

*INSTITUTO TECNOLÓGICO DE NUEVO LAREDO*

*INGENIERÍA EN SISTEMAS COMPUTACIONALES*

*INTELIGENCIA ARTIFICIAL*

*IMPORTE:* ING. BRUNO LÓPEZ TAKEYAS

*TEMA:* ALGORITMO MINIMAX

*EQUIPO1:*

*CENICEROS VAZQUEZ MARTHA DEYANIRA JAZMIN*

*RAMÍREZ RUIZ OCTAVIO*

*TORRES COLUNGA MARÍA DEL CARMEN*

*URIETA ACEVEDO BEATRIZ*

*VEGA HERNÁNDEZ MARGARITA ELIZABETH*

*NUEVO LAREDO, TAMAULIPAS., A 17 DE OCTUBRE DEL  
2005*

## CONTENIDO

|                                                                     |    |
|---------------------------------------------------------------------|----|
| INTRODUCCIÓN.....                                                   | 3  |
| DEFINICIÓN.....                                                     | 3  |
| REPRESENTACIÓN DE LOS JUEGOS                                        |    |
| Motivación para el estudio de los juegos .....                      | 4  |
| Características.....                                                | 4  |
| Elementos del juego.....                                            | 5  |
| Topologías del proceso de búsqueda .....                            | 5  |
| Búsqueda en el árbol.....                                           | 5  |
| Representación del conocimiento.....                                | 6  |
| EJEMPLO DE LA ESTRUCTURA DE UN ÁRBOL PARA EL ALGORITMO MINIMAX..... | 7  |
| PASOS DEL ALGORITMO MINIMAX.....                                    | 8  |
| CALCULO DE VALORES DE LA FUNCIÓN DE UTILIDAD.....                   | 9  |
| EJEMPLO DE APLICACIÓN.....                                          | 10 |
| BIBLIOGRAFIA.....                                                   | 16 |

# ALGORITMO MINIMAX

## INTRODUCCIÓN

Los juegos son uno de los campos de trabajo más antiguos de la IA.

En 1950, casi, en cuanto fue posible programar las computadoras, Claude Shannon, inventor de la teoría de la información Basado en la teoría de juegos de Von Neumann y O. Morgenstern, y Alan Turing elaboró el primer programa de ajedrez en 1951. Desde entonces, se han realizando continuos avances en el juego, al grado de que los sistemas actuales son capaces de desafiar a campeones mundiales humanos sin temor de que puedan hacer el ridículo.

Los primeros investigadores eligieron para su trabajo al ajedrez por varias razones. Una computadora capaz de jugar ajedrez seria la prueba viviente de una máquina que podía realizar algo para la que se consideraba era necesario tener inteligencia. Además, la sencillez de las reglas, y el hecho de que el programa pueda acceder totalmente al estado del mundo significa que es fácil representará Juego como una búsqueda a través de un espacio de posibles posiciones de juego.

La presencia de un oponente complica más el problema de decisión. El oponente introduce la *incertidumbre*, puesto que es imposible saber qué va a hacer. El contrincante se esforzará todo lo que pueda porque sus jugadas sean lo menos benignas para su oponente.

Pero lo que singulariza *realmente* a los juegos es que por lo común es excesivamente difícil resolverlos. El ajedrez, por ejemplo, tiene un factor de ramificación promedio de 35; las partidas frecuentemente llegan a 50 jugadas por cada contrincante, por lo que el árbol de búsqueda tiene 35100 nodos (aunque haya "sólo" aproximadamente 1040 posiciones legales *distintas*). La complejidad de los juegos trae aparejado un tipo de incertidumbre totalmente nuevo, no visto hasta ahora; la incertidumbre no es consecuencia de la información faltante, sino debido a que no hay tiempo suficiente para calcular las consecuencias exactas de una jugada. Sólo se intenta deducir lo que seria lo mejor con base en experiencias pasadas y proceder, sin estar completamente seguro de cuál seria la mejor acción. En este sentido, *los juegos se asemejan más al mundo real*.

## DEFINICIÓN

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

El funcionamiento de Minimax puede resumirse como elegir mejor movimiento para ti mismo suponiendo que tu contrincante escogerá el peor para ti.

El nombre del algoritmo deriva de considerar que, dada una función estática que devuelve valores en relación al jugador maximizante, éste procura maximizar su valor mientras que su oponente procura minimizarlo. En un árbol de juego donde los valores de la función estática están en relación al jugador maximizante, se maximiza y minimiza alternadamente de un nivel a otro.

