

# Inteligencia Artificial

## Juegos

Victor Angel Chambi Nina  
darkangelotaku\_bo@hotmail.com

### RESUMEN.

Este artículo se basa en la inteligencia artificial y en una de sus aplicaciones, qué creo que es una de las más interesantes los juegos o más conocida como: "Teoría de juegos".

Donde daremos una breve introducción a como cuando y donde se dieron las primeras aplicaciones de La inteligencia artificial en juegos, todo el ámbito teórico que involucra crear un juego con inteligencia artificial (no todo es tan divertido como párese también hay teoría y muy compleja), el desarrollo de este marco teórico y como se aplica en varios juegos conocidos y por ultimo una pequeña aplicación de la IA en un tradicional juego llamado tres en raya (Incluye código fuente y un ejecutable que pueden descargar desde mi correo XD).

### Palabras Claves

IA o AI, Gamer, DOTA, System Espert.

## 1. INTRODUCCIÓN

Siempre ha sido un reto para el hombre lograr que las maquinas piensen igual que nosotros, o al menos simular tal situación.

Estos pensamientos fueron surgiendo mucho antes de que la computadora aya sido concebida, ya que se remontan muy atrás en la historia. El mito del coloso de Rodas entre los griegos, las estatuas "parlantes" del medioevo, el androide de Von Kempelen que jugó al ajedrez con Napoleón, y el "motor analítico" de Charles Babbage que calculaba logaritmos, son sólo algunos de los ejemplos de este antiguo interés. En 1940 surgen varios trabajos publicados que hacían referencia a la IA pero que no tuvieron gran repercusión. Pero una década mas tarde en 1950 el científico matemático británico Alan Turing fue quien consolidó el campo de la IA con su artículo *Computing Machinery and Intelligence* ("Maquinaria Computacional e Inteligencia"), en el que propuso una prueba concreta para determinar si una máquina era inteligente o no, su famosa Prueba de Turing por lo que se le considera el padre de la AI. Dos de las contribuciones más importantes de Turing fueron el diseño de la primera computadora capaz de jugar ajedrez y, más importante que esto, el establecimiento de la naturaleza simbólica de la computación.

Como se ve desde sus orígenes la IA se relacionó con juegos como el ajedrez y las damas, probablemente debido a que los juegos de mesa constituyen modelos de situaciones reales en las que hay que calcular, solucionar problemas, tomar decisiones, corregir errores, recordar, etc. Muchos de los avances teóricos y metodológicos de la IA se dieron gracias a los juegos. Por ejemplo, Samuel diseñó en 1961 un programa que jugaba damas y que era capaz de aprender de sus errores, es decir, era capaz de

adaptar su comportamiento en relación a eventos pasados. Lo pasmoso de este programa fue que, aunada a su capacidad de aprendizaje la de memoria, con el tiempo consiguió derrotar invariablemente a su creador. El mismo resultado fue obtenido por Bernstein a través de un programa que jugaba ajedrez (Boden, 1977). Los grandes "retos" entre computadoras y seres humanos se multiplicaron, siendo el más famoso de ellos el que ocurrió entre Dreyfus (un enconado crítico de la IA) y el programa Machack, en el que Dreyfus fue derrotado en un juego de ajedrez de varias horas.

A pesar de que esta línea de investigación ha sido casi totalmente abandonada en la actualidad por los científicos, es grandemente explotada las compañías de juegos como. EA GAMES; UBISOFT, SIERRA, BLIZZARD, NAMCO compañía conocida por su juego TEKKEN para mencionar a algunos.



Figura 1. (Jin kasama de "TEKKEN 5")

## 2. MARCO TEÓRICO

La mayoría de las personas cuando escuchan la palabra videojuegos y AI piensan en que el juego ideal es aquel en el que no puedas diferenciar lo personajes que son maquinas de los que son jugadores, pero nadie se pone a pensar que esto enmarca marca un sin fin de estudios dentro del área de la inteligencia artificial. Para la mejor comprensión de este artículo daré unos pequeños conceptos de las ramas que abarcan los juegos en la IA.

