Tetris Reinforcement Learning Project

Richard Combes

Université Paris Saclay, Centrale-Supélec / L2S, France

January 9, 2026

Project outline

- Main goal: learn how to play the game of Tetris using reinforcement learning in small teams
- 6 sessions of 1h30 (+ unlimited time at home) where you'll develop algorithms as teams
- Competition during the last session, where teams will compete for the highest score
- A Tetris environment is provided

Teams

Students will be split into 6 teams

- Team 1: Akiki, Arrayech, Barbaut, Barki
- Team 2: Benssy, Bestaoui, Cazeres, Dagron
- Team 3: Guido, Hanachowicz, Kalla, Leduc
- Team 4: Leteurtre, Malgrain, Minini, Mussotte
- Team 5: Peters, Peyrot, Pillard
- Team 6: Pouliquen Fardoun, Schwinn Fagundes, Zidna

Teams with fewer than 4 students will be given a slight bonus to compensate.

Evaluation

The outcome of this project constitutes the final grade for the course (no written exam).
You will be evaluated on

- Performance: average performance of your proposed policy over 100 random episodes
- Thought process: the deductive process that lead to the proposed solution, and interaction with instructors
- Originality: how innovative the strategy you proposed
- Information gathering: you are encouraged to browse publications to come up with solutions
- Team work: the communication and synergy between the team members

You are allowed to use AI tools and packages, however: all use should be documented, and you should be able to explain all of the corresponding code line-by-line, and how the packages subroutines work internally. Plagiarism is not allowed.

Organization

- First session is today
- During each session, instructors will guide each team discussing proposed strategies, potential problems etc
- Last session (Friday 30/01/2026 at 3:30 pm) is when the evaluation will take place
- For the evaluation each group must prepare a 10 minutes (sharp !) presentation explaining the main idea behind their strategy, and then we will perform a competition
- You must come to the evaluation making sure your code runs correctly

Getting started: installation

Install the required Python packages using pip

```
> pip install tetris-gymnasium cv2
```

Download the code provided to you by e-mail.

Getting started: playing a game interactively

Play a game of Tetris to test the environment and GUI

```
> python3 play_tetris.py
```

The key mappings to play the game are listed below

- "q" = "move_left"
- "d" = "move_right"
- "s" = "move_down"
- "j" = "rotate_counterclockwise"
- "k" = "rotate_clockwise"
- " " = "hard_drop"

Getting started: visualizing some default policies

Visualize episodes with provided default policies

```
> python3 view_episode_policy_down.py
> python3 view_episode_policy_random.py
> python3 view_episode_policy_greedy.py
```

Those policies work as follow

- Down: always perform a hard drop
- Random: select actions uniformly at random
- Greedy: attempt to place the current tetromino in order to minimize a linear combination of the "maximal height" of the pieces and the number of "holes"

Once you have run those, read the code line-by-line.

Getting started: evaluating the default policies

Run several independent episodes with the default policy

```
> python3 evaluate_policy_greedy.py
```

Note: this script does not display the GUI.

One of the goals of the project will be to beat this baseline. As an exercise you can perform the same for other "simple" policies.

Getting started: reading the documentation

Read carefully the whole documentation at
`https://max-we.github.io/Tetris-Gymnasium/`
In particular:

- Readme `https://github.com/Max-We/Tetris-Gymnasium/blob/main/README.md`
- Example codes `https://github.com/Max-We/Tetris-Gymnasium/tree/main/examples`
- States, Actions and Rewards: `https://max-we.github.io/Tetris-Gymnasium/environments/tetris/`
- Action and Rewards mapping `https://max-we.github.io/Tetris-Gymnasium/utilities/mappings/`

Once you have read all of those you can start coding your learning algorithms.