \beta_t(s) + \mathbb{1}_{a\neq a_m}\delta_t\varepsilon_t$$

- Update Q-values

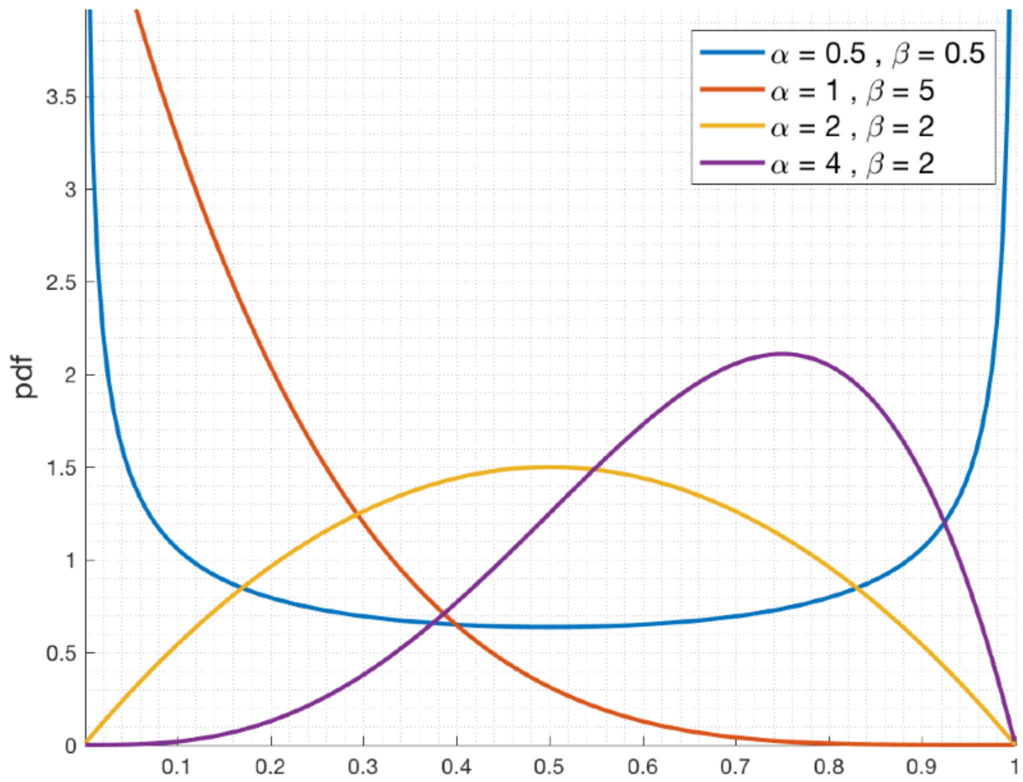


Figure : Beta p.d.f

- **EXPLICIT COMPLIANCE LEARNER**

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

- Procedure :

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

- Procedure :

Initialize (introduce bias):

$$\{Q_c(s,l), Q_c(s,d)\}$$

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

- Procedure :

Initialize (introduce bias):

$$\{Q_c(s,l), Q_c(s,d)\} \longleftarrow \text{listen and discard Q-values}$$

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

- Procedure :

Initialize (introduce bias):

$$\{Q_c(s,l), Q_c(s,d)\} \longleftarrow \text{listen and discard Q-values}$$

Sample from it (SARSA)

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

- Procedure :

Initialize (introduce bias):

$$\{Q_c(s,l), Q_c(s,d)\}$$

listen and discard Q-values

Sample from it (SARSA)

Update initial MDP

EXPLICIT COMPLIANCE LEARNER

- Learn the action-values of the MDP

$$\mathcal{A}_c(s) = \{'listen', 'discard'\}$$

- Gibbs sampling in this MDP drives the exploration of the initial MDP !

- Procedure :

Initialize (introduce bias):

$$\{Q_c(s,l), Q_c(s,d)\}$$

listen and discard Q-values

Sample from it (SARSA)

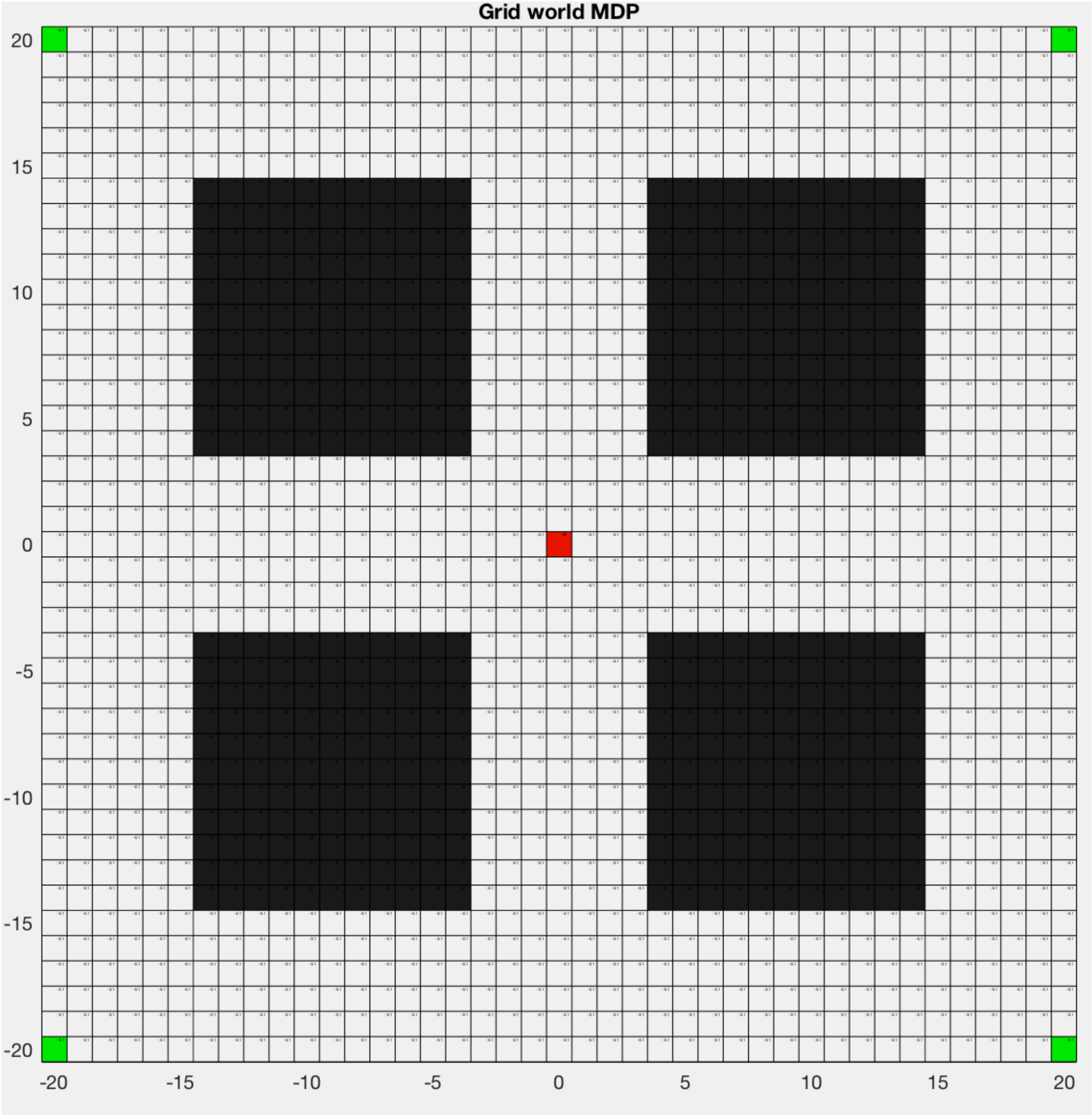
Update initial MDP

Update new MDP

$$\begin{cases} Q_c(s,l) \leftarrow \beta Q_c(s,l) + (1-\beta) Q(s,a_m) \\ Q_c(s,d) = \beta Q_c(s,d) + (1-\beta) \max\limits_{a \neq a_m} Q(s,a) \end{cases}$$

