

Traffic Modeling with Unity 3D

Isaiah Martinez
CSUN
Computer Science Department
isaiah.martinez.891@my.csun.edu

Jae Molina
CSUN
Computer Science Department
jae.molina.499@my.csun.edu

Anastasia Naydina
CSUN
Computer Science Department
anastasia.naydina.947@my.csun.edu

02/26/2024

Contents

1	Change History	2
2	Progress	2
3	Challenges	4

1 Change History

Version: 0.11

Modifier: Isaiah Martinez, Jae Molina, Anastasia Naydina

Date: 2/26/24

Description of Change: Simple Car model made. Looked at Related Works for process in utilizing Unity for traffic modeling. Looking for additional related works.

2 Progress

Where are we with design/implementation?

We've added a little scene for potential pathfinding including a goal post and starting post that a "player" would set up.

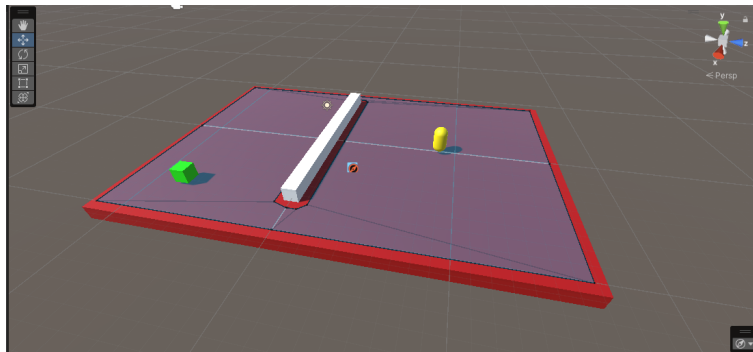


Figure 1: A screenshot of the current pathfinding scene.

The green cube represents the post that the initial start positioned yellow capsule would make its way towards, and the white column in between is an obstacle that the NavMesh has labelled as an obstacle. In the program, the yellow capsule navigates around the column and goes towards the green goalpost. The light blue mapping on the red floor is the current NavMesh Surface baked that tells the start capsule where to go.

Currently, the two posts are set at the beginning of the project. When the game starts, the posts will be wherever they were placed before starting the game during the scene. Moving forward, we will be implementing a UI that the user can place the green goal post and place the starting yellow point and the pathfinding leads to the end.

We discovered a useful free Unity Asset that will make the creation of the road easier. The asset is called EasyRoads3D Free v3 and can be found at <https://assetstore.unity.com/packages/3d/characters/easyroads3d-free-v3-987>

This asset can create complex road networks based upon nodes and that means we can automate the process by placing nodes at each intersection to create a road network.

car modeled which is a generalized Sedan look.

Below are the packages that have been installed to allow the MLAgents git repository to be cloned.

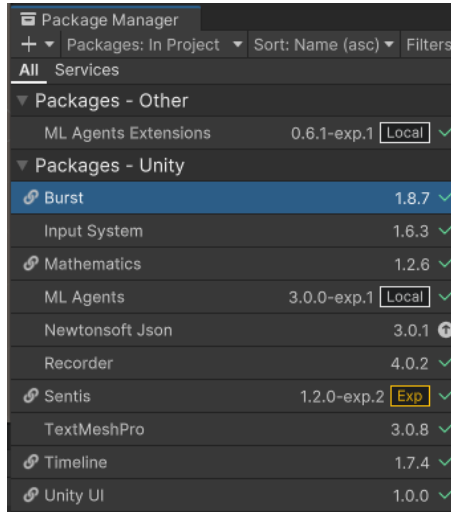


Figure 5: Screenshot of the downloaded packages.

3 Challenges

What challenges have occurred for this week?

Currently, there is an issue somewhere with the pathfinding code that it won't start moving towards the end goal post (green cube shown above) and there is an issue with the playerwalk variable, which hopefully is an easy fix. There was a challenge with implementing the NavMesh, since it is (or soon will be) a deprecated system in Unity that you need to go out of your way to download through package managing in Unity. The AI window is not a part of Unity by default, that we just needed to download that seperately.

Getting the MLAgents repository to be downloaded took lots of IT work. Environment variables were changed, new paths were added, and new pytohn installations were necessary. Additionally, there were a myriad of packages to install and load into the sample unity project provided by the MLAgents developers. All in all, it took a long time to configure and was a headache to straighten out.