INTELIGENCIA ARTIFICIAL

Universidad Nacional Mayor de San Marcos

Facultad de Ingeniería de Sistemas e Informática

Ing. Mg. Rolando A. Maguiña Pérez

Agenda

- Juegos inteligentes
  - Juegos como búsqueda
  - Funciones de utilidad y de evaluación
  - Estrategias de búsqueda
- Estrategia MINMAX
  - Algoritmo MINIMAX
  - Poda α - β
  - Resolución de Problemas

Juegos inteligentes humano-máquina

Ing. Mg. Rolando A. Maguiña Pérez

Clasificación general de juegos

Juegos en paralelo

Tijera, papel, piedra - sin azar, con manías y tendencias.

- Decisiones imperfectas

Juegos en serie (por turnos)

Damas, michi, etc.

Juegos con oponentes y por turnos

JUEGOS	Deterministas	No deterministas
Información perfecta	3-en-raya=Michi, otello=reversi, ajedrez, go	backggamon, monopolio
Información imperfecta	juego de inspección, hundir la flota	dominó, truco, bridge, póquer, escoba

Juegos con oponentes y por turnos

Los juegos como problemas de búsqueda

- Juegos bipersonales {computadora, humano}, y los dos quieren ganar
- Juegos por turnos y con información completa
- Hay un número finito de estados y decisiones
- No influye el azar
- Juegos de suma cero
- Restricciones de tiempo

Juegos con oponentes y por turnos

Los juegos como problemas de búsqueda

- Juegos inteligentes son un dominio útil para el estudio de la inteligencia de máquina
  - Existencia de reglas y metas bien definidas, y
  - Un ambiente muy estructurado
- A. Samuel: los juegos son un buen campo de estudio:
  - Requieren escasos conocimientos
  - Con información perfecta y sin azar aparecen patrones

Juegos con oponentes y por turnos

Los juegos como problemas de búsqueda

- Difieren de otros problemas de búsqueda:
  - presencia de un adversario

    Falta de información: incertidumbre frente a las acciones del adversario
  - espacios de búsqueda desmesurados
    - ajedrez

      espacio típico de búsqueda ~35^{100} nodos

      "sólo" 10^{40} posiciones válidas
  - a veces límites en el tiempo – énfasis en velocidad

Juegos con oponentes y por turnos

Los juegos como problemas de búsqueda

- Cada jugador tiene un modelo completo y perfecto del entorno, de sus movtos y de los de su oponente, y de los efectos de los mismos
- Ninguno de los jugadores tiene conocimiento de lo que el otro hará en una situación dada
- En juegos deterministas con información perfecta, asumimos que hay un juego perfecto y un contrincante óptimo

Juegos como problemas de búsqueda

Los juegos como problemas de búsqueda

- Juegos se pueden representar como una BEE
- Juego con dos participantes MAX (maximizante) y MIN (minimizante): los jugadores alternan sus jugadas
- Usa árboles de estados para representar las jugadas.
- Tarea consiste en encontrar el "mejor" mvto para MAX (MIN): algoritmo la proporcionará