### 2.1 Sistemas Expertos

Son programas de computación inteligentes que usan conocimientos y procesos de inferencia, para resolver problemas sumamente difíciles para un humano ya que requerirían de

mucha experiencia. En sí un Sistema Experto; emula la capacidad de experiencia de un ser humano para tomar decisiones. Aunque los Sistemas Expertos se centran el dominio del problema, no tienen conocimientos más allá de donde fueron programados, esto es, no relaciona ningún otro concepto, a menos que este específicamente indicado. A este conocimiento se le llama Dominio del Conocimiento y como ya se apuntó es muy cerrado. Los Sistemas Expertos se basan en las Redes Neuronales. Sus partes son: La Base del Conocimiento, el Motor de Inferencia, el Subsistema de Explicación y la Interfaz.

- **La Base del Conocimiento:** Aquí están almacenados todos los datos que el Sistema Experto tiene para tomar decisiones.
- **El Motor de Inferencias:** Contiene todos los procesos que manipulan la Base del Conocimiento, para deducir la información pedida por el usuario. (Por Ejemplo, resolución, encadenamiento atrás hacia delante, etc.).
- **El Subsistema de Explicación:** Analiza la estructura del razonamiento y da una explicación al usuario.
- **La Interfaz:** Que es el puente de comunicación entre el sistema y el usuario.

## 2.2 Agentes

Un Agente es algo que razona (Viene del latín *agere*, hacer). Sin embargo los agentes informáticos deben tener algo que los diferencie de los programas comunes. Entre ellos está tener controles autónomos, percibir su entorno, que persistan por un periodo de tiempo prolongado, que se adapten a los cambios y que sean capaces de alcanzar objetivos distintos.

### 2.2.1 Agente Racional

Es el que actúa con la intención de obtener el mejor resultado, o si hay incertidumbre, el más apropiado. Estos agentes se basan en Inferencias, aunque existen situaciones en las que no se puede hacer nada correcto, pero se debe tomar una decisión; también poseen Actos Reflejos, ya que son mucho más eficientes que una larga inferencia en algunos casos. Para pasar la Prueba de Turing, los sistemas deben tener estas características. Estudiar la IA en base a Agentes Racionales, es beneficioso ya que nos hace diseñar al agente en base a inferencias, una forma correcta según las leyes del pensamiento. Sin embargo, por cuestiones de espacio es imposible diseñar una personalidad perfecta. En la práctica se usa una racionalidad limitada, por razones de tiempo y espacio.

### 2.2.2 Agente Inteligente

Son capaces de percibir un medio ambiente, usando sensores y actuar usando actuadores, los sensores pueden ser de un agente robot: Teclado, red, detectores de luz, etc. Y los activadores pueden ser una pantalla o la red. Se dice que un agente tiene percepción si puede recibir múltiples entradas en cualquier instante. En general este agente tomará decisiones de acuerdo a todas las entradas que recibió. A esta secuencia se le ordena en una tabla, no obstante por lo infinita o grande que puede llegar a ser, se limitan algunas entradas. El Agente es manejado por el programa del Agente, se debe diferenciar entre la función que es una descripción abstracta en el programa con el propio programa, que es la implementación real.

### 2.2.3 Agentes de Estímulo-Respuesta

Estos agentes solo reaccionan de acuerdo a los estímulos que reciben del mundo exterior donde operan, esta es la forma más sencilla de IA. Tomemos por ejemplo el caso de un robot que puede moverse en una superficie cuadrícula, es capaz de percibir si las celdas están ocupadas o no y ser capaz de moverse a las desocupadas. El robot entonces dispondrá de 8 entradas que llamaremos S1, S2, S3, S4, S5, S6, S7, S8. Y que están numeradas en sentido horario. Además tiene 4 movimientos posibles:

- Norte: Se mueve una celda arriba.
- Sur: Se mueve una celda abajo.
- Este: Se mueve una celda a la derecha.
- Oeste: Se mueve una celda a la izquierda.

### 2.2.4 Características de un Agente

- **Percepción:** Según el ejemplo anterior hay 8 variables para ubicarnos, por tanto existen  $2^8=256$  combinaciones de valores posibles. Algunos pueden ser descartados, ya que no existen pasillos estrechos. Podemos elegir 4 características llamadas X1, X2, X3, X4.

- **Acción:** De las características definidas, lo siguiente que debemos hacer es darles alguna definición para que puedan cumplir su cometido.

