Modeling and simulation of complex systems - Project Report

Ta Quang Hieu - M22ICT002

February 2, 2024

1 Subject

Herd behavior and pathway tracks in sheep offer an intriguing lens through which I can explore the complexities of emergent phenomena from individual behaviors. In this report, I delve into the intricacies of herd behavior among sheep and the formation of pathway tracks, seeking to understand the interplay between individual actions and collective dynamics.

Hypothesis:

I hypothesize that the emergence of herd behavior and pathway tracks in sheep is a result of a combination of evolutionary instincts and adaptive responses to environmental challenges. Specifically, I posit that the evolutionary drive to minimize predator risks and optimize resource utilization has shaped individual sheep behaviors, leading to the observed collective phenomena of herding and the formation of distinct pathways.

Report Overview:

The report aims to address the fundamental question: "How do herd behavior and pathway tracks emerge from individual sheep behaviors?" To unravel this question, I will construct the report as follows:

- 1. Subject: how do I see and make the hypothesis of the problem
- 2. Model: what kind of model I choose?
- 3. Documentation of the base model
- 4. Extensions - what did I add to the model
- 5. General conclusion

Here is the link to the Github repository of the project: `https://github.com/Ctn3m0/Gama-project`

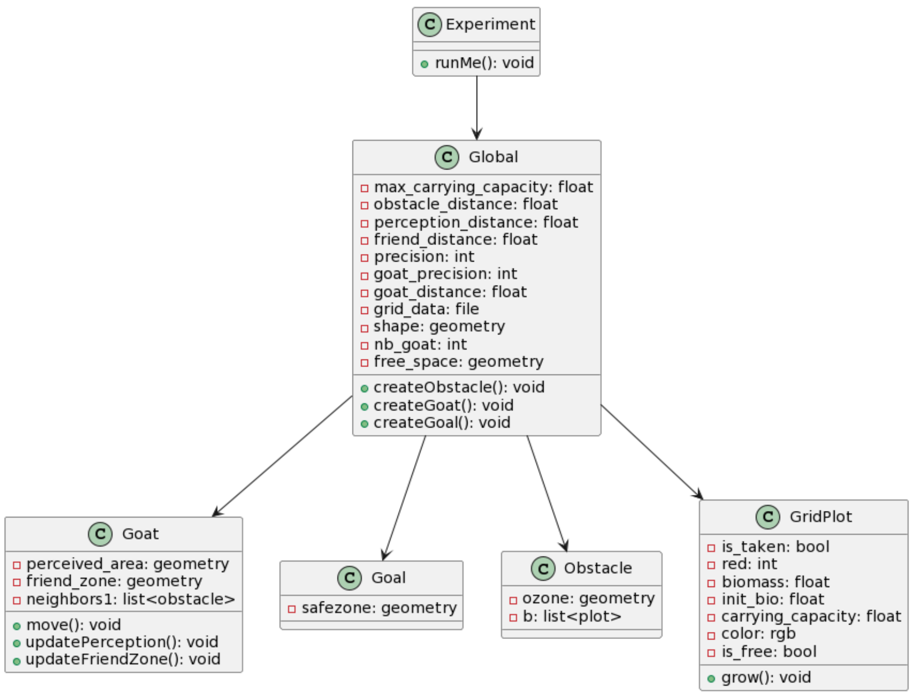

Figure 1: The UML model for the base model

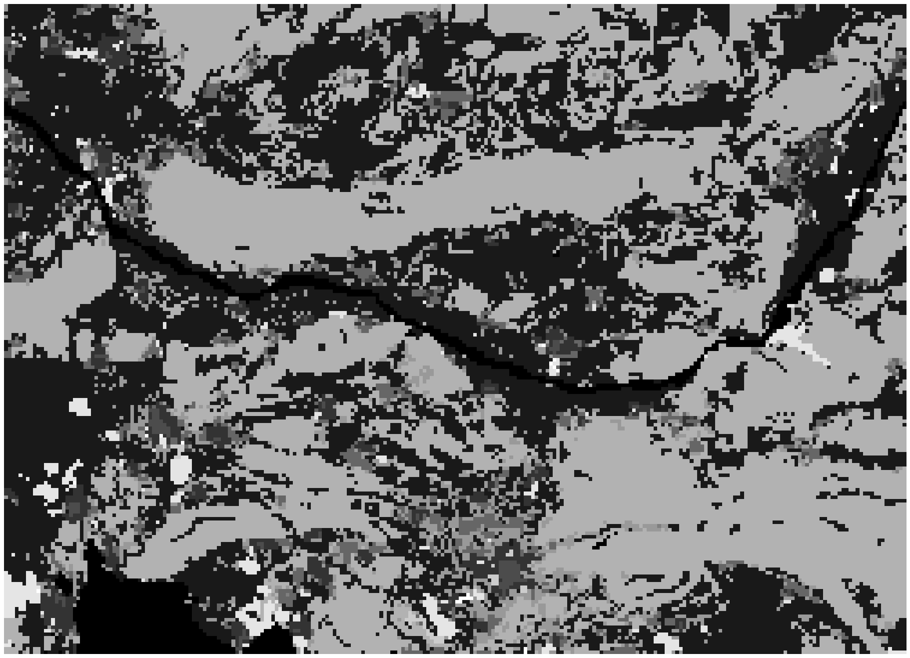

Figure 2: The world map

2 Model

Figure 1 shows the UML diagram of the model that I have chosen.

Regarding this project, I have decided to create three species: Goat, Goal, and Obstacle. For the map, I will use the old map from the project of **Predator and Prey** as in Figure 2

2.1 Agent Choice:

Goat as an Agent: The decision to represent individual sheep as agents in the model stems from the need to capture their autonomous and interactive nature. Each sheep agent possesses distinct attributes, such as its perceived area, friend zone, and responses to obstacles, contributing to the overall complexity of the model.

For the goat agent, I have several properties like this:

- perceived_area: show the area that the goat can see and react to the obstacle, this is used as a mechanism for the goat to dodge the obstacle along the way
- friend_zone: this attribute is somewhat like the perceived_area, but it is bigger and the purpose of it is to make the goat able to sense the neighbor

friend goat

