Practical Project nº 1 – Rational Agents

1. Goals

Design, implement and analyze rational behavior for reactive agents.

2. The Environment

In an environment defined by a closed two-dimensional grid there are 2 types of waste - yellow cells and toxic waste (red cells) - and one type of food (green cells). The percentage of environment cells that define waste or food is configurable: between 0% and 15% for each type of waste and 5% - 20% for food. The energy obtained by eating the food is also configurable, ranging between 1 and 50.
The environment also contains a configurable number of deposits (blue cells) where some agents can deposit garbage. The amount of deposits varies between 1 and 10. Food and waste must reappear in the world in such a way that the configured levels are maintained throughout the simulation.

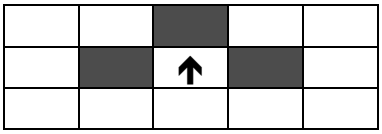
3. The Agents

There are two types of agents in the environment: the *Gluttons* and the *Cleaners* (configurable initial value). The main objective of these agents is to find food in order to maintain their energy levels, thus ensuring their survival. The *Cleaners* have a second objective - to clean the world of waste and allow all agents to survive. *Cleaners* gain energy when they eat and when they deposit waste in the dumps.

When created, all agents receive the same initial amount of energy (configurable value). In each iteration they lose a unit of energy. In addition, gluttons lose energy if they contact or sense waste (see further details). If the energy reaches values less than or equal to zero, the agent dies.

Characteristics of the Gluttons

- **Perceptions:** *Gluttons* can perceive the content of the cells that are immediately in front of them, on their left and on their right. The figure below shows which cells are perceived by the agent represented by the arrow (the agent is moving north).



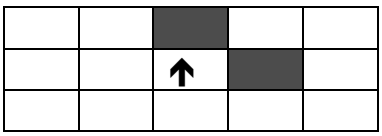
- **Actions**: *Gluttons* can move to the cell immediately in front of them, rotate 90º to the left or rotate 90º to the right. In each iteration, they can perform only one of these actions. For each perception, you must choose the action that optimizes the agent's survival. Each action takes a unit of energy from the *Gluttons*.

- **Characteristics:**

  - Are purely reactive agents
  - *Gluttons* automatically ingest the food found in the current cell. If this happens, the cell turns black and the agent's energy increases according to the configuration of the environment.
  - If *Gluttons* perceive a cell containing some type of waste, its energy should be reduced as follows:
    - Decrease 5% of your current energy if you perceive normal garbage
    - Decrease 10% of your current energy if you sense toxic waste
  - If *Gluttons* move to a cell containing some kind of waste, they die. The *Gluttons'* perceptions/actions should try to avoid these situations.

Characteristics of the Cleaners

- **Perceptions**: they can perceive the content of the cells that are immediately in front of them and to their right. The figure below shows which cells are perceived by the *Cleaner* represented by the arrow (the agent is moving north).



- **Actions**: *Cleaners* can move to the cell immediately in front of them, rotate 90° to the left or rotate 90° to the right. In each iteration, only one of these actions can be performed. For each perception, you must choose the action that optimizes the agent's survival. Each action takes one unit of energy from the *Cleaners*.

- **Characteristics:**
  - They are reactive agents with memory.
  - They can transport a limited amount of waste (configurable). Upon reaching this limit, the *Cleaners* have to find one of the deposits (blue cells) and dump the waste, before continuing the collection.

- Even with the deposit full, *Cleaners* can continue to eat food, receiving less energy from the food (see energy boost rule below).
- Waste collection is done automatically in the current cell (this cell turns black), unless the limit has already been reached.
- Memory: they have a single integer variable where the amount of waste they transport is registered. The update should be done automatically:
  - When collecting a normal waste it should be increased by one unit;
  - When collecting a toxic waste it must be increased by two units;
  - When finding the deposit, the variable resets to zero, the accumulated in the deposit is updated and the Cleaner energy level is increased by 10*number of cells deposited.
- *Cleaners* automatically ingest the food found in the current cell. If this happens, the cell turns black and the agent's energy increases according to the following rule:
  - If the number of wastes you transport is less than half the limit, the energy increase corresponds to the value indicated in the environment setting.
  - Otherwise, the energy increase corresponds to half the value indicated in the setting.
  -

4. Tasks to perform

The work to be carried out is divided into the implementation component and the experimentation/result analysis component.