Por último esta parte es para aquellos que quieren empezar en este mundo de los video juegos, es muy interesante y ambigua a la vez así que disfrútenla.

## 2.3 Búsqueda Heurística

Se llama Búsqueda Heurística debido a que usa conocimientos específicos del problema. Con esta búsqueda podemos encontrar soluciones más eficientemente en el tiempo más rápido posible sin repetir ninguna ciudad. No obstante existe un tope de 30 ciudades, ya que aún a 1  $\mu$ seg por solución, se tardaría 1020 años en resolver el problema. (El universo solo tiene 1,6 x 10<sup>10</sup> años).

Una solución común es de 9 ciudades, que nos da una combinación de  $9!=362\ 880$  viajes posibles. Como se ve, la Búsqueda Heurística nos da la ventaja de rapidez en estos casos. A la aproximación de la Búsqueda Heurística, la llamaremos: Búsqueda Primero el Mejor. Existen algunas variantes:

### 2.3.1 Algoritmo \*A

Es la forma de búsqueda Primero el Mejor más conocida, sirve para el pathfinding (Búsqueda de Caminos) y es muy usada en juegos. Un ejemplo es el famoso juego Pacman: Los fantasmas que persiguen a Pacman buscan el camino más corto, en lugar de aparecer en forma aleatoria en el Mapa del Juego, otro ejemplo es el Age of Empires, un juego de conquista de civilizaciones, los enemigos salvan obstáculos para llegar a la ciudad del adversario.

El Algoritmo \*A, no desarrolla un camino por interacción, sino que desarrolla varios caminos y elige los más prometedores. (Para ver un ejemplo del Algoritmo \*A véase el Anexo A).

### 2.3.2 Algoritmo MINIMAX

Se poseen 2 jugadores: MAX y MIN, primero jugará MAX y así seguirá el flujo hasta acabar el juego. Existe un árbol de juegos que los programas utilizan para calcular los Movimientos Legales (Permitidos). El valor MINIMAX será un valor que determine el estado final del juego (En el ajedrez: -1, 0, 1; que son triunfo, derrota o empate). En algunos juegos este valor MINIMAX puede ser muy alto (Como en el Backgamon, se estima algo de -192 a 192 valores).

Incluso el Tictac toe (3 en Raya), es complejo para armar el árbol de juegos. A los niveles del árbol se les llama capas las cuales detallaremos mas adelante de este artículo como una aplicación.

Para determinar la estrategia óptima a seguir se usan valores MINIMAX. MAX adoptará el valor máximo y MIN el mínimo. La decisión MINIMAX supone que los 2 jugadores son óptimos, si uno es un novato será fácilmente derrotado por esta decisión.

El Algoritmo MINIMAX es la forma computacional en la que se calcula el valor MINIMAX de cada estado de sucesos. Lo interesante del Algoritmo MINIMAX es que no calcula todo el árbol de juegos, sino que calcula solo un horizonte de k niveles.

Un cálculo detallado del árbol de juegos perdería tiempo, ya que como ejemplo, en el ajedrez se calculan algo de 35100 movimientos totales. El objetivo de MAX aquí es ganar a MIN tomando alguna decisión sobre la mejor jugada que se puede hacer. El Algoritmo MINIMAX no se usa como tal en la Industria de Juegos, ya que es en algunas ocasiones lento, en vez de este se usan versiones optimizadas que veremos a continuación.

### 2.3.3 Poda $\alpha$ - $\beta$

El resultado del Algoritmo MINIMAX es el valor MINIMAX que nos permite tomar la decisión más acertada para el horizonte de búsqueda fijado, no obstante hay ocasiones en el que la ramificación del árbol de juegos es muy alta. (En ajedrez se calcula algo de 35100 o 10154 movimientos). Sería un derroche tratar de probarlos todos, esta es la razón por la cual el Algoritmo MINIMAX no se usa como tal en la Industria de Juegos. Afortunadamente se puede simplificar el proceso, evitando algunos nodos. Esto se basa en que al momento de explorar se determina que el nodo no va a cambiar el valor MINIMAX, es obviado, ya que no nos va a servir de mucho. Estos nodos quedan "Podados", ahorrando así tiempo de proceso en los programas.

