

## Workshop Lecture 4

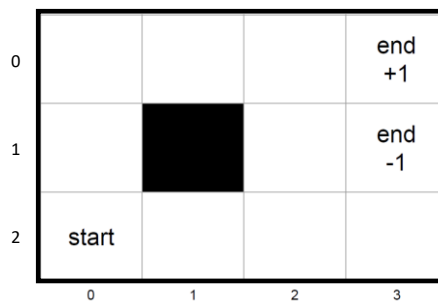
---

### Part A: Continuing with the simple maze

In the Blackboard folder for this week, you will see the python script “lecture4-simulation.py” under the workshop materials. Save locally, and run with (in terminal, cd to location of script):

```
python3 lecture4-simulation.py
```

This script provides the same simple maze as in weeks 2 and 3, class State (you are encouraged to tweak in whatever way you see fit to improve visualisation, usability, etc). There is one ‘agent’ implemented: IRLAgentPlus provides some changes to the IRLAgent class from week 3.



### Part B: Extending the IRL Agent

To enable the IRLAgentPlus, uncomment the relevant line in main, and run the simulation again. In contrast to providing feedback after every action as with IRLAgent, IRLAgentPlus allows the shaping of training in a different way – what is this? What are the implications of this difference?

Compare the performances of all three systems, and evaluate the differences in terms of speed of training, involvement of the human supervisor (you), etc. It may be worth adding graph plots of the relevant variables (e.g. using Matplotlib).

Consider the following potential extensions IRLAgentPlus:

- Can you improve the selection of a new action if the proposed one is rejected? (*consult the IRLAgentPlus.chooseAction method*)
- Instead of merely providing positive/negative reward for the present state, how could you explicitly guide which action the agent should take next? (*Note that this may require a change of learning algorithm*)
- How could you integrate explicit reward feedback into IRLAgentPlus?