

What is the difference between a grid graph, point graph and a navmesh?

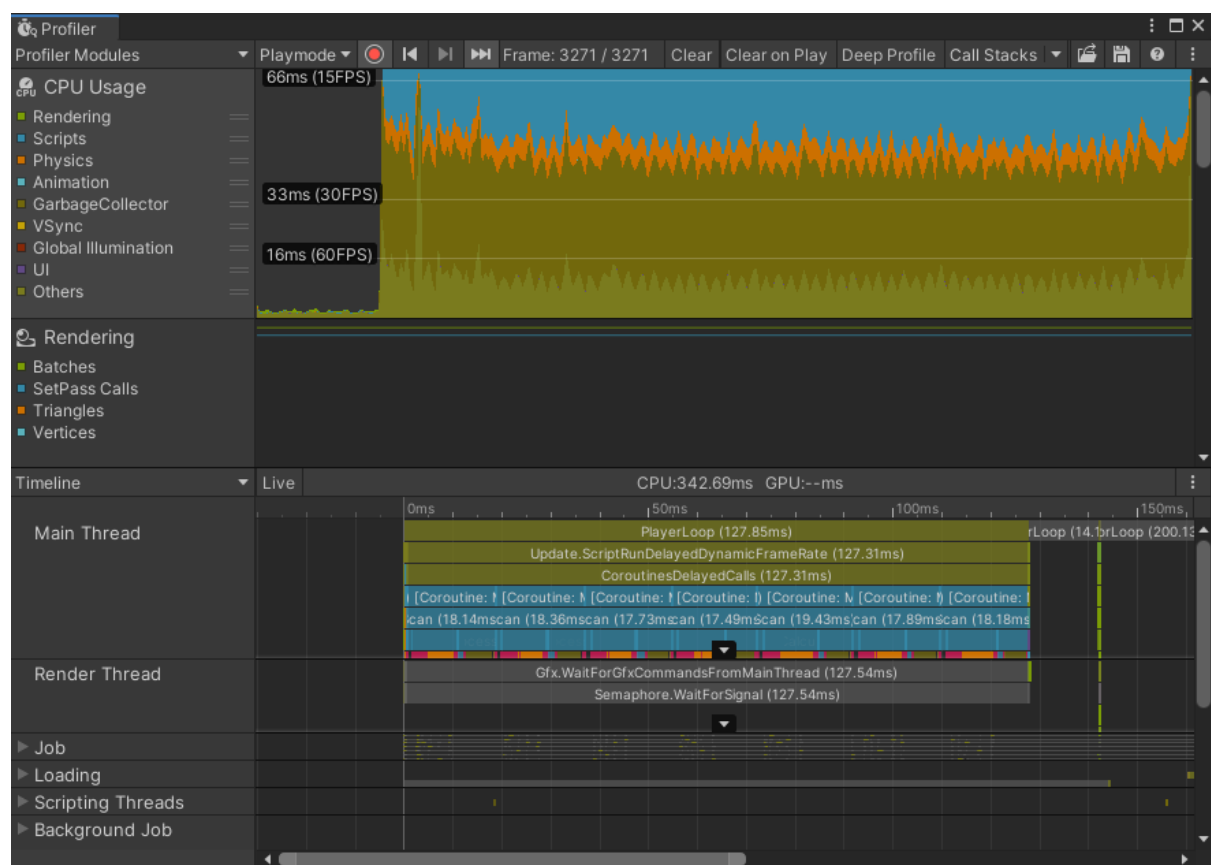
Grid graph is the most recommended graph for most scenes, It generates nodes in a grid pattern of Width*Depth. This graph is specialized for runtime graph changing, which is mostly used in RTS or Tower Defense games.

Point graph is the simplest of the graph types, but offers a variety of customization, consisting of user-placed points linking together. The connected nodes use raycast to link together.

Navmesh is a main type of graph, the pathfinding data uses triangle meshes instead of square grid or points. Navmesh is best used when the graph doesn't need a lot of changing during runtime. Since it has fewer nodes, it is faster than grid graph.

What is the biggest computer bottleneck for AI pathfinding? Use the Unity profiler to determine this

The biggest computer bottleneck for AI pathfinding is CPU as the scripts and pathfinding heavily rely on the usage of CPUs.



Describe the uses of Monte-Carlo Search Tree for strategy games and the use of combat artificial intelligence in FPS games

The method of Monte Carlo Tree Search works by predicting the path taken by the policy to reach the final winning zone, usually used in Real Time Strategy games. It's an algorithm to figure out the best set of moves which are; **Selecting, Expanding, Simulating and Updating to find final solution.** The method keeps repeating until it reaches the solution while it is learning. The Tree Search Algorithm is used to search each possible move that exists after a turn, visualizing a tree representation. FPS games use a pathfinding algorithm called Reinforcement Learning, it is an algorithm that learns how to achieve a complex objective over many steps and has a positive and negative kind of reinforcement learning method.

KU6 Describe available readymade AI solutions

A* pathfinding is a popular AI plugin for the Unity game engine.

1. Comparing to A* Pathfinding plugin, RonenNess Unity pathfinding plugin only works for Unity 2D tile-based scenarios.

<https://github.com/RonenNess/Unity-2d-pathfinding>

2. Comparing to A* Pathfinding plugin, Kuptsevykh's Unity DOTS Pathfinding works for both 2D and 3D which utilizes Unity3D DOTS for multi-core performance.

<https://github.com/Kuptsevykh/Unity-DOTS-Pathfinding>

3. Comparing to A* Pathfinding plugin, h8man's NavMeshPlus uses navmesh that is automatically generated from the scene geometry for 2D scenarios

<https://github.com/h8man/NavMeshPlus>

4. Comparing to A* Pathfinding plugin, Unity-open-tools' OpenPath pathfinding plugin uses an automated, simplistic path finding procedure and automatic world bounds calculation + adjustable scanning parameters.

<https://github.com/unity3d-open-tools/openpath>

5. Comparing to A* Pathfinding plugin, dbrizov's Unity-PathFindingAlgorithms plugin works similarly to A* but with more algorithms such as BFS, Greedy BFS and Dijkstra

<https://github.com/dbrizov/Unity-PathFindingAlgorithms>