El Algoritmo que implementa la poda  $\alpha$ - $\beta$  se llama Algoritmo  $\alpha$ - $\beta$ , y se basa en ajustar los límites llamados: límite  $-\infty$  y límite  $+\infty$ ; para el nodo raíz estos son de  $-\infty$  y  $+\infty$ ; ya que el valor MINIMAX estará entre esos 2 límites. Estos valores se propagan hacia abajo y se actualizan. Los nodos MAX actualizan su valor si  $v > \alpha$  y los nodos MIN actualizan su valor si  $v < \beta$ . La situación de Poda se detecta si un nodo satisface a  $\alpha \geq \beta$  y ello implica obviar la exploración del sub-árbol siguiente y devolver  $\beta$  como si se tratara de un nodo MIN. Si no existiera Poda (Es decir no se satisface la expresión  $\alpha \geq \beta$ ), el algoritmo se comportará como un Algoritmo MINIMAX normal.

### 2.3.4 Algoritmo MTD-f

El hecho de que la llamada al Algoritmo  $\alpha$ - $\beta$ , tenga los argumentos  $\alpha_0 = -\infty$  y  $\beta_0 = +\infty$ , significa que el valor MINIMAX  $v$  del nodo raíz, satisface a  $-\infty < v < +\infty$ . Esta es una ventana que se puede cambiar, no obstante al contrario de lo que queremos, esto consume tiempo considerable, por lo que si restringimos los valores habremos ahorrado una gran cantidad de tiempo. Si restringimos los valores a:  $\alpha = \beta - 1$ , podemos hablar de Ventana Nula, en contraposición a la clásica ventana infinita.

Una sola llamada al esquema de la Ventana Nula, nos dará una cota, no el valor MINIMAX, para hallarlo debemos usar distintas cotas para modificar los límites de  $-\infty$  y  $+\infty$ , para encontrar el valor MINIMAX. A este proceso se le llama Algoritmo MTD-f. (Memory-Enhanced Test Driver). Las cotas  $f^+$  y  $f^-$ , se inicializan a  $+\infty$  y  $-\infty$  respectivamente. Luego se comienzan a ajustar las cotas hasta que se la cota superior iguale o supere a la inferior; esto se hace en el procedimiento  $\alpha$ - $\beta$  con  $\alpha = \beta - 1$ , es decir búsqueda de Ventana Nula que devuelve  $g$ . Si  $g < \beta$ , ajustamos la cota a ese valor y sino ajustaremos la cota inferior. Este valor  $g$ , nos sirve para ajustar a  $\beta$  (Por lo tanto de  $\alpha = \beta - 1$ ) para la siguiente iteración. El número de estas dependerá de lo lejos que este la estimación inicial de  $f$  del verdadero valor MINIMAX del nodo raíz.

Muy bien ¡despierten! ahora ya tenemos conocimientos básicos sobre lo que se usa en un juego así que vayamos ahora a lo divertido ( $\tilde{n}$ ).

## 3. DESARROLLO

### 3.1 The Sims y la Vida Artificial

Como ya hemos visto la IA es una rama importante de la informática, Esto implica también modelar procesos biológicos para simular el ciclo de vida de seres vivientes, un ejemplo muy claro de esto son los juegos SimCity, Spore y Los Sims este último es el videojuego de PC más vendido de toda la historia. Su creador William Ralph Wright, Jr. (nacido el 20 de enero de 1960, Atlanta, Georgia) es un diseñador de videojuegos estadounidense y co-fundador del desarrollador, Maxis.

The Sims. Es un tipo de Virtual Dollhouse, o en español simulando a una familia viviendo en un suburbio.

El juego consiste en crear personajes Sims y "construirles" una vida, concretar sus aspiraciones, cumplir sus sueños, etc. El juego permite diseñarles a los Sims una casa y abastecerlas con todos los objetos que se requieran para satisfacer sus necesidades. Los Sims poseen un modo libre (Libre Albedrío) en donde los Sims realizan sus actividades por sí solos, en donde también puede intervenir el jugador dándole órdenes, las cuales serán respetadas o no por los Sims. El jugador deberá satisfacer las necesidades básicas de los Sims, como comer, dormir y bañarse, entre otras cosas. Para sustentar la economía de la casa, los Sims deberán buscar trabajo y ganar Simoleones (la ficticia moneda en el mundo sim), el juego ofrece una variada línea de trabajos, y el personaje podrá ir promoviendo en su cargo dependiendo de las habilidades que tenga y la cantidad de amigos. Existen diferentes tipos de habilidades como: creatividad, cocina, mecánica, estado físico, etc., que se podrán ir subiendo realizando diferentes acciones como pueden ser: leer

un libro, hacer ejercicio, mirar por un telescopio, cocinar, trabajar, ver televisión, etc.

