INSTITUTO SUPERIOR TÉCNICO

INSTITUTO DE SISTEMAS E ROBÓTICA

INVERSE REINFORCEMENT LEARNING WITH EVALUATION

Valdinei Freire da Silva[*,**]
Pedro Lima[*]
Anna Helena Reali Costa[**]

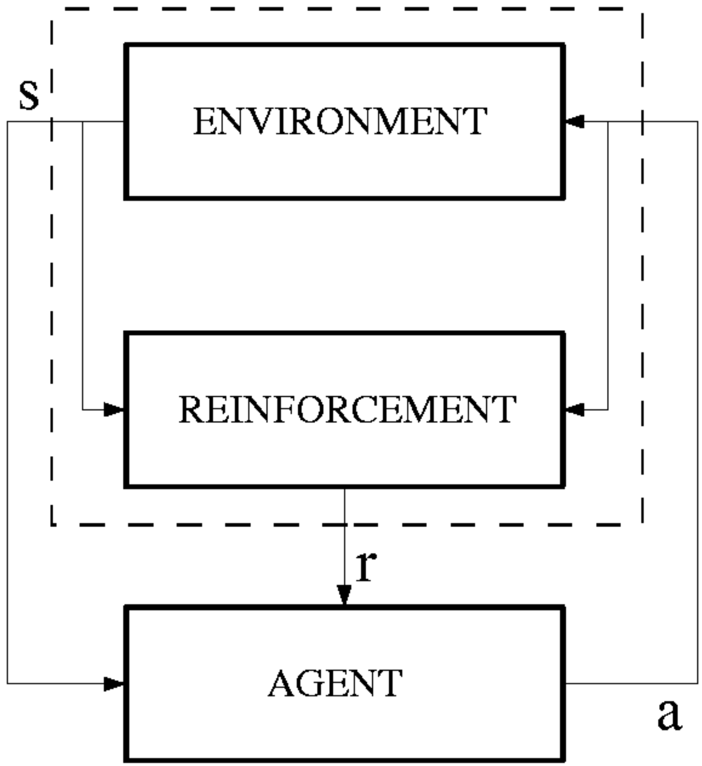
[*]Institute for Systems and Robotics
Instituto Superior Técnico
Lisbon, PORTUGAL

[**]Laboratório de Técnicas Inteligentes
Escola Politécnica – Universidade de São Paulo
São Paulo, BRAZIL

Schedule

- Reinforcement Learning
- Inverse Reinforcement Learning
- IRL with Evaluation
- Algorithms
- Experiment
- Conclusion

Reinforcement Learning
1-Environment Model



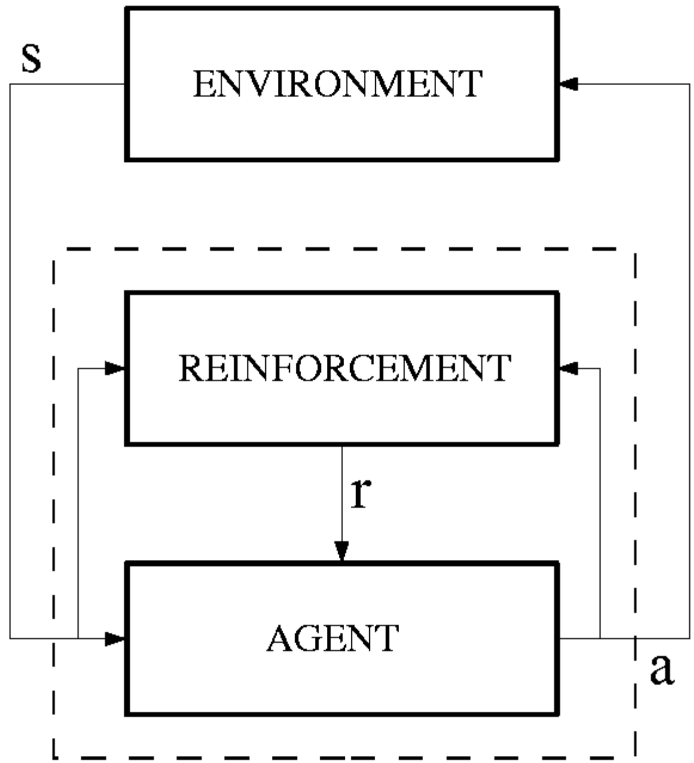
Reinforcement Learning
2-Properties

- Unknown and Stochastic Environment
- Inference Learning (trial and error)
- Partial Evaluation of each action (reinforcement)
- Sequential Problem (prediction)
- Objective: to obtain an action policy that maximise the sum of reinforcements

$$V = \sum_{t=0}^{\infty} r_t$$

- Solutions:
  - temporal difference (Markovian reinforcements)
  - policy search (evaluating a policy)

