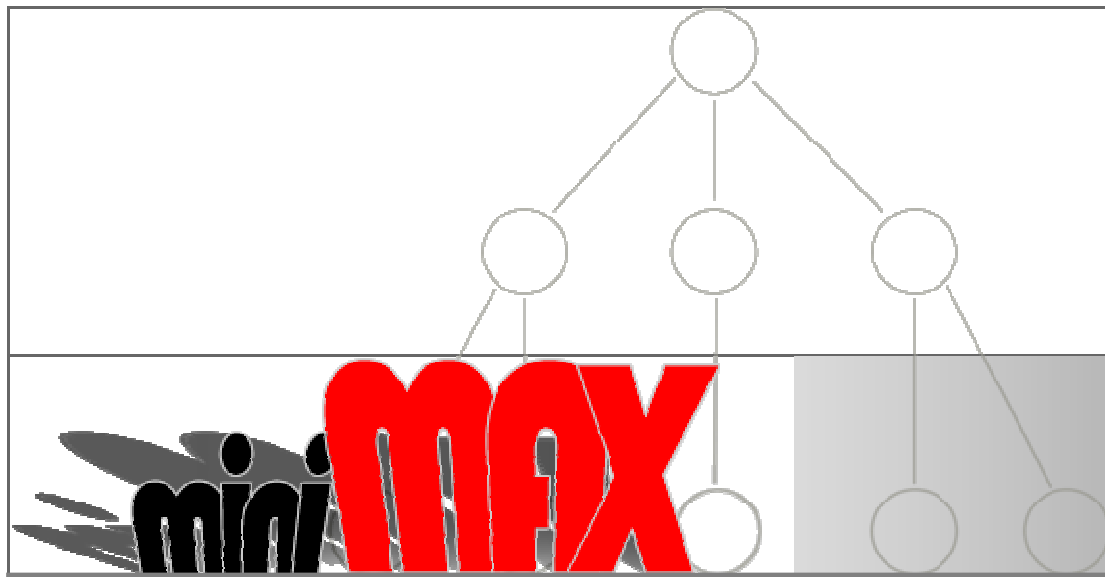


# INSTITUTO TECNOLÓGICO DE NUEVO LAREDO

Ingeniería en Sistemas Computacionales

Inteligencia Artificial

Ing. Bruno López Takeyas



|                                |          |
|--------------------------------|----------|
| Arturo Alejandro Castro        | 01100157 |
| Nadia Alejandro Castro         | 01100158 |
| Graciela Teresa García Vizcaya | 01100214 |
| Maria de los Ángeles Hernández | 01100227 |
| Maria Luisa Torres Lara        | 01100310 |

Nuevo Laredo, Tam., a 17 de Octubre del 2005.

## Introducción

Los juegos han ocupado la atención de las facultades intelectuales del ser humano, en ocasiones en grado alarmante. Juegos de tablero, como el ajedrez y el Go, en parte son interesantes debido a que en ellos se libra una lucha pura, abstracta, sin tener que pasar los trabajos y penas que implica organizar dos ejércitos y, con ellos, librar batallas. Esta característica de abstracción es lo que hace atractivos a los juegos para la Inteligencia Artificial.

Al hablar de estrategias de búsqueda de soluciones, consideramos al algoritmo Minimax como uno de los más importantes y de éste existen muchas variaciones donde intervienen diversos factores dependiendo de la naturaleza del juego y sus reglas.

A continuación se verá una explicación de ¿qué es el algoritmo Minimax? Y así como la estructura utilizada, sus funciones y una descripción de su funcionamiento.

También se verá de manera mas clara este funcionamiento aplicando el algoritmo en un juego imaginario.

Más adelante se encuentra una sección donde se muestran ejemplos de juegos donde se aplica el algoritmo Minimax.

## Reseña histórica

### Teoría de juegos

Para hablar de la historia de minimax hay que empezar con la teoría de juegos, por que de ahí se deriva este algoritmo de búsqueda para juegos de dos participantes.

La teoría de juegos o solo se refiere a juegos de tablero o de cartas, puede referirse a situaciones donde hay algo que negociar, o inclusive la guerra.

Un juego puede ser cualquier situación en la que intervienen dos actores o jugadores, en donde los intereses de los mismos están interconectados o interdependientes.

Los jugadores, se asume que son racionales ya que buscan maximizar sus utilidades, esto quiere decir que en un juego un jugador tratará de actuar de la manera que más le convenga y vaya de acuerdo con sus intereses, y de igual manera lo hará su oponente.

La teoría de juegos es usada incluso por los militares para resolver problemas de logística o tácticos. Pero la teoría de juegos va mucho más allá, puede ser considerada descriptiva y positiva, capaz de explicar y predecir. Puede aplicarse a conducta humana, conducta social, economía, milicia o política.

En teoría de juegos se consideran pioneros a Emile Borel, Oskar Morgenster y John Von Neumman, este último es el creador del teorema Minimax.

La teoría de juegos tiene un teorema fundamental, que es el teorema de Minimax cuyo creador es John Von Neumman.

