Leysen Floris

Long-term sci-fi vegetation  
progression simulator

Graduation work 2019-20

Digital Arts and Entertainment

Howest.be

Contents

[Abstract 2](#_Toc21353889)

[Introduction 3](#_Toc21353890)

[Research 5](#_Toc21353891)

[1. Model to represent the system 5](#_Toc21353892)

[2. Spreading and growth patterns 5](#_Toc21353893)

[2.1 Spreading patterns 5](#_Toc21353894)

[2.2 Grass-like growth 6](#_Toc21353895)

[2.2.1 Sod-forming versus bunching 6](#_Toc21353896)

[2.2.2 Warm versus cool season 6](#_Toc21353897)

[2.2.3 Growth stages 6](#_Toc21353898)

[2.3 Life cycles 6](#_Toc21353899)

[3. Surface detection 7](#_Toc21353900)

[Case study 7](#_Toc21353901)

[1. Introduction 7](#_Toc21353902)

[2. Engine choice 7](#_Toc21353903)

[Conclusion 7](#_Toc21353904)

[References 9](#_Toc21353905)

[Appendices 10](#_Toc21353906)

# Abstract

**An abstract explains the aim of the paper in very brief, (the methods, results, etc.). Maximum length 150 words**

# Introduction

**In the introduction, you write the background of your topic, explain the purpose of the paper more broadly, explain the hypothesis, and the research question(s).**

The intent of this research is to create a system that is not only a standalone simulation, but also an expandable framework to be usable inside a complete game.

Project title:

|  |
| --- |
| Long-term sci-fi vegetation progression simulator |

Research Question:

|  |
| --- |
| How can the progression of vegetation in a contained environment, based on seeding, be accurately simulated real-time in a game engine? |

Project outline:

|  |
| --- |
| **ENVIRONMENT** - Sci-Fi; feels familiar by rooting in our nature, but adding elements that don’t exist here - Inspired by No Man's Sky    Source: reddit.com/4yuhxz  **SCOPE** - Timespan: calculations based on a day passing, can be accelerated or decelerated. If slow growing vegetation like trees are part of the possible growths, simulations will need to span many years. If not, it should give good results after just several years.  - Environment bigger than a back yard, smaller than a forest - Test case will likely be executed in 2500-10000m² - It is assumed that the environment can sustain the chosen vegetation in terms of nutrition and temperature. The only environmental element influencing vegetation growth is placement of surfaces (e.g.: some can only grow on the sides of objects, or need light to grow) - Environments do not have to be realistic (refer to image’s floating slab of ground) - Seasons remain the same as on Earth, but duration can vary  **VEGETATION** All vegetation has predefined visuals to indicate various states of progression. Most vegetation has a seasonal cycle that changes its properties. (e.g.: grasses will stop spreading during winter season, generally seeds are only produced during spring season)  **Specific properties:** Spreading: - Can either reproduce or spread through roots in or on the ground (like our grasses) - Can only spread through reproduction (fruits, seed pods)  Survival: - Relies on soil for energy - Relies on light for energy - Draws from roots of nearby vegetation for energy - Draws from tree it is located on for energy - Relies on wind for energy  Growth locations (possible options): - Can only grow on trees (or other plants) - Can grow anywhere as long as it is pointed upwards (i.e.: not on sides or bottoms) - Take root above ground and are suspended there - Can grow anywhere as long as it is not pointed upwards - Can grow anywhere with room to grow upwards (like ivy)  Progression patterns (possible options): - Seed => seedling => singular plant or tree => grow => reproduce (back to grow) => die - Seed => seedling => singular plant or tree => spread through roots => grow => possibly reproduce (back to spread) => die (when conditions are unfavorable or after lifetime end)  Seasonal patterns (possible options): - Spring reproduction => summer growth => fall stagnation => winter pause - Spring growth => summer stagnation => fall pause => winter reproduction  **PREDICTED PROCESS** - In editor, before running: Create volume, defining which vegetation is allowed to grow in this area and the amount of those plants' seeds being present in the ground before simulating. Note: if there are no plants that can grow just by having a seed in the ground (i.e.: no conditions like needing another plant nearby), nothing will grow. It is also possible to have seeds or roots enter the volume at set intervals. They come from an unknown outside area and were taken here via wind, animals...  - At start: The volume will check all objects inside of it and spawn the amount of predetermined seeds spread out over the area.  - During running (early stage): The seeds will get a chance to grow. It is likely that most seeds will be eliminated unless the specific vegetation has seed with high chance of surviving. Some vegetation will appear.  - During running (mid stage): Progression pattern is followed; more vegetation will appear and grow bigger.  - During running (final stage): Vegetation will start dying, freeing up space for others to grow. The ones that take their place will grow and eventually die, too. This cycle is repeated indefinitely.  **INTERACTIONS**  Since vegetation won’t just sit tight until it withers, there are some situations that can spell an early end to its life. - Bigger plants might inadvertenly eliminate nearby other plants by taking away their sunlight - Some might not survive a certain season because it froze on a cold day  - Disease or predator damage (outside unknown factors), these can spread to other nearby vegetation - Vegetation cannot grow on a patch of ground another plant is already occupying (exception: vegetation of the type that grows on trees) - Withered plants enhance their soil and improve growth of vegetation that reoccupies their space  **VEGETATION GENERAL PARAMETERS**  - Resistance to disease and predators - Seed survival chance - Expected lifespan  - Growth rate (before counting factors that influence growth like sunlight)  **ENVIRONMENTAL PARAMETERS** - Disease/predator appearance rate  - Season durations - Possible vegetation and the amount of seeds present for each of those (refer to section “Process”) |

