# *Ipseity*

# An open-source platform for studying sequential decision problems in multiagent systems

Fabrice LAURI

IRTES-SeT

2014, 12th May
JFPDA, Liège, Belgique

# Outline

# Introduction

As pointed out by **[Kovacs, 2011]**, solving sequential decision problems has empirical aspects which involves:

- writing codes,
- conducting experiments, and
- comparing results obtained from alternative approaches.

There are a wide range of environments, agent architectures, decision-making algorithms and performance measures.

# Introduction: range of environments

Multi-agent environments are defined according to many features:

- assumptions on the state/action/time spaces
- deterministic or stochastic state transitions
- static or dynamic decision-making processes
- populated by independent learner agents, cooperative agents or competitive agents...

# Introduction: range of decision-making techniques

RL algorithms have many dimensions **[Sutton, 1998; Busoniu, 2010]**.

The decision-making process of learning agents may:

- be based on Value-Iteration, Policy Iteration or Policy Search
- be model-based or model-free
- be on-policy or off-policy
- be offline or online
- use a wide range action selection method
- use a wide range of function approximators

But there are also alternative approaches coming from other fields...

# Introduction: experiments and statistics

Besides, experiments may generate multiple statistics, such as:

- the agents' interactions over time,
- the discounted return per agent over time,
- the number of steps taken by some agent to reach a goal,
- the Q-function.

# Motivations

- There are currently no fully-featured experimental platforms, such as WEKA for Supervised Learning, dedicated to Reinforcement Learning.
- "Science is founded on replicability. Having access to the source code used by others is invaluable and makes for better science.", **[Kovacs, 2011]**
- Application of RL is still more an art than a science...

# Objectives

- *Ipseity* has been especially designed to facilitate the synthesis and the validation of *Cognitive Systems* within Multi-Agent Systems.
- Cognitive Systems embed a set of Artificial Intelligence decision-making techniques that may be executed alternatively or concurrently, in parallel or sequentially.
- *Ipseity* is targeted at a broad range of users interested in Artificial Intelligence in general, including industrial practitioners, as well as machine learning researchers, students and teachers.

# Available Elements

**Available environments:**

- Single-Agent: Acrobot, Cartpole, DoubleIntegrator, InvertedPendulum, MountainCar, RasendeRoboter, Rubik's Cube
- Multi-Agent: SmartGrid, Delirium2*

**Available Learning Techniques:**

- Online model-free: Q-Learning($\lambda$), Sarsa($\lambda$)
- Offline model-free: RCAL **[Piot, 2014]**

**Available Action Selection Techniques:**

- $\epsilon$-Greedy
- Softmax

# Available Elements

**Available Q-Memory:**

- Static (preallocated) Lookup Table (for finite state-action space whose size is known)
- Dynamic Lookup Table (for finite state space whose size is unknown, but "manageable")
- Linear Function Approximator of the $Q$-function: $\tilde{Q}(s,a) = \theta^T \phi(s,a)$

**Feature extraction methods:**

- Cerebellar Model Articulation Controller (CMAC)
- Customized feature vectors

# Available Elements

**Controllers for the environments:**

- Delirium2 (*SWI-Prolog*)
- RasendeRoboter (*SWI-Prolog*)

# Demo...

# Conclusion

- *Ipseity* is daily used as a course support in AI and RL.
- It has already been used successfully to manage power flows in simulated microgrids using multi-agent reinforcement learning **[Lauri, 2013b]**.
- *Ipseity* can be freely downloaded from:

  `http://www.ipseity-project.com`

  under a GNU *GPLv3* open-source licence.
- It is still an experimental platform: it currently provides mostly online model-free RL algorithms.
- Though *Ipseity* is highly modular and broadly extensible.

# Future Works

Possible enhancements of *Ipseity* include:

1. Allowing the user to easier combine the modules of a cognitive system
2. Scripting the experiments
3. Implementing efficient Policy Search algorithms for solving continuous sequential decision problems
4. Providing tools for facilitating the comparison between different decision-making techniques
5. Establishing a database for storing and ease the sharing of experiment data
6. Providing a web-interface for accessing the results of experiments under progress

# References


- **[Sutton, 1998]** Sutton *et al.*, "Reinforcement Learning: an introduction", MIT Press, 1998.
- **[Busoniu, 2010]** Busoniu *et al.*, "Reinforcement Learning and Dynamic Programming using Function Approximators", CRC Press, 2010.
- **[Kovacs, 2011]** Kovacs *et al.*, "On the analysis and design of software for RL, with a survey of existing systems", Machine Learning, 2011.
- **[Lauri, 2013a]** Lauri *et al.*, "Ipseity - A Laboratory for Synthesizing and Validating Artificial Cognitive Systems in Multi-Agent Systems", ECML, 2013.
- **[Lauri, 2013b]** Lauri *et al.*, "Managing Power Flows in Microgrids using Multi-Agent Reinforcement Learning", Agent Technologies in Energy Systems (workshop of AAMAS), 2013.
- **[Piot, 2014]** Piot *et al.*, "Boosted and Reward-regularized Classification for Apprenticeship Learning", AAMAS, 2014.