- neighbors1: This is a list of obstacles around the goat to trigger the algorithm to dodge the surrounding obstacle

Goal as an Agent: The decision to model the goal as an agent is rooted in the need to simulate the impact of a specific destination or objective on sheep behavior. The goal agent represents a target location within the landscape, influencing the movement and navigation decisions of individual sheep agents.

For the goal agent, I have a property like this:

- safezone: this is the zone that I have set around the goal, in this zone, the goat will no longer follow their friend but always head to the goal

Obstacle as an Agent: Modeling obstacles as agents extends the scope of interactions within the Agent-Based Modeling. Obstacle agents have properties and behaviors influencing the movement of nearby sheep agents. By representing obstacles as agents, the model can capture the nuanced responses of individual sheep to varying obstacle spatial distributions.

For the obstacle agent, I have several properties like this:

- ozone: this is the area where the obstacle takes, I used it to make it available for the goat to be able to spawn without being overlapped by the obstacle

- b: this is a list of plots which are overlapped by the obstacle

3 Documentation on base model

A GAMA model is composed of three types of sections:

- global: this section, which is unique, defines the "world" agent, a special agent of a GAMA model. It represents all that is global to the model: dynamics, variables, and actions. In addition, it allows to initialization of the simulation (init block).

- species and grid: these sections define the species of agents composing the model. Grid is defined in the project as shown in the UML diagram, the grid plot.

- experiment: these sections define the execution context of the simulations. In particular, it defines the input (parameters) and output (displays, files...) of a model.

In this part, I will provide brief user documentation of the base model through each section, i will first start with the Global section:

3.1 Global section

The global section represents a specific agent, called **world**. Defining this agent follows the same principle as any agent and is, thus, defined after a species. The world agent represents everything that is global to the model: dynamics, variables... It allows to initialization simulations (init block): the world is always created and initialized first when a simulation is launched (before any other agents). The geometry (shape) of the world agent is the grid data taken from the file **hab10.asc** in the includes folder.