El funcionamiento de Minimax puede resumirse a como elegir el mejor movimiento para ti mismo; suponiendo que tu contrincante escogerá el peor para ti.

Ofrece la siguiente gran ventaja matemática: supone que el adversario ha de jugar óptimamente pero no subóptimamente. Como hay muchísimas maneras de jugar mal y sólo una manera de jugar bien (aunque de vez en cuando puede haber dos jugadas indistinguibles en su ventaja comparativa), esta postura quita toda incertidumbre a lo que va a hacer el adversario.

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 se accede por movimientos legales aplicando sus reglas
- Nivel: Contiene todas las situaciones posibles para cada uno de los jugadores.

El algoritmo Minimax es un procedimiento recursivo y el corte de la recursión esta dado por las siguientes condiciones:

- A) Gana algún jugador
- B) Se han explorado N capas, siendo N el límite establecido
- C) Se ha agotado el tiempo de exploración
- D) Se ha llegado a una situación estática donde no hay grandes cambios de un nivel a otro.

## **REPRESENTACIÓN DE LOS JUEGOS**

### **MOTIVACION PARA EL ESTUDIO DE LOS JUEGOS**

- Crear programas de ordenador para jugar.
- Emular el razonamiento humano en un ordenador.
- Construir sistemas que sean capaces de tomar decisiones en un entorno adverso.

### **CARACTERISTICAS**

- Juegos bipersonales.
- Los jugadores mueven alternativamente.
- La ventaja para un jugador es desventaja para el otro.
- Los jugadores poseen toda la información sobre el estado del juego.
- Hay un número finito de estados y decisiones.
- No interviene el azar (dados, cartas).

## ELEMENTOS DEL JUEGO

### Jugadores:

- MIN: Humanos.
- MAX: Maquinas.

### Los juegos bipersonales se representan con:

- Estados iniciales. Que incluye la configuración inicial del tablero y una indicación del jugador que comienza moviendo.
- Un conjunto de operadores. Que definen los movimientos legales que los jugadores pueden realizar.
- Prueba terminal. que determina los casos en que el juego ha finalizado. Los estados en los que el juego ha finalizado constituyen el conjunto de estados terminales.
- Función de evaluación estática. Asigna un valor numérico al resultado del juego.

### Nodos:

- Tablero+jugador+valor

### Movimientos:

- Generar movimientos.
- Verificar el movimiento humano.
- Procedimiento de decisión de la maquina.

## TOPOLOGIAS DEL PROCESO DE BUSQUEDA

Dependiendo de la estructura que definan los operadores el espacio de estados puede ser:

### **Un árbol.**

- más sencillo de manejar.
- Más consumo de memoria (estados duplicados).

### **Un gafo (con o sin ciclos).**

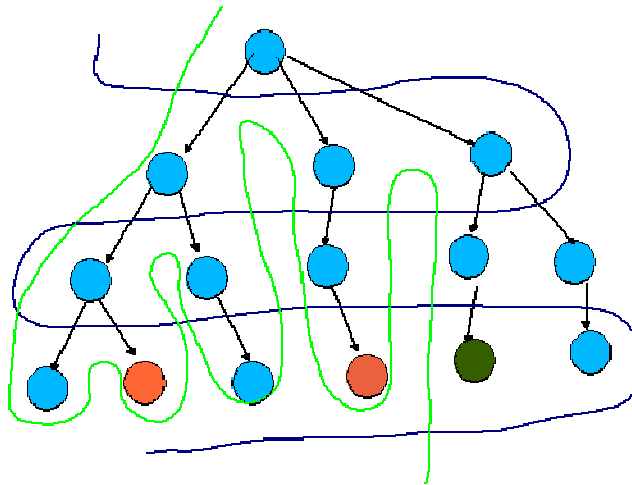
- Ahorro de memoria.
- Generaciones complejas (comprobar existencia de estados).

## BUSQUEDA EN EL ARBOL

### **Mecanismos de búsqueda:**

Resulta necesario recorrer el árbol y existen dos formas básicas para hacer esto:

El recorrido en verde refleja una búsqueda a profundidad mientras que el recorrido azul refleja una búsqueda hecha a amplitud primero.