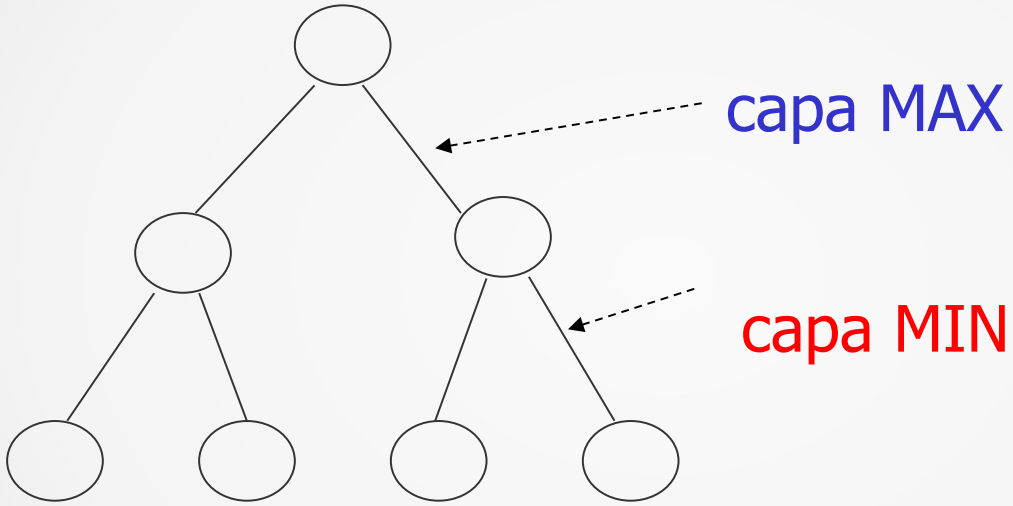
Juegos como problemas de búsqueda

Elementos de un juego

- Un juego es un tipo de problema de búsqueda:
  - Jugadores: Máquina (MAX) vs Humano (MIN)
    MAX y MIN alternan sus jugadas
  - Estado inicial
  - Conjunto de operadores
  - Prueba de parada (estados terminales ganadores)
  - Función utilidad (función de evaluación)

Juegos como problemas de búsqueda

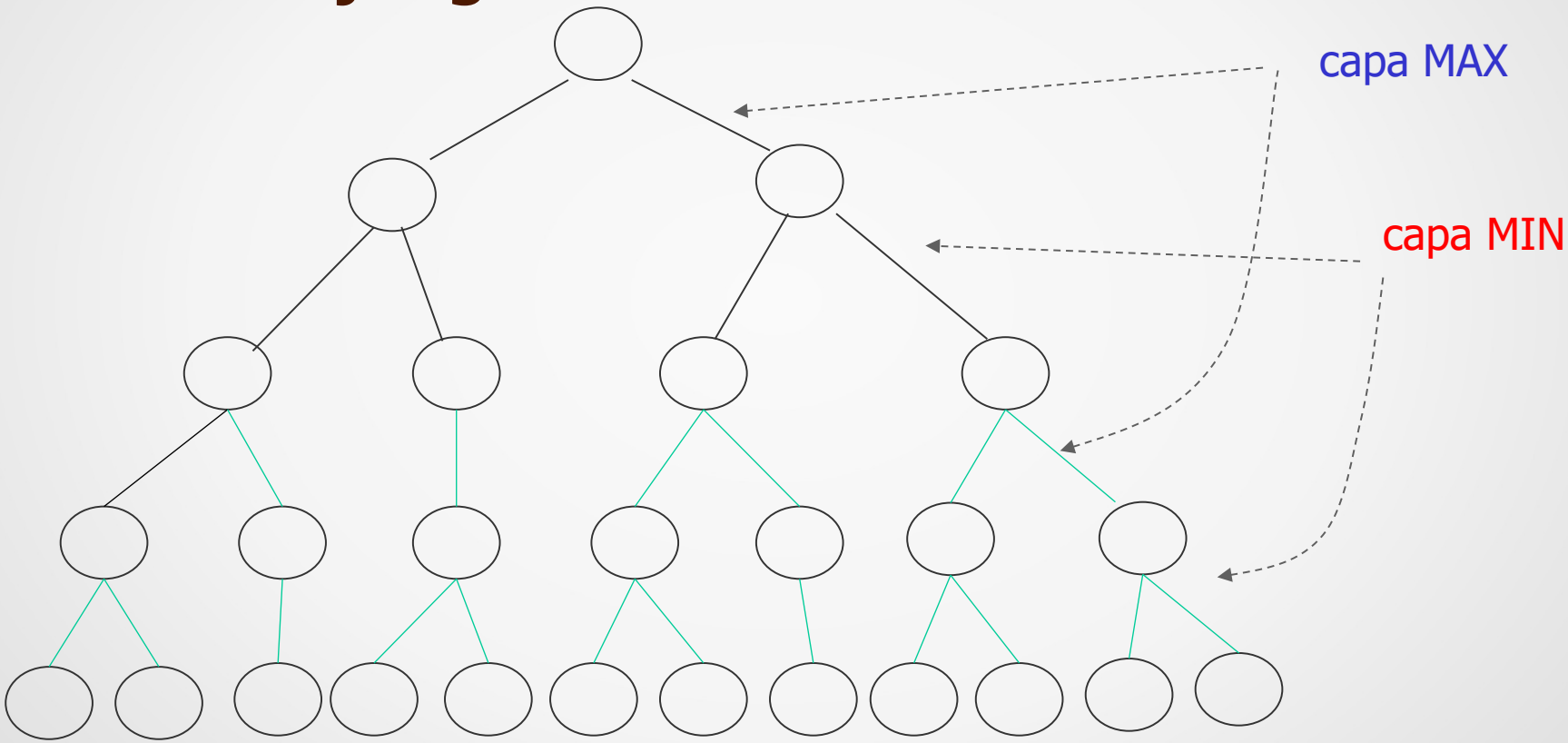
Árbol de juego



problema de decisión más complejo que en los problemas de búsqueda definidos anteriormente!