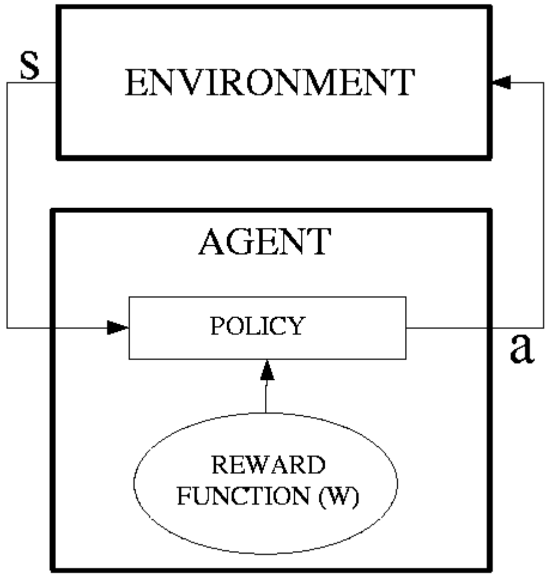
Reinforcement Learning
3-Programming an Agent



Reinforcement Learning
4-Reinforcement Function

- Most famous algorithms consider Markovian reinforcements
  - features (score, hit a wall, find a resource, etc.)
  - weight vector
  - additive, linear and independent
- How to describe different reinforcement functions?
  - collecting water with a finite size glass
  - Possible solution: use of history (POMDP)
- How to discover unknown reinforcement function?
  - What the value of a score in soccer game when the game is: 1x0, 0x0, 1x0, 2x0, 3x0
  - Possible solution: preference elicitation

Inverse Reinforcement Learning
1-Definition



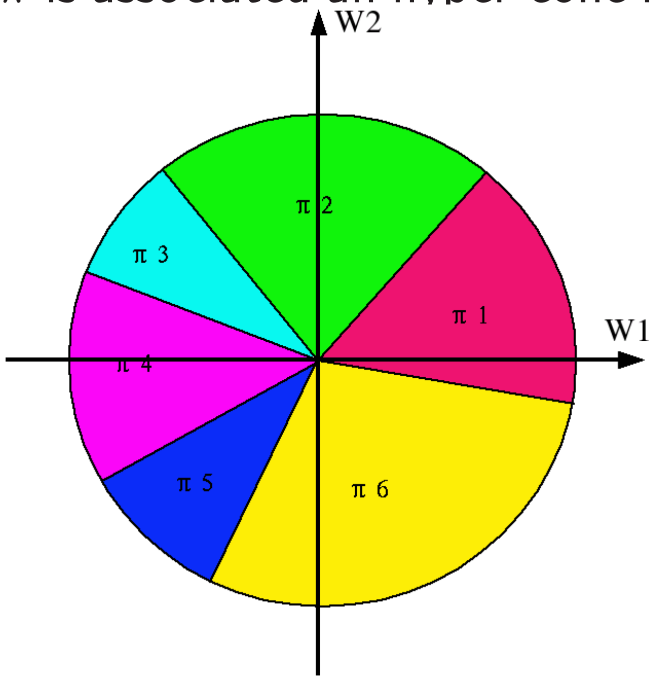
- Given the agent's policy ($\mathcal{S} \rightarrow \mathcal{A}$) determine the weight vector $\mathbf{W}$
- Given the agent's behaviour (history of pairs (s, a) summarised by a feature vector μ) determine the weight vector $\mathbf{W}$

