Report Project Modeling and simulation of complex systems

Heiko VIEL

February 1, 2024

1 Introduction

This report is presenting the project for the course Modeling and simulation of complex systems. The subject from my project was Driftwood : Self-regulating access to natural resources. On the coast of a distant country, people compete for the gathering of driftwood brought to the shore by storms. Whoever is first onto a stretch of the shore after high tide is allowed to take whatever he wishes up to its carrying capacity and to gather it into a pile above the high-tide line. To indicate ownership, piles are marked by placing two stones on their top. The wood it contains is then regarded as the property of a driftwood collector. Only wood pile owners always respect pile ownership. Collectors having not yet established a pile can head towards wood piles and "steal" the wood it contains, but only when no pile owner is observing them.

The goal of this project is to create an Agent-based Model with Gama that will describe the situation just above. With the different experience I will do, we will be able to know if it's possible, without any external enforcement, to reach a stabilized situation.

My hypotheses are that it's not possible to reach a stabilized situation when there is no external enforcement, the collectors are not able to protect there pile if they are currently collecting in there zone. Adding some monitoring on the model may help to regularize the system and prevent the other collectors to steal.

In this report, there are 4 different models, the first one is the base model that will represent the collectors that collect and create their own pile. The 3 others are 3 extensions based on the first model. The second will implement the ability to steal wood that are contained in piles from other collectors. The second extension is the implementation of external authorities that will monitor the piles already created. The last one is the integration of collectors groups to see if there is an impact if some collectors help other.

2 Base Model

2.1 UML Diagram

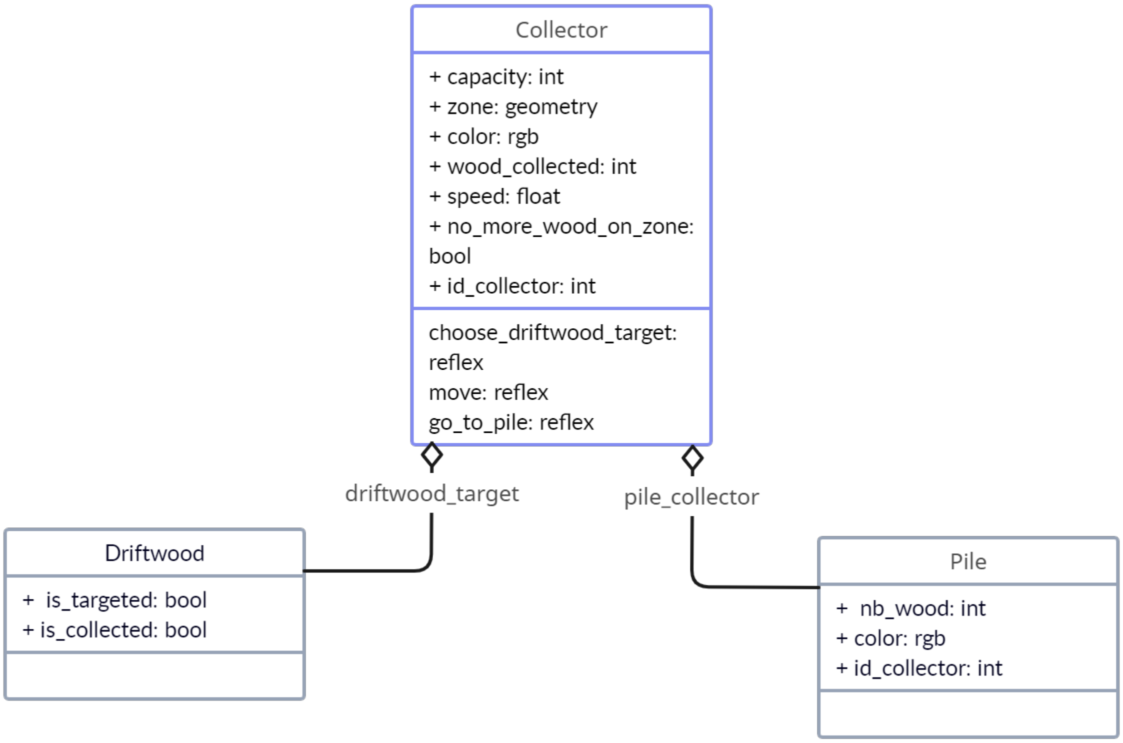

Figure 1: UML Diagram

You can see above the UML Diagram for the basic model, this diagram is the root and we will use it to implement the extensions by adding attributes into species or create new species.

