ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

EPFL

CS-430

Intelligent Agents

Report - A Rabbits Grass Simulation

Authors: (Group 34)
Stéphane Cayssials
Khau Anh Nghia

Student Numbers:
272048
223613

October 4, 2016

1 Implementation

1.1 Assumptions

- We assume that the grass amount bounded in each cell is 1000 (MAX_ENERGY_OF_GRASS).
- We assume that for each move, the rabbit loses one unit of energy.
- We assume that when the simulation starts, the total amount of grass is 30 (INIT_NUM_GRASS) and each agent has 10 units of energy (INIT_ENERGY_AGENT) when he is created.
- We assume that when a rabbit reproduces, he shares a half of his energy to his child.
- We assume that in this simulation, the color of a rabbit is white and the one of grass is green.

1.2 Implementation Remarks

There are two important methods:

1. public void move() in RabbitsGrassSimulationAgent class. This method decides how a rabbit can move in each step of the simulation. First we find a list of all available neighbours (the cell that are not occupied) and then choose the direction randomly on the list. After that we move the rabbit into new cell and he can earn some energy if he find some grass inside. Rabbit will not move if all four cell (NEWS) are occupied. If the rabbit move to somewhere he loses one unit of energy, otherwise he dont lose anything.
2. public void execute() in the inner class RabbitsGrassSimulationMove that are defined in RabbitsGrassSimulationModel class. This method defines what will the simulation change(do) in each step. Since we have these requirements stated in the assignment, first we check if there is any rabbit on the space, if not we stop the simulation. Then we spread some grass in the space with an amount of GRASSGROWRATE and for each rabbit, we will check if he has enough energy (greater than 0) to live, if not he has to die. Otherwise (rabbit has enough energy), we call method move() for this rabbit for moving and check again if he has enough energy to reproduce (because maybe after moving he win some energy in new cell). If this is the case, we call methode reproduce() to create a child.

2 Results

2.1 Experiment 1

2.1.1 Setting

NumberAgent: 35, BirthThreshold: 15, GrassGrowRate: 30, WorldXSize = WorldY-Size = 20.

2.1.2 Observations

The simulation is stable. When the value of NumberAgent is big, they will eat most of grass then the value of total amout of Grass decrease. So now we have many agent and an small amount of grass leads some agents will die. In the same time, grass will be planted with GrassGrowRate on each simulation step and at a given moment, we will have a small the number of agent and a big number of total amount of grass and so on...

2.2 Experiment 2: much NumberAgent

2.2.1 Setting

NumberAgent: 70, BirthThreshold: 15, GrassGrowRate: 30, WorldXSize = WorldY-Size = 20.

2.2.2 Observations

There are so many agent in the space, at the beginning the number of agent decreases very very fast (there are only 3 agents in the space at one given moment) because there is not enough grass for a big number of agent. But after that, when the agent reproduces the situation is stable like the first Experiment. It is possible that at the beginning, all the agents will die and the simulation will stop !!!

2.3 Experiment 3: much NumberAgent + small GrassGrowRate

2.3.1 Setting

NumberAgent: 70, BirthThreshold: 15, GrassGrowRate: 15, WorldXSize = WorldY-Size = 20.

2.3.2 Observations

Simulation stop at the beginning due to there is no agent. The GrassGrowRate is too small leads the number of grass grows slowly and there is not enough grass for all agent. So the number of agent die increase exponential.

2.4 Experiment 4: Increase size of space

2.4.1 Setting

NumberAgent: 35, BirthThreshold: 15, GrassGrowRate: 30, WorldXSize = WorldY-Size = 40.

2.4.2 Observations

This situation is also stopped fastly due to there is no agent. There are many "free" spaces without grass where agent move in and get zero energy (at the same time his energy decrements by one). So the number of agent decrease fastly after a few of step.