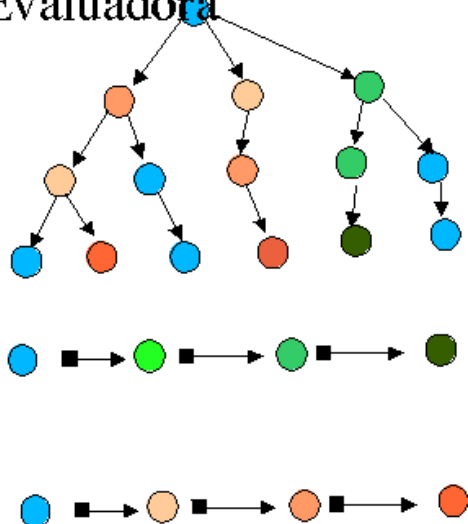


Consideremos que el nodo ganador del juego es el verde y el rojo el perdedor.

Una solución más interesante es la de hacer una búsqueda con cierto conocimiento (una implicación) más que con información, por que si ya sabemos como llegar no tiene sentido que le ande dando vuelta por las ramas.

La forma de obtener conocimiento ya lo sabemos es a través de una función evaluadora SUMATORIA (  $c \cdot X^n$  ) } que nos estime que tan lejos o que tan cerca estoy del lado bueno como se muestra a continuación:

## FUNCION Evaluadora



El diagrama como tal desea indicar que para llegar al nodo verde ganador debe ser posible conocer que tan verde es nuestro camino, y por el otro lado que tan rojo es. Esto una vez mas es conocimiento que inyecto al sistema por medio de la función que me indica entonces el

manejo de la implicación: Ante un Antecedente Entonces un Consecuente  
IF x THEN y.

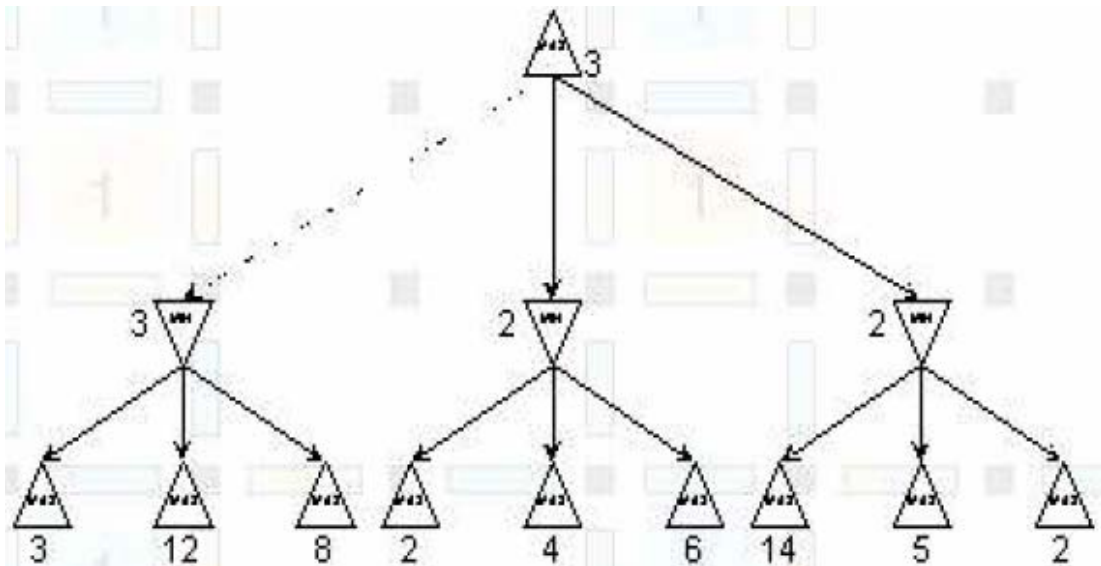
### Representación del conocimiento.