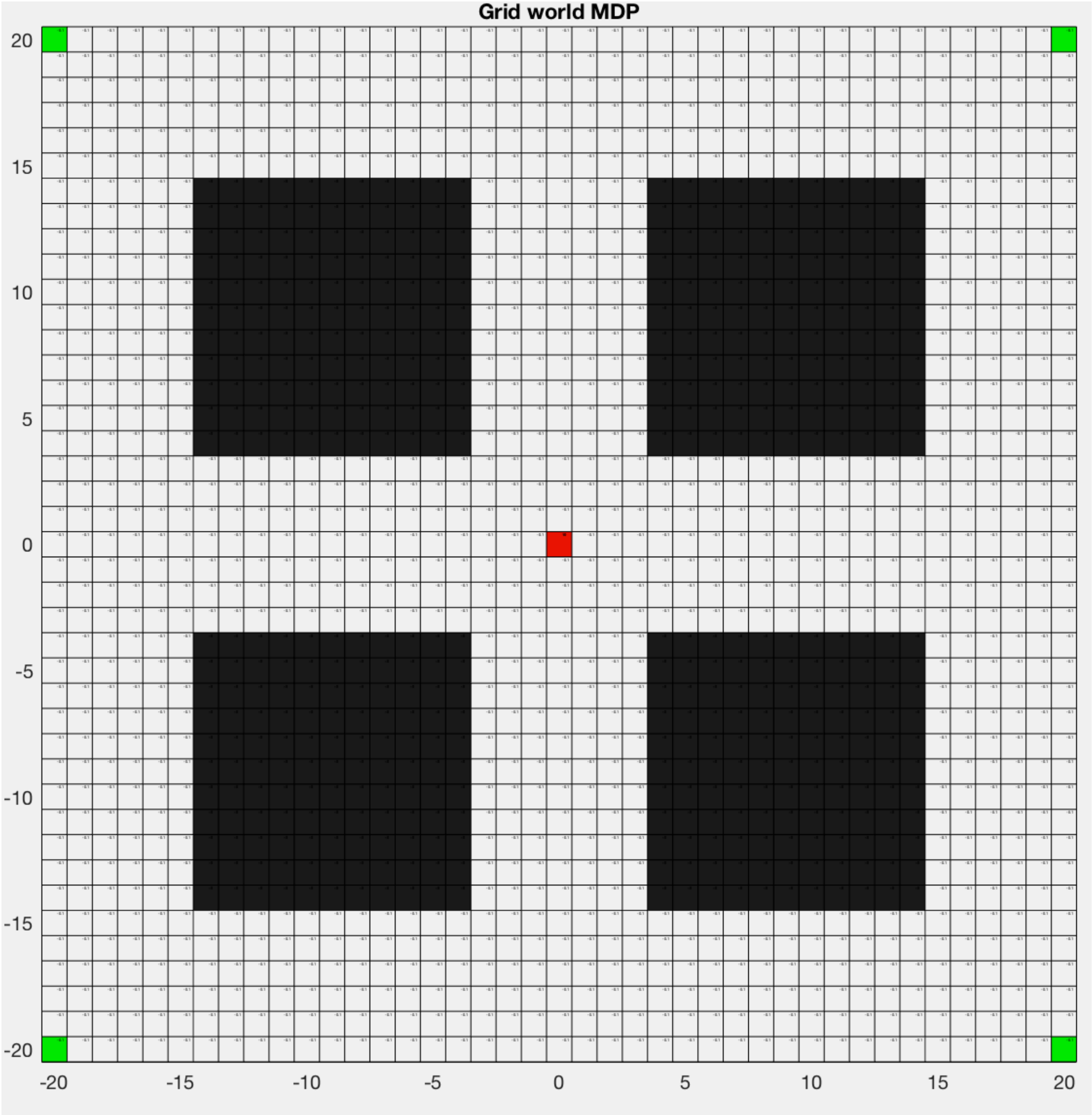
Results

- MDP



Results

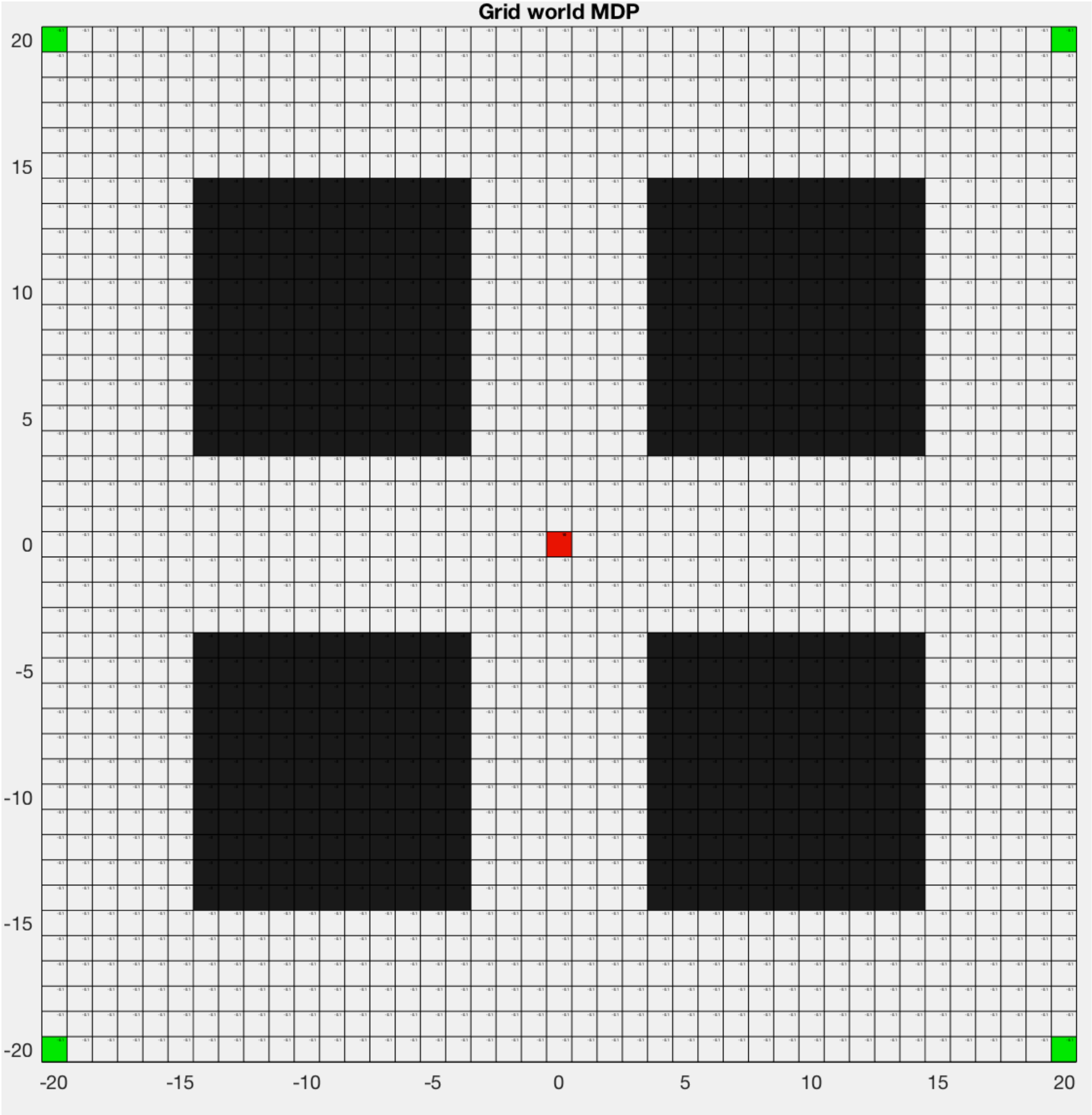
MDP



- Finite MDP

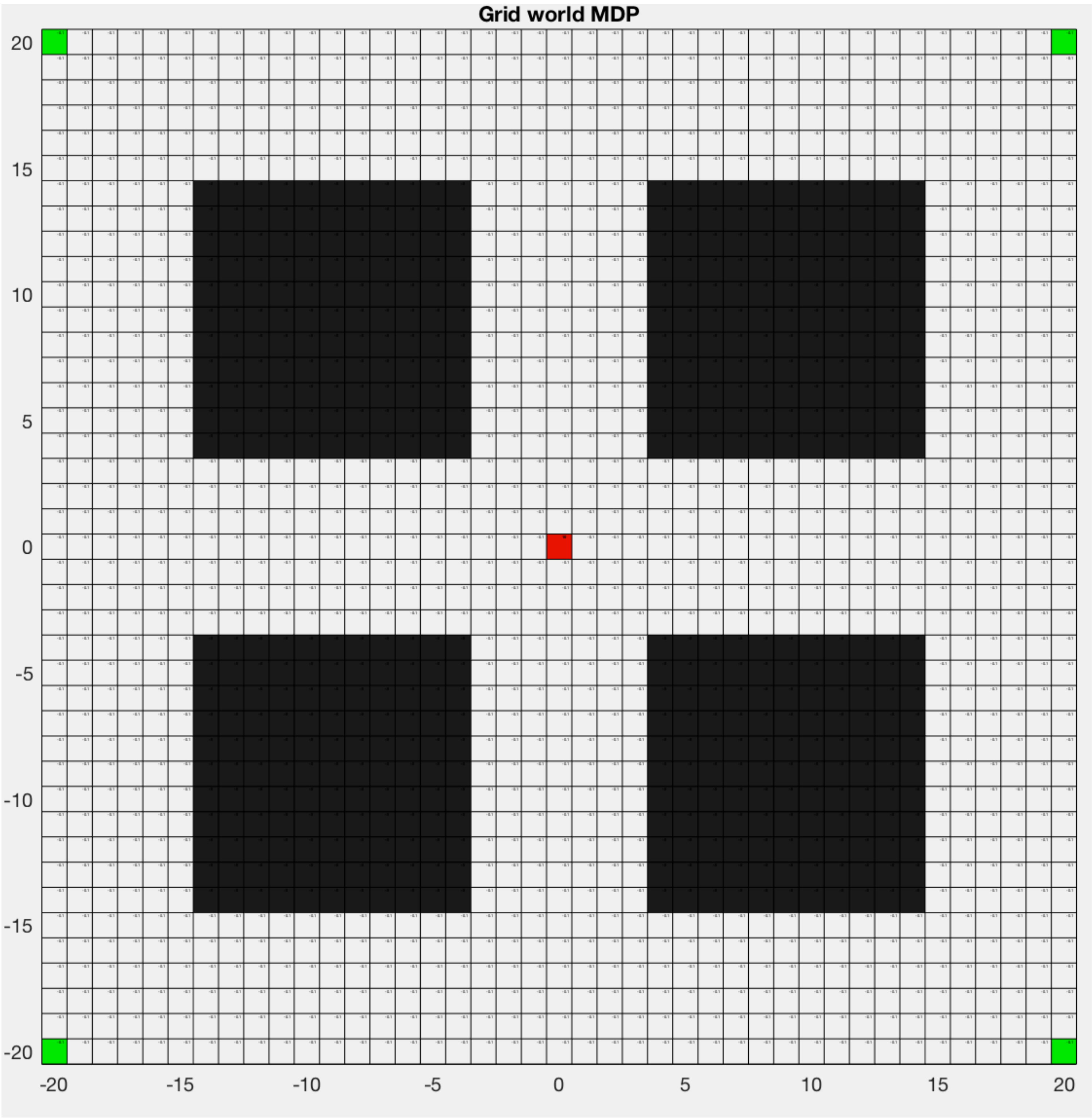
Results

MDP



- Finite MDP
- From green cell to red cell as quickly as possible

MDP

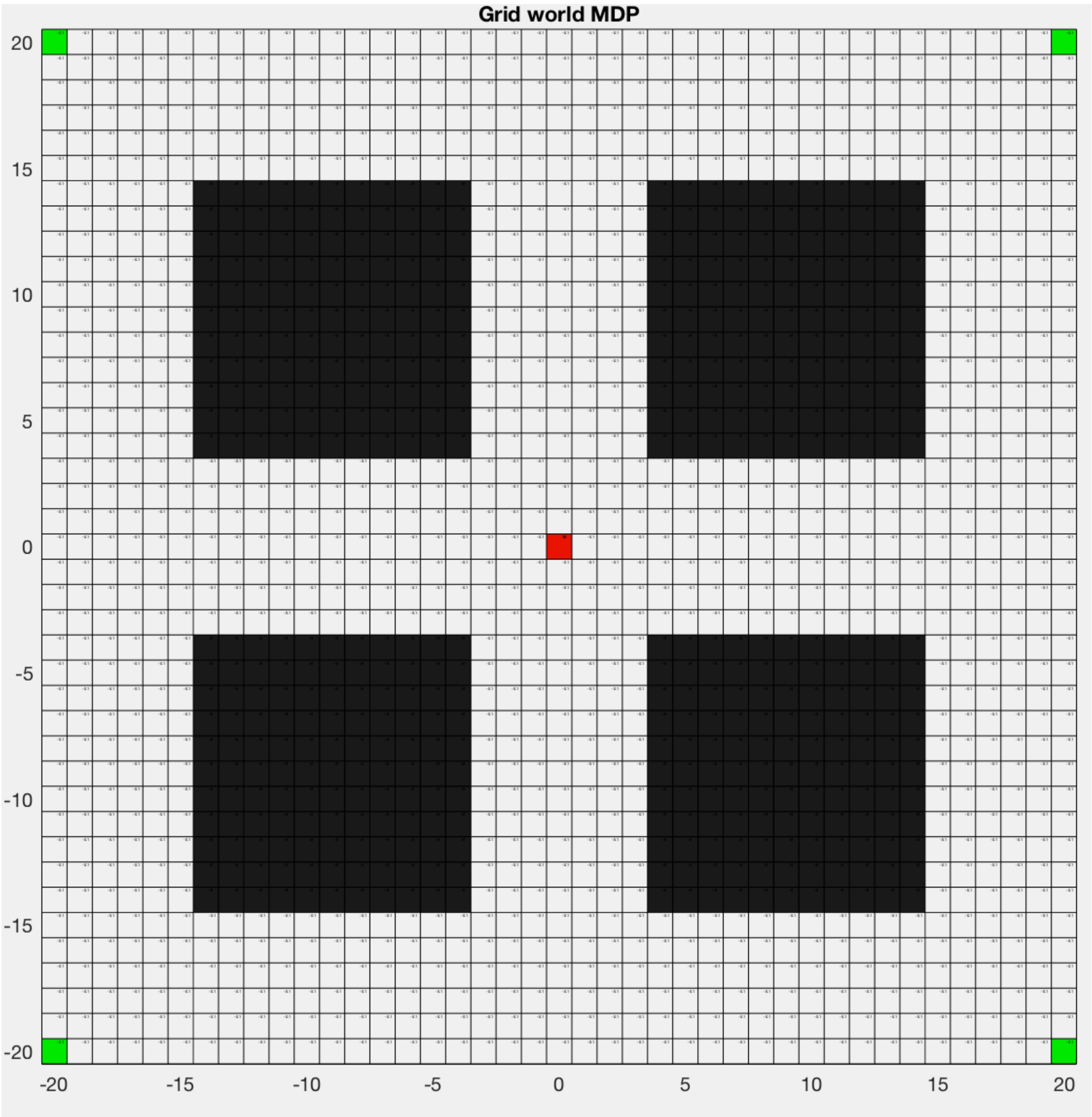


- Finite MDP
- From green cell to red cell as quickly as possible
- Stochastic :

$$\mathcal{P}^{a}_{s,s'} = \begin{cases} 0.9 & \text{if } s' = a(s) \\ 0.1 & \text{otherwise} \end{cases}$$