### Teorema Minimax

John Von Neumman, en un famoso artículo publicado en 1928, presento una solución aplicable a una amplia clase de juegos. Considérese el juego que se ejemplifica en la figura de abajo, que involucra a dos jugadores. Las interacciones que realizan son conocidas como juegos de suma cero, por que la suma de los puntos de cada resultado es igual a cero. John Von Neumman estudio estos juegos y probó que pueden ser resueltos en la misma manera.

|    | C1   | C2   | C3     |
|----|------|------|--------|
| R1 | -2,2 | 1,-1 | 10,-10 |
| R2 | -1,1 | 2,-2 | 0,0    |
| R3 | -8,8 | 0,0  | -15,15 |

La solución propuesta por Von Neumman esta basada en la idea de que cada jugador debe maximizar el valor del peor resultado posible. De esa manera:

- Si R escoge R1, el peor resultado es -2, que ocurre cuando C juega con C1.
- Si escoge R2, el peor resultado es -1 cuando C escoge C1.
- Finalmente, el peor resultado para R si escoge R3 es -5 cuando C opta por C3.

De estos malos resultados, el mejor de R es el -1 que se obtiene cuando juega con R1. Esto es llamado la estrategia maximin de E, esto es, R maximiza su mínimo. Debido a esto, la estrategia maximin de R es dada por R1 con un resultado de -1. Investigando las opciones de C para la misma perspectiva, obtenemos la estrategia C1 como su maximin con un resultado de +1.

Hay que hacer notar que la suma de los valores maximin de los dos jugadores es igual a cero. Debido a esto el teorema de John Von Neumman esta en todos los juegos de suma cero, esto es en cada juego de dos personas con interacción de suma cero, la suma de los valores maximin de los jugadores es igual a cero.

La razón por la que Von Neumman pensó que los jugadores podían racionalmente escoger ser tan pesimistas esta relacionada con su teorema. Hay que recordar que en un juego de suma cero tu triunfo es el fracaso del otro y viceversa. Ser pesimista significa, en el contexto de los juegos de suma cero que el jugador es realista y que esta conciente de el hecho de que su oponente siempre tratará de hacer el máximo daño posible a él. ¿Por qué? , por la razón de que el triunfo de uno significa la pérdida del otro.

John Von Neumman estaba convencido de que ningún jugador racional jugaría por menos de lo que le permita tener el resultado maximin mejor, y que si los dos jugadores estaban determinados a evitar un resultado debajo de su maximin, ninguno esperarían recibir o trataría de recibir más que su resultado maximin.

## Teoría del Equilibrio

Hay que hacer notar que la solución maximin es la misma que el la teoría del equilibrio de Nash, ya que la suma de los valores maximin de los jugadores debe ser igual a cero. Siempre habrá un positivo y un negativo.

John Nash a los 21 años escribió una tesis de menos de treinta páginas en la que expuso por primera vez su solución para juegos estratégicos no cooperativos, lo que desde entonces se llamó "el equilibrio de Nash", que tuvo un inmediato reconocimiento entre todos los especialistas.

El punto de equilibrio de Nash es una situación en la que ninguno de los jugadores siente la tentación de cambiar de estrategia ya que cualquier cambio implicaría una disminución en sus pagos. Von Neumann y Oskar Morgenstern habían ya ofrecido una solución similar pero sólo para los juegos de suma cero. Para la descripción

formal del problema y su solución, Nash utilizó funciones de mejor respuesta y el teorema del punto fijo de los matemáticos Brouwer y Kakutani.

En los años siguientes publicó nuevos escritos con originales soluciones para algunos problemas matemáticos y de la teoría de juegos, destacando la "solución de regateo de Nash" para juegos bipersonales cooperativos. Propuso también lo que se ha dado en llamar "el programa de Nash" para la reducción de todos los juegos cooperativos a un marco no cooperativo.

## Claude Shannon

Claude Shannon es considerado el principal en utilizar el concepto de Minimax, además de la teoría de información y la negentropía.

Claude Elwood Shannon (30 de abril de 1916 (Michigan) - 24 de febrero de 2001) recordado como "el padre de la teoría de la información".

Los primeros años de su vida los pasó en Gaylord, donde se graduó de la secundaria en 1932. Desde joven, Shannon demostró una inclinación hacia las cosas mecánicas. Resaltaba respecto a sus compañeros en las asignaturas de ciencias. Su héroe de la niñez era Edison, a quien luego se acercó bastante en sus investigaciones.

En 1932 ingresó en la Universidad de Michigan, siguiendo a su hermana Catherine, Doctora en matemáticas. En 1936 obtuvo los títulos de ingeniero eléctrico y matemático. Su interés por las matemáticas y la ingeniería continuó durante toda su vida.

En 1936 aceptó la posición de asistente de investigación en el departamento de ingeniería eléctrica en el Instituto de Tecnología de Massachusetts (MIT). Su situación le permitió continuar estudiando mientras trabajaba por horas para el departamento, obteniendo como resultado la calculadora más avanzada de esa era.

En ese momento surgió su interés hacia los circuitos de relevadores complejos, sumado a su gusto por la lógica y el álgebra booleana. Estos nuevos intereses pudo desarrollarlos durante el verano de 1937, que pasó en los laboratorios Bell en la ciudad de Nueva York.