Pero no todo es trabajo en la vida de los Sims, ellos también necesitan diversión. Cuando no la tienen se deprimen; las consecuencias son innumerables desde malos tratos con personajes cercanos hasta la muerte.

Los Sims también se relacionan con demás personajes, y dependiendo de sus gustos personales podrán formar o no amigos.

En la versión original el personaje sólo podía permanecer en su hogar, pero con las actualizaciones surgidas, el Sim puede ir a diferentes lugares públicos donde hay más personas con las que relacionarse.

En las primeras versiones hay muchas limitaciones, como que los niños nunca crecen, los Sims no tienen edad, y no existe un concepto de semana: los adultos van todos los días al trabajo y los niños todos los días a la escuela.

Otros personajes pueden llegar como visitantes a la casa del Sim, pero estos no pueden ser manejados por el jugador, aunque le dan sociedad al personaje.

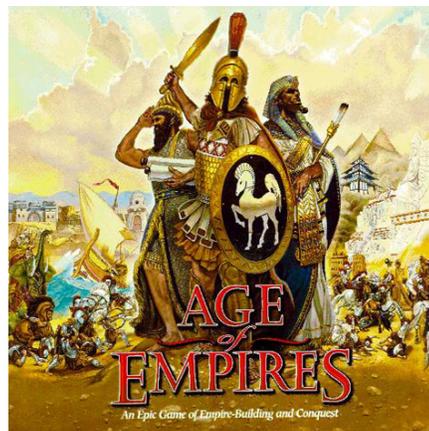
No hay un verdadero objetivo en el juego, ya que los Sims nunca mueren a no ser que se queden sin movimiento, se ahoguen, se quemén, se electrocuten o les agarre un virus (contraído de la mascota cerdo de guinea, que solo puede suceder si no se limpia su jaula). Lo interesante aquí es que los SIMS son una verdadera obra de arte dentro de lo que es la IA en Teoría de juegos ya que usan todos los conceptos de agentes, sistemas expertos por mencionar algunos, un ejemplo simple de dar es que en la segunda versión de sims **SIMS2**, los personajes habrían la puerta en ropa interior, Ahora la inteligencia artificial de Los Sims 3 ha sido completamente revisada.

Los Sims ahora llevarán a cabo los siguientes puntos en consideración. Tomarán una decisión si algo es apropiado antes de cometer una acción. Como un agente de estímulo respuesta.

Como por ejemplo: Los sims ya no abrirán la puerta con ropa interior; se cambiarán antes de abrir la puerta para recibir a sus visitas.

Y para terminar con los sims podemos decir que ya se lanzó el año 2002 una versión On-Line en que personas del mundo interactúan entre sí en un ambiente totalmente virtual, pero que la mayoría de los jugadores lo convirtieron en salas de chat o salones de foros dado que el juego no tenía todas las ventajas de cambios que el juego original.

Ahora en 2008 se ha creado un proyecto llamado EA Land en el que el juego de Los Sims Online se ha hecho gratuito.



**Figura 3. Inteligencia artificial en batallas épicas con “AGE OF EMPIRES”**

En una entrevista para la revista GameSpot UK el desarrollador Bruce Shelley creador de AGE OF EMPIRES nos explica sobre la inteligencia artificial en este juego.

GameSpot UK: ¿Qué hay nuevo en Age of Empires II: Age of Kings. en cuanto al sistema de juego?

Decís que habéis mejorado la Inteligencia Artificial (AI) ¿Cómo exactamente?

