Simulating human civilisation and the evolution of society based on factors such as culture, religion, and technology – Project plan

James Benjamin Brimelow Gorman - 100505349

College of Engineering & Technology, University of Derby

100505349@unimail.derby.ac.uk

1. Introduction And Hypothesis

History is a divisive subject, and rightfully so. Stories from centuries past still define the day-to-day life of people across the world – simple lines drawn by forgotten men have resulted in chaos and oppression, while small mistakes have resulted in prosperity for millions. Any discussion of this topic will inevitably end in disagreement, some idolize figures of the past, while others resent them – regardless though, it is undeniable the impact some people have had on the world. It is quite interesting then, how these actions that have changed the lives of thousands, are so easily represented on a simple piece of paper – conflicts of family, money or faith reduced to simple points on a map.

This idea has fascinated me for the last decade of my life, and in October of 2017 I decided to combine this passion with my love for software development, developing a program designed to make a new history – starting with a random map populated with various peoples that would build new nations which would rise and fall as conflicts with those around them occurred. When this project was finished, it successfully met my initial concept, but fell short of what I had wanted to achieve – populated with the mistakes of a more amateur developer, the project was error-prone and inefficient, nevertheless it was completed, and I considered it my best project id ever completed for years afterwards.

The year now is 2021, and I have learned a lot more about both software development and history over the years. I’ve found that even the ideas and concepts of that project were flawed in many ways, particularly relating to how the nations of the world acted. This is why over the next year I will be revisiting this concept, not only creating a new and better simulation, but also changing the focus in a way that I believe gives more meaning to the project – answering the hypothesis “*Can a simulation be used to believably portray a new world history, using artificial representations of factors that have defined real history, and what methods can and should be used to achieve this objective?*”

1. Aims and Objectives

To expand on the concept discussed above, the specific aim of this project is to develop a simulation of a world map that portrays the birth, expansion, and recession of nations over a period of time. Starting from an empty map, over time the simulation will develop a populated world map which has been defined by socio-economic factors such as culture, access to resources and faith as well as ideas like international relations, technology and geography. This artefact will also be heavily documented; presenting algorithms used, comparisons to existing projects (Primarily my own, as I have an intricate knowledge of how it functions and the flaws and advantages of these functions) as well as extensive logs of testing whenever a key point in development is reached. At the end of the development cycle, I will also record the accuracy of the program, demonstrating instances where I believe the model has particularly shown evidence to support my hypothesis, as well as incidents where it has failed to meet my goals.

It is my hope that this project will at the very least serve as an interesting experiment demonstrating how computing can be used to represent the world around us. I also believe that as the game industry is currently leaning heavily towards procedurally generated content, this project could see use as a world map generator – a service that may have relevance in a number of different genres, ranging from strategy games, roleplaying games and even to roguelike titles.

If the project is particularly successful, it could see use as an educational tool which, while not portraying real history, could demonstrate what factors have had an impact on our world’s history. I believe this is an especially important concept for the modern day because it is my opinion that a significant amount of pre-GCSE history is taught at a surface level – what happened and the effects that came of it – whereas the reasons for why are often ignored for unimportant topics. Considering factors that lead to key points in history is a necessary topic for fully understanding the gravity and reason for conflicts in history – such as how the mountains of Afghanistan have aided its survival against foreign powers or how the shortage of domestic animals in the Americas lead to the plagues which weakened the native’s ability to fight against the colonial powers.

In terms of objectives for the project, I believe there are 5 key milestones that will need to be met for this production to truly meet its potential – including both development and review. These key objectives are as follows:

* Design the basic functions of the simulation and create basic tools for the simulator to implement when necessary
* Implement world generation and the preliminary factors that will come into play as the simulation progresses
* Allow the simulation to progress through time and evolve
* Create the ability to save progress and thoroughly test the system and its functionality
* Compare the final build of the project to the hypothesis provided
  1. Design & Basic tool implementation

The key point of this milestone is to define how the system will function before implementation begins properly. This includes listing what variables will be a key part of the system (for example, what properties will need to exist after world generation?) as well as deciding what algorithms to use for important parts of the system (for example, how will world generation be handled?). While this document will outline some pre-existing concepts for the system design, the design stage will consist of an in-depth analysis of the requirements of the project, and how this will be achieved. The design stage will be complete when the following are produced to a degree I am personally satisfied with:

* Documentation of key factors and features to include within the software
* Rough sketches of the user interface
* Flowcharts and conceptualizations of key system components (While forgoing discussion of algorithms)
* Definition of algorithms that will be used for key system components

Simultaneously to this I will be working with the simulation’s primary engine – Unity – to develop some essential but simple tools for the functioning of the program, a process which will hopefully reduce distractions during the development stages. The following points must be implemented to a functional degree before this stage can be considered finished:

* Basic camera movement against a 2D plane
* Zoom controls
* Polygon rendering
* Polygon interactivity
* User interface implementation
  1. World Generation

After the design phase is concluded, the focus of development will shift to the first half of the implementation procedure – World generation. This, while on paper only representing a fraction of the features I intend to implement, will be a large chunk of the development cycle due to its importance and complexity. As the world map will serve as the backdrop for the entire simulation, as well as define the main factors the artificial intelligence will use in its decision making, it needs to be created to a satisfactory degree. This stage will also include significant documentation of the methods used and any testing applied on the final build of the world generation methods.

