TÉCNICO LISBOA

Autonomous Agents and Multiagent Systems

Projects must aim to develop a multi-agent system (MAS) with its mathematical formulation and decision-making process. Students are free to choose the programming language to develop their system.

1. PROJECTS

Projects can be in two formats: reimplement an existing work, or propose a new MAS problem.

Reimplement an existing work

More frequently than desired, research results cannot be reproduced by other authors for reasons such as a lack of description of all the needed details or unavailable data. In this project, a paper is selected, implemented, and a critical analysis of the results must be done. This includes the answer to the questions:

- Are the results the same?
- Does the paper include all the parameters needed to reproduce the results?
- Are the conclusions the same?

In the appendix, we suggest a list of papers; some are too long, some are too short. The concrete set of results that are going to be reproduced need to be selected in the beginning.

Evaluation of these projects is not based on the match between the original results and the obtained ones, but on the critical analysis of the differences. For an example of a replication study of an exploration method in reinforcement learning, see https://github.com/AugustinChrtn/Reproduction/blob/master/article.pdf

Grading will be based on the following criteria: description of the problem; identification of limitations of the original research; identification of limitations on the description of the results and simulations of the original paper; critical analysis of the differences between the obtained results and the original ones.

New MAS problem

In this option, you can suggest a new domain and solution. Note that, many times, the students focus too much on creating the domain, i.e. to make the game, with little time left to consider the solution and, most importantly, to analyze the results. Ideally, a simulator already existing should be used.

The projects need to clearly target at least one chapter of the course (ideally two), and the solution needs to compare at least 2 different algorithms (besides a greedy algorithm).

Evaluation is based on the complexity of the problem (not of the simulator), and the critical analysis between alternatives and the clear identification of limitations and possible solutions.

Grading will be based on the following criteria: description of the problem; complexity of the solutions; critical analysis of the results; critical comparison between the alternatives included advantages and disadvantages; identification of limitations of the solutions provided and possible solutions; the amount of topics from the course that is used in the project *(from at least 2 chapters that need to be explicitly described).*

Depending on the courses selected by the students, some other topics can also be considered for the projects, namely: multi-agent reinforcement learning, deep learning, and human-A.I. interactions.

2. Organization

2.1 Project Proposal and Presentation

A project proposal needs to be submitted (end of week 2), including the plan (limit of 2 pages of text, plus figures and references, using the *ACM Primary Article Template* (2-column article)), and a 15 minute presentation will be done the following week. Sections would normally include;

- An **introduction** that includes motivation, related work, problem definition and relevance, and objectives of the project;
- A description of the **approach** with the specification of the environment, (multi)agent system, and system architecture. This section should also include text to explain why the design choices are adequate to address the problem;
- A description of the **empirical evaluation**. Define a set of metrics that can validate the project's objectives.

For the projects of type "***Reimplement an existing work"*** *this description is simpler has the main focus is on what is going to be replicated.*

2.2 Final Delivery

The final delivery (see dates in fenix) with the following elements:

1. **Full source code** and **executable** with a README that explains how to run the system
2. **Final report** (limit of 4 pages of text, plus figures and references, using the *ACM Primary Article Template* (2-column article)).
3. A 1-to-2 minutes **video** demonstrating the agents or algorithms *in-action* (N.B.: a video of text outputs within a terminal (system console) is not accepted).

Be prepared to make a 15-minute presentation of the project (including a presentation of the video) followed by a 5-minute session of questions and answers.

3. Project Examples

A computational model for the Downs–Thomson paradox
(https://en.wikipedia.org/wiki/Downs%E2%80%93Thomson_paradox) or related problems

A computational model for the question "Why do competitors open their stores next to one another?" (https://www.youtube.com/watch?v=jILgxeNBK_8&ab_channel=TED-Ed)

A computational model for "The Better Boarding Method Airlines Won't Use"
(https://www.youtube.com/watch?v=oAHbLRjF0vo)

A computational model for "The Simple Solution to Traffic"
(https://www.youtube.com/watch?v=iHzzSao6ypE)

A computational model for all the problems related with elections, e.g. how the minority can win, …
(https://www.youtube.com/playlist?list=PLqs5ohhass_RN57KWlJKLOc5xdD9_ktRg)

4 Tobacco companies a real life example of Game Theory application
(https://www.youtube.com/watch?v=iguBgJP4Cwo&ab_channel=BusinessLearningPresentation)

Game theory challenge: Can you predict human behavior? - Lucas Husted
(https://www.youtube.com/watch?v=MknV3t5QbUc&ab_channel=TED-Ed)

Public Goods Games
(https://www.youtube.com/watch?v=kw9shrf-6U4&ab_channel=SystemsInnovation)

Papers to reproduce

Markov games as a framework for multi-agent reinforcement learning
(https://courses.cs.duke.edu/spring07/cps296.3/littman94markov.pdf)

Ad Hoc Autonomous Agent Teams: Collaboration without Pre-Coordination (change to a computational one)
(https://www.cs.utexas.edu/~pstone/Papers/bib2html/b2hd-AAAI10-adhoc.html)

Social diversity and promotion of cooperation in the spatial prisoner's dilemma game
(https://www.semanticscholar.org/reader/f64c6ef4459846b231055fa746b17aa6cab814f5)

Anytime algorithms for multiagent decision making using coordination graphs
(https://ieeexplore.ieee.org/abstract/document/1398426)

Nash Q-Learning for General-Sum Stochastic Games
(https://www.jmlr.org/papers/volume4/hu03a/hu03a.pdf)

Properties of winning Iterated Prisoner's Dilemma strategies
(https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1012644
https://www.youtube.com/watch?v=7wVkXS821ig)

Game Theory, Evolutionary Stable Strategies and the Evolution of Biological Interactions
(https://www.nature.com/scitable/knowledge/library/game-theory-evolutionary-stable-strategies-and-the-25953132/)

Testing Mixed-Strategy Equilibria When Players Are Heterogeneous: The Case of Penalty Kicks in Soccer
(https://pricetheory.uchicago.edu/levitt/Papers/ChiapporiGrosecloseLevitt2002.pdf)