Esto se logra por medio de estructuras de datos que almacenen cuando menos la siguiente información:

- Para cada Estado
- Numero [ Integer ]
- Representación [ "oxo---xxo" ]
- Formula de evaluación [ Suma ( c X n ) ]
- Reglas de transferencia para otros estados [ p ---> px ]
- Tipo de Estado [ ganador, perdedor, prohibido ]
- Elementos [ jugadores, tablero ]

Esta es solo una lista inicial, en realidad la forma genérica de esta representación es una gramática libre de contexto

### EJEMPLO DE LA ESTRUCTURA DE UN ÁRBOL PARA EL ALGORITMO MINIMAX

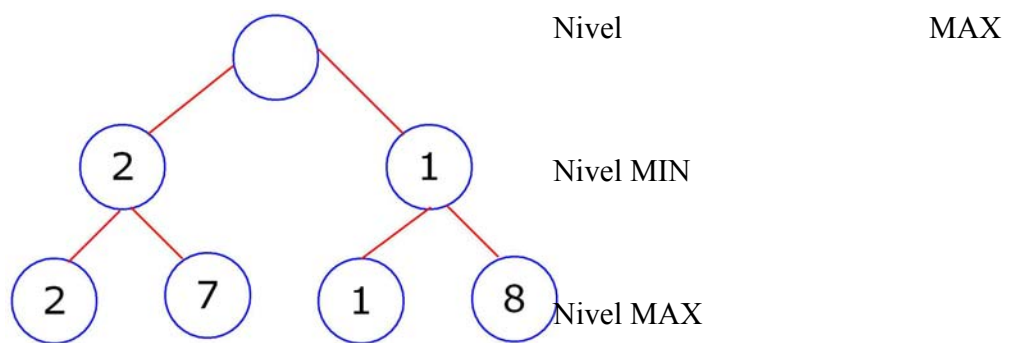


- El nodo raíz representa la posición de inicio del tablero.
- Cada nivel representa el turno.
- Cada círculo representa una jugada.

## PASOS DEL ALGORITMO MÍNIMAX

1. Generación del **árbol de juego**. Se generarán todos los nodos hasta llegar a un estado terminal.
2. Cálculo de los valores de la **función de utilidad** para cada nodo terminal.
3. Calcular el valor de los nodos superiores a partir del valor de los inferiores.  
Alternativamente se elegirán los valores mínimos y máximos representando los movimientos del jugador y del oponente, de ahí el nombre de Minimax.
4. Elegir la jugada valorando los valores que han llegado al nivel superior.

- El algoritmo explorará los nodos del árbol asignándoles un valor numérico mediante una función de utilidad, empezando por los nodos terminales y subiendo hacia la raíz.
- Colocar 0 ó 1 en los nodos terminales dependiendo si gana MIN o MAX
- La función de utilidad definirá lo buena que es la posición para un jugador cuando la alcanza.
- Se requiere de una estrategia que garantice llegar a estados terminales ganadores independientemente de lo que haga el oponente.
- Un valor positivo indica la ventaja de un jugador y uno negativo la ventaja del otro.
- El jugador que espera valores positivos se conoce como *maximizador*.
- El jugador que espera valores negativos se conoce como *minimizador*.
- El *maximizador* busca movimientos que lo conduzcan al mayor número positivo.
- El *minimizador* busca movimientos que lo conduzcan al menor número negativo.
- Por ejemplo:



- El maximizador:
  - ✓ Puede esperar llegar a un valor de 8
  - ✓ Sabe que el *minimizador* puede escoger un movimiento que lo lleve a un valor de 1
- Desde el punto de vista de el *maximizador*, el *minimizador* puede escoger 2 ó 1
- Los resultados de un nivel determinan la acción y el resultado del nivel inmediato superior

## CÁLCULO DE VALORES DE LA FUNCIÓN DE UTILIDAD

Calcular el valor *minimax* del nodo J:  $V(J)$

Si J es Terminal,  $V(J) \leftarrow ev(J)$

SINO

    Genera los sucesores de J:  $J_1, J_2, \dots, J_n$

    Evalúa  $V(J_1), V(J_2), \dots, V(J_n)$  de izq a der

    SI J es nodo *Max* ENTONCES

$V(J) \leftarrow \text{Max}[V(J_1), V(J_2), \dots, V(J_n)]$

    SI J es nodo *Min* ENTONCES

$V(J) \leftarrow \text{Min}[V(J_1), V(J_2), \dots, V(J_n)]$

### EJEMPLO

- Funcionamiento de minimax en un árbol generado para un juego imaginario.
- Los posibles valores de la función de utilidad tienen un rango de [1-9].
- En los movimientos del contrincante suponemos que escogerá los movimientos que minimicen nuestra utilidad
- En nuestros movimientos suponemos que escogeremos los movimientos que maximizan nuestra utilidad.
- *1er Paso*: Calcular los nodos terminales, en verde.
- *2do Paso*: Calcular el cuarto nivel, movimiento MIN, minimizando lo elegido (5, 2y 1).
- *3er Paso*: Calcular el tercer nivel, movimiento MAX, maximizando la utilidad (5, 9).
- El segundo nivel es un movimiento MIN (5, 3y 1).
- Finalmente llegamos al primer nivel, el movimiento actual, elegiremos el nodo que maximize nuestra utilidad (5).



