

Aims

- Research relevant topics and papers.
- Create or Use an existing framework that can be used to develop a procedurally generated map.
- Design an evaluation algorithm taking into account size, path of the payload and fairness.
- Develop a application that can procedurally generate escort mission maps.

Methodolgy

Procedural map generation

The map generation combines several techniques to create the map these include; Binomial Generation, Height-Maps , Gradient Noise and Dijkstra's algorithm.

Evaluation Algorithm

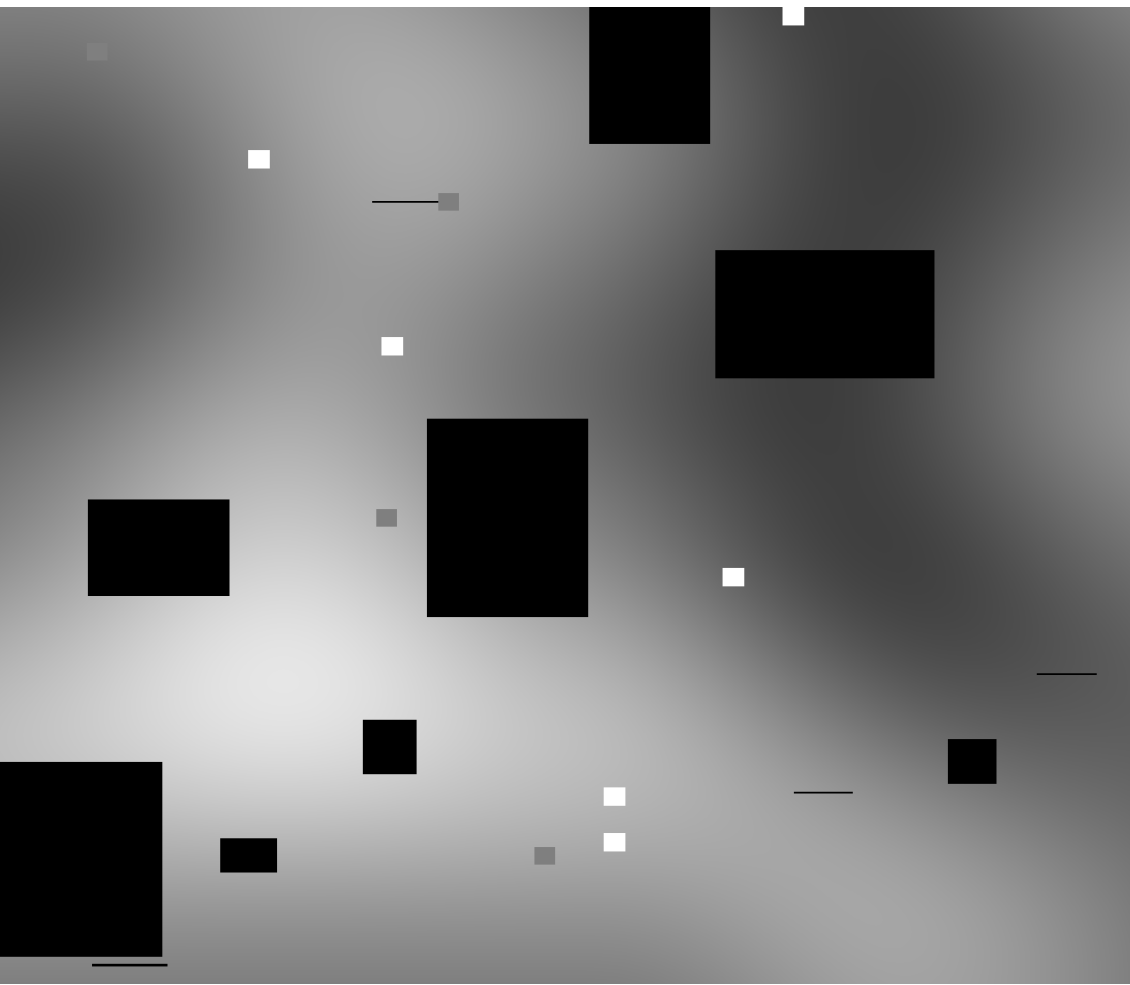
The algorithm measures three separate things; Size of map, Length of payload and fairness of the map each are represented by their own part of the formula shown below. The algorithm returns a higher number the worse the map is with zero being the best.