Results

MDP



- Finite MDP

- From green cell to red cell as quickly as possible

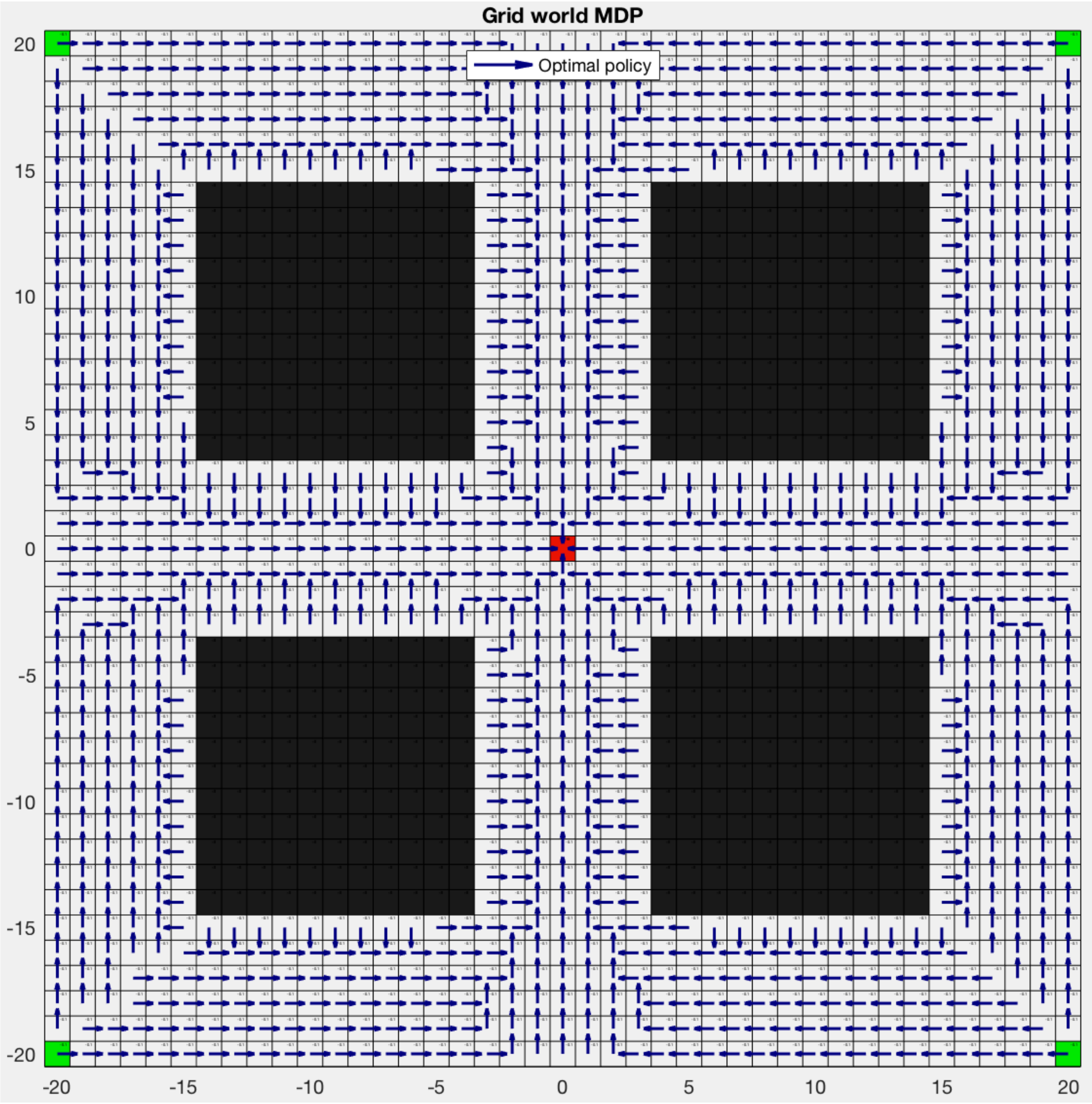
- Stochastic :

$$\mathcal{P}_{s,s'}^{a} = \begin{cases} 0.9 & \text{if } s' = a(s) \\ 0.1 & \text{otherwise} \end{cases}$$

- *Value iteration* for computing optimal policy (ground truth)

Results

MDP

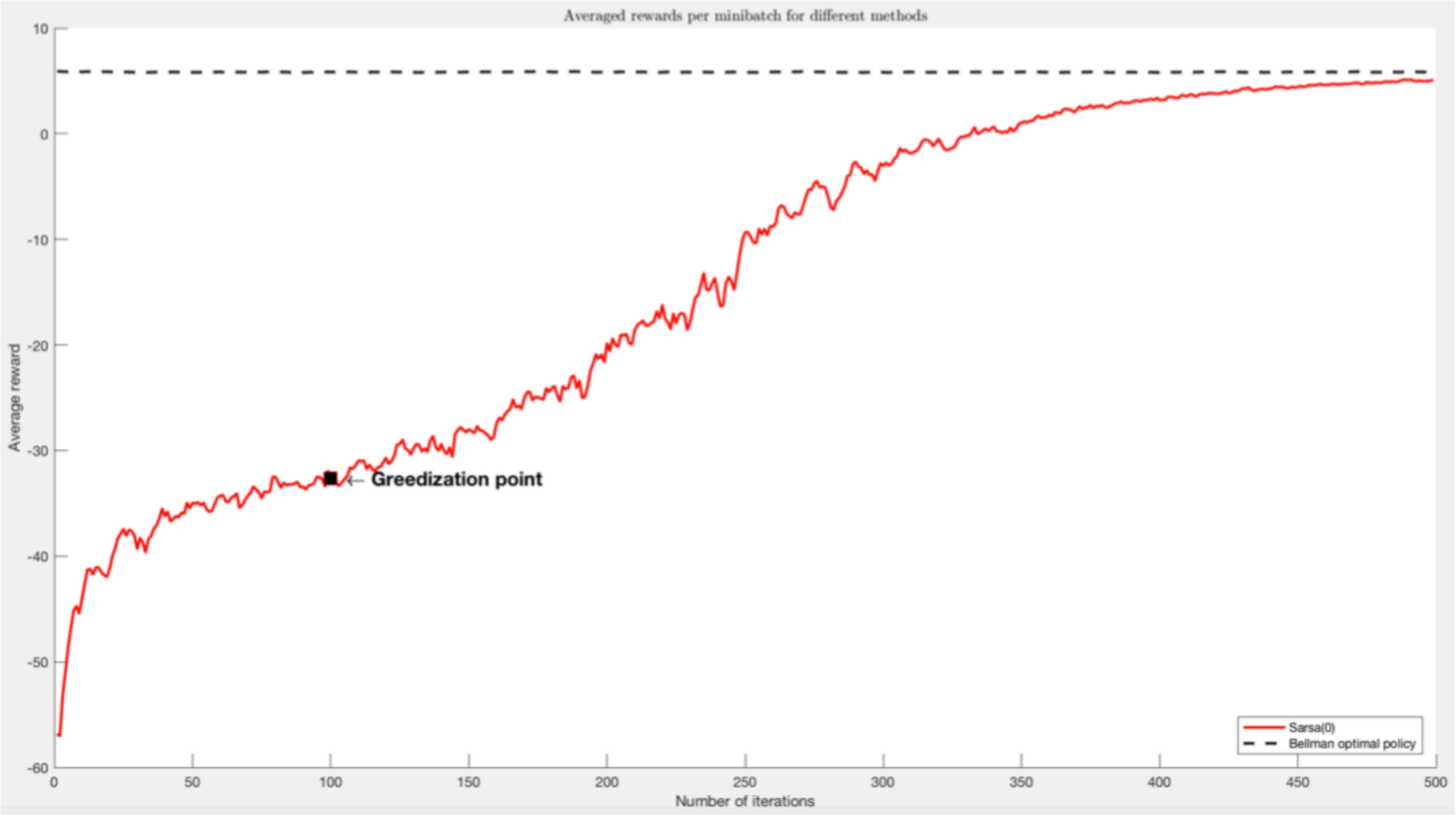


- Finite MDP
- From green cell to red cell as quickly as possible
- Stochastic :

$$\mathcal{P}^a_{s,s'} = \begin{cases} 0.9 & \text{if } s' = a(s) \\ 0.1 & \text{otherwise} \end{cases}$$

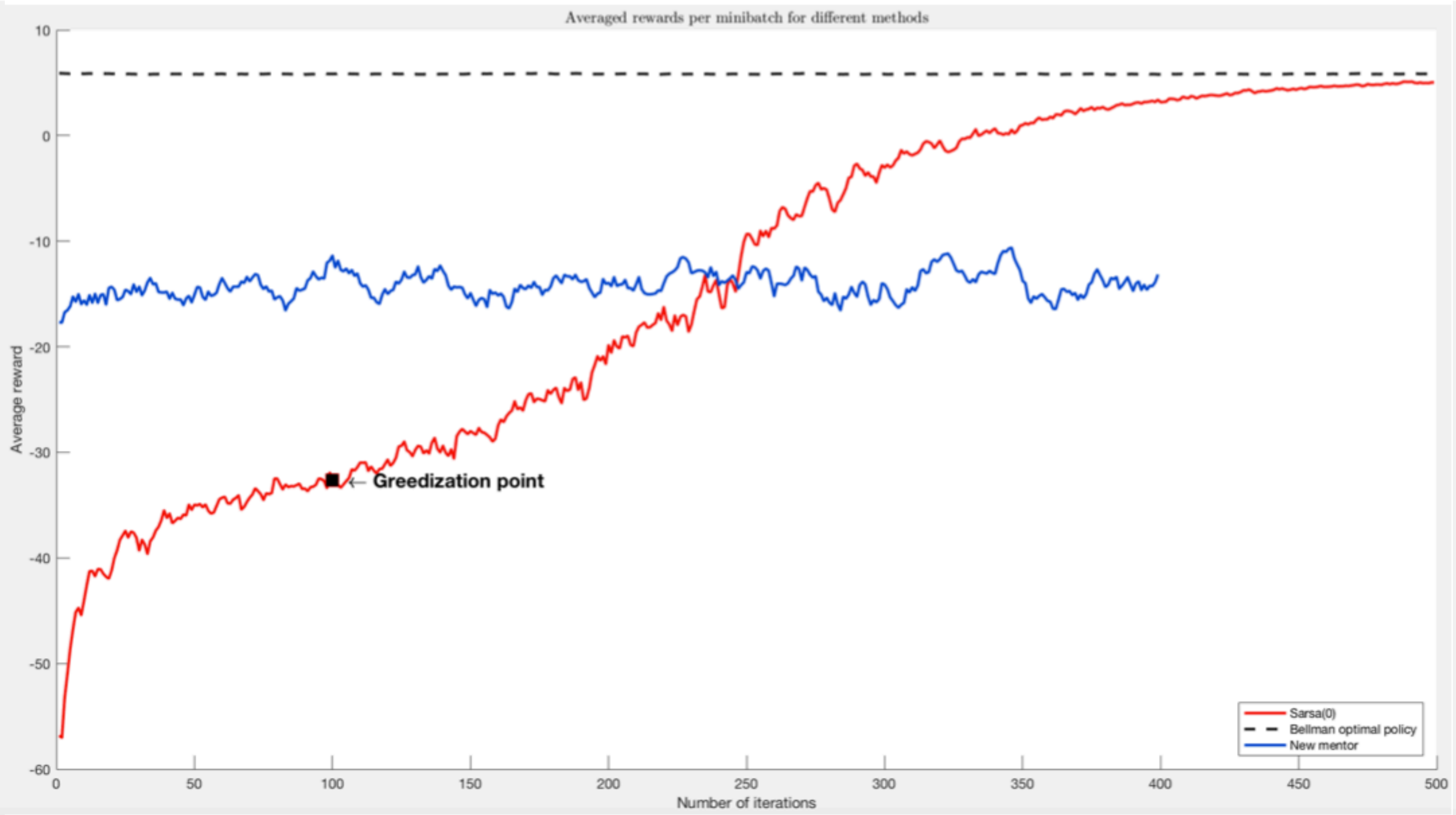
- *Value iteration* for computing optimal policy (ground truth)