En su tesis doctoral en el M.I.T., demostró cómo el álgebra booleana se podía utilizar en el análisis y la síntesis de la conmutación y de los circuitos digitales. La tesis despertó un interés considerable cuando apareció en 1938 en las publicaciones especializadas. En 1940 le fue concedido el Premio a ingenieros americanos del *Instituto Americano Alfred Nobel* de Estados Unidos, una concesión dada cada año a una persona de no más de treinta años. Un cuarto de siglo más tarde H. H. Goldstine, en su libro "Las computadoras desde Pascal hasta Von Neumann", citó su tesis como una de las más importantes de la historia... que ayudó a cambiar el diseño de circuitos digitales.

Durante el verano de 1938 realizó trabajos de investigación en el M.I.T. y le fue concedida la beca Bolles cuando trabajaba como ayudante de enseñanza mientras realizaba un doctorado en matemáticas.

En 1940 estudió un master en ingeniería eléctrica y se doctoró en filosofía matemática.

Shannon pasó quince años en los laboratorios Bell, una asociación muy fructífera con muchos matemáticos y científicos de primera línea como Harry Nyquist, Brattain, Bardeen y Shockley, inventores del transistor; George Stibitz, quien construyó computadoras basadas en relevadores y muchos otros más.

Durante este período Shannon trabajó en muchas áreas, siendo lo más notable todo lo referente a la teoría de la información, un desarrollo que fue publicado en 1948 bajo el nombre de "Una Teoría Matemática de la Comunicación". En este trabajo se demostró que todas las fuentes de información (telégrafo eléctrico, teléfono, radio, la gente que habla, las cámaras de televisión, etc, ... ) se pueden medir y que los canales de comunicaciones tienen una unidad de medida similar. Mostró también que la información se puede transmitir sobre un canal si, y solamente si, la magnitud de la fuente no excede la capacidad de transmisión del canal que la conduce, y sentó las bases para la corrección de errores, supresión de ruidos y redundancia.

En el área de las computadoras y de la inteligencia artificial, publicó en 1950 un trabajo que describía la programación de una computadora para jugar ajedrez, convirtiéndose en la base de posteriores desarrollos.

A lo largo de su vida recibió numerosas condecoraciones y reconocimientos de universidades e instituciones de todo el mundo.

Shannon debe ser considerado dentro de los personajes más importantes del siglo XX, aunque su nombre es desconocido para el público en general.

Claude Shannon falleció el 24 de febrero del año 2001, a la edad de 84 años, después de una larga lucha en contra la enfermedad de Alzheimer.

En 1950 Claude Shannon publicó un artículo de ajedrez en computadora titulado "Programando una Computadora para que Juegue Ajedrez". Describiendo como una máquina o computadora puede hacerse que juegue racionalmente un juego de ajedrez. El proceso en el cual hacía que la computadora decidiera que movida realizar es un proceso minimax, basado en una función de evaluación de una posición de ajedrez dada. Shannon dio un ejemplo a grandes rasgos de una función de evaluación donde el valor de una posición negra era sustraído al de una posición blanca. El Material era contado de acuerdo a el valor relativo usual de las piezas de ajedrez, 1 punto para los peones, 3 puntos para los caballos o alfiles, 5 puntos para las torres y 9 puntos para la reina. El considero algunos factores posicionales, sustraer .5 de cada peón doble, o cada peón que va hacia atrás, o cada peón que esta aislado. Otro factor posicional en la función de evaluación era la movilidad, agregar 0.1 puntos a cada movida legal disponible. Finalmente considero el valor artificial de 200 puntos al rey.

Shannon explicó que esta formula era solo a grandes rasgos y que había muchas otras cosas que considerar e incluir, pero explicó el valor de 200 puntos para el rey o el jaque mate de la siguiente manera:

Debido a que 200 puntos es un valor mayor que la suma máxima de todos los otros valores, el procedimiento minimax valorará al jaque mate arriba de cualquier otra cosa y sacrificará todo el Material posible para prevenir ese jaque mate. El valor es arbitrario a final de cuentas, cualquier número mayor que la suma de todos los otros valores estaría bien.

## Algoritmo Minimax

En teoría de juegos Minimax es un método de decisión para minimizar la pérdida máxima esperada en juegos con adversario y con información perfecta. Minimax es un algoritmo recursivo.

### Teoría de juegos

La **teoría de juegos** es un área de las matemáticas que utiliza modelos para estudiar interacciones en estructuras formalizadas de incentivos (los llamados *juegos*). Sus investigadores estudian las estrategias óptimas así como el comportamiento previsto y observado de individuos en juegos. Tipos de interacción aparentemente distintos pueden, en realidad, presentar estructuras de incentivos similares y, por lo tanto, representar conjuntamente un mismo juego.

