TÉCNICO LISBOA

Inteligência Artificial para Jogos

2018-2019

3.º IAJ Project –Decision Making and MCTS

The third project of the IAJ Course will consist on the work performed for Lab 5, 6 and 7, plus a small number of extra requirements that will be described next (we will call them Secret Levels). For this second project, you will need to write a report up to 5 pages, explaining and justifying the decisions made by your group in terms of implementation, describing performance tests performed and corresponding analysis. **Explain the rationale and describe the heuristics used**. Additionally, you should also include any analysis and discussions made for the requirements in Lab 5 6, 7.

You should submit a zip file with both Unity source code and with the report (a pdf/doc file) via Fenix, until 23:59 of November 26th. We will accept submissions after the deadline, but with a 0.5 value penalty for each hour after the deadline.

Secret Level 1 – Optimizing World State Representation

- The implementation used to represent a World Model is flexible but it is not the most efficient one. Create an alternative World Model implementation similar to F.E.A.R's system, where you have a fixed size array (or more than one fixed size array) with all data that represents the state of the world (instead of using a recursive model of dictionaries). You will need one array position for each property, and another for each consumable resource and enemy (to represent whether it was already picked up /destroyed or not). You can also remove the goals if you are not using them in MCTS). Alternatively you can use a class with Properties defined as part of the class, and make a copy of the instance whenever a new child is generated. Choose the easiest one for you to implement.
- **Test both versions and compare them in terms of efficiency.**

Secret Level 2 – Limited Playout MCTS

- Another form of making MCTS more efficient is to limit the depth of a playout (usually by a defined depth) and using an heuristic function to estimate the quality of a state (or likelihood of victory) as a reward when backpropagating the reward of a playout. Implement this variation. Describe and explain the rationale for the heuristics implement in the report.

Secret Level 3 – Comparison of MCTS variants

- Compare the variants of MCTS implemented: MCTS, MCTS+BiasedPlayout, MCTS+Limited+Biased Playout. Analyze factors such as processing time, number of iterations performed and quality of behaviour (which should be measured in terms of win rate%). Select one of the variants as your main algorithm and justify.

Secret Level 4 – Additional Optimization. What else?

- Select 1 relevant efficiency optimizations that can be made in your code and implement it. Justify why it was important. Analyze and discuss the efficiency gains obtained.