GENERATING MENTORS



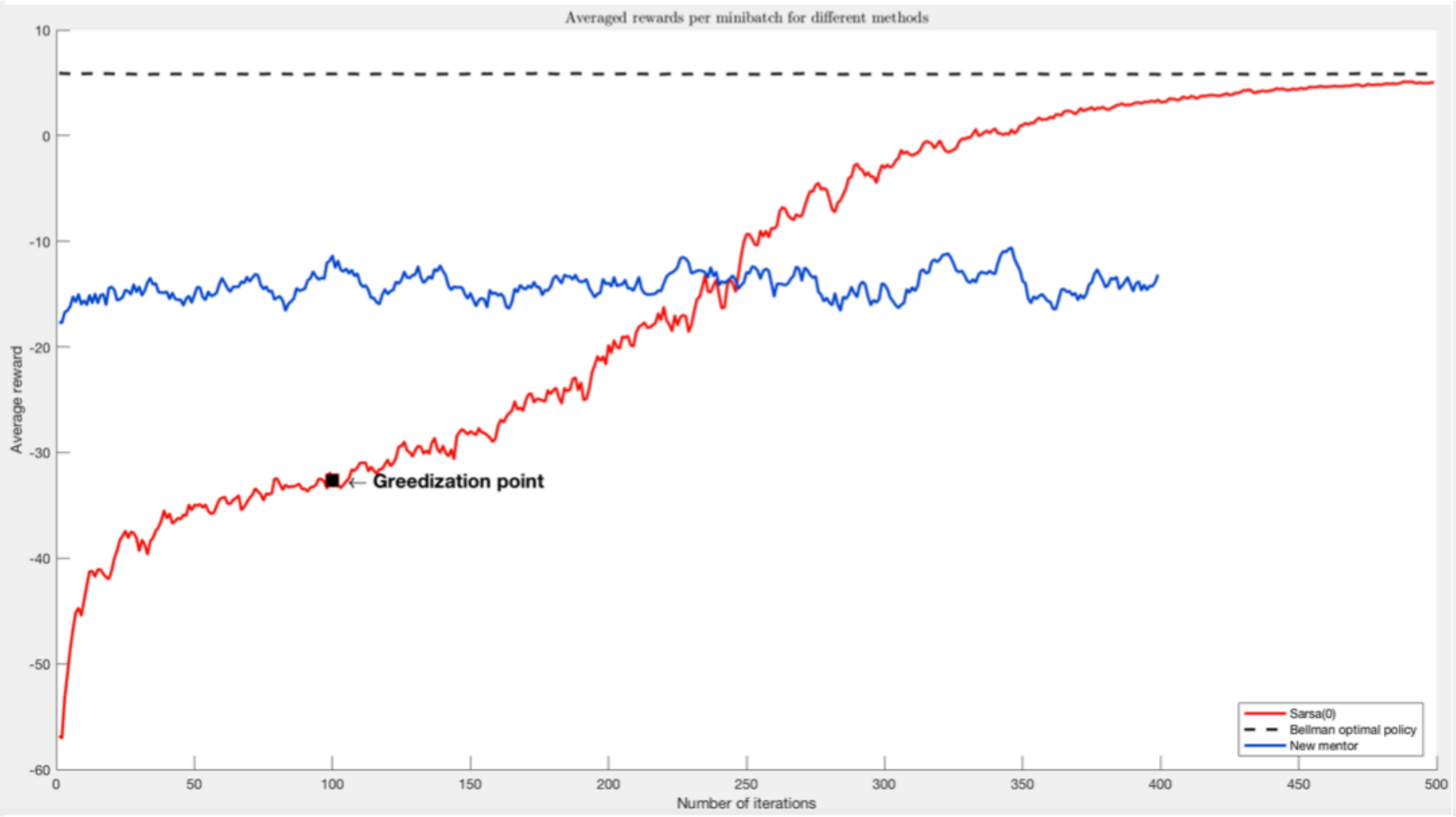
Results

GENERATING MENTORS



Results

GENERATING MENTORS



- Fairly strong hypothesis : **one mentor recommandation for every state**

Results

- **VANISHING COMPLIANCE** (The naive way)

Results

■ **VANISHING COMPLIANCE** (The naive way)

Figure : Learning Curve (Teacher 1)

Results

■ **VANISHING COMPLIANCE** (The naive way)

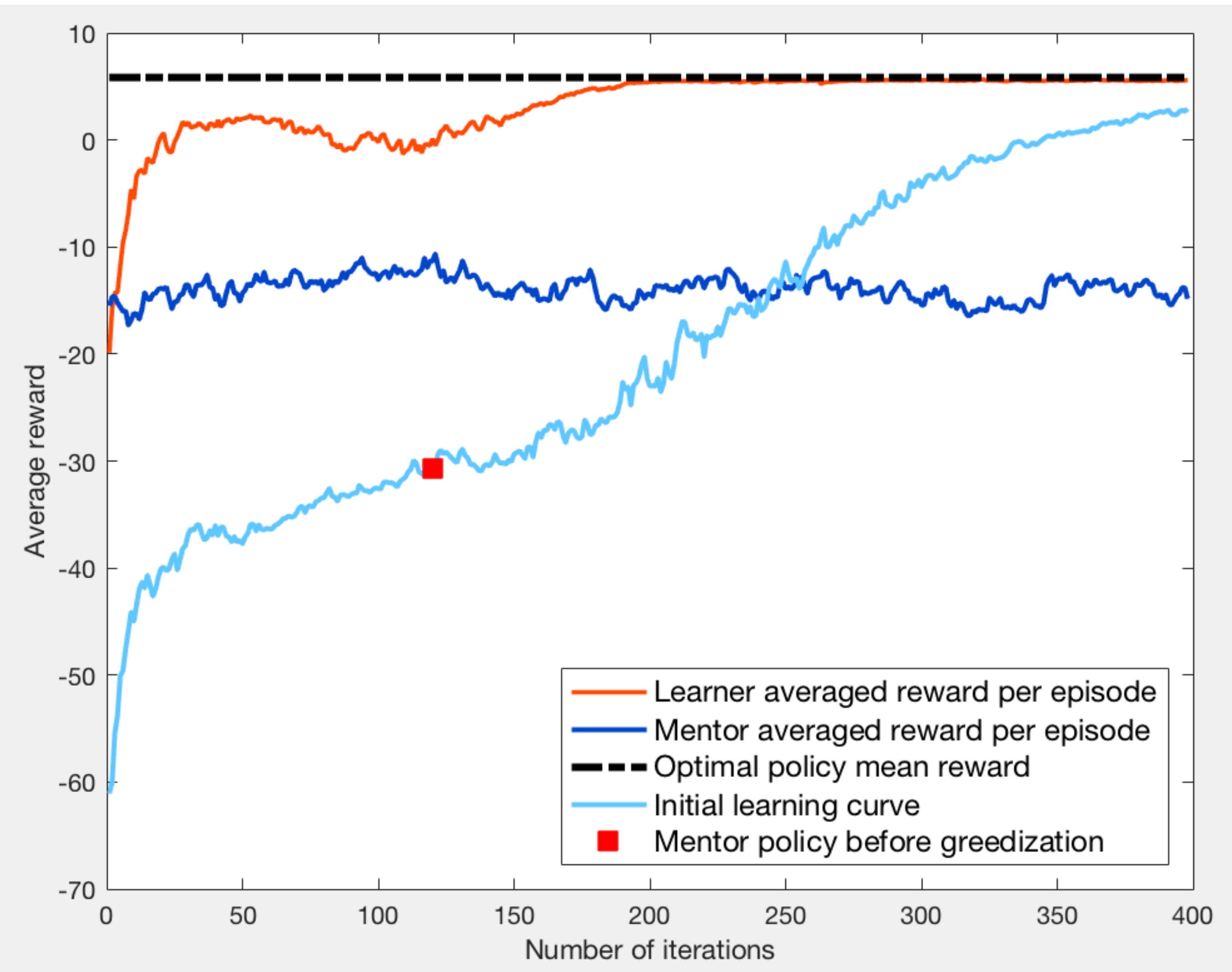


Figure : Learning Curve (Teacher 1)

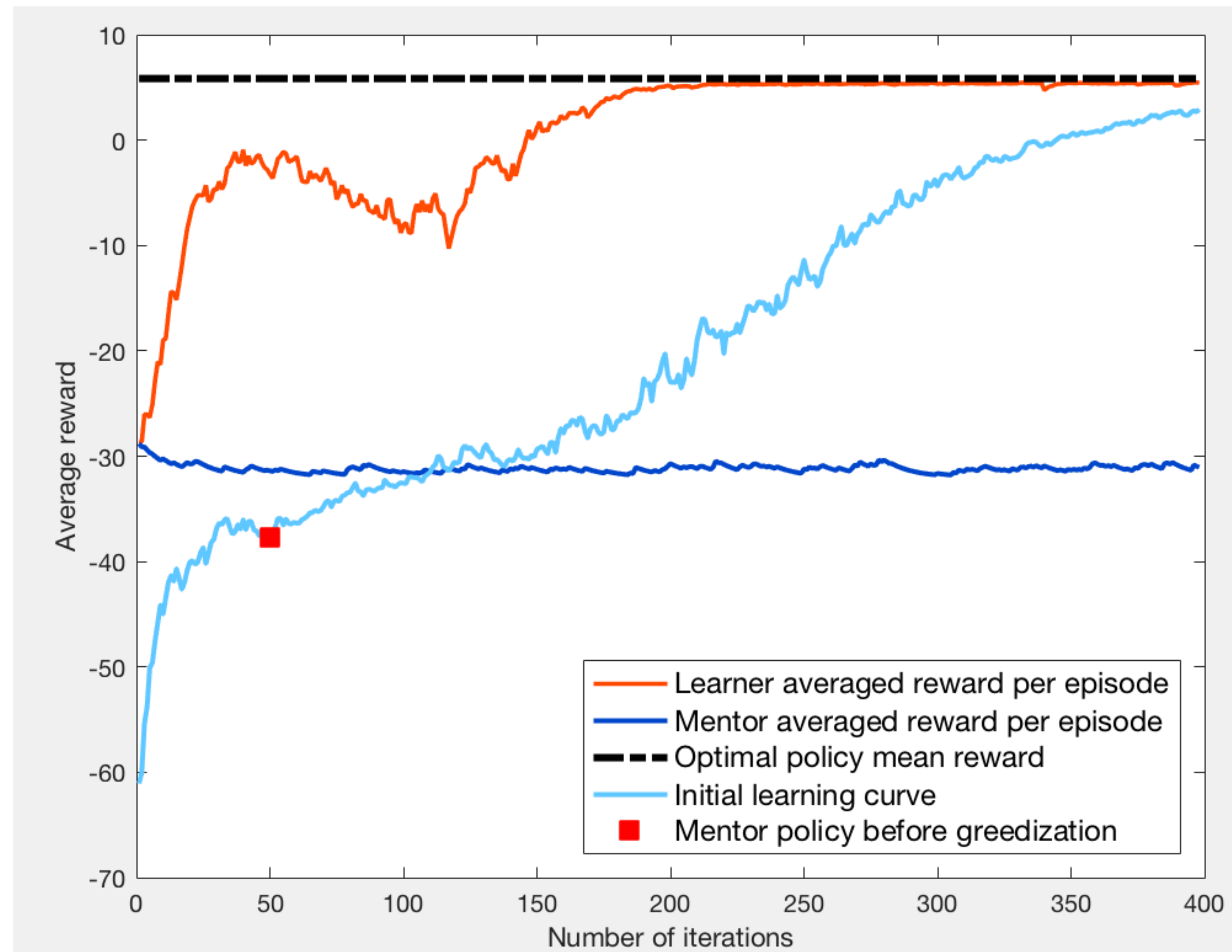


Figure : Learning Curve (Teacher 2)