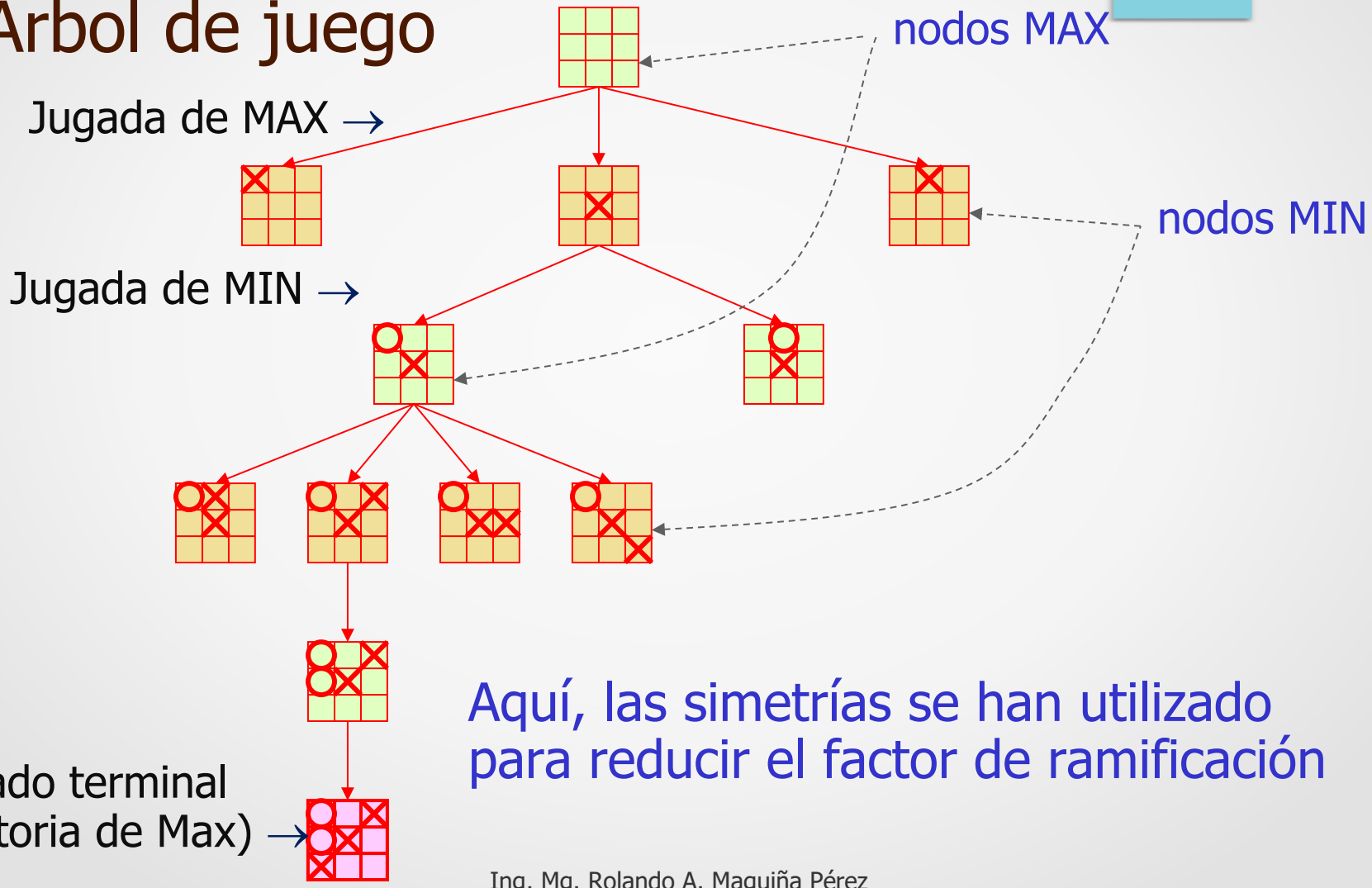
Juegos como problemas de búsqueda

Árbol de juego



Juegos como problemas de búsqueda

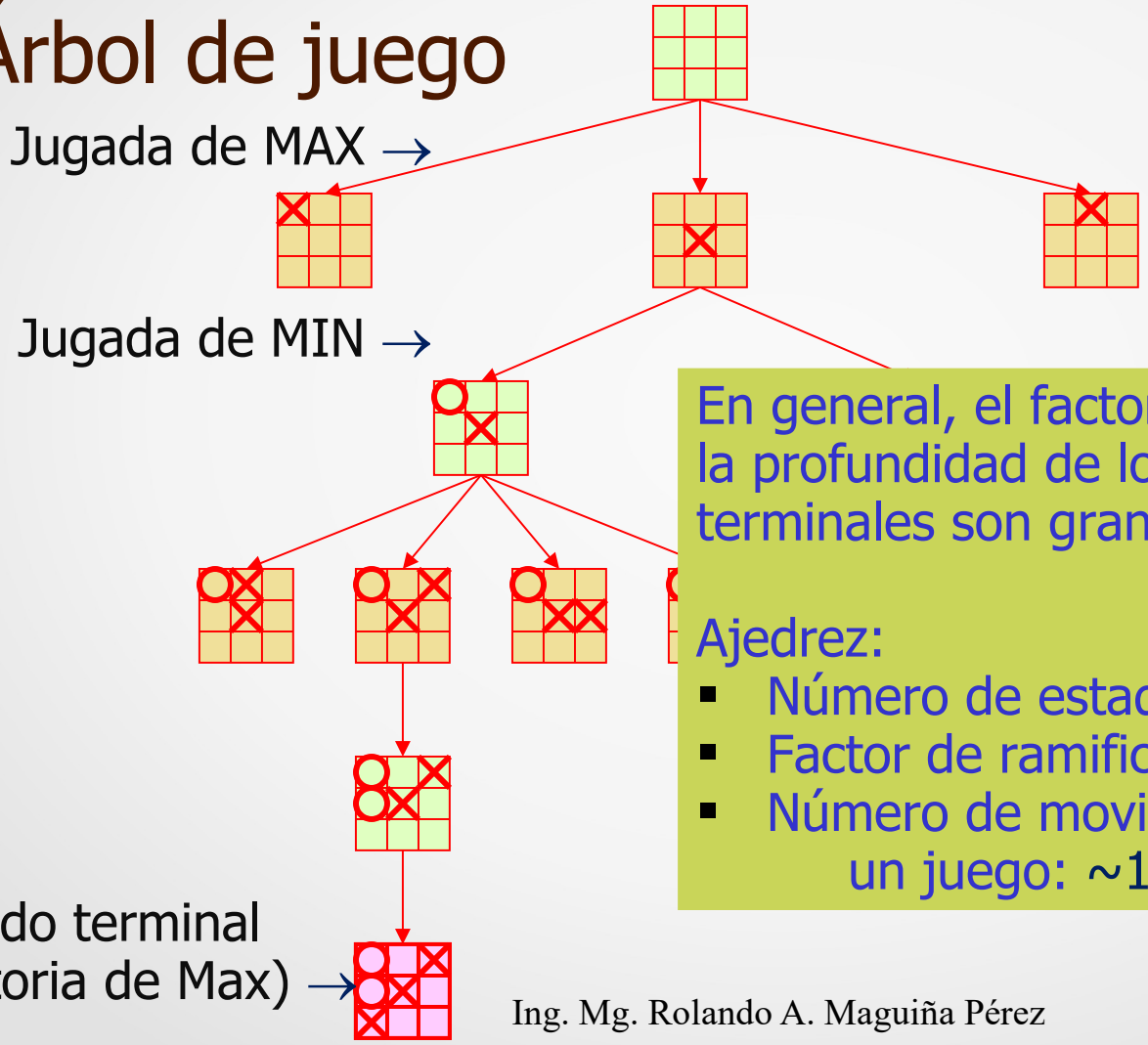
Árbol de juego



Aquí, las simetrías se han utilizado para reducir el factor de ramificación

Juegos como problemas de búsqueda

Árbol de juego



En general, el factor de ramificación y la profundidad de los estados terminales son grandes

Ajedrez:

- Número de estados: $\sim 10^{40}$
- Factor de ramificación: ~35
- Número de movimientos totales en un juego: ~100

Juegos como problemas de búsqueda

Árbol de juego

- Enfoque computacional: es imposible generar el árbol de juego completo, con algunas excepciones
- Se debe modificar las condiciones de parada en base a:
  - tiempo de búsqueda,
  - máxima profundidad en los nodos, o
  - espacio de almacenamiento limitado