### ¿Qué es?

Minimax es un algoritmo recursivo y dicha recursión se da por alguna de las situaciones siguientes:

- Gana algún jugador.
- Se han explorado N capas, siendo N el límite establecido.
- Se ha llegado a un situación estática donde no hay grandes cambios de un nivel a otro

El espacio de estados se representa mediante árboles alternados, donde:

- **Nodo.** Representa una situación del juego.
- **Sucesores de un nodo.** Situaciones del juego a las que accede por movimientos legales aplicando sus reglas.
- **Nivel.** Contiene todas las situaciones posibles para uno de los jugadores.

En árbol Minimax es lo mismo que un **árbol de juego**. De hecho no es ni siquiera un árbol es en realidad un algoritmo que es usado en un árbol de juego.

El algoritmo Minimax es una manera muy clara de transformar un árbol de juego en datos que una computadora puede analizar para que haga una decisión inteligente acerca de cual es la movida que es mejor en el momento.

El algoritmo Minimax es diseñado para dos jugadores: **Min** y **Max**. El algoritmo trabaja como sigue: Max empieza y por lo mismo tiene la posición dominante y será el agresor en el juego. Cada vez que haga una movida, el escogerá la mejor movida para si mismo. Min esta a la defensiva, y cada vez que haga una movida tratara de poner a Max en la peor posición posible.

El procedimiento Minimax es un procedimiento de búsqueda hacia adelante.

Suponga que tenemos un analizador de situaciones que convierte todos los juicios acerca de las situaciones de tablero en un numero simple de calida global. Suponga también que los numeros positivos, por convención, indican la ventaja de un jugador y los negativos la ventaja del otro. El grado de ventaja aumenta con el valor absoluto del número.

El proceso de cálculo de un número que refleje la calidad de tablero se conoce como evaluación estática. El procedimiento que hace el cálculo es un evaluador estático y el número que calcula se conoce como resultado de la evaluación estática.

El jugador que espera obtener numeros positivos se conoce como jugador de maximización o maximizador. El otro jugador recibe el nombre de jugador de minimización o minimizador.

### Estructuras utilizadas

Un árbol de juegos es una representación, o sea, un árbol semántico en el que

- Los nodos representan configuraciones de tablero;
- Las ramas representan jugadas.

### ¿Para que sirve?

En juegos bipersonales el algoritmo más usado es el denominado algoritmo Minimax.

Como ya mencionamos que los dos jugadores se participantes se llaman Max y Min el algoritmo minimax sirve para determinar la estrategia optima para Max, y decidir así cual es la mejor jugada.

## funcionamiento

### Planteamiento general

- 2 jugadores: MAX y MIN (MAX mueve primero):
- Estado inicial
  - Posición del tablero e identificación del primer jugador a mover
- Función sucesora:
  - Lista de pares (movimiento, estado) indica cada movimiento legal y su estado resultante
- Función objetivo:
  - Determina cuándo se acaba el juego (en nodos objetivo o terminales)
- Función de utilidad (función u):
  - Definida en nodos terminales (valores numéricos)
  - Resultado del juego. Por ejemplo:
    - +1 si gana MAX
    - -1 si gana MIN
    - 0 si empate (tablas)
- OBJETIVO. Tiene por objetivo decidir un movimiento para MAX.
- HIPÓTESIS
  - ♦ Jugador MAX trata de maximizar su beneficio (función de utilidad).
  - ♦ Jugador MIN trata de minimizar su pérdida.
- MINIMAX-VALUE(n) =
  1. UTILITY(n) Si n es un nodo terminal
  2.  $\max_s \rightarrow \text{Sucesor}(n) \text{ MINIMAX-VALUE}(s)$   
Si n es un nodo MAX
  3.  $\min_s \rightarrow \text{Sucesor}(n) \text{ MINIMAX-VALUE}(s)$   
Si n es un nodo MIN

## Pasos del algoritmo Minimax

El algoritmo minimax trabaja de la siguiente manera:

2. Generar árbol entero hasta nodos terminales
3. Aplicar función  $u$  a nodos terminales

Va hacia abajo hasta cada nodo hoja y analiza el estado del juego en ese punto. Entonces utilizan un algoritmo heurístico para producir un número. Un número alto significa que el estado es bueno para Max y malo para Min, y un número bajo significa que el estado es malo para Max y bueno para Min.

**Heurística:** En computación, dos objetivos fundamentales son encontrar algoritmos para la mayoría de casos buenos tiempos de ejecución y buenas soluciones, usualmente las óptimas. Una heurística es un algoritmo que ofrece uno o ambos objetivos; por ejemplo, normalmente encuentran buenas soluciones, aunque en ocasiones no hay pruebas de que la solución no pueda ser arbitrariamente errónea; o se ejecuta razonablemente rápido, aunque no existe tampoco prueba de que deba ser así.

El algoritmo analiza todos los nodos hojas del árbol de juego. El estado donde el jugador Max pierde se hace cero o  $-\infty$ , y los estados donde el jugador Min se hacen uno o  $\infty$ , o el valor que indique empate de acuerdo a las reglas del juego. También pudiera recibir otros valores heurísticos.

4. Propagar hacia arriba para generar nuevos valores de  $u$  para todos los nodos
  - minimizando para MIN
  - maximizando para MAX

Después de haber sido completado, el algoritmo Minimax hace una búsqueda backtracking a través del árbol. Siempre que sea el turno de Min, ella selecciona la movida que tenga la calificación mas baja, siempre que sea el turno de Max, el selecciona la movida con la que tenga la calificación mas alta.