■ **VANISHING COMPLIANCE** (The naive way)

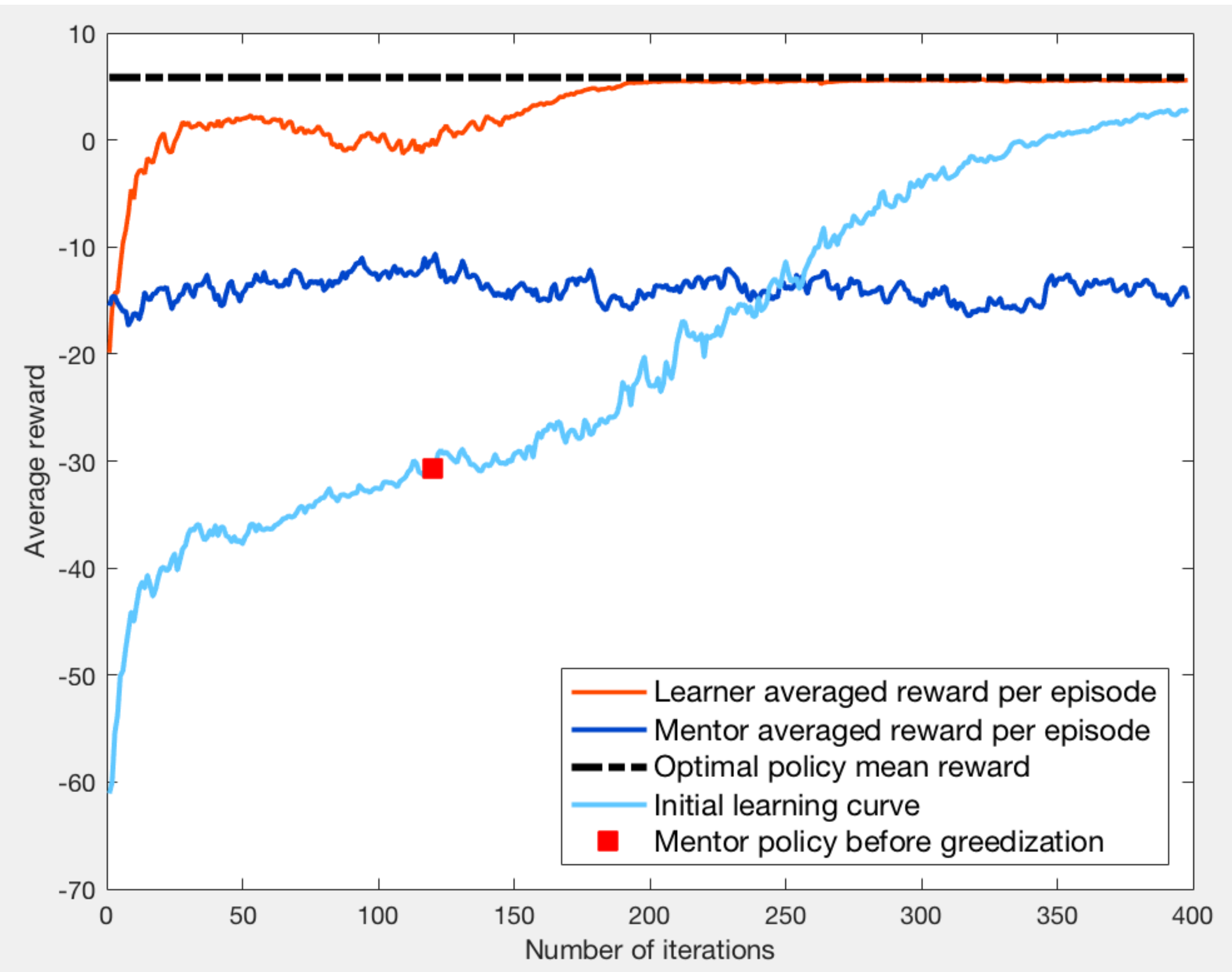


Figure : Learning Curve (Teacher 1)

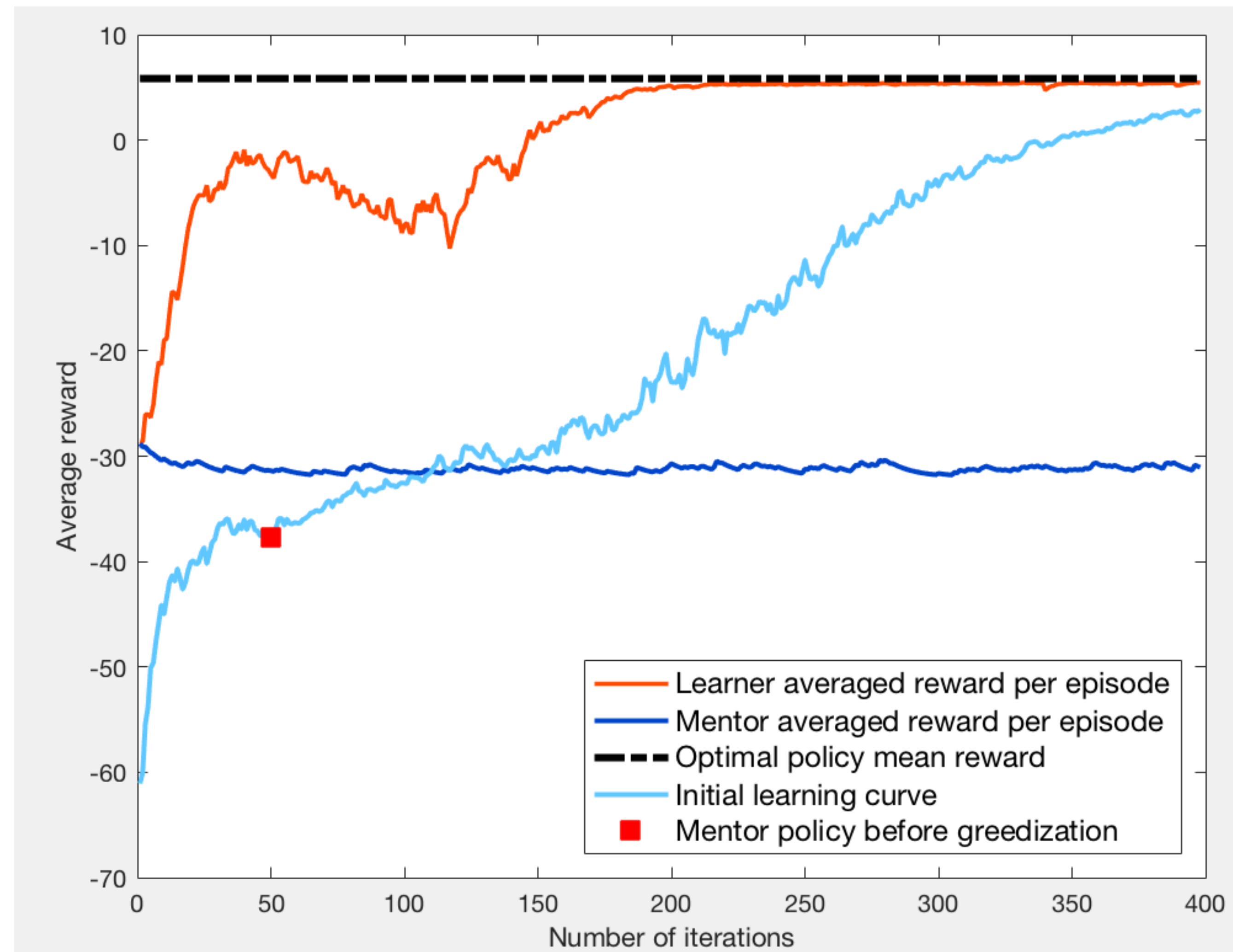


Figure : Learning Curve (Teacher 2)

- Too much time spent exploring around good solutions !