Global attributes

In the current model, I will only have a certain number of goats thus I need to hold this number in a global or world's variable of type integer (int), I also define some other global variables as follows:

- max_carrying_capacity: the maximum number for the biomass which I will use later in the second extension.
- obstacle_distance: the distance that the goat will start to know that there are obstacles around.
- perception_distance: the distance that the species is able to "see" and dodge the obstacle, this is set with the facet **parameter: true** to be able to be changed during the experiment.
- friend_distance: the distance that the goat will start to know that there are friends around.
- precision: used for the masked_by operator (default value: 120): the higher the most accurate the perception will be, but it will require more computation.
- free_space: space where the agent can move.
- the geometry **shape** is set to be enveloped by the grid_data taken from the file mentioned above

For the initiation step in the global **init**, I have done several things as follows:

- creates a number of obstacles and then masks the plot of the world that is being overlapped by it to be come taken.
- creates the goat with the number defined above in the spot that is still free (for this project, I have faced the problem when the number of goats can only be **around 45** if you increase the number of goats more than that, the model will become really slow).
- create the goal where all the goats will come to in a free plot.

3.2 Goat Agent

The goat agent will be able to move since I have set the skills: [moving] to it. For the attribute of this agent, there are several things to note:

- perceived_area: this is the attribute that sets the area that the goat is able to see and dodge the obstacle.
- friend_zone: this is the zone where the goat is able to sense its friendly goat around to follow them to the goal.
- And finally the list of obstacles **neighbors1** shows the surrounding obstacle at the distance that has been set in the global.

For the aspect (which shows the visual of this species) of this species, I have made the goat has a default shape of a circle with the size of 150 and white color, then I made 2 other aspects to show the vision of this goat to sense the obstacles and friends mentioned above.

Regarding the reflex, which shows how well the goat acts in each cycle, I have defined these reflexes:

- update_perception: the perception of the goat will be updated after each cycle because it might be masked by the obstacle with the precision defined in the global. The perception is shown as a cone with the area intersection by the circle of the size perception_distance defined in the global and the degree of 60.
- update_friendzone: this is somewhat like the update_perception above but the degree is now 100 and the area is the intersection by the circle of the size friend_distance in the global. This zone is made by another goat that is inside it and the precision is defined in the global.
- Finally is the reflex move, which defines how the goat will react in each cycle. In this reflex, each time the goat moves, it will be left behind a trace in red for the plot it had passed by. If there is no obstacle in the neighbors1 list, the goat will go to the goal at a random speed. And if now, there are obstacles inside the list, the goat will check, if there is a friend around, follow him and increase the speed; if the is an obstacle, dodge it (by going around the perceived_area)

3.3 Goal Agent

For the goal agent, I only set up a few simple things: The safezone where the goat will stop following each other and then create two aspects to show the shape of the goal which is a square of size 1000 and the safe zone is also a square with the size 15000. Both of them have the color blue.

3.4 Obstacle Agent

For the obstacle agent, I also set up a few things: the ozone shows the area that it is occupying and list b shows the plot being overlapped by the obstacle to be set as taken. The shape of the obstacle is shown in the init step in the global, for the base model it is just some rectangle with a fixed size.

3.5 Grid Plot

This plot will define the area of the world, in the base model, there is nothing much to care about the thing inside it. I will explain it later in extension 2 where I introduce the biomass to the map. By default now, it is just a black map.

3.6 Experiment

An experiment block defines how a model can be simulated (executed). Several experiments can be defined for a given model. For this project I named the experiment as **runme**, inside this experiment is the mechanism of how well the experiment is being displayed (inside the output): the grid plot will be displayed first and then the goat, goal, and obstacle, there is a thing that I want to note:

- **species goat aspect: perception transparency: 1;** If you want to see the perception area of the goats you can change the transparency to 0.5 and also for other zones like friend, safezone..

Let's have a glance at the map of the base model in Figure 3.