Bruce Shelley: Hemos reescrito completamente las rutinas de IA. Usamos "Expert Systems", la IA será experta en el sentido de cómo responde a lo que está ocurriendo en el juego. Seguirá varios cientos de reglas, y está programada para hacer ciertas cosas bajo ciertas condiciones de juego. Así, por ejemplo, al principio del juego puede estar programada para ir a la siguiente edad tan rápidamente como pueda. Cuando esté allí, podría tomar una decisión sobre si su próxima meta será atacar a alguien o ir a la siguiente edad, por ejemplo, y lo que hará dependerá en parte de lo que haya encontrado al explorar el mapa. Si ha encontrado un enemigo muy cerca, puede tomar la decisión de atacarle. O, si no lo ha encontrado, puede pensar: "Bien, estoy relativamente seguro" y tomar la decisión de concentrarse en avanzar de edad y construir una fuerza militar. La idea general es que jugará su juego basándose en lo que ha aprendido al explorar el mapa, bastante parecido a como lo haría un jugador humano.

Cómo actúe dependerá también de que civilización tiene; cada civilización tendrá ventajas y debilidades. Entonces, a medida que el juego progresa, los programas de IA la llevarán por diversas bifurcaciones. Por ejemplo, podría planear construir una Maravilla: recogerá los recursos necesarios y levantará una muralla defensiva. Pero antes de que termine su Maravilla, yo construyo una. Ahora, la IA se da cuenta de que nunca me alcanzará y que mi Maravilla estará terminada antes que la suya.

Entonces para de trabajar con la maravilla, cambia la lista de edificios a construir y concentra su atención en destruir mi maravilla.

Queremos hacer la IA más como un jugador humano, de forma que sientas que estás jugando contra alguien listo.

Ahora como punto final serraremos con broche de oro.

### 3.2 Warcraft 3 y Dota



Figura 4. WARCRAFT 3 y DOTA con Inteligencia artificial

Al no encontrar mucho material sobre la Ai en warcraft o dota la mayor parte de este punto va por mi cuenta y por mi experiencia propia, así que disculpen si omito algo o cometo algún error.

Warcraft III Reign of Chaos es un videojuego de estrategia creado por Blizzard Entertainment y es la tercera parte de la serie Warcraft. Además de continuar la historia del mundo épico medieval de Warcraft se distingue de sus predecesores por incorporar dos importantes cambios: el paso a los gráficos 3D y la aparición de dos nuevas razas. Los elfos nocturnos (nightelf) y los muertos vivientes (undead). Warcraft III es uno de los juegos de estrategia más laureados por la comunidad de Gamers y es reconocido como uno de los mejores en la historia. Warcraft fue lanzado un 3 de julio de 2002 y la expansión fue lanzada en julio de 2003.

El mapa original de DotA fue creado por un gamer y creador de mapas conocido como "Eul" mediante el editor de mapas de Warcraft III: Reign of Chaos. Luego del lanzamiento de la expansión The Frozen Throne, que añadió nuevas características al editor de mapas, Eul no actualizó el escenario, por lo que fueron otros autores quienes se encargaron de realizar dicha actualización, y actualizar posteriormente el juego.

Numerosas variantes del juego se han realizado basadas en el mapa original, pero Allstars es la versión que volvió popular el juego. Este escenario fue desarrollado por un autor bajo el alias "Guinsoo". A partir de la versión 6.x, otro autor con el nombre de "IceFrog" se puso a cargo del desarrollo del escenario, corrigiendo errores y añadiendo o actualizando características del juego.

El juego tiene un fuerte apoyo comunitario, mantenido mediante foros oficiales. Allí, los usuarios pueden comentar ideas para nuevos héroes u objetos, e indicar sugerencias de todo tipo. El mapa puede obtenerse de forma gratuita en diversos sitios de descarga, y en Battle.net mediante transferencia directa al iniciar una partida. Al igual que cualquier escenario creado con el

editor de mapas de Warcraft III, la distribución de DotA no puede realizarse con fines comerciales de ningún tipo. En el juego se puede apreciar el tributo a estos dos gamer con dos items que tienen sus nombres. Algunas versiones del escenario con oponentes controlados por inteligencia artificial han sido desarrolladas.

Estas versiones no son oficiales sino que fueron creadas por aficionados al juego, se puede ver versiones que van desde algoritmos simples, hasta otras con la Ai bien estructurada. Una de estas y la que analizaremos es la del Gamer "RGB".

Quien creo varios mapas de AI, el ultimo realizado es el DotaAllstars 6.54Ai\_v1.14. Este mapa tiene un AI muy bien elaborado que va desde pequeños detalles, hasta grandes algoritmos, que crean un Ai que puede ser comparadas con un gamer de a nivel medio, el ai contiene 3 niveles de dificultad que va desde fácil, normal, y demente.