ADAPTIVE LEARNERS

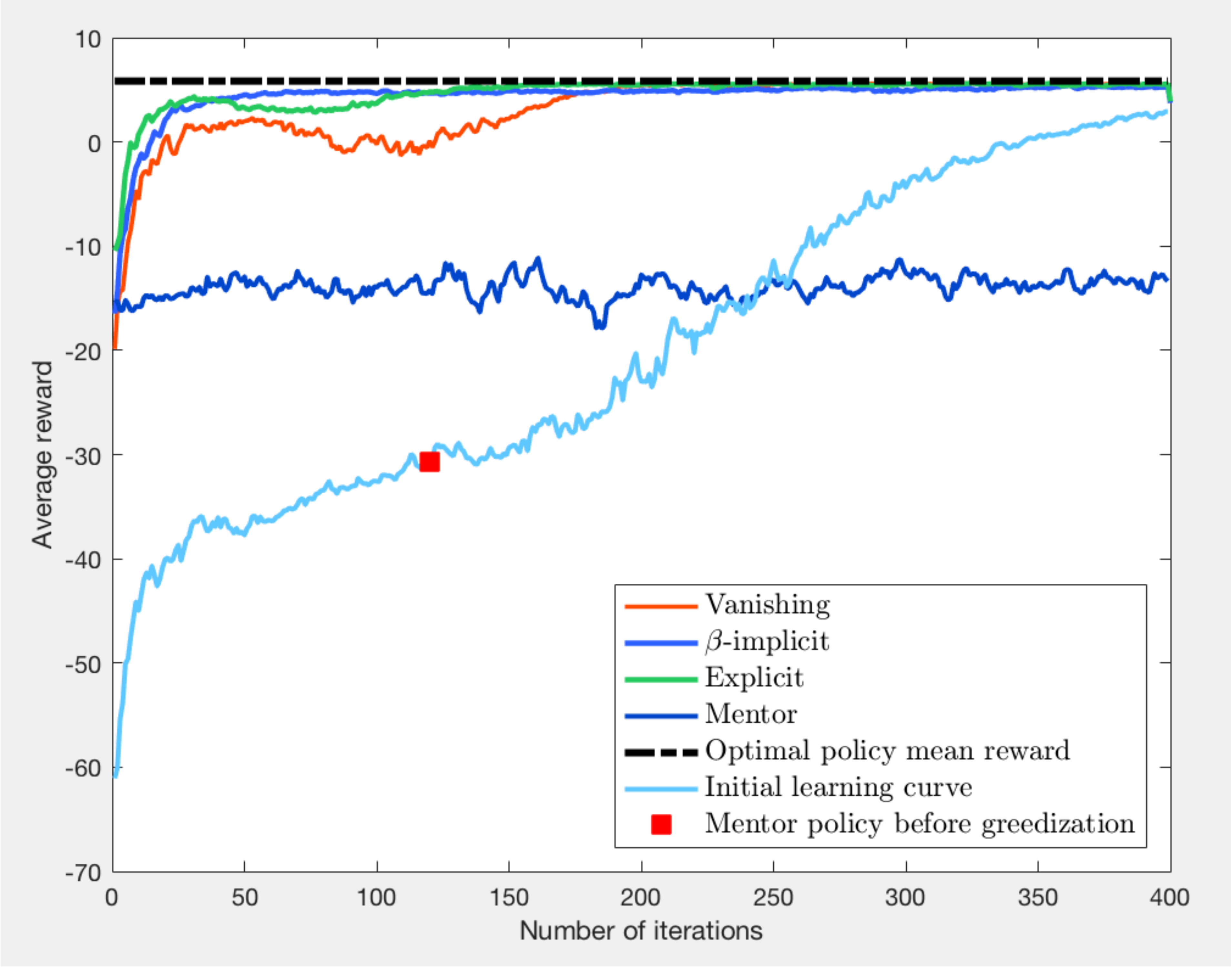


Figure : Learning Curves (Teacher 1)

Results

ADAPTIVE LEARNERS

ADAPTIVE LEARNERS

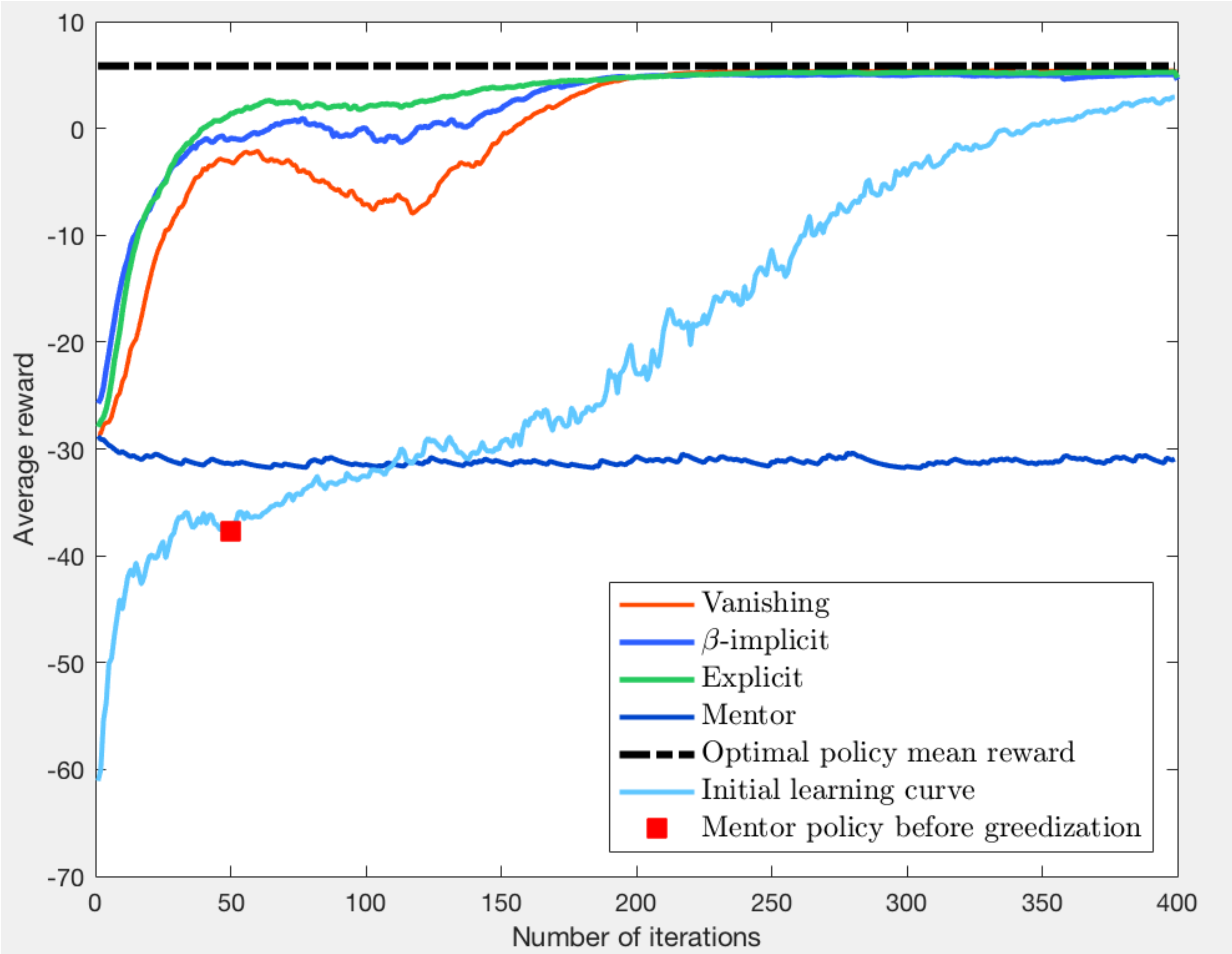


Figure : Learning Curves (Teacher 2)

ADAPTIVE LEARNERS

- Mentor optimality : linear scaling between random policy and optimal policy reward

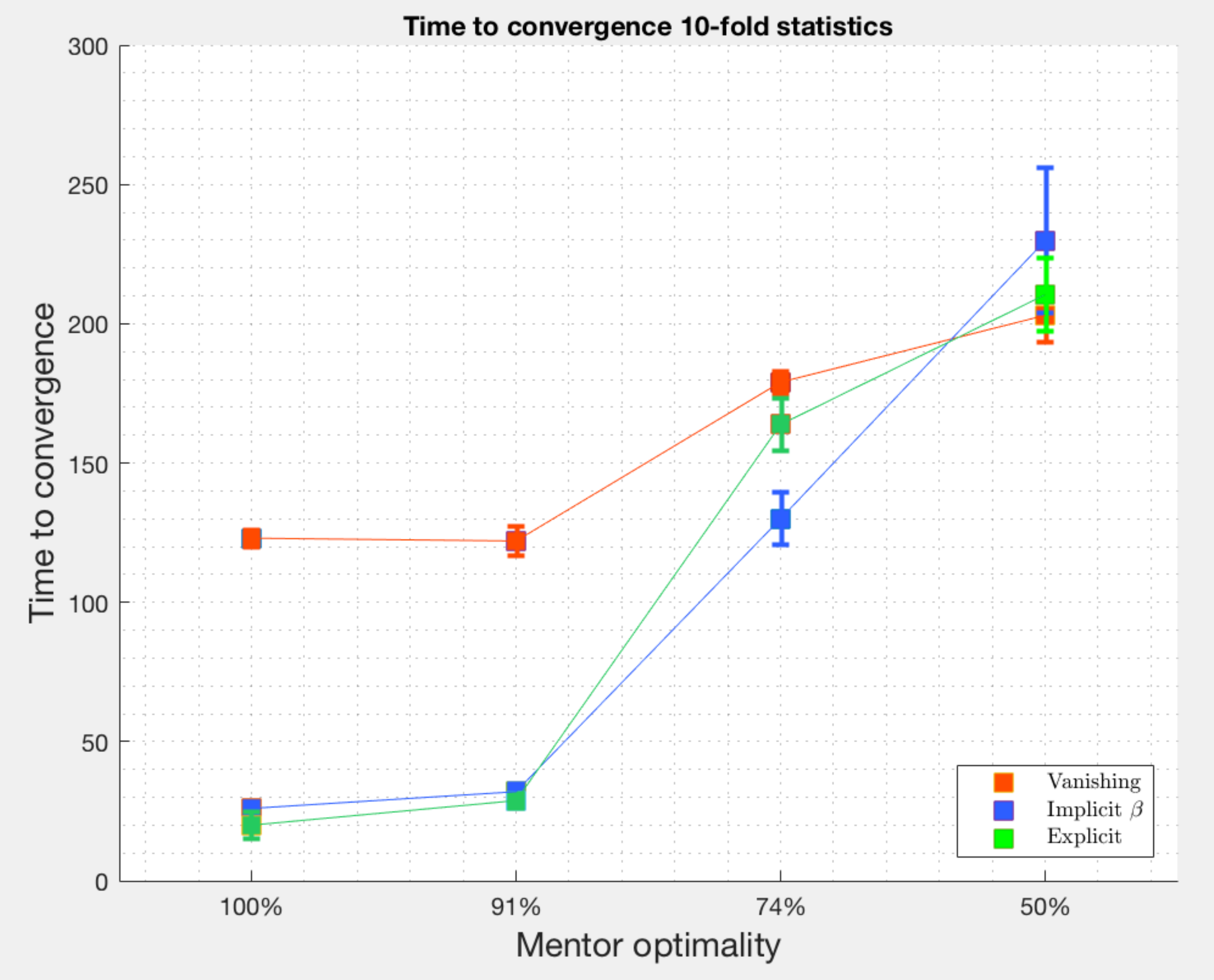


Figure : Time To Convergence

ADAPTIVE LEARNERS

- Mentor optimality : linear scaling between random policy and optimal policy reward

