

Complex game design

What the system is:

This system is for a gpu multithreaded pathfinding system for AI. This system is separated into two components, an AI agent component that can be attached to a gameobject to provide movement and pathfinding. The second component is a node graph generator, a node graph is a collection of position points with distances between them for the AI agent to follow along for pathfinding. Both of these systems will offload their node processing to the graphics card of the users / players computer to increase performance. These two systems work together to create a standalone pathfinding package for unity.

What is the purpose:

This system's purpose is to allow for automatic node creation for the user to reduce the time spent manually placing them individually, and for fast pathfinding that does not affect cpu performance.

How the node graph system will assist the user in making pathfinding nodes is with the ability to select or group gameobjects of certain tags or layers as "pathfindable", from there the user can press a bake nodes button in the inspector and the node graph will be made, this will then be accessible by the AI agent to use for pathfinding. The process of making the nodes will be done with the cpu creating the nodes at each given point (based on user defined distances, to include vertices of objects selected etc.), from there the list of nodes is given to the gpu which will open a thread for each node to cycle through every other node to find those that can be connected to it, creating the node graph.

The pathfinding system will use the created node graph from the generation system (or user placed node graph) to find the shortest route to its destination, the way this will be done is on the start of the application the node graph will be put into gpu memory with an identical copy put on the user's computer's ram, when an agent calls the pathfinding function the cpu will send the position of the ai agent and the desired position, then the gpu will start processing nodes with the a star pathfinding algorithm, when the gpu finds a node that has multiple connections it will open a new thread allowing many nodes to be processed at the same time. This is done on the gpu to allow the cpu to have more performance to use on other game systems.

What unity libraries will I be using:

This system will be using the following unity engine libraries, the general library, the editor library and the events library. The general library is needed for basic unity functions such as monobehaviour, the editor library is needed as the node graph system will be its own editor object and the events library will be needed as this system will be run on events rather than in an update function.

What mathematical operations would my system use to function as intended:

Basic vector math such as distances between two vector threes

(square root of $(a^2 + b^2 + c^2)$) will be used for the most part as the system will be based on the distance between position vectors. Float differences will be used for getting height differences between positions.

What advanced algorithms are required to be implemented:

A star pathfinding algorithm will be used as the main pathfinding algorithm. This is because it is faster than using breadth first search and similar simple pathfinding algorithms. This algorithm will have changes made to it to allow for parallel processing of nodes without overlapping.

How will the system be modular:

As this system will be a universal pathfinding solution it can be used for many different game ideas without having to change the system itself, there will be many options to change how the system works such as the number of nodes per distance and the height restrictions between nodes. The AI agent will have its movement abilities to be greatly altered through its speed, turning speed, breaking speed, acceleration and stopping distance, allowing for it to be used for pedestrians, enemies, vehicles and many others without having to be reworked.

How will it be integrated into an application:

The system will be a unity package that is downloadable from the asset store, when added to the users project, they will import the package.

There will be a custom inspector tab they can add for the node graph generation, in this inspector the user can change what objects are walkable, the object tags that are walkable or not walkable, the number of nodes, the height restrictions between nodes and etc. from there the user can press a bake node graph button to create the node graph for the pathfinding algorithm.

To create the AI agent the user can add a pathfinding component to a game object. The agent will have functions accessible from the component to control its movement, to follow a path and to stop its movement based on inspector values that can be changed in script.