2.2 User documentation

I will now explain how the basic model runs. When you initialize the experiment, you can initialize several parameters: the number of collectors, the shore distance, the number of driftwood on the shore and the capacity max that can carry each collector. Before you run the start button, the model will create the collectors and put them randomly on the map, same for the driftwood they will be put randomly on the shore. The shore is also divided into equal zones depending on the number of collectors. So each collector will have one zone where he will be able to collect the driftwood. At the moment the expriment starts, the collectors will go into their zone to start collecting, when they will reach their capacity they will go back to create a pile, then they will use this pile to put the wood they collect after. They will do these actions until the zone has no more wood on it. At the end they will go back on their pile, and the console will print the number of wood collected for each collector.

In this case, the system reaches a stabilized situation because it relies on the random generation of pieces of wood in the zones.

3 Extension 1 : ability to steal

3.1 UML Diagram

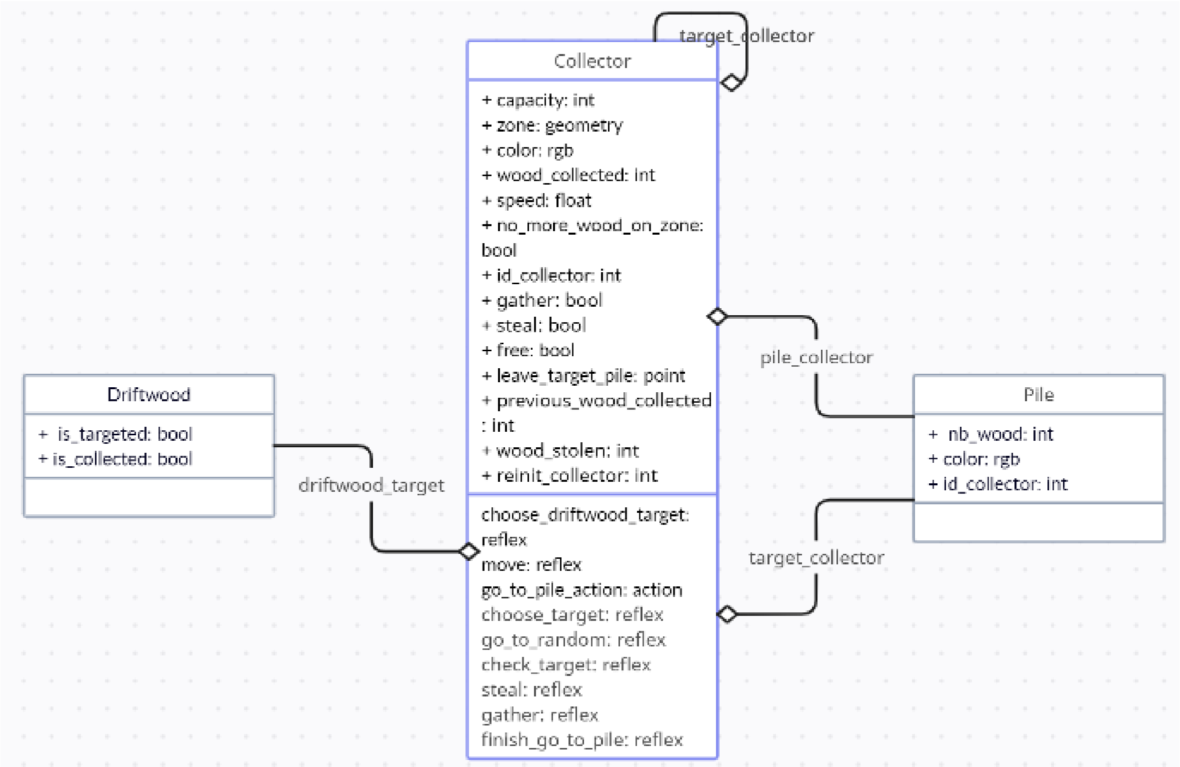

Figure 2: UML Diagram

As you can see, we add some attribut and reflex into colletor species to implement the ability to steal.

3.2 User Documentation

Into the experiment, there are new parameters that you can modify with the other from the previous experiment : the number of zones and the distance to do monitoring. When you start the experiment, depending on the number of zones register, some collectors will have one and other will have no zone (this is to represent that some collectors came before others and choose their own zone where only the owner of the zone can collect, so the other collectors without a zone are only stealers). When you start the experiment, collectors having a zone will choose between wait and steal other or go collect woods into their zone. Others will automatically wait to steal pile from owners. When a pile is created, the owner is unable to steal to respect the pile ownership. But for example if the owner is leaving it (distance of monitoring), collectors without pile can steal the wood on his pile, if there is no wood anymore into the pile, the pile is destroy. So the owner can steal again because he has no pile anymore. Someone that has no zone can only steal and he will never be stolen because he will steal until he reach his capacity and then he will create his pile and stay on it (because he can't do any additional action). There are several condition inside the model to not stop the model running or to prevent some non logical action. For example, when all the collectors want to steal at the same time, no pile can be created so the model is blocked, to prevent this I put a condition where if the collectors having a zone want to steal, the last one will always continue to collect, so he will create a pile after and unlock everyone. Also I put an additional condition that unlock a collector that want to steal if during 200 cycle he wasn't able to find a target.