Method and approach:

|  |
| --- |
| Analyze spreading patterns of grass-likes and reproductive types (fruits, seed pods)  Figure out which model can be used to represent the vegetation Influence of light on plant growth and calculating average time spent in light for an object (over 24h)  Detection of surfaces inside a volume (for vegetation placement)  Find optimization techniques for large interactive systems intended for use in a game  Investigate real life vegetation response to seasonal changes  Weighing engine options |

Deliverables:

|  |
| --- |
| - Engine package aimed to be used in games as a means of evolving environments throughout the passing of time, without placing any vegetation beforehand  - Package has customizable parameters to influence the simulation (see project outline)  - A video showing simulation of a few years  - Videos showing partial test cases |

# Research

## Model to represent the system

Before considering which model is best, it is important to define the characteristics of the intended complete system. Expandability and maintainability are crucial, so adding a new plant, removing one or updating an existing one should not necessitate any changes to the system. It is also necessary to keep complexity of individual vegetation elements low, while keeping an intricate larger scale evolving environment.

One model that has been extensively used in biological simulations is an agent-based model [[1]](#AgentBasedBiology). Agents define it and the system itself is kept simple, only being a framework for the agents to get information from. It’s a model based on emergence, meaning the complete environment possesses features that cannot be observed in its agents. Those two elements make it a perfect fit for our goal, though other possibilities were not ruled out.

A multi-agent system is one that is highly similar to an agent-based model, though both its agents and environments are of greater complexity than the aforementioned model [[2]](#AgentBasedModel). That distinction makes it a less viable model and it can be discarded.

Cellular automaton is a grid-based model, with each cell of that grid being in a certain state. Rules are defined for the updating of states within cells, customarily maintaining those rules over time and remaining the same between cells. This model has seen use in biological simulations, too, but for decidedly simpler results like patterns. Since the aim is to have many varying plants with different rules, this model was also deemed inappropriate.

Considering these advantages and disadvantages, the agent-based model was determined to indeed the best fit for the planned system.

## Spreading and growth patterns

### Spreading patterns

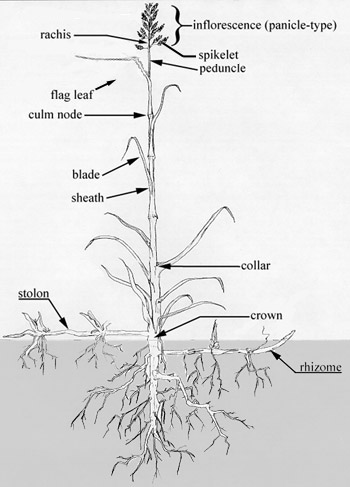
Currently no research is available on accurately predicting patterns of which direction grasses will spread their roots and at what pace, thus eliminating the possibility of considering this element for the simulation. Precisely determining where reproductive seeds will end up is influenced by too many factors – wind, animals, gravity… – to be usable. As a result, both root-type spreading, and seed-based spreading are based on assumptions of maximum and minimum expected distance from their source, unless specified otherwise by the rules of a specific plant.

### Grass-like growth

There are numerous types of grasses, sometimes having vastly different properties from each other. It is therefore more appropriate to analyse those properties individually, allowing combination at will. Ignoring non-grass specific traits – i.e.: height and lifespan – the differentiations can be limited to sod-forming versus bunching and warm versus cool season grasses. Reproductive spreading is omitted, too, due to not being exclusive to grasses.

### Sod-forming versus bunching

In nature, grasses of the sod-forming type form a large interconnected network both above ground – a connection like that is called a stolon – and below ground – called a rhizome. This manner of spreading allows new shoots – plants that are individually sustainable, but part of the network – to grow from a stolon, rhizome or even right at the ground level of another plant [[3]](#GrassManageMent).

 Figure 1: Stolon and rhizome [[4]](#BasicsOfGrassGrowth)

Bunching types, unlike sod-forming grasses, can only grow new plants at the soil surface of an already present plant. Grasses with this property cannot grow as closely packed as its counterpart, though the bunches themselves can be. This does increase the odds of an invasive plant appearing in the empty space between two clumps.