5. Elección jugada con máximo valor de  $u$

Finalmente, el nodo raíz es evaluado. Max necesita escoger el nodo con el valor más grande.

## Decisiones imperfectas en juegos de dos adversarios

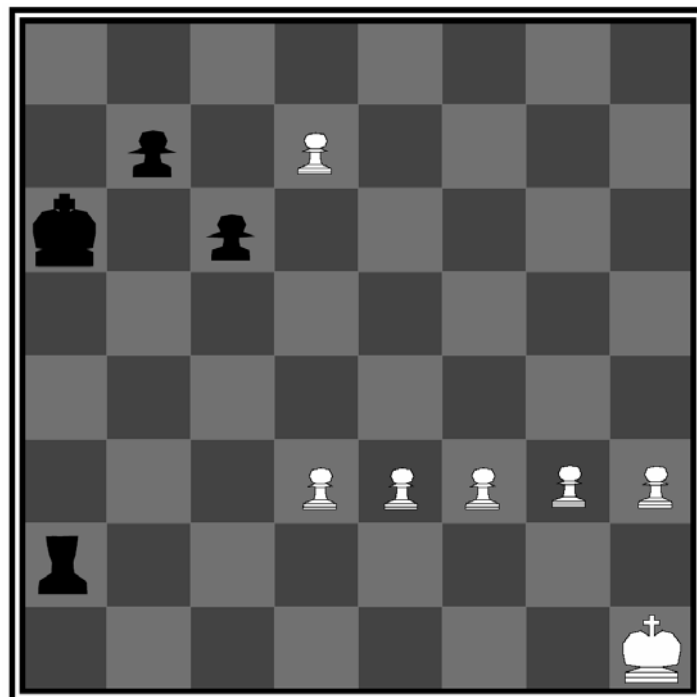
Dada una función de evaluación  $f$ , se puede aplicar una búsqueda *minimax* con límite de profundidad:

- Se elige un límite de profundidad
  - Observación: el límite puede tener una posición desventajosa en un nivel más abajo.
- Se pueden elegir sucesivos límites de profundidad.
- El límite de profundidad se debería aplicar sólo a posiciones "inactivas".
  - En ajedrez, serían por ejemplo posiciones en las que es poco probable que existan capturas

### Problema del horizonte

Surge cuando el programa se enfrenta a una acción del oponente, inevitable y que causa serios perjuicios.

Ejemplo: en la figura anexa, peón blanco amenaza convertirse en dama. Torre negra amenaza con jaque. La ventaja actual es negra y la inmediata futura es blanca (evaluación calidad piezas).



Black to move

## Exploración y evaluación

El procedimiento de exploración visto separa por completo el proceso de generación del árbol de exploración y la evaluación de posiciones.

Se puede reducir el esfuerzo requerido si se hace evaluación de los nodos finales y se llevan hacia atrás esas evaluaciones con la generación el árbol

## Función de evaluación para el juego del gato

- Se define la función de evaluación:

$$f(s) = NMAX(s) - NMIN(s)$$

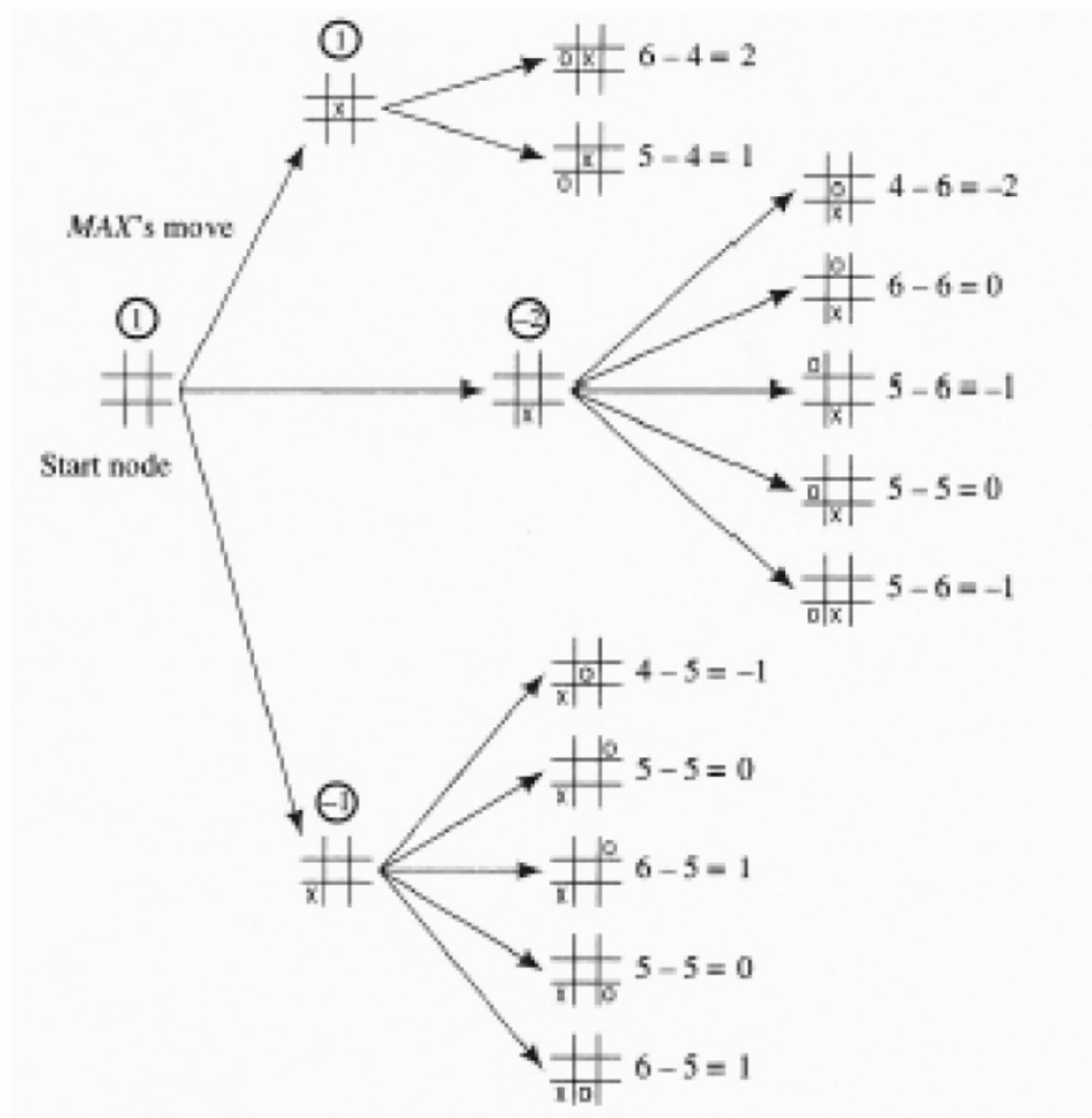
donde:

**S:** Situación o distribución del tablero

**f(s):** Función de evaluación del tablero (nodo del espacio de estados)

**NMAX(s):** No. de filas, columnas o diagonales abiertas para MAX (donde aún puede ganar)

**NMIN(s):** No. de filas, columnas o diagonales abiertas para MIN (donde aún puede ganar)



## Juegos en los que aplica el algoritmo Minimax

A continuación se presentan los juegos en los que puede aplicarse el algoritmo Minimax en su implementación en computadora.

### Ejemplos de juegos:

Gato, conecta cuatro, damas, ajedrez, Go, Nim y Othello

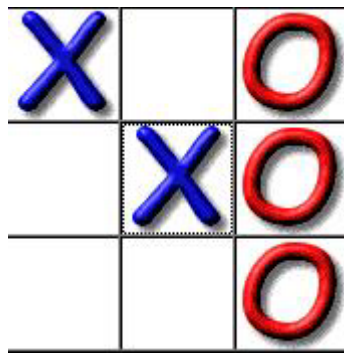
### Juegos que no aplican búsqueda de soluciones

Bridge, Solitario, Backgammon

#### ♦ Tic-tac-toe

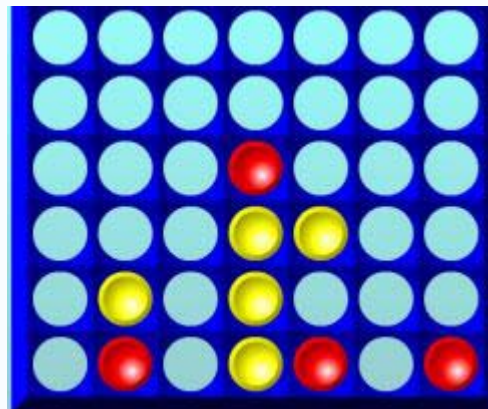
**9! = 362880 partidas**

Es un juego de dos jugadores y se juega en un tablero de 3X3. Un jugador juega con las X y otro con O y gana aquel que logre unir 3 de sus marcas en fila, horizontal, vertical o diagonal. Si depende de la estrategia y de la habilidad del jugador pero se conocen todos los movimientos posibles.



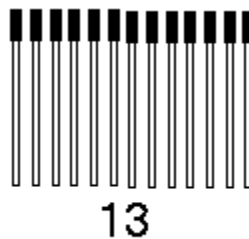
### ♦ Conecta cuatro

Conecta cuatro es un juego de mesa de dos jugadores en donde el objetivo es ser el primero en tener cuatro de sus fichas conectadas en una línea horizontal, vertical o diagonal. El juego es jugado en un tablero de 7 columnas y 6 filas, el tablero es puesto en posición vertical. Cada jugador tiene 21 discos de un solo color. Los jugadores se toman turnos para poner uno de sus discos en una columna vacía. El disco cae hasta el espacio desocupado mas bajo del tablero. Puede llegar el juego a empatar si el tablero se llena completamente y nadie ha conectado cuatro de sus discos en una línea. Es un juego de dos jugadores, se saben los posibles estados, se puede llegar a un empate y depende de las movidas del jugador y no del azar.