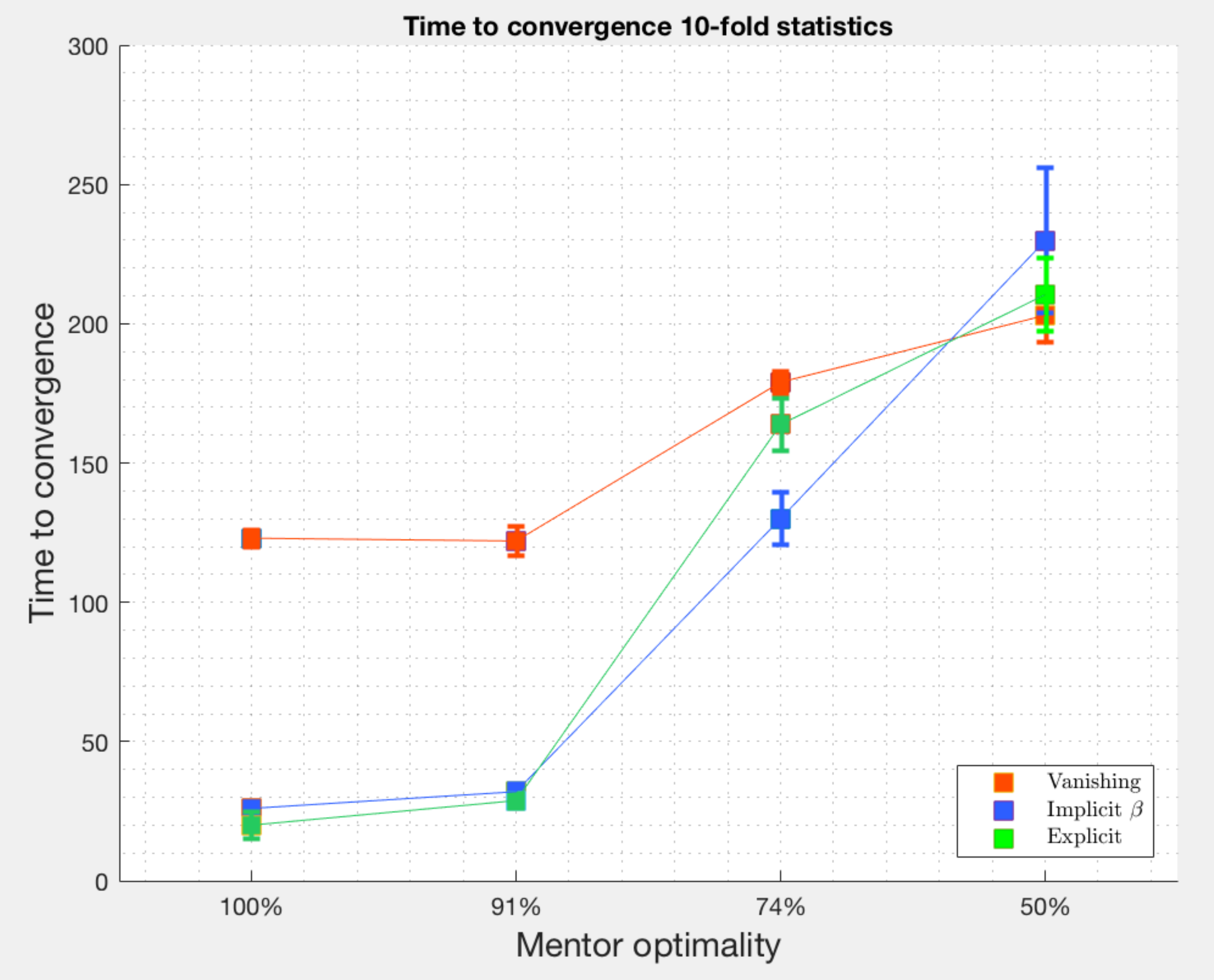


Figure : Time To Convergence

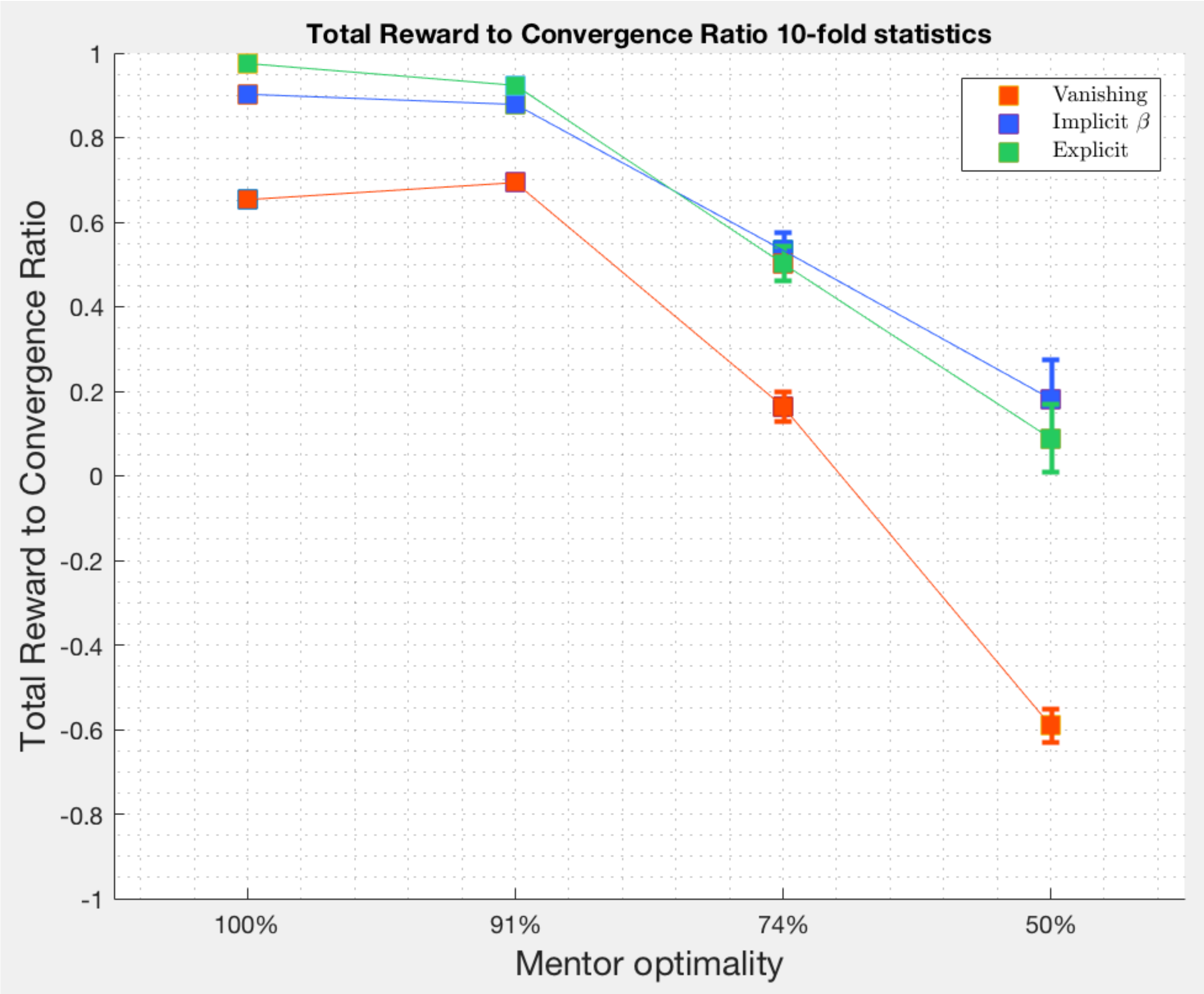


Figure : Reward Ratio to Convergence

ADAPTIVE LEARNERS

- Compared to *classical learners*

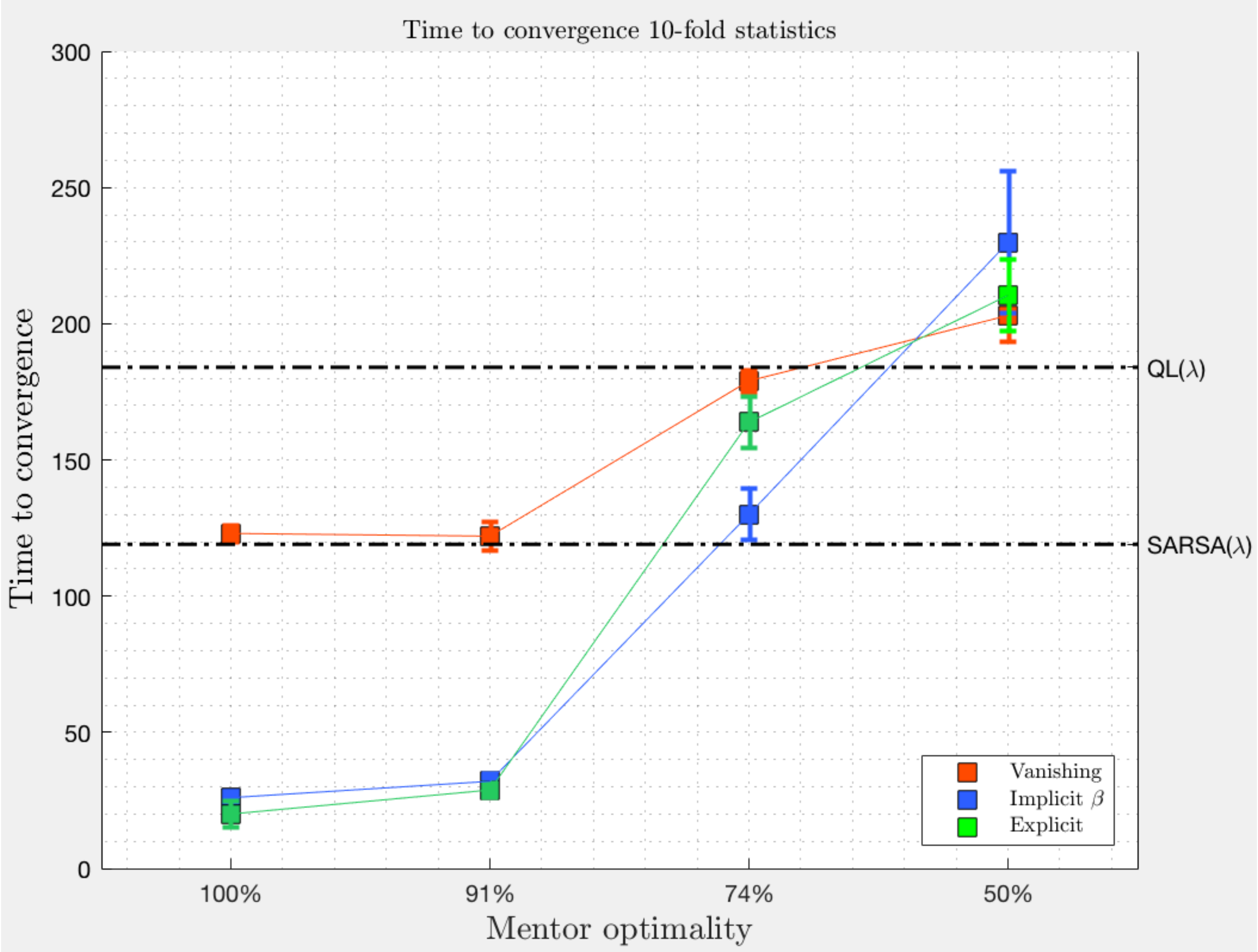
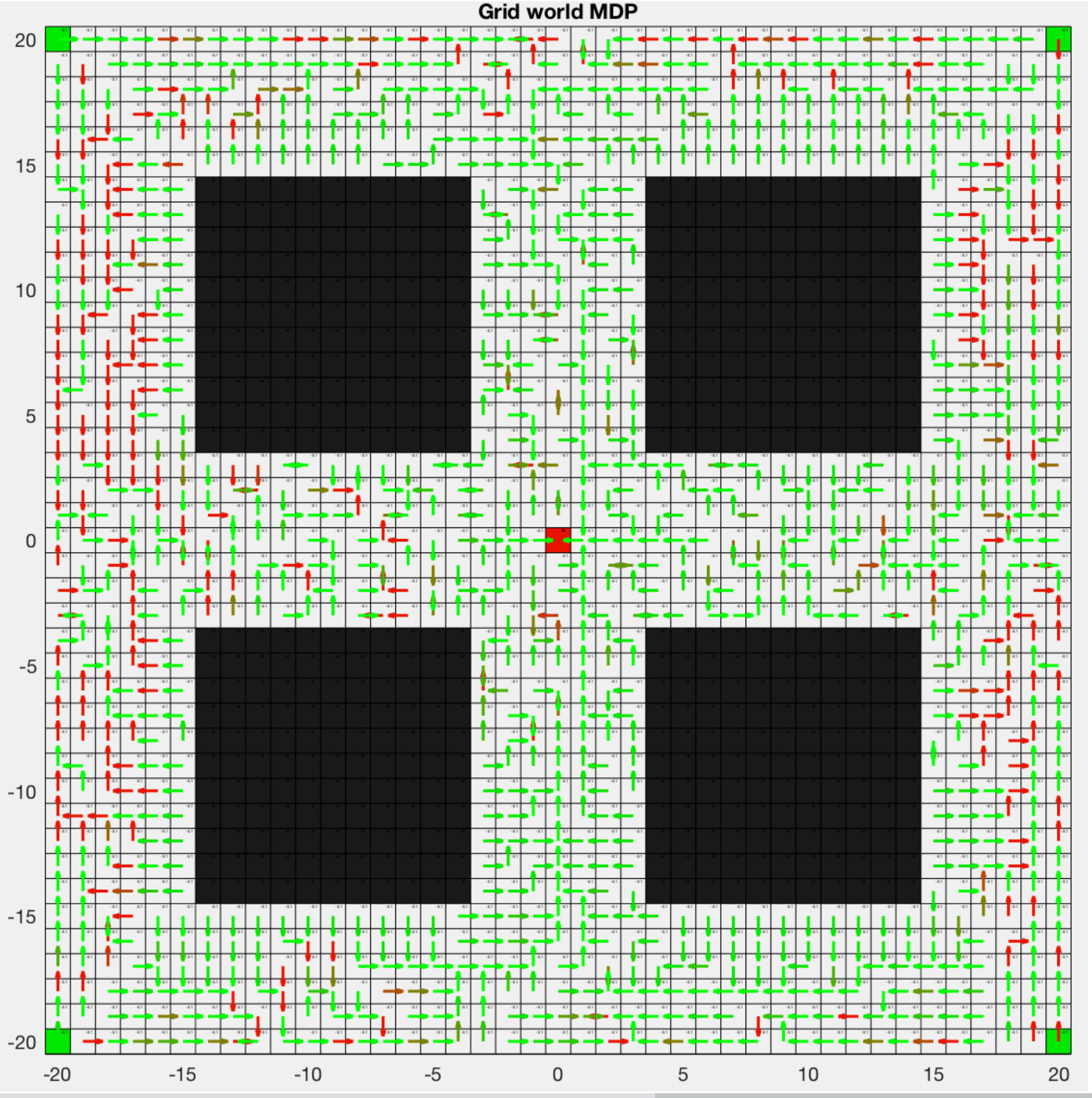