Inverse Reinforcement Learning
2-Analytic Solution

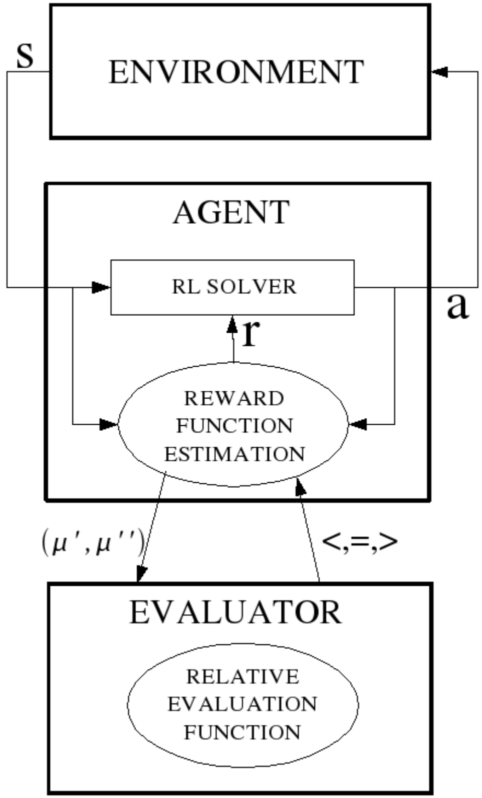
- Characteristic of the set of solutions [Ng and Russell,00]:

$$(T_{\pi^*} - T_a)(I - \gamma T_{\pi^*})^{-1} \cdot R \geq 0 \text{ for all } a \in \mathcal{A}$$

- to each policy π is associated an hyper-cone in the weight space



Inverse Reinforcement Learning with Evaluation 1-Definition



- given relative evaluation of measurements of some agent's behaviours over time, determine the weight vector $\mathbf{W}$ of the relative evaluation.

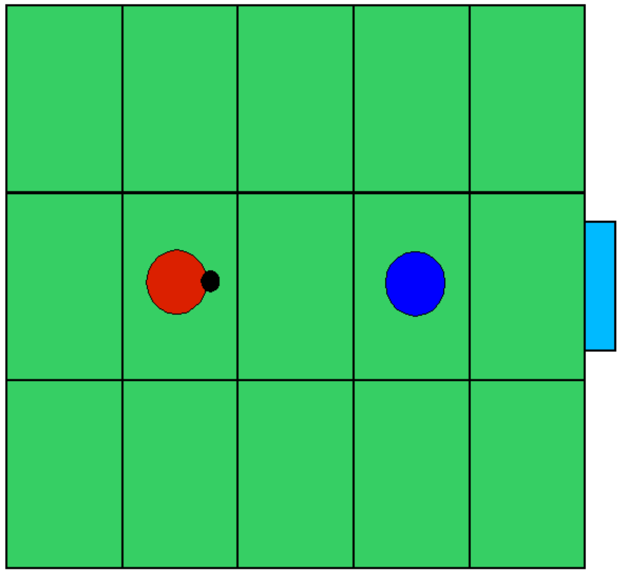
Inverse Reinforcement Learning with Evaluation
2-Local Search IRLE

- Objective: find out a weight $\mathbf{W}$, where $\pi^*_{\mathbf{W}}$ beats any other $\pi^*_{\mathbf{W}'}$
- Hypothesis: the evaluator can average the behaviours presented
- Algorithm (Local Search):
  - given $\mathbf{W}$ the current best weight
  - execute $\pi^*_{\mathbf{W}}$ during T time step
  - choose a neighbour $\mathbf{W}'$ of $\mathbf{W}$
  - execute $\pi^*_{\mathbf{W}'}$ during T time step
  - if $\pi^*_{\mathbf{W}'}$ is better evaluated than $\pi^*_{\mathbf{W}}$, updates $\mathbf{W} \leftarrow \mathbf{W}'$
- Heuristic:
  - when the neighbour is better, keeps the same direction
  - choose neighbours with different policies
  - choose direction that respect the last evaluations

Inverse Reinforcement Learning with Evaluation
2-Expected IRLE

- Objective: find out the mean weight $\hat{\mathbf{W}}$ that averages all $\pi^*_{\mathbf{W}}$ that respect answer constraints
- Algorithm (Q-Learning):
  - choose a weight vector $\mathbf{W}$ that satisfy all known constraints
  - update the mean weight $\hat{\mathbf{W}}_{t+1} = \hat{\mathbf{W}}_t + t^{-1}(\mathbf{W} - \mathbf{W}_t)$
  - choose a action a and execute it
  - if the run has finished, ask for an evaluation
  - update the known constraints
- Problems:
  - number of constraints very large (choose the most common)
  - constraints can be non-linear (try satisfying the most)
  - average must be normalised (expected utility theory)