4 Extensions

4.1 Extension 1

In this extension, I have extended the agent-based model to incorporate variations in the obstacle types, such as different sizes, and shapes to assess their distinct impacts on pathway emergence and utilization by the herding sheep. For this extension, I made the shape and size of the obstacle now become random by using this script **shape ¡- flip(0.6) ? rectangle(1000+rnd(2000), 1000+rnd(2000)) : circle(2000 + rnd(1000));**. Making the obstacle more complex like that created some challenges for the goats, during the simulation experiments in this extension, some goats are now stuck with some complex architecture of the obstacle. You can observe this in Figure 4 where there are 2 goats are stucking with the obstacle and cannot find the way out.

4.2 Extension 2

In this extension, I have introduced environmental factors, there will be the biomass attracting the goat into the model to examine how it interacts with

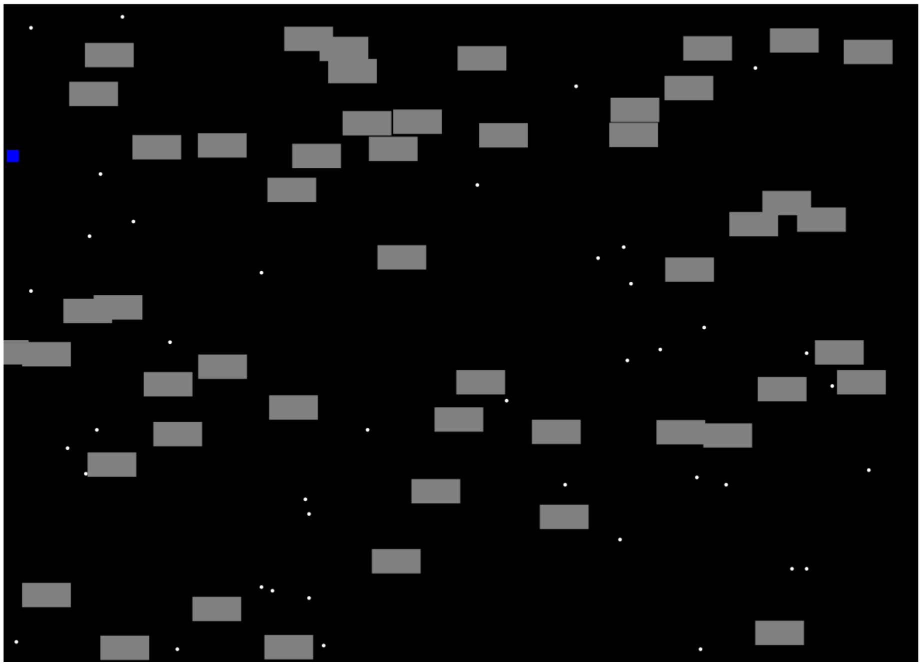

Figure 3: The base model map

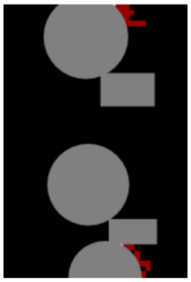

Figure 4: The stucking goat

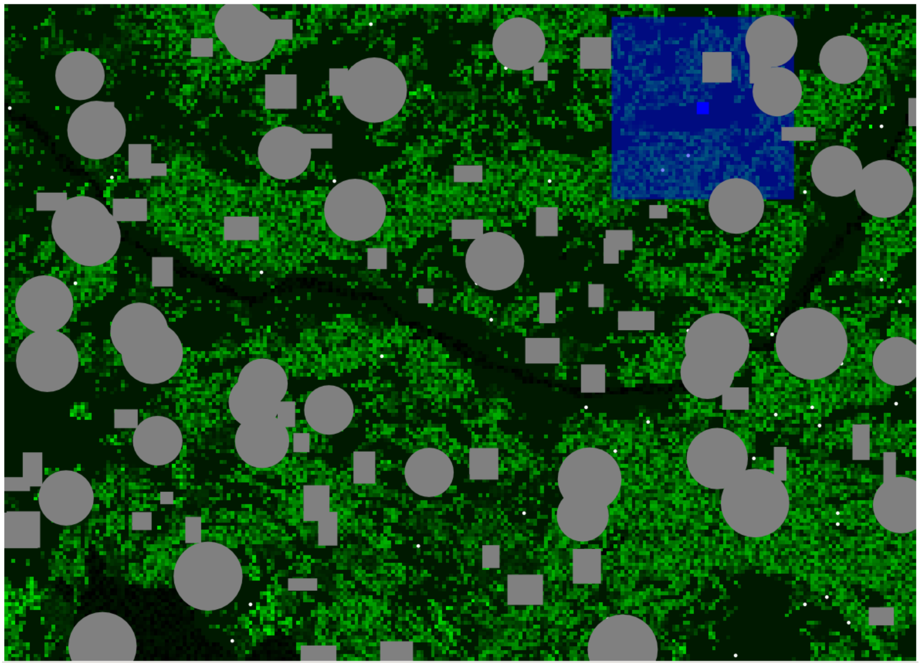

Figure 5: The biomass map of extension 2