### ♦ Rocas o Nim

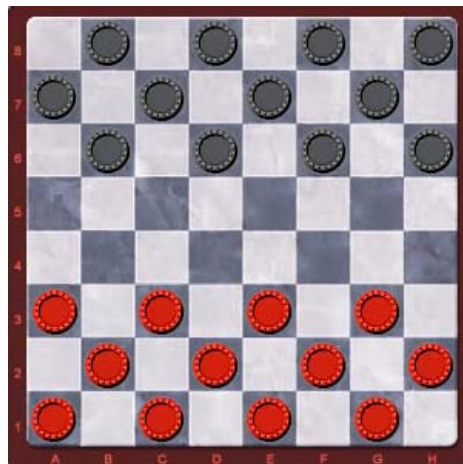
Es un juego de dos jugadores. En este caso consiste en que tenemos 13 cerillos (pueden ser piedras, monedas, etc). Cada jugador tiene su turno y puede tomar 1,2 o 3 cerillos. El jugador que tome el último cerillo, gana.



### ♦ Damas (Checkers)

$10^{31}$  partidas

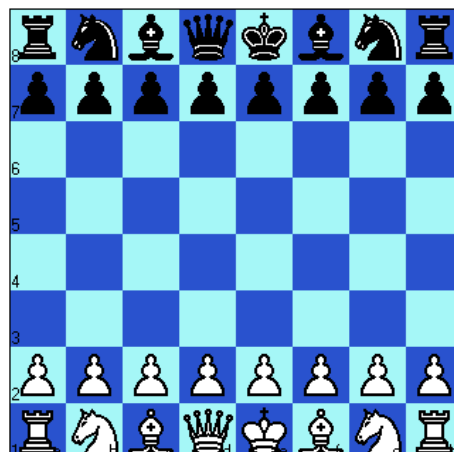
Es un juego para dos jugadores. Se juega en un tablero de 8X8, con cuadrados oscuros y claros, y cada tablero debe empezar con el cuadrado oscuro en la esquina izquierda de cada jugador. Las piezas se mueven solamente en los cuadros oscuros. Es parecido al ajedrez pero más sencillo ya que todas las piezas tienen el mismo movimiento y gana aquel que logre capturar las piezas de su oponente. Se conocen todos los posibles movimientos y depende de las habilidades de ambos jugadores, se puede llegar a un empate.



### ♦ Ajedrez (Chess)

$10^{123}$  partidas

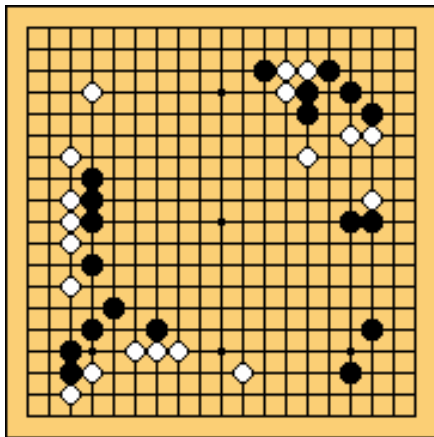
El ajedrez es un juego de dos jugadores, donde a un jugador se le asignan piezas blancas y al otro negras. Cada jugador dispone de 16 piezas al empezar el juego: un rey, una dama o reina, dos torres, dos alfiles, dos caballos y ocho peones. El objetivo del juego es capturar al rey del otro jugador. La captura no se completa nunca, pero una vez que el rey es atacado y no puede escapar de esa captura, se dice que es un jaque mate y el juego finaliza.



## ♦ Go

**$10^{360}$  partidas**

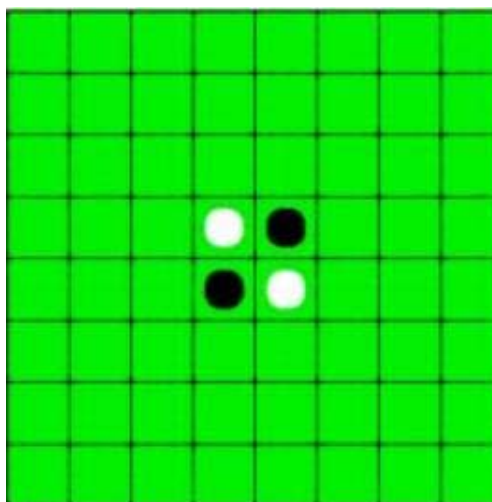
Es un juego de estrategia, de dos jugadores originado en la Antigua China. Es clasificado como un juego de mesa abstracto y se dice que se asemeja a la guerra. Los jugadores, blanco y negro, pelean para maximizar el territorio que controlan, buscando rodear grandes áreas del tablero con sus piedras, para capturar a sus oponentes que invaden esas áreas y para proteger a sus propias piedras de ser capturadas. Las estrategias pueden llegar a ser sutiles y sofisticadas. Las guerras verdaderas terminan cuando los participantes llegan a un acuerdo. En go también, los jugadores deben llegar a un acuerdo de que el juego ha terminado.



