

Redes Neuronales-Neuralbot

Vanessa Ruth Miranda Chavez
Vanessa_bolivia@hotmail.com

RESUMEN

El Quake 2 de id SOFTWARE, es el resultado de la evolución de género de los arcades 3D impuesto hace ya varios años por aquel Spear of Destiny/Wolfstein 3D y sobre todo por el Doom. Pasando por una enorme cantidad de programas se ha llegado hasta éste, el más avanzado de hoy día. Se habla ya del Quake III (lo distribuirá Activision y saldrá en el lugar de niveles extras para Quake 2) y de algunos sucesores como Unreal o Blood 2.

Palabras clave:

Aprendisaje, juegos de disparo, gen cromosoma, pesos, algoritmo genético, bot.

1. INTRODUCCIÓN

La inteligencia humana ha maravillado a los hombres desde el principio de los tiempos, siempre ha tratado de imitarla, igualar y mecanizarla para sus propios propósitos.

Comenzó por desarrollar algoritmos capaces de resolver problemas específicos, se interesó en aplicar la lógica matemática en la resolución de dichos problemas, y es aquí donde comenzó a desarrollarse la inteligencia artificial. El término IA se acuñó en Dartmouth, en 1958, durante la conferencia convocada por McCarthy a la cual asistieron, entre otros, Minsky y Simon.

Una de las ramas de la IA es redes neuronales de la cual hablaremos de una aplicación muy importante que se desarrolló con la ayuda de algoritmos genéticos.

2. MARCO TEORICO

La historia del juego sigue el formato habitual para los juegos en primera persona de id Software: unos portales a realidades de seres malignos han sido abiertos, y tu eres el único que puede viajar a través de ellos para cerrarlos. En el caso específico de Quake, la otra realidad está inspirada por muchas influencias, destacando las novelas de Howard Phillips Lovecraft. (el jefe del final del juego es el mismísimo Shub-Niggurath)

A pesar del paso del tiempo, Quake todavía conserva su estatus de clásico del género. La prueba de esto es que el juego ha sido expandido en niveles, modificaciones, elementos gráficos, personajes, historias, armas, decorados, etcétera, que sentaron precedente para la tecnología en los videojuegos actuales.

Sorprende aún cuantos juegos modernos usan la tecnología del juego, hecho en gran parte debido a que Id Software liberó el código de las diferentes versiones de la saga.

Quake es un juego de acción en primera persona que fue publicado por id Software (**id Software** es una empresa estadounidense desarrolladora de software en el campo de los videojuegos) el 31 de mayo de 1996. Introdujo algunos de los mayores avances en el género de los juegos en 3D: utiliza modelos tridimensionales para los jugadores y los monstruos en vez de sprites bidimensionales; y el mundo donde el juego tiene lugar está creado como un verdadero espacio tridimensional, en vez de ser un mapa bidimensional con información sobre la altura representado en tres dimensiones. También incorporó la

utilización de los mapas de luz y las fuentes de luz en tiempo real, descartando la iluminación estática basada en sectores de los juegos anteriores. Ofreció, en su tiempo, uno de los motores físicos más realistas programados para un videojuego hasta la fecha. Muchos creen que proporcionó la plataforma para la revolución de las tarjetas gráficas 3D independientes, "GLQuake" fue la primera aplicación que, en esos días, demostró la capacidad verdadera del chipset "Voodoo" de 3DFX. El impacto del motor del Quake puede aún sentirse en nuestros días.

La mayoría de la programación del motor del Quake fue realizada por John Carmack. Michael Abrash, un especialista en optimización del rendimiento de los programas, fue contratado para ayudar a que el motor de representación por software fuera posible teniendo en cuenta la velocidad. La banda sonora y efectos de sonido fue compuesta por Trent Reznor.

Quake y sus secuelas *Quake II* y *Quake III Arena* han vendido más de 4 millones de copias juntos.

Cuando nos referimos a primera persona quiere decir: Un **juego de acción en primera persona**, llamado *First Person Shooter (FPS)* en inglés, es un subgénero de los juegos de disparos que se desarrolla desde la perspectiva del personaje protagonista.