Juegos como problemas de búsqueda

Estrategias de juego

- Estrategias para las jugadas de la máquina:
  - búsqueda ciega
  - búsqueda informada (Primero el Mejor)
  - algoritmo minimax y sus variantes

Juegos como problemas de búsqueda

Escoger una acción: idea básica

1) Utilizando **el estado actual como estado inicial**, construir el árbol de juego uniformemente hasta la profundidad máxima h (llamada horizonte), factible dentro del límite de tiempo

2) Evaluar los estados de los nodos hoja

3) Propagar hacia atrás los resultados desde las hojas hasta la raíz y elegir la mejor acción suponiendo lo peor para MIN

→ algoritmo Minimax

Estrategia Minimax

Conceptos básicos

- El objetivo del análisis del árbol es encontrar qué valor asignamos al nodo raíz (valor minimax)
- Se decide sólo cuál próxima jugada nos toca hacer (las otras jugadas a veces ni se analizan)
- En algunos casos la función puede devolver valores como PIERDE, GANA o EMPATA
- Usando este enfoque "minimax", podemos asignar valores a cada estado del árbol de juego!

Juegos como problemas de búsqueda

Función utilidad y función de evaluación

- Función utilidad (función de resultado)

  Asigna un valor numérico al resultado obtenido en un juego.

  En el ajedrez es +1, -1, 0 para representar victoria, derrota y empate, respectivamente en el juego.

- Requerirían generar todo el árbol de juego

- Es reemplazada por la función de evaluación

Juegos como problemas de búsqueda

Función utilidad y función de evaluación

- Imposibilidad de generar todo el árbol de búsqueda: generar árbol hasta un determinado nivel de profundidad y en ese nivel aplicar alguna función de evaluación f(N)

Shannon , 1950

- f(N) devuelve un valor numérico que indica cuán bueno es un determinado estado

Por convención, valores altos son buenos para MAX y malos para MIN, y valores bajos, al contrario

Juegos como problemas de búsqueda

Función utilidad y función de evaluación

- Función de evaluación
  - Producen una estimativa de la utilidad de un juego correspondiente a una determinada situación
  - Es una heurística que estima cuán favorable es para MAX

    Función f: estado s $\rightarrow$ valor numérico f(s)
  - Requisitos:
    - Debe coincidir con la función de utilidad en los estados terminales.
    - Cálculo no debe ser muy demorado.
    - Debe reflejar las posibilidades reales de ganar

Juegos como problemas de búsqueda

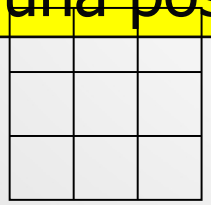
EJM. Michi (Tres en raya)

Función de evaluación propuesta:

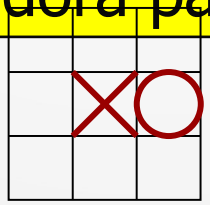
f(s) = (Nº filas, columnas y diagonales abiertas para MAX) – (Nº filas, columnas y diagonales abiertas para MIN)

Si s es una posición ganadora para MAX entonces: f(s) = ∞

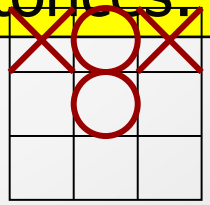
Si s es una posición ganadora para MIN entonces: f(s) = -∞



8–8 = 0



6–4 = 2

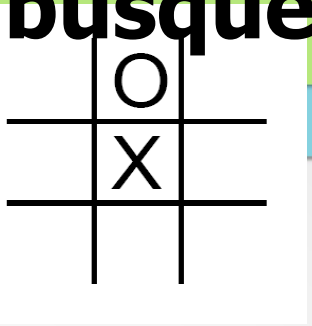


3–3 = 0

Juegos como problemas de búsqueda

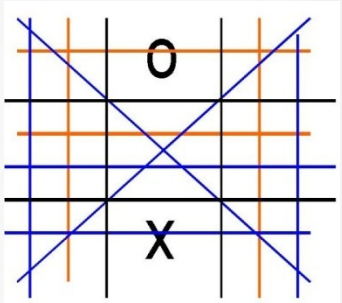
Si s representa la posición:

$f(s) = 6 - 4 = 2$

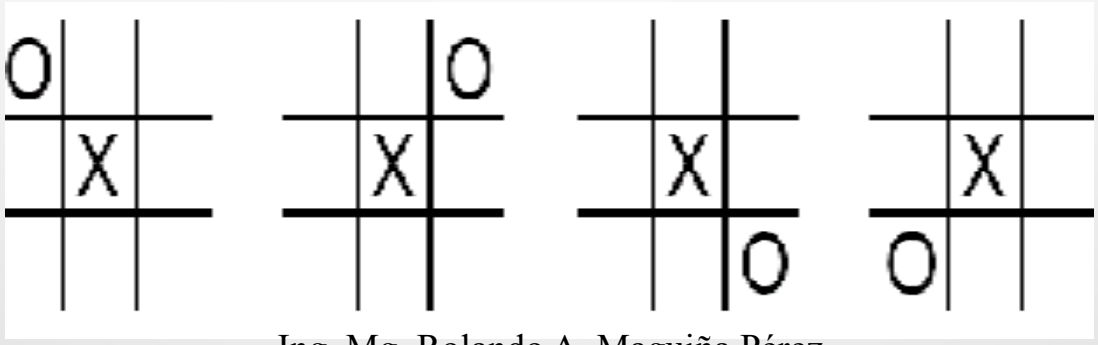


Para esta posición:

$f(s) = 6 - 6 = 0$



Se puede utilizar las simetrías del juego para generar las posiciones sucesoras.



Juegos como problemas de búsqueda

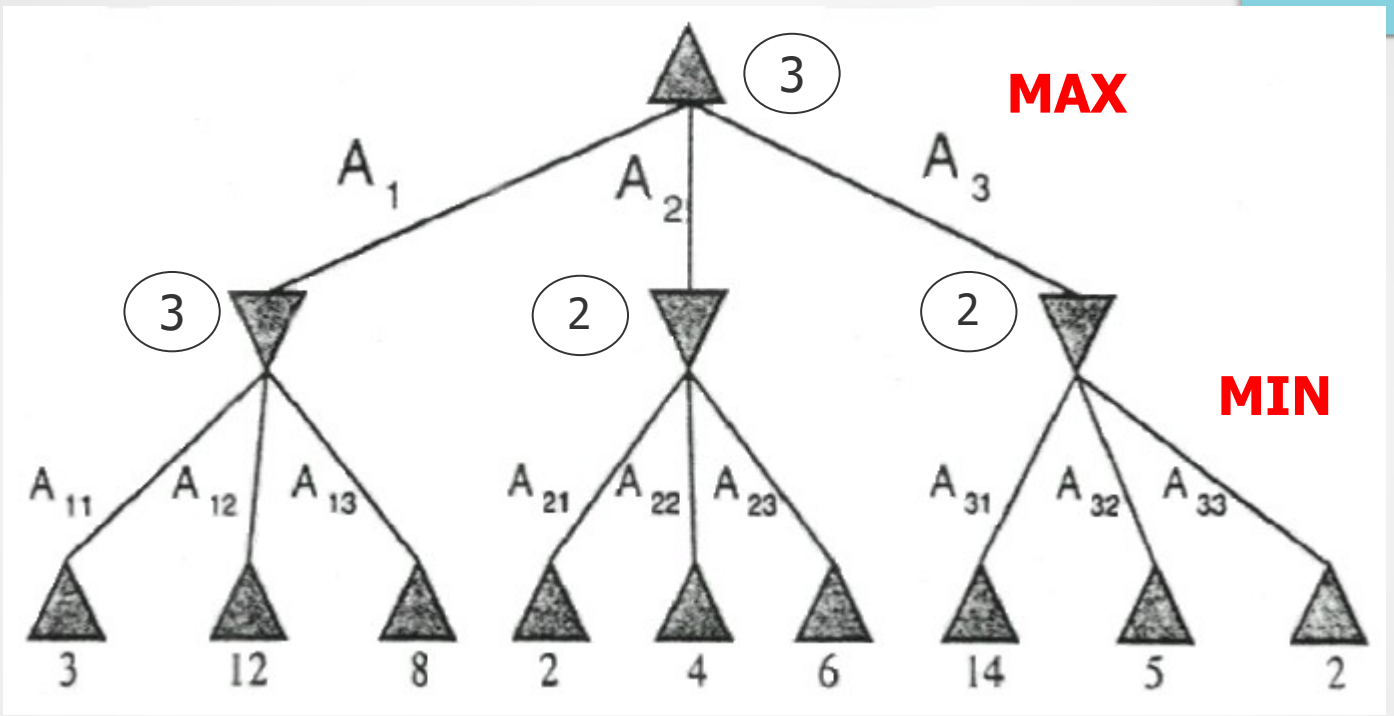
Construcción de funciones de evaluación

- Por lo general, una suma ponderada de "características":

$$f(s) = \sum_{i=1}^{N} w_i c_i(s)$$

- Las características pueden incluir:
  - Cantidad de piezas de cada tipo
  - Cantidad de movimientos posibles
  - Número de casilleros controlados