## EJEMPLO

Los Juegos bipersonales en la IA

- » Son problemas con contingencias
- » En ocasiones pueden tener una ramificación alta
  - por ejemplo en ajedrez,  $b \approx 35$
- » Puede haber limitaciones de tiempo
  - Entorno semidinámico

En la resolución se utilizan:

- » Funciones de evaluación
  - Evalúan los operadores utilizados por cada jugador.
  - Ayudan a decidir el resultado del juego y las mejores estrategias para cada jugador.
- » Métodos de poda
  - Simplificación de la búsqueda.

### El Gato: un ejemplo sencillo

- Para ejemplificar el algoritmo, consideremos el juego del gato. En este juego podemos usar la siguiente función de evaluación para un tablero  $t$ :

$$E(t) = N_A(t) - N_C(t)$$

donde  $N_A(t)$  es el número de filas, columnas o diagonales abiertas para MAX (donde aún puede ganar) y  $N_C(t)$  es el número de filas, columnas o diagonales abiertas para MIN.

- Si  $t$  es un tablero ganado por MAX,  $E(t) = \infty$  y si es un tablero perdido,  $E(t) = -\infty$  (aquí en vez de  $\infty$  podríamos haber ocupado otro valor).
- La figura 1 muestra el algoritmo Minimax con un árbol de profundidad 2. La figura 2 muestra otro árbol con profundidad 2, pero luego que MIN ya ha jugado en una de las posiciones del tablero. Finalmente, la figura 3 muestra la última etapa de la búsqueda.

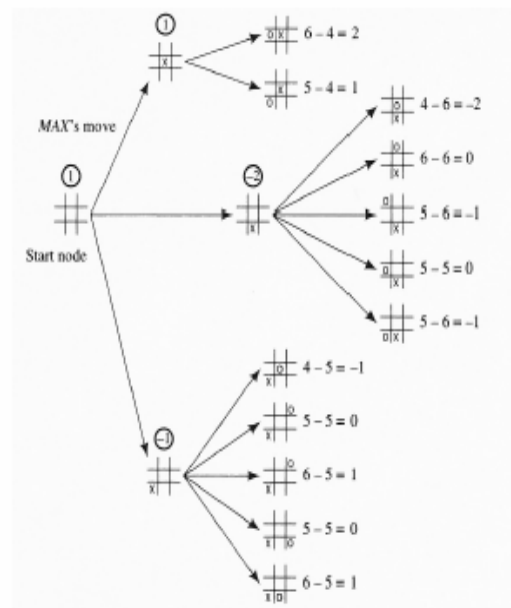


Figura 1: Primera etapa en la búsqueda del gato

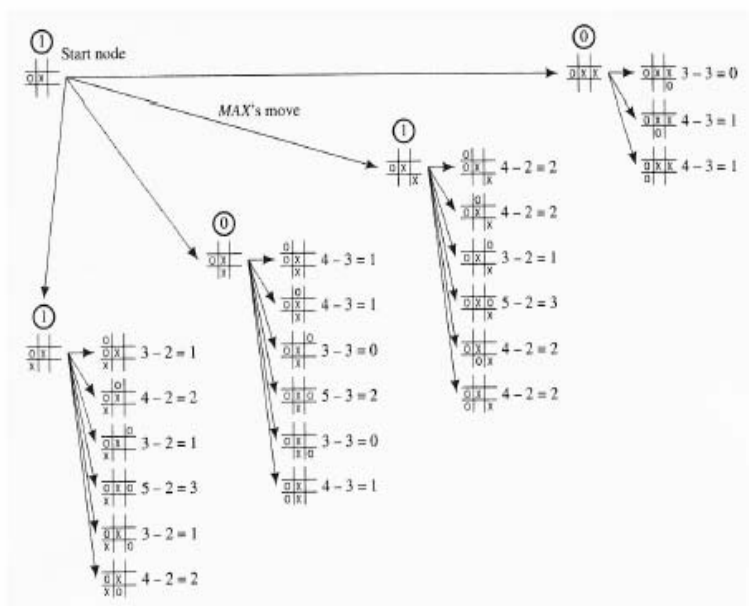


Figura 2: Segunda etapa en la búsqueda del gato



supone también que cada compañía no puede emplear mas de uno de estas actitudes o estrategias al mismo tiempo y que las tres estrategias tienen idénticos costos.