When there is no more wood on the shore, everyone will go to their pile even if the have no wood, it was easier to do this method to count every wood collected for each collectors. At the end, the console will print the number of wood collected, if the collector has a zone, and the number of wood he stole during the experiment.

In this experiment, most of the time collectors that have no zone are able to steal easily, because they will only do this. Other with a zone are less impacted because they can also collect, so they will steal less time.

4 Extension 2 : External authorities

4.1 UML Diagram

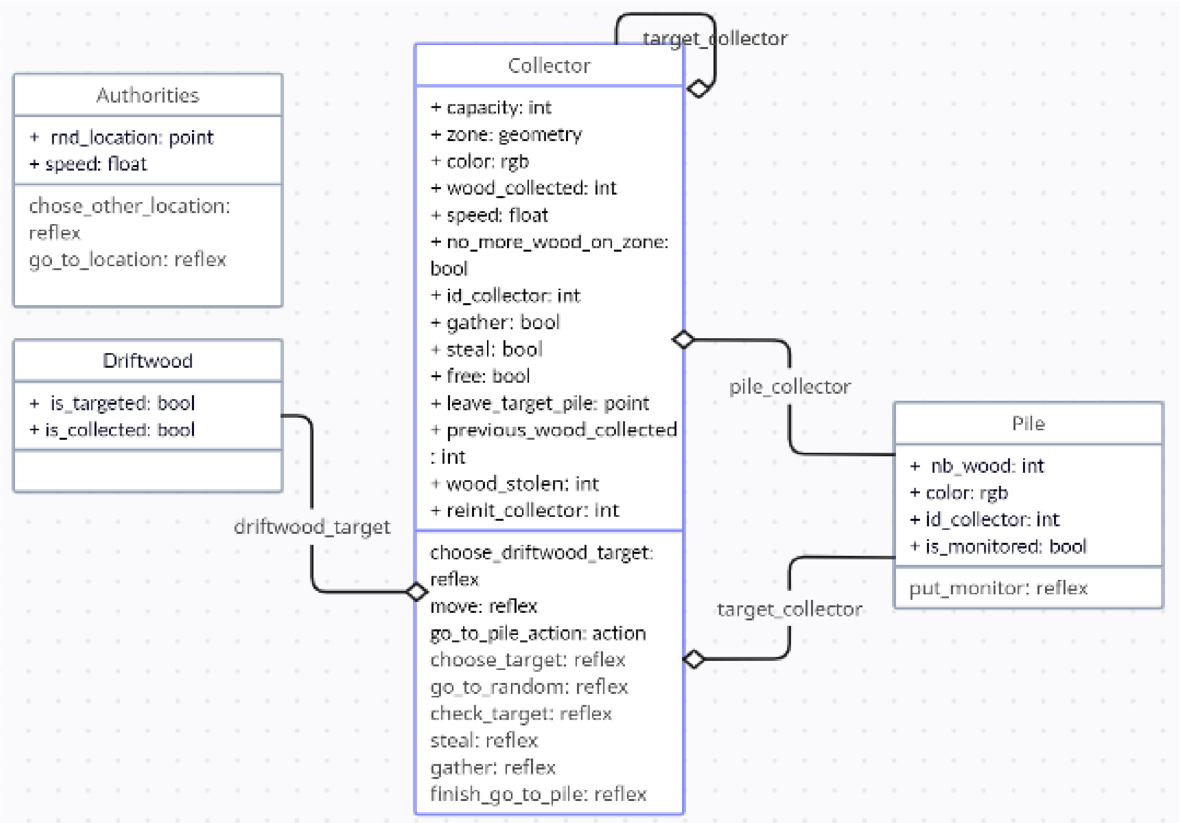

Figure 3: UML Diagram

For the second extension, I had some authorities to the model, so there is an additional reflex inside the pile to check if there is an authorities next to it or not.

4.2 User documentation

In this experiment, there is a new parameter: the number of authorities that will be put inside the model. The goal of the authorities will be to monitor the pile that have been already created by walking randomly on the map. If they are at a distance less than the distance to monitor, then the pile will be monitor and no one can steal it while it is monitored. Otherwise the system will work the same as the first extension.

Adding authorities do not really stop steals but it reduces it a little bit, when each collectors has a zone, they will do less steal because we reduce the number of piles available to steal, so they will change mind and keep collecting inside their zone. If there are collectors without a collecting zone on the shore, they will only focus on steal other so most of the time at the end, there will be pile available for them because they will wait until one is free, they can't change mind about steal or collecting.