Estrategia Minimax



Árbol de juego (dos capas) generado por MINIMAX

posibles jugadas de MAX → A_1, A_2 y A_3

evaluación de nodos terminales según función de evaluación

evaluación de los otros nodos mediante MINIMAX

Algoritmo Minimax

1) A partir del estado actual, expandir el árbol de búsqueda hasta la profundidad de corte (límite de profundidad)
2) Aplicar la función de evaluación a cada nodo "hoja" del árbol de búsqueda
3) Propagar hacia atrás los valores obtenidos en la evaluación de los nodos "hoja", aplicando (una capa a la vez hasta llegar al nodo raíz) sgte criterio :
   3.a) Un nodo MAX obtiene el máximo de los valores de evaluación de sus sucesores
   3.b) Un nodo MIN obtiene el mínimo de los valores de evaluación de sus sucesores
4) Elegir la jugada de MAX (MIN) con la que obtiene el valor más alto (bajo) de los valores propagados (decisión Minimax)

Algoritmo Minimax

Procedimiento Minimax (Situación Profundidad)
Si Profundidad= p_{max}
entonces devolver evaluación(Situación)
Sino
    Si ganadora(Situación)
    entonces devolver $+\infty$
    Sino
        Si perdedora(Situación)
        entonces devolver $-\infty$
        Sino
            Si empate(Situación)
            entonces devolver 0
            Sino
                S= sucesores(Situación)
                L=lista de llamadas al MINIMAX ($S_i \in$ S, Profundidad +1)
                Si nivel-max-p(Profundidad)
                entonces devolver max(L)
                Sino devolver min(L)

Fuente: "IA" R. Aler/D. Borrajo/A.Silva Univ Carlos III Madrid - España

Juegos como problemas de búsqueda

Algoritmo del juego

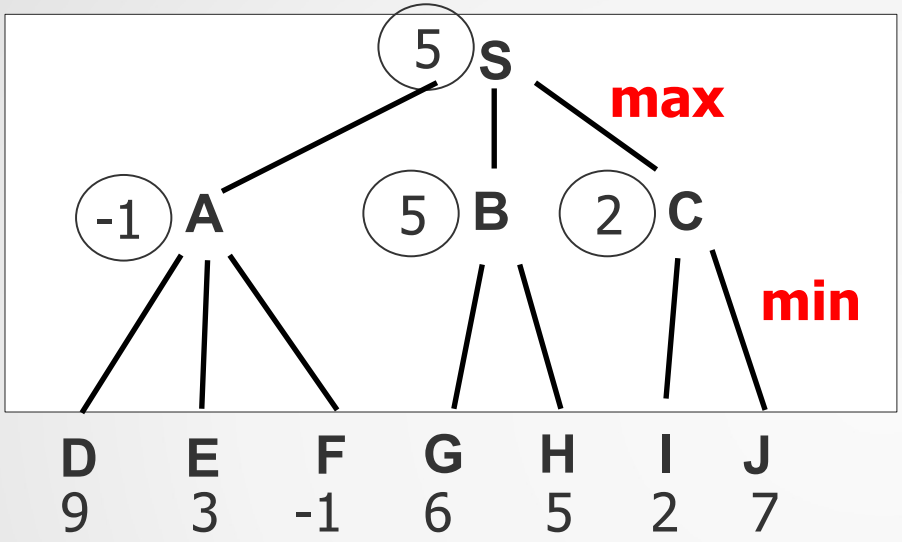
Repetir hasta llegar a un estado terminal

1) Seleccionar movimiento usando Minimax
2) Ejecutar movimiento
3) Observar el movimiento de MIN

Notar que en cada ciclo, el árbol de juego grande construido hasta el horizonte h se usa para seleccionar solo un movimiento.

