Adam Kaufman

IGME-202-04 Final Project Documentation

Erin Cascioli

Project Theme:

My project is a battlefield! In it, two armies, ne of brave knights, and the other of evil skeletons clash at each other endlessly. In the scene, there are a total of 6 flocks, 3 per side, all with 4 npc’s in each flock. All of the flocks start out on their own sides, and spread out on their side of the battlefield. When the scene starts, the npc’s of a flock will run toward each other, and then start moving toward the main battlefield, or the middle area between the two sides. Once they have arrived in this zone, they will begin wandering around, while staying within the zone. They will continue to wander around as flocks until they see an enemy, at which point they will stop wandering, and will immediately start pursuing the enemy. Once the npc and the enemy have arrived at each other and clash (collide), the faster one will win and the other npc will die. When an npc dies, it will start to fall over, and then once it falls over, it will die, and a replacement soldier will come running. In addition, as they are doing all of this, they are also avoiding rocks that act as obstacles which are scattered around the environment. And as an added bonus, occasionally when a good amount of an npc’s flock dies, if it is still alive it may start to flee back towards its side until re-enforcements arrive.

Algorithms and how and when they work:

**Containment**:

In my scene, I have two overall large containment fields, one encompassing the entirety of the battlefield (including the sides of either team), and one that acts as the main battleground (the middle or neutral part of the battlefield). The npc’s spawn on their own sides, outside of this smaller containment field, so they are automatically being forced to get back into it. The outer one serves the purpose of just in case they leave the middle zone, they will have a significant amount of extra weight on them to not leave the battlefield entirely. The way containment works, is I have a bounding box collider set up in my unity scene, which has a script on it, that when an npc enters or leaves the bounding box, a bool value is switched, at which point the npc is told they need to go back to an arbitrary point that is hard coded in. For the neutral battleground it is the center, for the overall battlefield it pushes them towards their sides respective spawn point.

**Wandering**:

In the scene, it may be hard to miss sometimes, but the npc’s will wander as a flock, as in one wander force is determined per flock, when they are within both containment fields, and they can’t find any enemies to attack. The reason I give a single wander force to all members of a flock is so that flocks will stay together more easily, and so that wandering doesn’t conflict with the flocking behaviors and the npc’s can still wander around. The wandering algorithm used works using perlin noise. It generates two values using perlin noise, one for x and one for z. and uses those values to create a wander force. This results in smooth wander movements.

**Pursuing**:

When an npc finds an enemy npc, it will start pursuing it. In order to make the difference between pursue and plain seek, I add together the targets position and velocity vectors, and feed that to my pursue function, resulting in it pursuing its target. Each npc will pursue individual targets, rather than pursing or being pursued as a flock. This may not be obvious, but all npc’s are effectively being pursued rather than seeked.

The way other things work:

This scene takes full advantage of the flock code and obstacle code provided. The “fleeing” when an npc’s flock mates die is not actually anything other than cohesion telling it to go back to the centroid, which has just moved significantly behind it since the flock mates respawn.

The way respawning actually works in code however, is that once the npc has finished “falling over”, it is then teleported back to its teams spawn point, essentially faking the npc dying and another one immediately coming in as a replacement.

All of the flocks are managed by the game manager class, and are categorized into different teams.

Not all algorithms are active at once. This was a design choice. For example, npc’s receive no wander force whatsoever unless they are both within the containment zones, and they have no npc to pursue. An npc cannot pursue another enemy npc unless it is within the middle zone. And lastly, a containment force will go away immediately after the npc reenters the zone, as to keep npc’s from piling up at or near certain points.

The Camera:

The camera at default starts out following one of the knight flocks, using the flocks’ centroid as the follow position. You can scroll through all of the flocks using the left and right arrow keys. In addition, if you scroll past the flocks, there is also a ghost camera that you can control and fly to whichever vantage point you wish to see the carnage. (I did not script that camera, credits to follow). When in this free moving camera mode, use wasd keys to move around, and press the escape key to make the mouse cursor visible again. Warning: You can go through terrain and objects in this mode so don’t wander off to far.

Things that may look a little weird:

NPC’s do still sometimes shake around or begin to bug or shake out, but this is always short lived and they get right back to work.

Also, there is occasional model clipping between members of a flock. This is an issue with the model size, and its arms being held out, if I make the separation distance any larger, it will compromise the rest of the scene’s look.

NPC’s of different flocks on the same team will run into each other from time to time. This is because npc’s never actually acknowledge their fellow team members unless they are in the same flock. I am aware that it may look weird but I feel that it adds a sense of urgency to the npc’s, as if all they can think about is the battle, and cannot be bothered with other team members. Especially when an npc is “running away” and runs into another npc.

Credits for assets I used but did not make:

Knight: <https://www.assetstore.unity3d.com/en/#!/content/17292>

Skeleton: <https://www.assetstore.unity3d.com/en/#!/content/17495>

Free roam camera script: <https://www.assetstore.unity3d.com/en/#!/content/19250>

Terrain Assets: (the ones provided by you for the initial in class unity tutorial)

Ground Texture Pack: <https://www.assetstore.unity3d.com/en/#!/content/13001>

Stone Texture PacK: https://www.assetstore.unity3d.com/en/#!/content/12962