4.1 Implementation

The implementation is divided into two phases:

a) Base Model – replicate the entire description made in the previous sections. Create all components in the interface that allow you to configure characteristics of the environment and agents, as described in 2. and 3.
In cases of omissions, or where some ambiguity may arise, students should opt for a sensible solution, justifying their choice in the report.

b) Improved Model – at this stage, students are free to propose and implement changes to the base model that aim to improve the agents' performance. The originality of the proposals will be valued and the choice of techniques and strategies must be based on the report. Some suggestions students can explore:
   a. change perceptions: number of perceptions, order in which cells are analyzed, ...
   b. change memory: What if the *Cleaners* had a larger, smaller, or other type of memory? Foodies with memory?
   c. change the *Cleaners'* properties: what if there were cleaners specialized in a single type of waste?

d. create new types of agents
e. create environment cells with other functions and characteristics
f. ...

4.2 Experimental Study

This task consists of carrying out experiments to test the implementations made.

Performance metrics must be defined: number of agents after X iterations, % of experiences with total extinction of agents, among others.

4.2.1 Experimental study of the Base Model

An experimental plan for the base model must be elaborated, analyzing the parameters that can affect the agent's performance.

Important: Carrying out simulation experiments presupposes formulating hypotheses and carrying out tests to confirm their validity.

Example of a hypothesis

- *Hypothesis*: The amount of food influences the agents' survival.
- *Confirmation of the hypothesis?* Vary the parameter "amount of food" using different values and check what happens to the agents' survival. Was the hypothesis confirmed or not? Justify the results obtained.

Similar to the example given, students should formulate the set of relevant hypotheses. Some examples of changes you can study in the base model:

a) survival of gluttons without cleaners
b) survival of cleaners without gluttons
c) influence of the number of agents of each type
d) influence of the number of deposits
e) influence of the amount of food and garbage in the environment
f) others...

NOTE: Averages of at least 10 repetitions of each experiment must be presented

4.2.2 Analyze the improved model

An experimental plan for the improved model should be drawn up. The idea is to verify whether, for equivalent configurations between the base model and the improved model, the proposed strategies improved the agents' performance.

For each proposed strategy, you should compare the metrics with the equivalent base model and check whether the agents' performance has changed. Justify the results.

NOTE: Averages of at least 10 repetitions of each experiment must be presented.

5. Evaluation criteria

- Base model implementation (30%)
- Improved model implementation, correctness, originality (30%)
- Quality of the experimental study - analysis of at least 3 hypotheses for each model (30%)
- Quality of documentation, presentation and defense (10%);

6. Report

In the report to be delivered, all proposals and changes implemented and the configurations tested must be described and justified; describe for each task the different strategies implemented;

The questions to which the carrying out of simulation experiments could lead to finding answers must be clearly identified. The report must also contain a careful analysis, which helps to clarify the results obtained;

Tip: you can attach global result files from all the experiments performed to the job (Excel file, for example). In the report, present the most relevant results, presenting the hypotheses formulated and the results obtained.

The report should be about 10 pages long.

7. Rules

The work must be carried out in groups of two students. In exceptional cases it can be done individually.

Presentation: The work must be presented in the practical class(s) scheduled for the presentations. Students must confirm with the teacher of their practical class the day and the form of presentation and defense of this work (in person, remotely, other).
Each group has 10 minutes to present and justify the main options taken. In this presentation, they can use the materials they consider most suitable (PowerPoint, NetLogo models, drawings on the board, among others).

Material to be delivered:
NetLogo file(s), report in pdf format, presentation slides (if any), EXCEL file with results.

Delivery method:
The work must be delivered via Moodle until 23:00 on November 12, 2021. The presentations will be in the practical classes from the 15th to the 26th of November, according to prior registration to be carried out in Moodle.

All files must be compressed in a .ZIP file whose name must identify the name and student number of the group members, for example:
AnaSilva_2017011111_JoaoMelo2018022222.ZIP

Quotation: 2 values (in 20)