## ♦ Othello

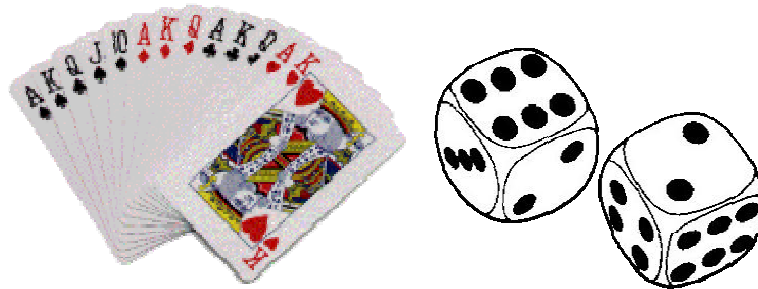
Algoritmo alfa-beta para búsqueda en juegos

Para este juego se utiliza un tablero 8x8 en el que dos jugadores (colores blanco y negro) van colocando fichas alternativamente. Inicialmente están ocupadas las 4 casillas centrales como se muestra en la figura



## JUEGOS QUE NO APLICAN BUSQUEDA DE SOLUCIONES

Los juegos que se juegan con una baraja no aplican el algoritmo Minimax, ya que se requiere que la baraja se revuelva para repartirla y no conocemos sus posibles estados que dependen del azar, así que depende de la suerte más que de la habilidad del jugador o jugadores. Así también en todos los juegos de mesa donde se utilicen dados para llevar a cabo jugadas, ese juego es impredecible y depende absolutamente del azar.

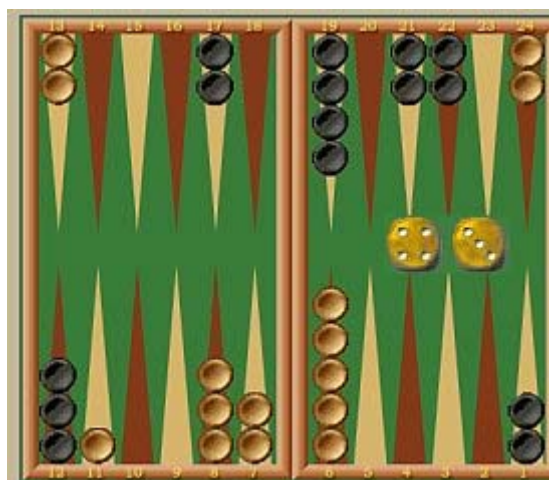


### ♦ Solitario

Es un juego de un solo jugador. También tiene incertidumbre ya que depende de cómo se barajan las cartas. Requiere también destreza de parte del jugador, pero también depende en que orden las cartas se acomodan (ya sea boca abajo o boca arriba).

### ♦ Backgammon

Es un juego de dos jugadores con incertidumbre. Los jugadores tiran un dado para determinar que movidas hacer. Estos juegos son buenos para explorar hacer desiciones en problemas con adversario que involucran suerte y destreza.



## Bibliografía

- **Data Structures for Games Programmers**  
Ron Penton  
Bivitec  
Capítulo 15
- **Inteligencia Artificial**  
Un enfoque moderno  
Stuart Rusell-Peter Norvig  
Prentice Hall
- **Game Theory: A Critical Text**  
Shaun Hargreaves Heap  
Manis Varoufakis
- **Game Theory: Concepts and Aplications**  
Sage University Papers Series  
Quantitative Applications in the Social Sciences  
No. 07-041
- **Inteligencia artificial** : Una nueva síntesis  
Nilsson N.  
Capítulos 7, 8, 9, 12
- **<http://www.ii.uam.es/~fdiez/docencia/04-05/IA/juegos.ppt>**
- **"<http://es.wikipedia.org/wiki/Minimax>"**
- **AI-Search Algorithm Animation Project (RMIT University)**  
Applet que demuestra el funcionamiento de los algoritmos de búsqueda estudiados en clase.  
<http://www.cs.rmit.edu.au/AI-Search>

# Índice

|                                                            |           |
|------------------------------------------------------------|-----------|
| <b>Introducción .....</b>                                  | <b>2</b>  |
| <b>Reseña histórica .....</b>                              | <b>3</b>  |
| Teoría de juegos .....                                     | 3         |
| Teorema Minimax .....                                      | 3         |
| Teoría del Equilibrio.....                                 | 4         |
| Claude Shannon .....                                       | 5         |
| <b>Algoritmo Minimax .....</b>                             | <b>8</b>  |
| Teoría de juegos .....                                     | 8         |
| ¿Qué es? .....                                             | 8         |
| Estructuras utilizadas .....                               | 9         |
| ¿Para que sirve? .....                                     | 9         |
| <b>funcionamiento .....</b>                                | <b>10</b> |
| Planteamiento general .....                                | 10        |
| Pasos del algoritmo Minimax.....                           | 11        |
| Decisiones imperfectas en juegos de dos adversarios .....  | 12        |
| Exploración y evaluación .....                             | 13        |
| Función de evaluación para el juego del gato .....         | 14        |
| <b>Juegos en los que aplica el algoritmo Minimax .....</b> | <b>16</b> |
| <b>Bibliografía .....</b>                                  | <b>21</b> |