For this milestone to be completed, I will need an algorithm able to produce geographical maps with realistic continents and terrain. It is my intention at this stage that this will use the “Perlin noise” algorithm developed by Ken Perlin, a commonly implemented method for this style of world generation in both 2D and 3D space. This procedure will need to include these important geographical features:

* Height mapping
* Land and sea generation based on surface heights
* Temperature
  + A defined polar region at the north or south of the map
  + An equator which defines the temperate regions of the map
* Rivers and mountains using surface heights as a guideline
* Biomes using the temperature, proximity to water and height of a location as well as additional Perlin noise masks to decide aspects like forest density
* Resource distribution

The land will then be divided based into different polygons of varying size (depending on the geographical features of a location – for example, low flora locations with a consistent height mapping and extreme temperature will often have a dispersed populace, therefore they will be represented by a larger polygon. On the other hand, a location with access to water and average temperatures will have a more concentrated population and therefore a smaller polygon). These polygons will represent different populations on the map, each of which can form their own nation or be taken by another. After these “Provinces” are generated, they will need the following properties initialised (Though note that these are not the only variables a province will store, but rather just those that need to be given values at world generation):

* Vertices
* Colour
* Name of the provincial capital
* Population (This factor will be independent of the previously defined population density though will use geographical information once again)
* Culture (A shared property between a random number of connected provinces)
* Cultural region (A constant property that is shared between many provinces, in a similar geographical region. All provinces in the same culture will be in the same cultural region, and this variable will act loosely like continents in real life)

While many of the aspects of this stage may be subject to change as development progresses, the listed properties above are what I believe are essential components, and therefore will mark the progress of this stage of development. A completed system should include all of the listed objectives in some form, and development into later stages should not occur until these points are met. Additionally, during this time documentation should continue, including discussing methods used to meet these goals and how they compare to similar projects. Finally, extensive testing should be completed and noted, and any high-priority issues fixed before proceeding.

* 1. Time and simulation

After world generation is complete, the actual simulation of time can begin. Starting from the map generated previously, provinces should be able to start to form their own nations, fighting with nearby provinces to expand their influence. It is my hope that by the end of this stage the simulation will over time mimic the evolution of society from the start of history. While how history progresses should be variable, a perfect model should follow a path similar to real world history – the events I have classified as relevant are listed below, in the order they can be expected to occur in. It should be noted that these events are not a strict list of requirements, and a successful simulation may only exhibit a number of these events, this is simply a guideline of how I currently expect the simulation to act.

1. Dawn of civilisation - A number of large powers form while the majority of the world remains in a tribal or dispersed state
2. Classical period – Religions should form and spread quickly between different provinces.
3. Iron age - Through any means, at least one large power should be overtaken or splinter into different states
4. Early medieval period – Small states, either remnants of a former great power or newly formed, should become more common, and should seek to unify provinces of their cultural groups.
5. Late medieval period – Many cultures should be entirely owned by a single nation; wars should be frequent between states.
6. Renaissance – A large technology gap should exist between different regions on the map. The majority of the map should now have nations on it.
7. Colonialism – States with better technology should begin to subjugate smaller states
8. Industrialisation – The technology gap should begin to even out, and some of the colonial states should lose territories. Wars should become less frequent.
9. Modern age - Technology gap should be almost entirely even. Wars that occur should involve a number of powers at once. All available land should be occupied by a nation.

In order for the simulation to be able to reach any of these goals, I believe the following functions and features are necessary:

* Time should be able to progress, ideally at different speeds depending on user input.
* Ability for nations to form
* Provincial information should update over time, such as the land becoming more populated as time progresses.
* Ability for nations to declare war on others and take land for themselves.
* Artificial intelligence which dictates how nations will act, using knowledge of the world around them to influence their decisions.
* Nations should be able to develop technology to give them advantages which should spread to neighbours over time.
* Religions should be able to form and spread.
* Nations should have relationships between each another, influencing how they will act towards the other, potentially even resulting in alliances. This should be impacted by factors such as cultural differences, technology gap and religion.
* National leaders who change over time and have personalities values that apply modifiers to decisions nations make – a confident leader will be more inclined to go to war but will be more likely to pick an impossible battle than a cautious one.

These points will serve as the goalposts for when the simulator itself is completed. Most if not all should be completed in some form before production can continue, though the specifics of how each will function (particularly in relation to how war will be modelled) will be better detailed throughout the design stage of development. Much like world generation, all content should be documented and thoroughly tested before production proceeds.

* 1. Saving and final testing

This stage will primarily consist of artefact testing, but it is important to make note of the save feature – which should be fully implemented by this stage to store both the generated world and the properties at the time of saving. This should also include a loading mechanism to make use of saved information, after which simulation should be permitted to continue.

In reference to testing, a full review of features in the model should be completed in acceptance, edge-case and erroneous environments. All of this testing data should be appended to the existing testing documentation produced over the course of the project. The targets of testing should include all features designated as necessary in this document, as well as any vulnerable or sufficiently important modules, additionally, any input mechanisms of any form should receive review to ensure they function as expected. Finally, due to the nature of this project as a simulation, a number of instances of the software running normally should be documented in detail to identify any anomalous results.

* 1. Review

This final objective will be to improve or complete any documentation that has not yet been finished by this stage, as well as to write a full review of how the project compared against my hypothesis, as well as what objectives outlined in this document were fulfilled. This will also include a full comparison against the simulation I produced years back as well as any projects with similar objectives or features.

* 1. Gantt Chart

1. References