• **Cuando hemos tenido acceso a algunos juegos de computadoras, podemos habernos topado con algunas de estas afirmaciones en sus manuales:**

- Motor de inteligencia artificial mejorado.
- Nuevo motor de inteligencia artificial.
- Basado en el motor de inteligencia artificial del juego.

• **¿Qué significa eso?**

- El programa es inteligente?
- El juego tiene un motor como el de un automóvil?
 - El juego tiene características humanas

• **Adicionalmente ¿porqué todos (absolutamente todos) los juegos no promocionan la tenencia de motores de IA?**

• Normalmente los juegos que sí hacen mención a su "motor de IA" son del tipo:

- First person shooter (matanza): Counter Strike, Quake, Rainbow Six, etc.
- Estrategia: Diablo I & II, Warcraft, StarCraft
- De mesa: Ajedrez (Frist, ChessMaster)
- Deportes: Fútbol (FIFA, Winning Eleven)

• **¿Porqué en este tipo de juegos sí se puede hablar de motores de IA y en el resto no?**

La respuesta es sencilla:

En los juegos en los que se requiera cierto tipo de comportamiento inteligente, será necesaria la implementación de algún tipo de algoritmo computacional que simule dicho comportamiento.

En ese tipo de juegos (FPS, estrategia, de mesa) se requiere de tácticas, de estrategias, de movimientos ordenados, de búsquedas, de recorridos

Estas son actividades que los seres humanos realizan para

Resolver problemas sean simples o complejos

Comportamiento inteligente

3. DESARROLLO

3.1 ¿Qué es Neuralbot-Quake II?

Un bot es un programa que simula a un jugador humano. El Neuralbot es un bot para el juego Quake II que utiliza una red neuronal para decidir su comportamiento y un algoritmo genético para el aprendizaje. Es muy fácil probarlo y ver su evolución.

Un bot (diminutivo de *robot*) es un programa informático que realiza funciones muy diversas, imitando el comportamiento de un humano. En sitios wiki, como Wikipedia, un bot puede realizar funciones rutinarias de edición. En otros sitios, como Encarta, el bot puede responder a cuestiones sobre el propio contenido del sitio (tcc bots conversacionales

- Diminutivo de robot. Es un programa informático que realiza diversas funciones imitando a humanos.

- **En videojuegos:**

Es un programa capaz de jugar por sí mismo el juego en cuestión.

- **Importante:** Es una definición funcional

Quake II es un motor y juego de acción en primera persona desarrollado por id Software y distribuido por Activision en 1997. Aunque usa el nombre de Quake, también de id Software y es considerado, de hecho como su secuela, su historia se aparta totalmente del ambiente mítico-medieval visto en Quake para adentrarse en la ciencia ficción.

Quake II utiliza un sistema de red cliente/servidor similar al de Quake y posee todas las demás características del motor de Quake y varias adicionales. A diferencia de Quake, donde no se obtuvo aceleración por hardware sino por parches posteriores, *Quake II* fue lanzado con soporte nativo de OpenGL.

3.2 Desarrollo de un bot para Quake 2 basado en Redes Neuronales y Algoritmos Genéticos



Figura 1. Bot de juego

- Bot totalmente autónomo para el videojuego **Quake II**. Se implementó en 1999 por **Nick Chapman**. Fue mejorado hasta el año 2000.
- Es, probablemente, el único bot que aprende.

- Utiliza una red neuronal y un algoritmo genético.
- Propósito del proyecto de Neuralbot o Experimento de inteligencia artificial.



Figura 2. Bot en acción

El comportamiento de la red neuronal usada en Neural Bot es determinado por los pesos (fuerzas) de todas las sinapsis.

Ejemplos:

- IN: 'foot in lava' OUT: 'jump'. Mayor peso
- IN: 'wall in front' OUT: 'fire rocket launcher'. Peso negativo
-

Entonces, un nuevo comportamiento es aprendido cuando el patrón de los pesos de sinapsis es cambiado. Cada bot tiene su propia red neuronal con un único set de pesos de sinapsis.