$$\frac{(E(t_c) - \frac{d_c}{v_c})^2}{k_1} + \frac{(E(t_p) - \frac{d_p}{v_p})^2}{k_2} + \frac{(\sum(\frac{o_{aw}}{d}) - \sum(\frac{o_{dw}}{d}))^2}{k_3}$$

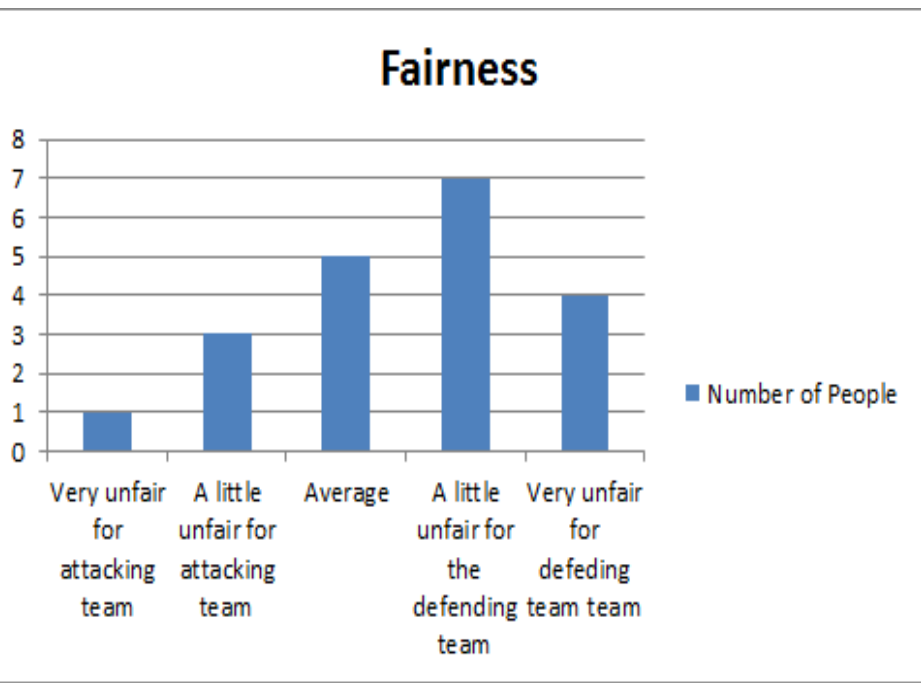
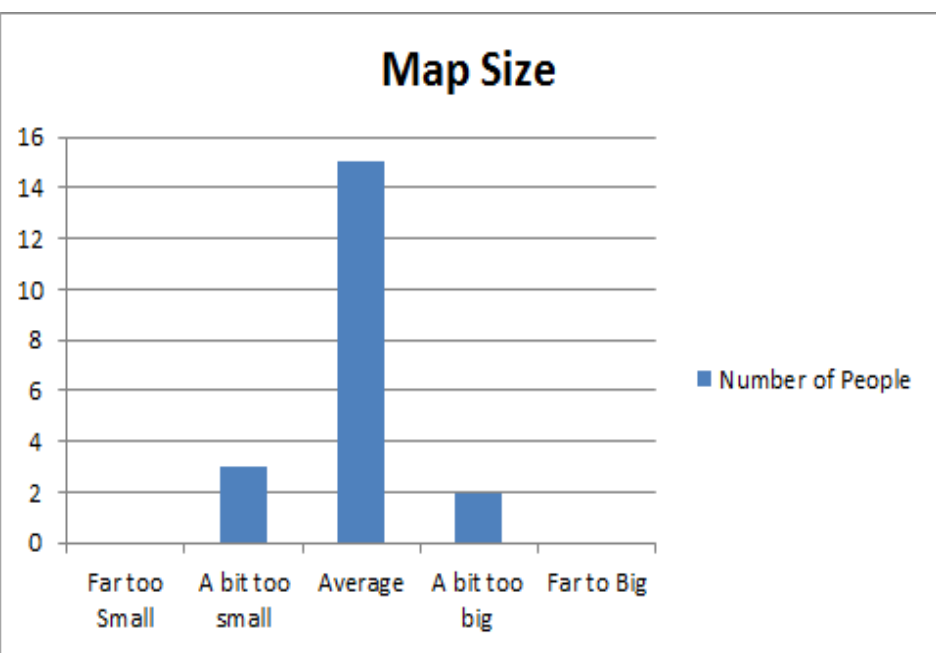
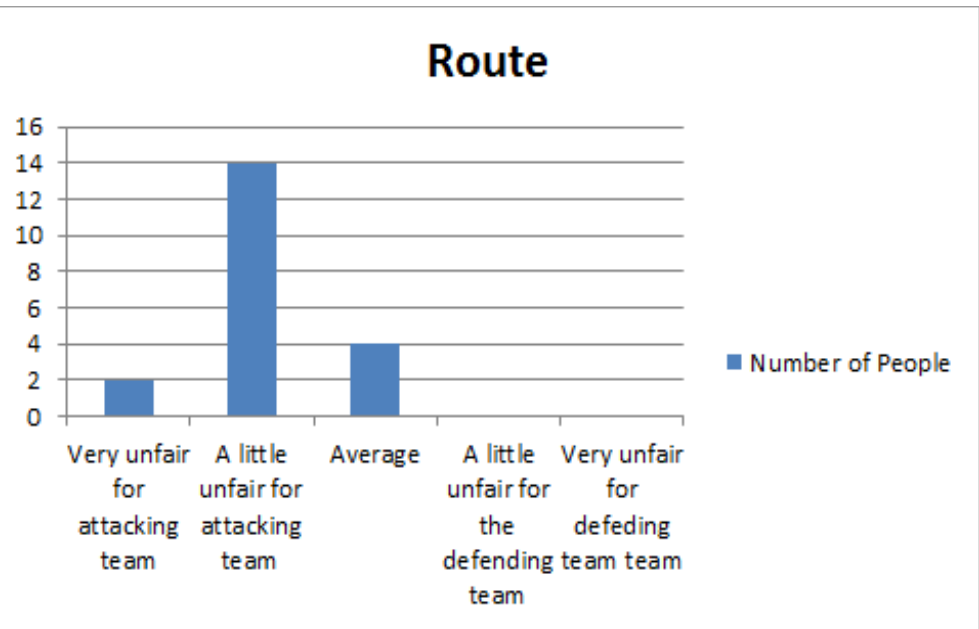
Results