obstacle spatial distribution in shaping the emergence of pathways in the landscape. The world map now becomes like in the Figure 5 In this extension, now I will explain the properties inside the grid plot that I had stated before, for the grid plot, I have introduced some attributes like follows:

- is_taken: This is a boolean attribute to show whether this plot has been occupied by the obstacle.
- The integer **red** is initially set to be 0 when none of the goats has passed it.
- biomass: this is an attribute to show the number of "food" inside the plot and if it is bigger than 5, the goat will be attracted.
- carrying_capacity, this had been set and explained in the global.

The default aspect of all plots is that their shape is all square with a size of 100 and has a black border, the color is randomly set with regard to the biomass and max_carrying_capacity. This grid plot has a reflex call grow that will start to have an effect when carrying_capacity is not 0, the biomass will continue to grow with regard to the growth_rate to compensate for the biomass that had been eaten by the goat.

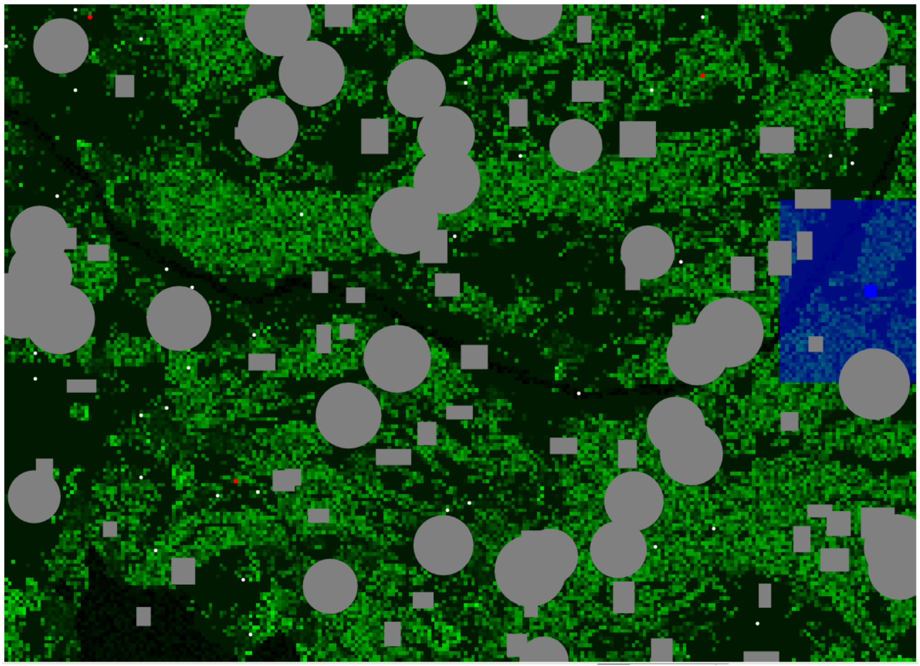

Figure 6: Introduction new species

Having the biomass around the map, the goat now cannot focus to go to the goal, some time because of this food, the goat becomes stuck and wanders around which leads to the running time being much longer.