La descarga de este Ai esta disponible en [www.getdota.com](http://www.getdota.com) y nos vamos a la pestaña AI maps.

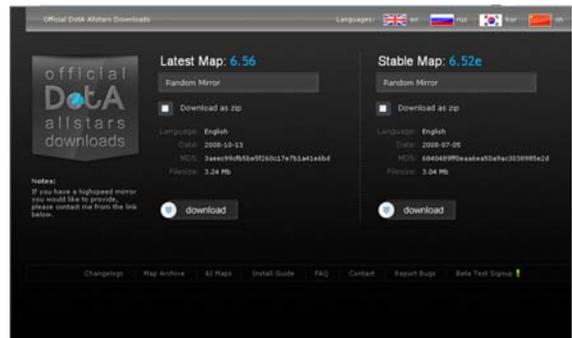


Figura 5. Página oficial de descarga de DOTA

## 4. APLICACIÓN

Como ya había dicho en una de las aplicaciones de las aplicaciones de la inteligencia artificial en juegos que les voy a presentar es el clásico juego de tres en raya creado por Francisco I. Leyva y descargado en <http://www.panchosoft.com>.

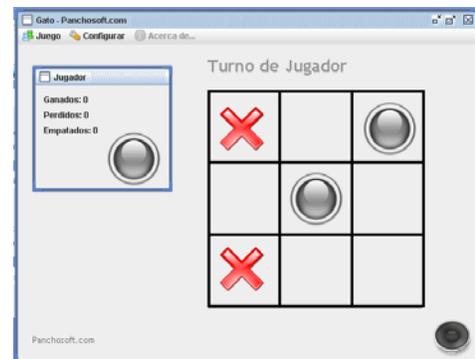


Figura 6.

Este juego fue creado en J# con el algoritmo MINMAX, la implementación del código encuentra en anexos de esta revista.

Pero ahora daré una breve reseña de como funciona:

- Nro. de jugadores: 1 jugador que juega con la persona o 2 personas.
- El algoritmo MINMAX esta dado para que nunca se pueda ganar, se empata o se pierde.
- Incluye estadísticas de los puntajes.

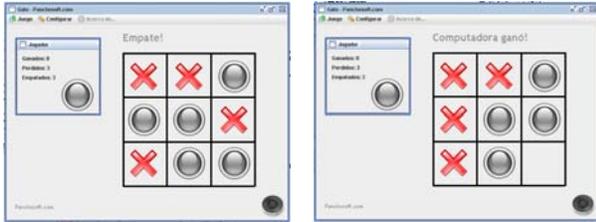


Figura 7.

## 5. CONCLUSIÓN

En conclusión podemos decir que la mayor aplicación de la IA es en juegos duela a quien le duela y que cada vez la comunidad de gamer se va asiendo más y más grande.

Por eso hay que tomar conciencia que los juegos forman una gran parte de la sociedad y que aumentan la inteligencia los reflejos y que familiariza a los niños y jóvenes con el mundo de las tecnologías.

## 6. BIBLIOGRAFÍA

- [1] Universidad Católica Santo Torino De Mogrovejo. APLICACIONES DE LA INTELIGENCIA ARTIFICIAL, Disponible En: [http://www.emagister.com/Monografia Introducción a la inteligencia artificial Gratis - emagister\\_com.htm](http://www.emagister.com/Monografia%20Introducci3n%20a%20la%20inteligencia%20artificial%20Gratis%20-%20emagister_com.htm). Leído El: 28 De Octubre De 2008.
- [2] Teoría de Juegos con Inteligencia Artificial: Ajedrez, <http://www.seccperu.org/>, leído en: 28 de octubre de 2008.
- [3] Cursos de video juegos, disponible en, <http://www.mailxmail.com/curso/informatica/cursodevideojuegos>, leído el 1 de noviembre de 2008.
- [4] Historia de la inteligencia artificial, disponible en , <http://es.wikipedia.org>, leído el, 1 de noviembre de 2008.
- [5] Orígenes de la inteligencia artificial, disponible en, <http://www.biografica.info/>, leído el 1 de noviembre de 2008