Experiment
1-Scenario

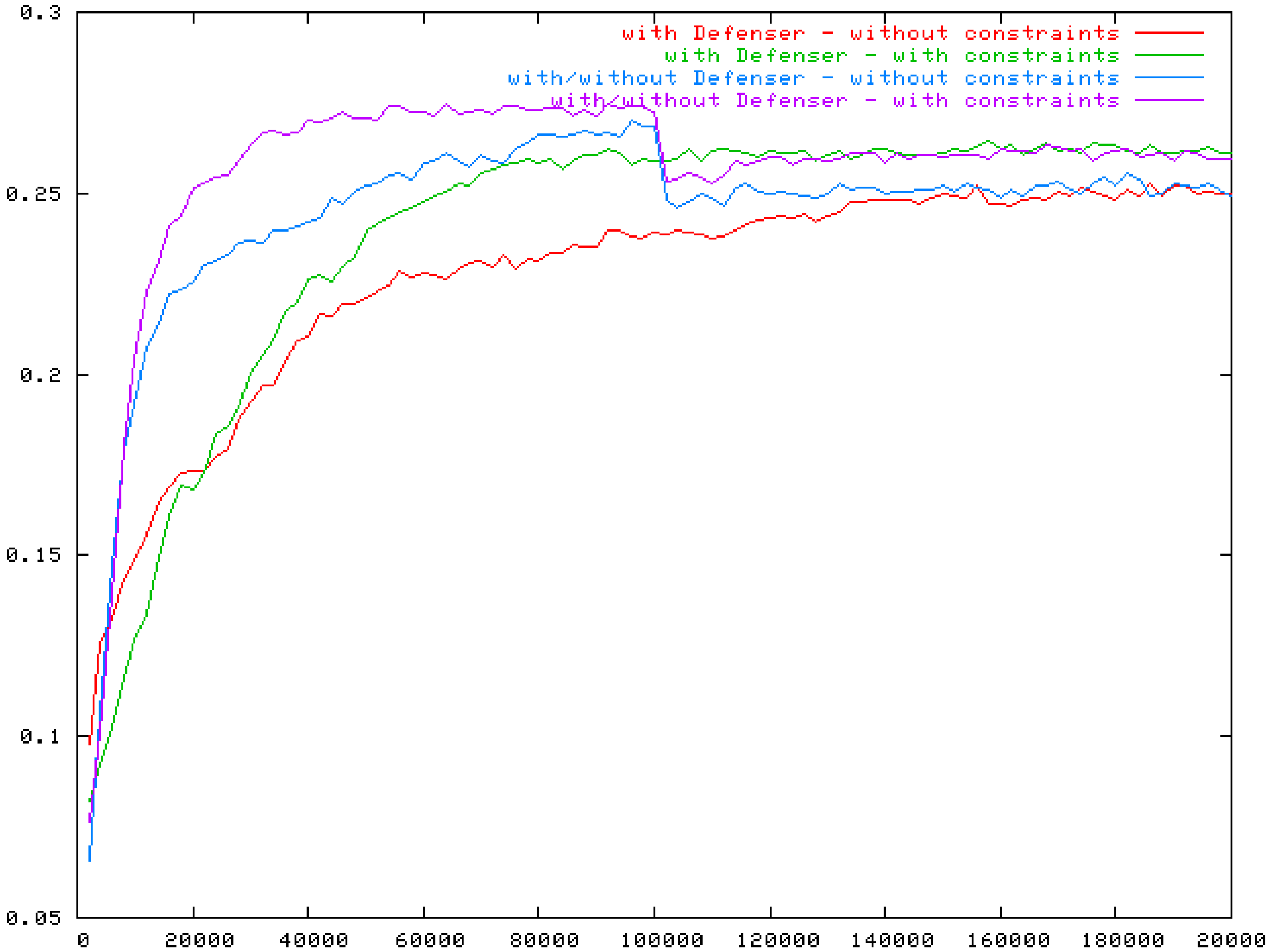


- Attacker (red) must learn to score as many as possible per time (average reinforcements $\neq$ sum reinforcements)
- Defender (blue) tries deterministically to intercept the attacker
- Attacker score with probability $.5^{(D-1)}$, where D is the Manhattan distance to the goal
- A new run start when ball is kicked or defender intercept attacker