3.3 Implementación de la red Neuronal del Bot

```
typedef struct
{
    int generation;
    float MAX_WEIGHT;// = 1;
    float MIN_WEIGHT;// = -1;
    float MAX_CHARGE;// = 3;
    float THRESHOLD; // = 1, the charge that has to build up
on
    a neuron before it
    starts conducting
    float DISCHARGE_AMOUNT;// = 0.1;
    int HIDDEN_NEURONS;// num hidden neurons per layer
    int SYNAPSE_LAYERS;
    Synapse Synapse_Array[60][60][4]
    Neuron Neuron_Array[60][5];
    qboolean shorttermlearning; //is short term learning enabled?
    float STL_learningrate;
    float totalstlweightchange;
    float reinforcement;
    int num_neurons_in_layer[5];
}NeuralNet
```

4. APLICACIONES

4.1 Planteamiento del Bot Mediante Algoritmos Genéticos

- **¿Para qué sirve?**
 - Para gestionar el “aprendizaje” de los bots: ¿qué tan certeros son al momento de disparar? ¿qué tanta destreza poseen?
 - **¿De qué manera?**
 - Modificando la Red Neuronal (RNA) de cada bot.
 - **Generación de la Población Inicial**
 - Se produce al introducir los bots en el juego.
 - **Evaluación de la población**
 - Se evalúa el “fitness” de cada bot: ¿qué tan bueno es el bot?
 - Fitness → número de muertes producidas por el bot.
 - Parámetros para considerarlo “bueno”
 - **Proceso de selección**
 - Se eligen dos “PADRES” (mejor fitness)
 - Se genera un cromosoma por cada padre.
 - Los pesos de las RNA se codifican
 - **Operadores genéticos**
 - **Crossover:** Ambos cromosomas son “cortados” y “combinados” en ciertos puntos *aleatorios*. Se forman los
 - **cromosomas de los bots hijos.**
 - **Operador de mutación**
 - Se eligen **dos** bots hijos.
 - Criterio de elección:
 - Menor **fitness**.
- Proceso de **decodificación:**
CROMOSOMAS → PESOS de RNA.
- Se reemplazan los pesos de las RNAs.
 - Algunos pesos de la RNAs son modificados.
 - Las modificaciones son aleatorias.
- **Algunas Consideraciones adicionales**

- **Factor aleatorio:** Múltiples resultados pueden ocurrir.
- Parámetros de entrada determinantes
- Generación de hijos (crossover, número de hijos, mutación de neuronas).

5. CONCLUSIONES

El juego por lo menos en el modo easy, es bastante acertada, es decir, ni nos dormimos jugando ni nos veremos obligados a utilizar el modo GOD en las últimas pantallas (esto último no quedaba más remedio que ponerlo en los últimos niveles de Quake 1) Los niveles tienen un alto nivel de elaboración, y no se vuelven en absoluto monótonos (de echo, habrá veces que nos colcaremos delante de alguna zona y nos quedaremos contemplándola en diversas partes del juego). Nosotros ya nos lo terminamos hace bastante, y nos hemos quedado muy satisfechos. Con respecto a las armas, son bastante buenas. La que más nos gusta es una que lanza balas de aluminio a velocidad supersónica dejando un rastro de humo en forma de espiral. Como arma más potente hay una que es muy similar al BFG-9000 del viejo Doom Y los enemigos esta vez gozan de una inteligencia artificial mucho mayor: hasta el punto que esquivan nuestros disparos, se agachan para evitar los misiles, se ocultan detrás de las paredes y nos persiguen. Tienen un aspecto mucho más "redondeado" que en Quake, pues están formados por un número mayor de polígonos.

6. BIBLIOGRAFIA

- [1] Goldberg D.E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley
- [2] [2] Bauer, R.J. (1994) Genetic Algorithms and investment strategies Wiley Finance
- [3] Banzhaf W. Reeves C. (editors) (1999) Foundations of Genetic Algorithms.5 Morgan Kaufmann Publishers.
- [4] www.monografias.com