5 Extension 3 : Group implementation

5.1 UML Diagram

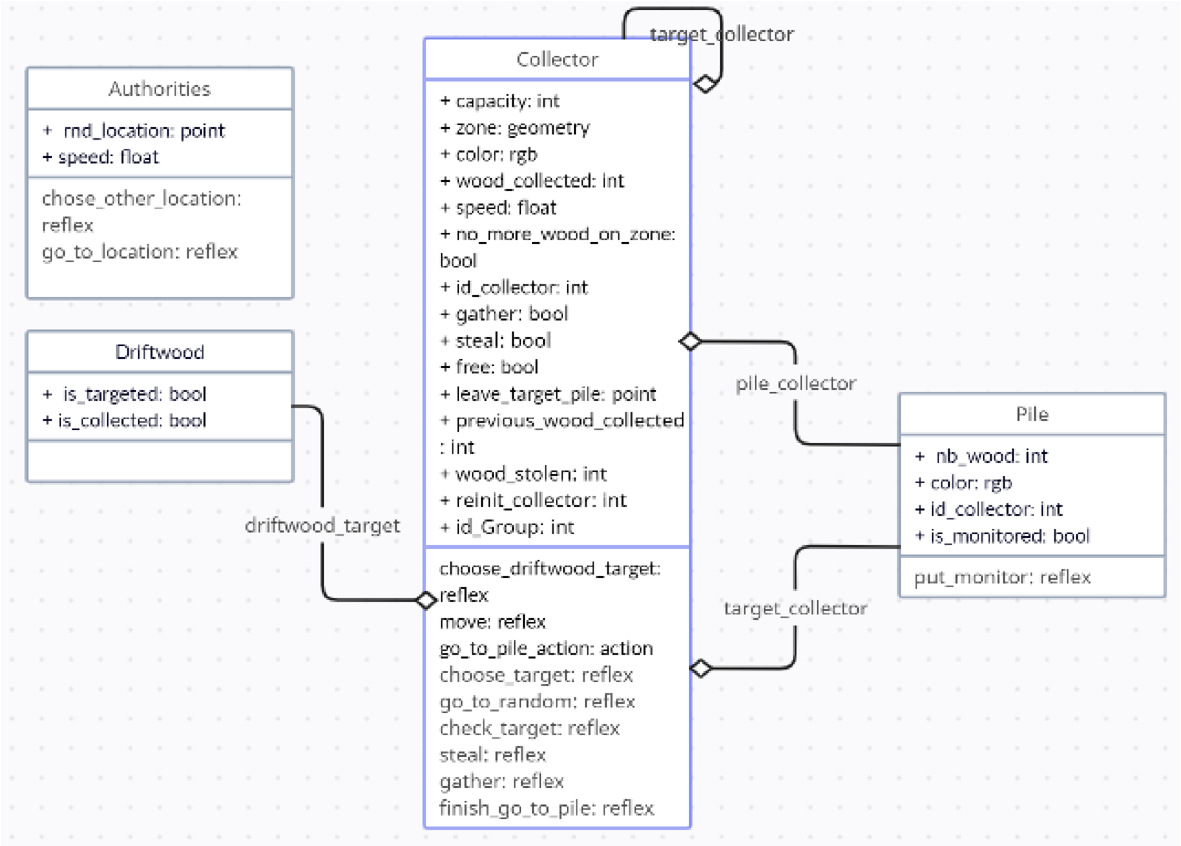

Figure 4: UML Diagram

In this model, I added one attribute inside Collector species : the id of the collector's group.

5.2 User Documentation

There is one additional parameter you can modify : the size of groups. At the beginning, each collector will be integrated inside. The groups will allow them to help each other. For example, collectors will not be able to steal wood from other collectors in the same group. Also if one collector is close to a pile that belong to a collector from the same group, the pile will be monitored.

Groups have an impact when the total collectors are divided into big groups (for example 20 collectors and size group is 10, so there will be only 2 groups) because more we have groups inside the model more it will reproduice the same thing when there is no group. If the group size is big you can add additional monitoring so it helps to prevent woods to be stolen.

6 Conclusion

To conclude, we generated a model to simulate the collection of driftwood. We have carried out several experiments with the aim of obtaining a stable system that limits wood theft. First of all, we tried to reproduce the system presented to try to have an example that comes as close as possible to reality. Then we implemented different options to see the evolution of the system.

This model does not fully reflect reality but it gives us an idea of potential actions that can be implemented to limit wood theft. The addition of external or group authorities can have an impact on the stability of the system. However, there are still collectors who manage to steal. What is certain is that without external reinforcement the system cannot be stable.

To try and add other options to the system, it is possible to add collectors that arrive after a certain time in the experience to see how the experience is changed. Also to get a stable situation, maybe add external authorities who monitor in a different way than the one implemented in my system.