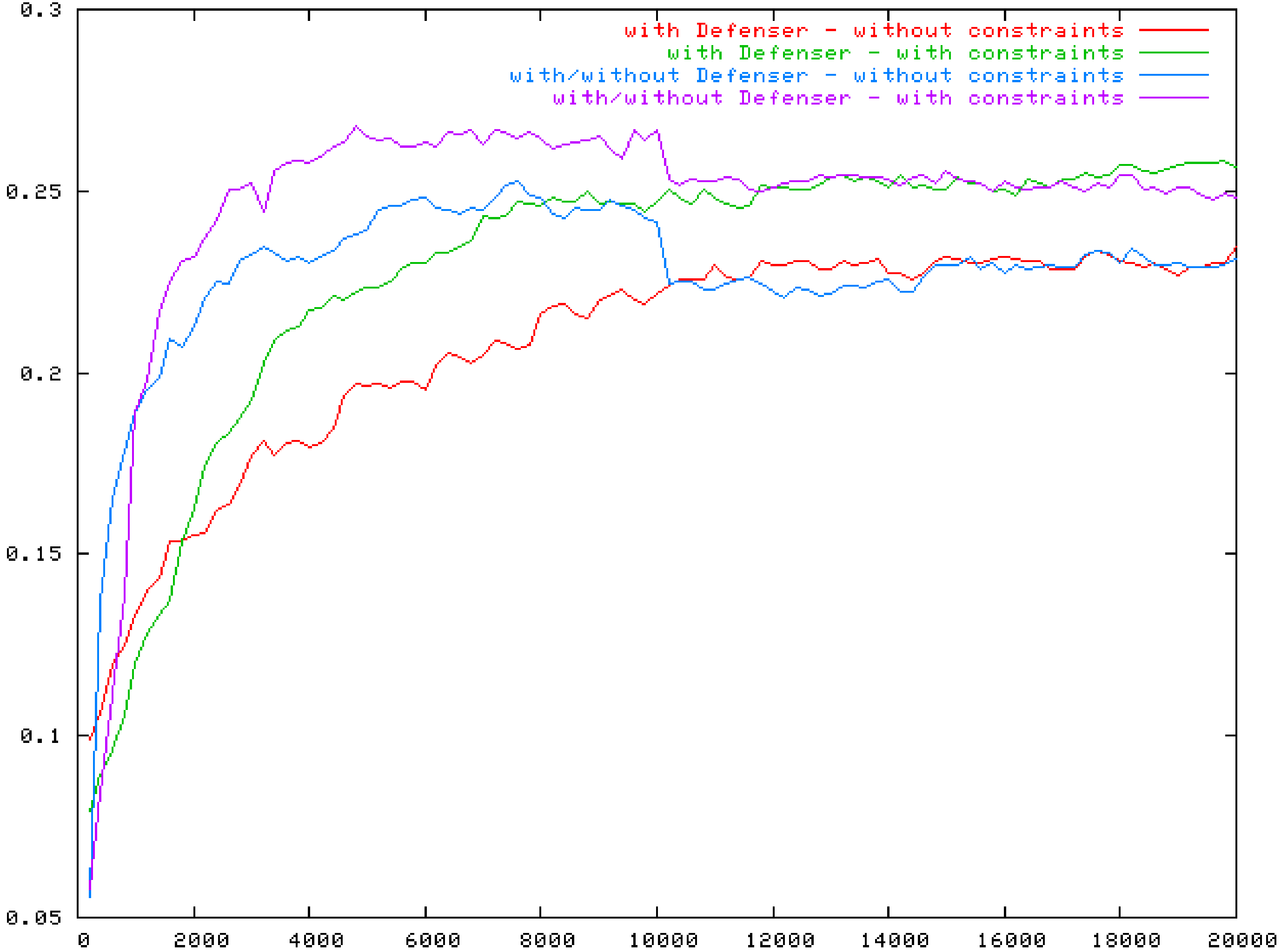
Experiment
2-Local Search

- Experiment 1:
  - Without and with defender
  - Without and With heuristic based on 10 constraints
  - Solving an MDP based on model
  - period $T = 100$ and $T = 1000$
- Experiment 2:
  - fixed learning time 20000 steps
  - different periods $T = 100$, $T = 200$, $T = 500$ and $T = 1000$
- Experiment 3:
  - solving the RL problem during execution

