

GRADO EN DISEÑO Y DESARROLLO DE VIDEOJUEGOS

DESARROLLO DE JUEGOS CON INTELIGENCIA ARTIFICIAL

Práctica grupal 1:

Algoritmo Q-Learning

**Profesores**

Alfredo Cuesta

Dan Casas

**Autores: Grupo X**

Pedro Casas Martínez

Adrián Vaquero Portillo

**Fecha de entrega**

18 de noviembre de 2019

**Índice**

[1. Descripción del algoritmo empleado para solucionar el problema. 1](#_Toc24097172)

[2. Características de diseño e implementación 1](#_Toc24097173)

[3. Dificultades y obstáculos en el desarrollo del algoritmo 1](#_Toc24097174)

[4. Discusión sobre los resultados obtenidos. 2](#_Toc24097175)

# Descripción del algoritmo empleado para solucionar el problema.

El algoritmo empleado para la práctica es el Q-Learning, un algoritmo de aprendizaje automático por refuerzo. Esto quiere decir que para decidir la mejor acción en función del entorno el algoritmo recibe una cierta recompensa predefinida dependiendo de las decisiones que tome durante su etapa de aprendizaje.

En concreto, con Q-Learning se genera una tabla de valores que indican la calidad de cada acción posible para todos los estados del entorno donde por defecto se inicializan todos los valores al mismo. Después, se simulan una serie de episodios en los cuales se asigna un estado inicial aleatorio y se intenta llegar al objetivo mediante acciones tomadas al azar. Conforme se van tomando estas acciones aleatorias se modifica el valor de calidad de la acción en ese estado ( ) en función de la recompensa obtenida ( ) y del valor de calidad máximo del estado siguiente ( ). El cálculo del nuevo valor de calidad también depende dos parámetros predefinidos:

* Alfa: ratio de aprendizaje, define cuánto se conserva del valor Q actual y cuánto de la modificación calculada.
* Gamma : ratio de descuento, define la importancia del máximo valor de calidad del estado siguiente.

Por todo esto, la fórmula de la regla de aprendizaje para calcular el nuevo valor de calidad en el proceso de aprendizaje tras tomar una acción al azar queda definida como:

# Características de diseño e implementación

## **Descripción general**

En nuestro programa hemos definido un array tridimensional llamado *tablaQ*, que almacena los valores de calidad mediante los cuales se deciden las acciones del agente, en nuestro caso hacia qué dirección se mueve nuestro personaje. Los dos primeros índices de este array se corresponden con las coordenadas *x* e *y* del agente en el entorno, y el tercero se corresponde con la acción. Por lo tanto, sus dimensiones son el ancho por el alto del entorno por las acciones posibles en cada estado, es decir, 14x8x4.

La primera vez que se llama a la función que define el siguiente movimiento del personaje, se comprueba si existe el fichero en el cual se encuentra la tabla-Q de la semilla actual. Si no existe, se ejecuta nuestra función *AlgoritmoQ*, que genera la tabla-Q, y se crea un fichero con el nombre de la semilla en el cual se escribe la tabla generada. Si el fichero ya existía, se leen sus valores y se guardan en la tabla-Q. Posteriormente, se emplean estos valores en la función cada vez que es llamada para decidir cuál será el siguiente movimiento del personaje.

## **La función *AlgoritmoQ***

La función *AlgoritmoQ* recibe como parámetros la información del entorno (*boardInfo*) y la posición de la meta (*goals*). Primero definimos las dimensiones de nuestro array *tablaQ* en función del parámetro *boardInfo*. Tras esto, comienza un bucle for que se ejecuta tantas veces como episodios definamos en nuestra constante global *numEpisodios*. En cada iteración del bucle, se calcula un estado inicial aleatorio, y comienza un bucle while en el cual se simulan acciones aleatorias hasta que se llega a la meta, y se actualiza la tabla-Q tras cada acción mediante la [regla de aprendizaje](#_Descripción_del_algoritmo) especificada en el apartado de descripción del algoritmo.

## **Las funciones *LeerFichero* y *EscribirFichero***

Usamos estas dos funciones para leer el fichero y almacenar sus valores de calidad en nuestro array *tablaQ* y para escribir en un nuevo fichero los valores de calidad almacenados en nuestro array respectivamente. La nomenclatura de nuestros ficheros es el número de la semilla seguido de “.txt”.

En la primera ejecución del método *GetNextMove*, que se llama para definir el próximo movimiento del personaje, se genera la ruta y el nombre del archivo en una string y se comprueba si existe un fichero con ese nombre y en esa ruta. Si es así, se ejecuta LeerFichero, que recibe *boardInfo* y la string *nombreArchivo.*

# Dificultades y obstáculos en el desarrollo del algoritmo

HABLAR SOBRE EL CLAMPEO DEL ALGORITMO Q

HABLAR SOBRE DESACTIVAR SALIR DEL WHILE AL TOCAR UN MURO

CONFUSIÓN ORIENTACIÓN EJE Y, ORIGEN ENTORNO

Alicia es una chica llena de inseguridades que no hace mucho vivió una experiencia social que le hizo replantearse sus problemas y querer hablarlo con alguien. Decide quedar una noche en un bar con Marco, un amigo de la infancia, para explicarle su situación. Alicia comienza a explicarle lo ocurrido a Marco, que escucha atentamente.

Todo parece ir bien, hasta que un ladrón irrumpe en el establecimiento y la utiliza como rehén. El ladrón amenaza con matarla si no le dan todo el dinero y apunta con su pistola a Alicia. Mientras observa el arma del ladrón apuntándole a la cabeza, recuerda su intento de suicidio. Se ve a ella misma de nuevo en aquella terraza, preparada para saltar al vacío mientras contempla el atardecer. Cuando todo estaba a punto de terminar, un pájaro voló cerca y le hizo perder el equilibrio, haciendo que por poco cayese. En ese momento sintió miedo, y, finalmente, escogió no saltar. Se da cuenta entonces de que no lo hizo porque, en realidad, quería vivir. Se reincorpora y vuelve al lado seguro de la barandilla, y echa un vistazo al atardecer con miedo por lo que le pueda deparar el futuro, pero por primera vez en mucho tiempo con determinación para superar cualquier cosa.

Decidida a escoger su destino por ella misma, le da un cabezazo al ladrón, que pierde el arma por la conmoción. Marco entonces la recoge y le apunta mientras Alicia coge su teléfono y llama a la policía.

Después de que la policía arrestase al ladrón, Alicia y Marco esperan a que les tomen declaración. Alicia le confiesa a Marco como casi se quitó la vida un día y que de aquella experiencia aprendió que, aunque fuera infeliz, valía la pena vivir y luchar por una vida mejor. Marco, sorprendido, se da cuenta de que Alicia es la persona más increíble y fuerte que conoce y no duda en expresárselo.

Marco abraza a Alicia, que, tras muchos años, no solo no quiere morir, sino que además es feliz. Después de esta experiencia, Alicia cambia para siempre, no volverá a mirar atrás con tristeza, sino que mirará al futuro con optimismo, valorando lo bueno que hay en su vida. Se podría decir que en verdad la antigua Alicia si que murió, pero tan solo para dar paso a una nueva etapa, a una nueva Alicia.

# Discusión sobre los resultados obtenidos.

En el esqueleto plasmaré un diagrama de acciones contextualizadas dentro de su