

ESNE – INGENIERÍA DEL CONOCIMIENTO: IA

WESTERN IA

PROFESOR: LUIS PEÑA SÁNCHEZ

Alejandro Benitez
ESNE DDVJ 3.3

Índice

Objetivo global de la práctica.....	2
Objetivo parcial. Primera entrega	2
Descripción del proyecto.....	2
Descripción del algoritmo empleado para solucionar el problema	3
Diario de desarrollo	4

Objetivo global de la práctica

El objetivo de la práctica es implementar distintos algoritmos de IA dentro de un entorno de Unity3D. Se plantea un escenario que irá creciendo a lo largo del curso que pretende englobar distintas técnicas de las que explicarán en teoría en un escenario concreto.

Objetivo parcial. Primera entrega

- Definición de la estructura de datos para la toma de decisiones de la información relevante del escenario.
 - Zonas de cobertura
 - Estado del jugador y de los cowboys
 - Otros elementos que se consideren importantes para el desarrollo de la IA.
- Primer algoritmo de toma de decisiones para controlar a los cowboys. Presentado el procesamiento y preparación de los datos para la toma de decisiones.

Descripción del proyecto.

El proyecto consta de una única escena. En ella hay una escena básica montada. Consta de un plano(suelo) con dos tipos de objetos en ella. Casas y barriles.

Las casas son coberturas completas mientras que los barriles son coberturas parciales.

Además, hay tres enemigos y un jugador.

Tiene una interfaz que muestra información como los FPS, memoria utilizada y una TopView con el campo de visión del jugador.

Los enemigos son capaces de esconderse del jugador cuando este los ve.

Dependiendo del arma que tienen equipada, se moverán más o menos lentos.

Descripción del algoritmo empleado para solucionar el problema

Problema: Que los enemigos se escondan al ser vistos.

Para resolver este problema, empecé con lanzar rayos desde el jugador a todas las coberturas para saber si había una visión directa de ellas o no. Al ver que era muy engorroso y poco optimo, indagué en internet a ver como lo solucionaba la gente. Tenía en mente hacer un cono de visión para determinar lo que pudiera ver el jugador. Encontré en YouTube un [tutorial](#) que lo resolvía de una forma que me gustaba. Hice el tutorial y lo adapté a lo que necesitaba.

Mi objetivo era tener dos listas que guardasen las coberturas seguras e inseguras y que el enemigo buscase en la lista de coberturas seguras cual era la más cercana a él.

Para ello he implementado las clases siguientes:

- BallCover

Estructura de datos que conforma una cobertura.

- FieldOfView

Esta clase pertenece al Player. Es la encargada de gestionar las coberturas y meter cada cobertura en las listas. Estas listas están en una clase con el patrón Singleton para que cualquier clase que lo requiera pueda consultar la información de las coberturas.

- CoverManager

Gestiona y guarda todas las coberturas de la escena.

- EnemyLocomotion

Gestiona los movimientos del enemigo.

- EnemyCoverManager

Gestiona y procesa las coberturas.

- EnemyDecider

Gestiona la toma de decisiones del enemigo

- EnemyWeapon

Gestiona el arma y lo que ello conlleva.

El funcionamiento es el siguiente:

FieldOfView rellena con las coberturas las listas que tiene CoverManager.

EnemyCoverManager procesa estas coberturas y en uno de sus métodos devuelve la cobertura segura más cercana.

EnemyLocomotion gestiona el movimiento para ir a esa cobertura.

EnemyDecider decide que acción realizar. Por el momento solo busca se mueve a la cobertura mas cercana cuando es visto por el jugador.

EnemyWeapon establece la velocidad de movimiento máxima al coger un arma

Diario de desarrollo

Empecé el proyecto a los pocos días de que se nos dijera de que iría la práctica. Por tanto, tenía ya algo hecho. Hice la generación de coberturas alrededor de los obstáculos y el código de la carpeta “./Assets/Scripts/OldScripts not used ATM”. Seguí con la practica e hice el tutorial antes mencionado. Por esto no he utilizado el proyecto base que se nos proporcionó. No obstante, miré los scripts y utilicé el PCLocomotion por ejemplo. También me base en la estructuración y nombres de los scripts.

He tenido algunos problemas a la hora de adaptar el tutorial que seguí para hacer el Field of view a lo que necesitaba. Y también a la hora optimizar el código. Tenia un bucle de orden dos con operaciones costosas como Distance o Magnitude. Esto hacían que el frame rate fuera de 50-55 FPS. Tras rehacer esta parte ahora va a 60FPS constantes.

Tengo problemas de lógica del juego, como que el enemigo va a la cobertura mas cercana sin importarle si tiene que pasar por enfrente del jugador. No me preocupa mucho porque espero aprender como solucionar estos errores.

Quiero añadir que me está gustando mucho la practica y la llevo con motivación porque es la primera vez que programo IA. Y me hace ilusión ver como los enemigos se esconden cuando te ven.