### Warm versus cool season

While both types employ the same method of energizing themselves through photosynthesis, the construction of their leaves is different. The deep science behind these behaviours is unimportant, but the result of it is.

Grasses meant for a cooler season are more active – growing and reproducing – during the more temperate seasons of spring and autumn, while growth is slowed down in the warmer season of summer and they might even become dormant. The warm season grasses, on the other hand, are at their peak productivity during the hot summer season. In terms of nutritional value, the cool season grasses are more densely packed.

### Growth stages

When a seed meets the right conditions to germinate, a single leaf will start growing. In this vegetative stage, leaves will continuously keep appearing and dying. Photosynthesis is the plant’s main purpose now. Changing of climate can cause the next stage. Here, the grass prepares itself for reproduction and reduces energy spent on other tasks like growing. With time, the final stage of flowering is entered. At this point, growth is completely halted, while flowers spawn. Reproductive spreading happens and depending on if the type of grass it might go dormant until the next growth season or die away entirely.

Other plants share their growth stages with grasses for the most part, with the main difference being they can still grow during the reproduction-related stages.

### Life cycles

There are three main types of plant life cycles: annuals, biennials, and perennials. The annuals go through all their growth stages just once, within a period less than a year, after which they die. Biennials need about two years to fully grow and die, using the first year to stock up on nutrients and reproducing in the second.

The perennial variation is a tad more complex than the others, possibly surviving indefinitely until struck by rot or other outside factors. While they can keep growing forever, their advancement is restricted to their growing season. Reproduction can happen every year, though if it does, the period of growth is reduced. Not all plants of this type can live endlessly, the only true requirement to be considered perennial is to survive for longer than three years [[5]](#PerennialPlants).

## Surface detection

The goal is to detect which surfaces are inside a defined volume and store them for later use in plant placement. Determining which objects are inside a volume is trivial when working in a game engine, so the only unknown is how to extract surface data from it.

One option is to check every individual triangle on the mesh and interpret each of those as a surface, then determining their orientation. This method introduces one major problem; if working on meshes with significant triangle counts, determining which surface is appropriate and available gets highly complex and can take a long time, especially when the variable of other plants taking up space is accounted for.

A second option would be to determine a surface only at the exact time something needs to spawn, e.g.: raycasting from an approximated location and spawning on the hit position. This method is promising, since it allows users to make changes to the environment without having to recalculate where everything is positioned. One great concern is keeping track of which spots other plants are already occupying, since there is no predetermined set of locations.

Since both have a serious flaw, but boast great advantages, too, some middle ground had to be found. Basing placement on a grid solves the problem of maintaining information on already inhabited locations. If the location of plants follows this defined grid, the issue of no longer looking natural appears. This could be solved by making the grid cells extremely small, yet doing as such defeats the purpose of it more simply and efficiently storing locational data. Instead, the grid can be interpreted as cells to remember which cells are occupied and which plant is in that cell.

This does not provide a solution to the actual placement, though. Simplifying the mesh-based surfaces, creating simple rectangular boundaries for each object inside yields a similar result. In irregular geometry, this results in floating vegetation, which is then solved by executing a raycast to place the object on the surface.

# Case study

## Introduction

## Engine choice

Since the focus lies on making a system that is useable in a game, it is only natural to pick an engine that is both widely used and accessible. This already narrowed down the choice to Unreal or Unity. Unreal has the advantage of beautifying the scene and vegetation more easily, while the strength of Unity lies in its ease of use and shorter development times – also for developers who want to expand on the system. Because of the intended style of vegetation being sci-fi, the generally less realistic style of Unity is no longer a major negative, thus triumphing over Unreal’s realism.

# Conclusion

**repeat the main topics, discuss your main findings, discuss the end result.**



Figure 1 : MAKING OF THE HOBBIT: THE DESOLATION OF SMAUG – LAKETOWN (WETA DIGITAL, 2014)

# References

[1] An, G. , Mi, Q. , Dutta‐Moscato, J. and Vodovotz, Y. (2009), Agent‐based models in translational systems biology. WIREs Syst Biol Med, 1: 159-171. doi:[10.1002/wsbm.45](https://doi.org/10.1002/wsbm.45)

[2] Agent-Based Model. (n.d.). In Wikipedia. Retrieved September 30, 2019, from: <https://en.wikipedia.org/wiki/Agent-based_model>

[3] Oregon State University. Discuss the basics of grass growth. Retrieved 1 October 2019, from the Oregon State University website: <https://forages.oregonstate.edu/nfgc/eo/onlineforagecurriculum/instructormaterials/availabletopics/management/growth>

[4] Oregon State University. How does grass grow? Developmental phases. Retrieved 7 October 2019, from the Oregon State University website: <https://forages.oregonstate.edu/regrowth/how-does-grass-grow/developmental-phases>

[5] Schacht, W., Sandall, L., Fick, W., Perennial Plants, Retrieved 7 October 2019, from: <https://passel2.unl.edu/view/lesson/50d18f39cff4/2>

# Appendices