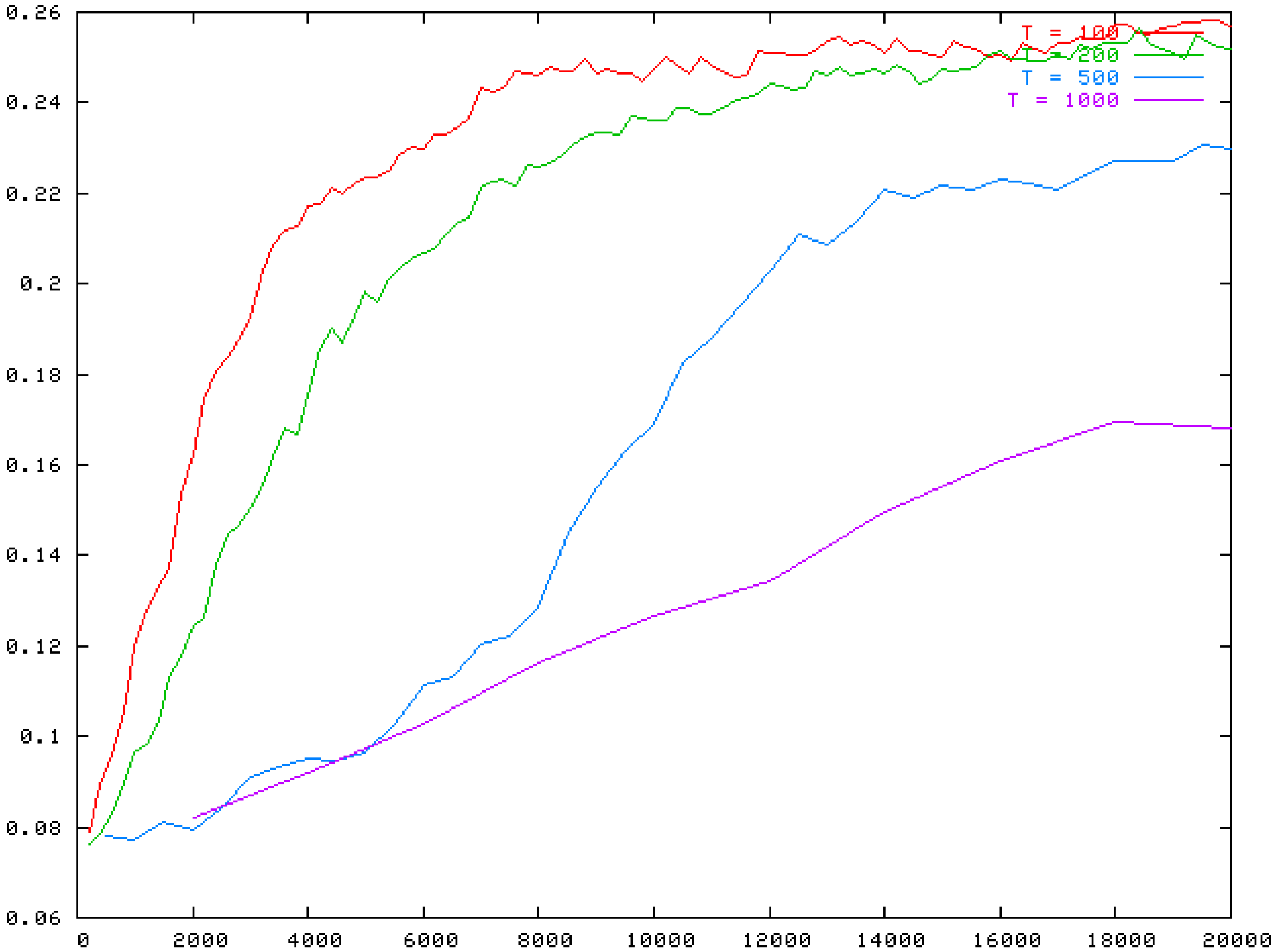
Experiment
2-Local Search ($T = 1000$)