Por estos supuestos, hay un total de  $3 \times 3 = 9$  combinaciones posibles de movimientos, y cada una es capaz de afectar a la parte del mercado en una forma específica. Por ejemplo, si A y B sirven refrescos durante el viaje, se dice que A perdería 10 por 100 de la parte del mercado a favor de B, lo que puede indicar que los refrescos de B son mas para los gustos de los clientes, igualmente, si A anuncia y B, por ejemplo, sirve refrescos, se supone que A ganaría 20 por 100 del mercado en perjuicio de B; evidentemente, la publicidad en televisión parece ser más eficaz que servir refrescos.

Ahora, por cada una de las 9 combinaciones puede determinar ganancias o pérdidas del mercado para A como se indica en la siguiente matriz de pagos.

|                      | <b>b<sub>1</sub></b> | <b>b<sub>2</sub></b> | <b>b<sub>3</sub></b> |
|----------------------|----------------------|----------------------|----------------------|
| <b>a<sub>1</sub></b> | -10                  | -11                  | -1                   |
| <b>a<sub>2</sub></b> | 9                    | -8                   | -6                   |
| <b>a<sub>3</sub></b> | 20                   | -10                  | -13                  |

#### 14. Estrategias maximin y minimax

El enfoque conservador a la elección de la mejor estrategia es suponer lo peor y actuar de conformidad con ello. Así según este enfoque y con referencia en la matriz de pagos. Si A decide sobre la estrategia  $a_1$ , supondría que B escogerá la estrategia  $b_2$ , reduciendo con ello el pago  $a_1$  para A a un valor mínimo o de seguridad de  $-11$ . Análogamente, los valores de seguridad para  $a_2$  y  $a_3$  son  $-8$  y  $-3$ , respectivamente.

Obsérvese que los valores de seguridad para los distintos movimientos que puede hacer A son los mínimos de filas. Dados estos valores mínimos, hará bien en emplear aquella estrategia que da el máximo de estos valores de seguridad mínimos. En el ejemplo A debe adoptar  $a_2$  y aspira a un pago de  $-8$  a B. Esta regla de decisión, que conduce a la elección del mayor de los valores mínimos en que puede resultar cada estrategia, se llama estrategia maximin.

La compañía B, según esta actitud conservadora, supondría que por cada una de sus acciones, la respuesta de A será tal que la ganancia de A en parte del mercado es la máxima posible. Por ejemplo, si B emplea la estrategia  $b_1$ , supondría que A adoptara la estrategia  $a_3$ , la cual dará la peor pérdida posible para B. Análogamente, los peores pagos para  $b_2$  y  $b_3$  son  $-8$  y  $-1$ , los máximos valores en las columnas 2 y 3, respectivamente. Así, vemos que el máximo en cada columna es el peor pago por un movimiento correspondiente hecho por B. El mejor de estos peores pagos es claramente el valor mínimo de estas cifras mas altas. Esta cifra  $-8$  en la columna 2, correspondiente a la estrategia  $b_2$  y el movimiento contrario  $a_2$ . Por tanto, la emisión óptima, llamada estrategia minimax de B, es  $b_2$ .

Se puede observar según la regla maximin de A y la regla minimax de B el pago es  $-8$ . Esta cantidad se llama valor del juego es positivo, se dice que el juego es a favor de A: si negativo, favorece a B; y si cero, se dice que el juego es equitativo. La solución de nuestro problema da un pago de  $-8$ , que indica que el juego favorece a B por que B gana 8 por 100 del mercado a expensas de A.

## **BIBLIOGRAFIA**

- Inteligencia artificial un enfoque moderno  
Stuart Russell, Peter Norving  
Ed. Prentice Hall
- [www.itnuevolaredo.edu.mx/takeyas](http://www.itnuevolaredo.edu.mx/takeyas)
- <http://dmi.uib.es/~abasolo/intart/capitulo2.doc>
- [www.inteligenciaartificial/minimax/1/3\\_2Minimax.htm](http://www.inteligenciaartificial/minimax/1/3_2Minimax.htm)