Map Generation

Map Statistics:
Number of Buildings = 12
Number of Health Packs = 4
Length of map = 15644.4
Speed of player = 50
Time = 312.887 seconds
Speed of Payload= (150)
Length of Payload route = 25423
Theme = Desert
Number of Sides = 0



The map key is : Black squares are buildings, grey squares are health-packs and white squares are the route that the payload will take. The darker the gradient of the landscapes the lower the map is. The attackers spawn is the top right corner while the defending team are the bottom left corner.



The survey and algorithm here are for the map above. The survey asked the participants what their opinion of certain aspects of the map were. Their responses were made into the graphs and compared with the algorithm.

Algorithm's Calculations

Map 1	Initial	Total
Size	-252	1270
Path	13.3	177.78
Fairness	-198.3	1572.92
Algoritmh Total		3020.7

Background

Co-operative First Person Shooters

This is a sub-genre of first person shooters . This type of game heavily relies on teamwork, with games being won by holding onto an objective or pushing a payload. These games are enjoyable to a wide player base, usually offering a variety of play styles or different characters to use.

Maps in Video Games

In video games, maps are the term used to describe the area that the player can move around in. Different games will have different types of maps depending on how the developer wants the player to play the game. For instance an RPG like Skyrim will usually have a large map for the player to explore the world and find their own path.

Procedural Generation

Procedural generation is a technique that can be used to generate data algorithmically and can be used to greatly reduce the time taken for the creation of large systems within the game. This can lead to a less predictable game and can also reduce the memory size of the game.

Pathfinding

Path-finding is way for a computer to calculate the shortest route between two points. It does this by searching a graph by starting at one vertex and evaluating the neighbouring nodes. The most common of the path-finding algorithms is Dijkstra's algorithm.

Evaluations

An evaluation is part of the development cycle where the company judges how well the company is doing. It is a very important stage in the development cycle that if performed wrong can seriously

Conclusion

Discussion of Results

As shown in the graphs the majority of people agreed that while the maps are not perfect they are playable. When compared to the algorithm their are some similarities between the two sets of results. Both the Survey and the algorithm agreed that the defending team have a slight edge in the fairness, they also agree the path is biased against the defending team, However where the disagree is in the size of the map where most people thought the map size was fine the algorithm believes that the map size is far too big

Fulfillment of Aims and Objectives

This project has resulted in successful implementation of a procedural map generator that adheres to the aims and objectives that were stated during the IPO

Future Work

Several optimization's could be made to the procedural map generator including parallelizing the noise generation using OpenMP or CUDA. The evaluation algorithm could also benefit from several additions to the equation these could include using different parts of map to test fairness including looking at choke-points.