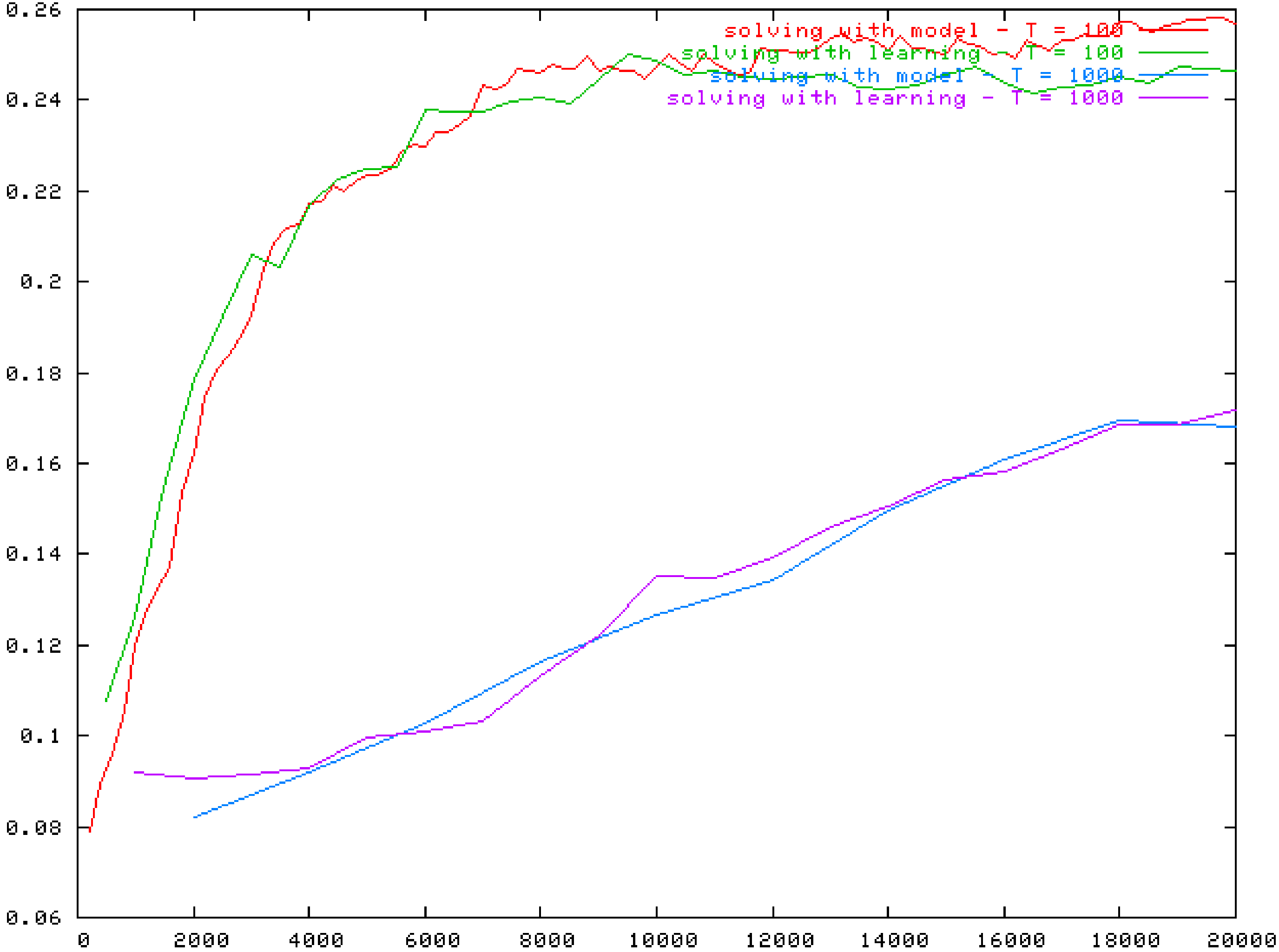
Experiment
2-Local Search ($T = 100$)



Experiment
2-Local Search (20000 steps)