Figure : Time To Convergence

ADAPTIVE LEARNERS

- What is actually learnt ?

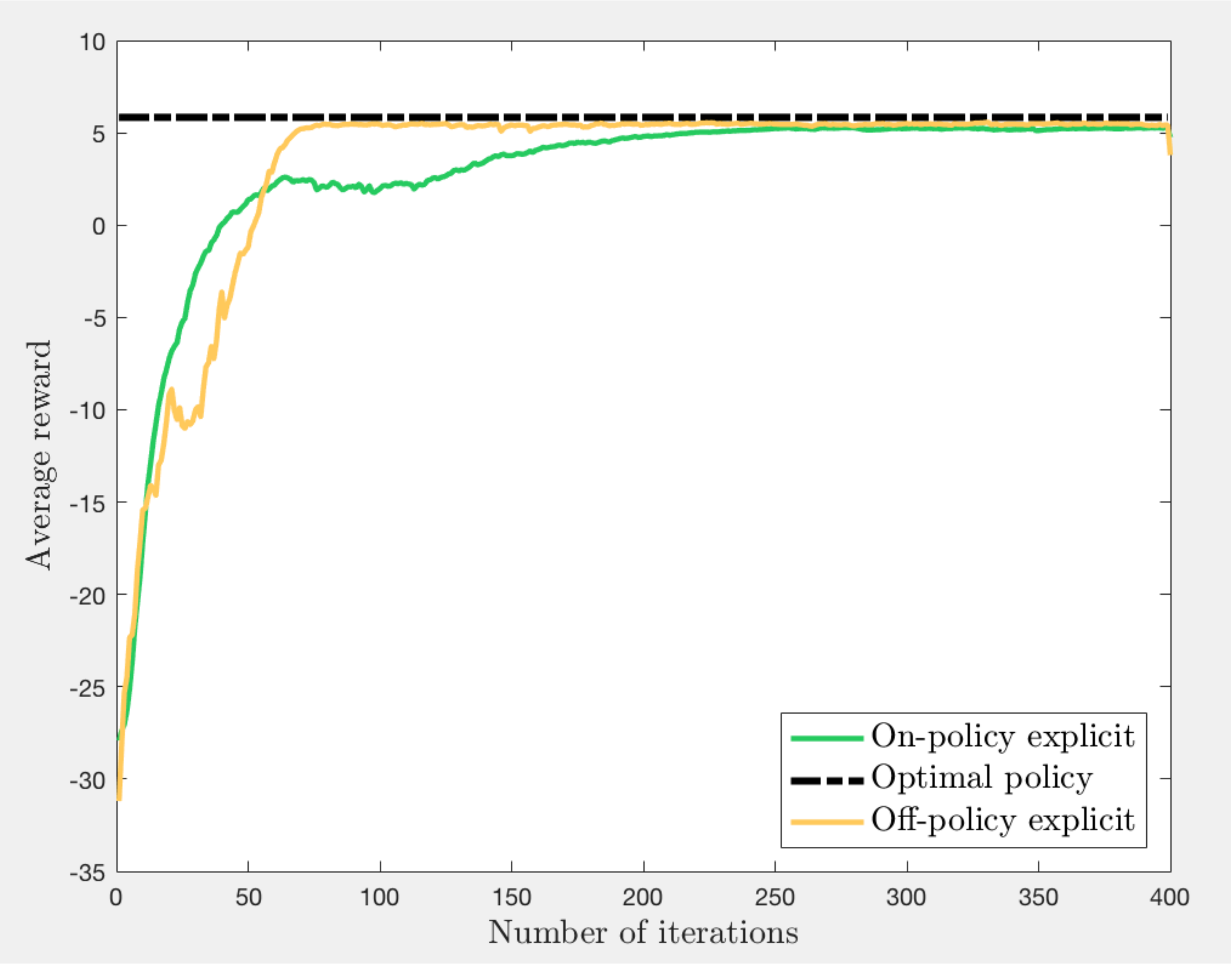


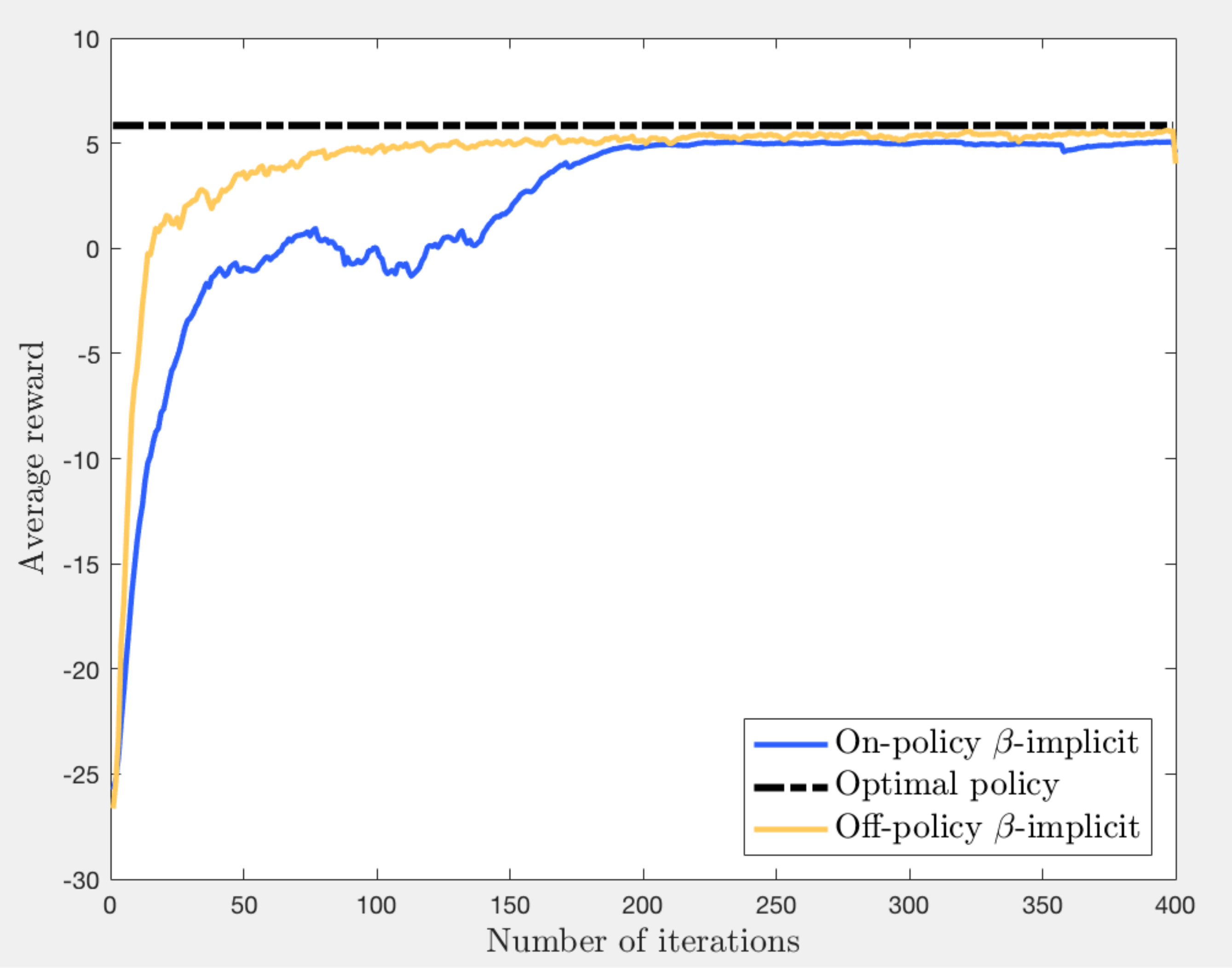
- Compliance heat-map
- Poor teacher recommandations back propagate too far
- The learner tries to circle the teacher instead of fixing it !

Results

IMPROVEMENTS

- Can off-policy learning improve this ?





- The learning now fixes the suboptimal regions !

Conclusion

■ **WHAT WE DID :**

- Provide **adaptive-compliant exploration policies**

■ **WHAT'S NEXT :**

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers

WHAT'S NEXT :

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers
  - Evaluate the optimality of a teacher

WHAT'S NEXT :

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers
  - Evaluate the optimality of a teacher
  - Extract useful informations

WHAT'S NEXT :

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers
  - Evaluate the optimality of a teacher
  - Extract useful informations
  - Speed-up the learning

WHAT'S NEXT :

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers
  - Evaluate the optimality of a teacher
  - Extract useful informations
  - Speed-up the learning

WHAT'S NEXT :

- Still some work to do !

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers
  - Evaluate the optimality of a teacher
  - Extract useful informations
  - Speed-up the learning

WHAT'S NEXT :

- Still some work to do !
  - Generalize to sparse recommandation

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
  - Learn from suboptimal teachers
  - Evaluate the optimality of a teacher
  - Extract useful informations
  - Speed-up the learning

WHAT'S NEXT :

- Still some work to do !
  - Generalize to sparse recommandation
  - Implement eligibility traces

Conclusion

WHAT WE DID :

- Provide **adaptive-compliant exploration policies**
    - Learn from suboptimal teachers
    - Evaluate the optimality of a teacher
    - Extract useful informations
    - Speed-up the learning

WHAT'S NEXT :

- Still some work to do !
    - Generalize to sparse recommandation
    - Implement eligibility traces
    - Test in continuous MDP

THANK YOU FOR YOUR ATTENTION !