4.3 Extension 3

In this extension, I have introduced a new species called dog which will scare goat and examine how this new species interacts with spatial distribution. Much of the dog species properties are just like what I have for the goat species but for the move reflex. The dog is marked by a red circle with a size of 200 as you can see in Figure 6

The species dog doesn't have a specific goal; it just goes randomly around the space. Inside the reflex move of the dog, I set a list a, which is a list of goats that is inside the perceived_area of the dog (note that, I have set the perception_distance of the dog 3 times larger than the goat). Once the goat knows that it has been in the eyesight of the dog, it will try to run to the goal at a speed much faster than the original one, leading to some disappearance of the red trace as you can see in Figure 7

Now, with the appearance of the dog (I have only created 3 dogs into the map), once caught in the sigh of the dog, the goat will no longer be attracted by the food anymore, but it will try to reach the goal as quick as possible until

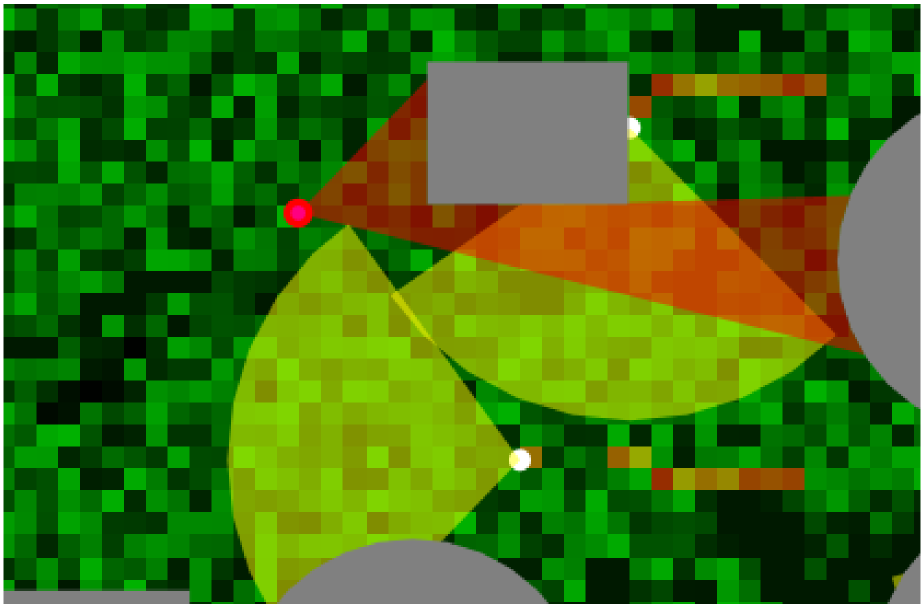

Figure 7: The goat runs fast so that the trace disappears

get out of the sigh of the dog

5 General conclusion

Throughout this project, I have been able to create an agent-based modeling approach that provides for exploring the emergence of herd behavior and pathway tracks in sheep. The model presented in this report leverages individual sheep agents, obstacles, and environmental factors to simulate complex interactions leading to collective phenomena. The model's extension to include variations in obstacle types and sizes enhances its realism, leading to scenarios where sheep may become stuck or face challenges in navigating complex obstacle architectures. The introduction of environmental factors, such as biomass, further extends the model's capabilities by simulating real-world influences on sheep movement. In the last extension, I introduce a new species, the dog, which adds an additional layer of complexity to the model. The dog species, with its ability to scare and influence the behavior of the sheep, demonstrates the flexibility of the agent-based approach in simulating multi-species interactions. The report provides documentation on the base model, including explanations of species attributes, and behaviors, and the global section, serves as a comprehensive guide for users to understand the fundamental components of the model. Each species's role, such as the goat, goal, and obstacle, is clearly defined, highlighting their significance in shaping emergent behaviors.

For further implementation, the project still needs to be optimized since it runs really slow, and also the number of goats I can have now is a maximum of 45, if it is bigger than that the program will freeze. I need to find a way to determine the parameters that enable a good compromise between execution speed and performance, after that I want to create some real-life scenarios but I am still struggling with it in the extension-4 file.