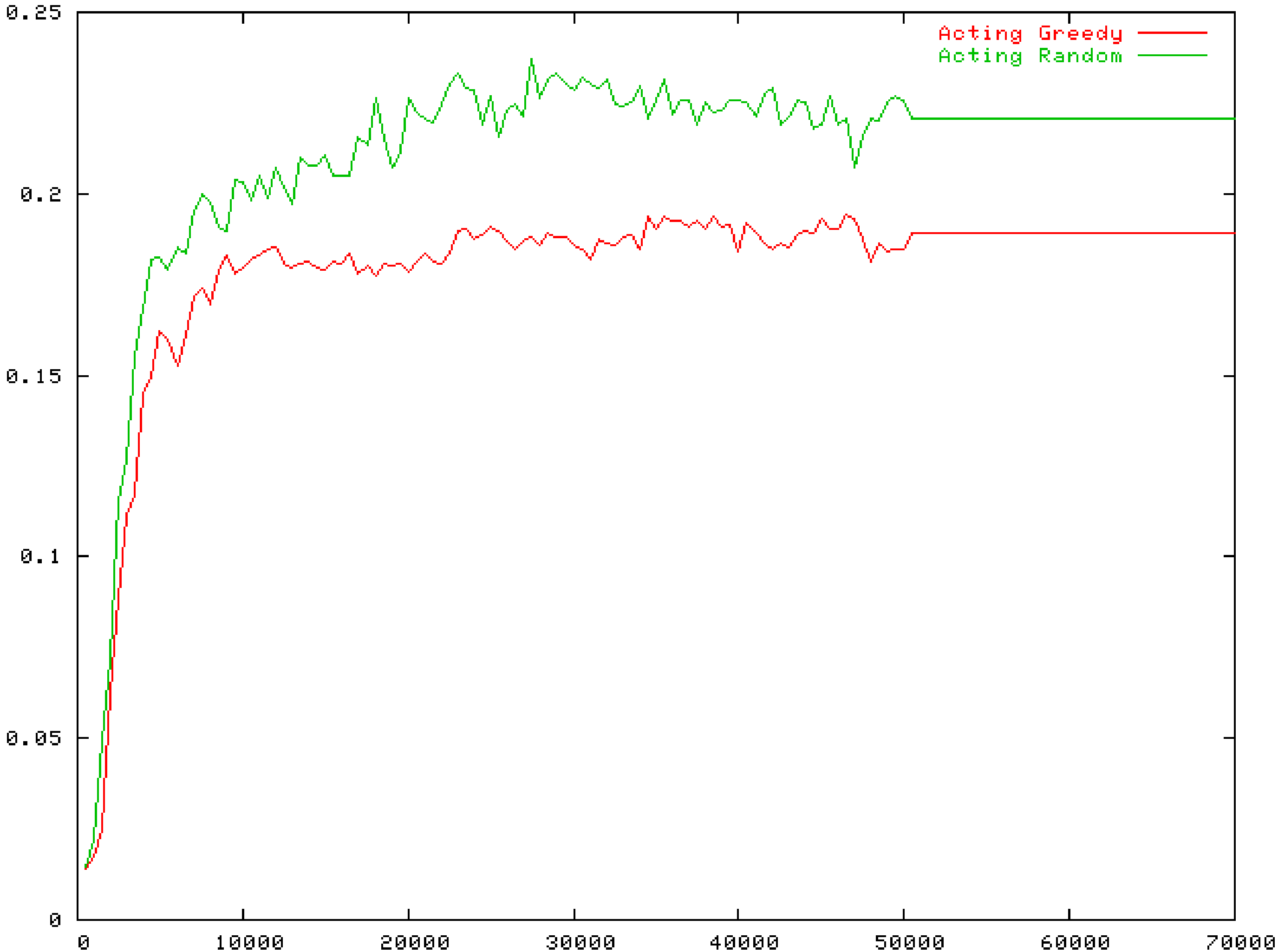
Experiment
2-Local Search (Learning through execution)



Experiment
3-Expected IRLE

- Experiment:
  - environment without Defender
  - considers 20 most common feature vectors
  - learns with Q-Learning algorithm through 50000 steps
  - acting randomly or ϵ-greedy
  - transferring to environment with Defender

Experiment 3-Expected IRLE



Conclusion

- Preference Elicitation
  - Abstraction from Environment
  - Transfer of objectives
- Problems
  - It is necessary too many evaluations
  - It is not useful against human evaluators
- Future Works
  - Trying to show behaviours that give more information

Bibliography

- Keeney, R. L. and H. Raiffa (1976). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs.* Wiley. New York.
- Ng, Andrew Y. and Stuart Russell (2000). Algorithms for inverse reinforcement learning. *In: Proceedings of the Seventeenth International Conference on Machine Learning.*
- Ross, Sheldon M. (1970). *Applied probability models with optimisation applications.* Holden-Day. San Francisco.
- Russell, Stuart (1998). Learning agents for uncertain environments (extended abstract). *In: Proceedings of Eleventh Annual Conference on Computational Learning Theory.* ACM Press.
- Watkins, Christopher J. C. H. (1989). Learning from Delayed Rewards. PhD thesis. University of Cambridge.