Todo se repite nuevamente en el siguiente ciclo

Estrategia Minimax



- Árbol de búsqueda generado por MINIMAX (hasta nivel 2)

❖ Obtención de los valores en nodos del nivel 1

Valor en nodo A: min(9,3,-1)= -1

Interpretación: si oponente alcanza este nodo, escogerá el movimiento asociado al arco de A hasta F

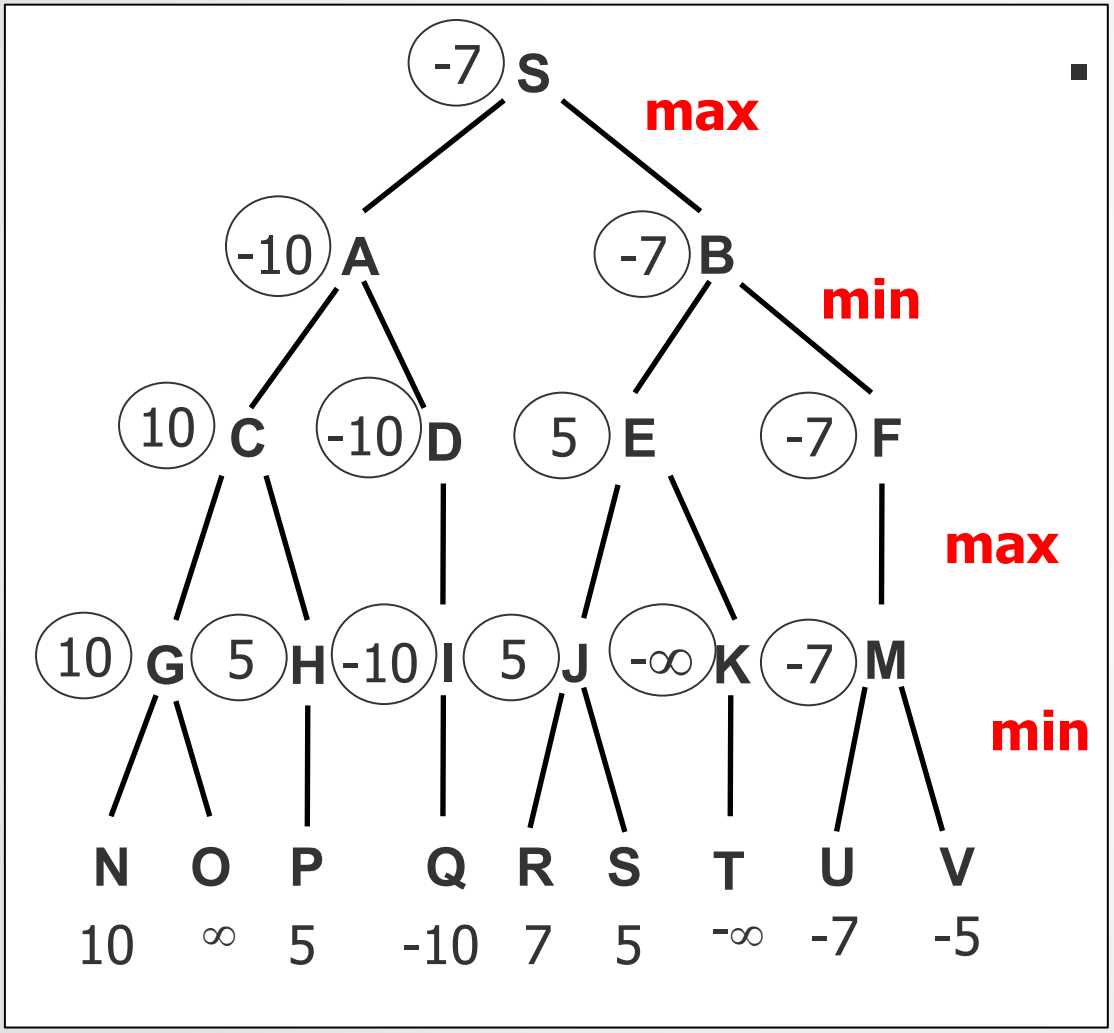
Similarmente, el valor en B es 5, y en C, 2

❖ Obtención de los valores en nodos del nivel inmediato anterior

Valor en nodo S: max(-1,5,2)= 5

Interpretación: valor minimax en nodo raíz es 5 y movimiento seleccionado es el asociado al arco de S hasta B

Estrategia Minimax



- Árbol de búsqueda generado por MINIMAX (hasta nivel 4)

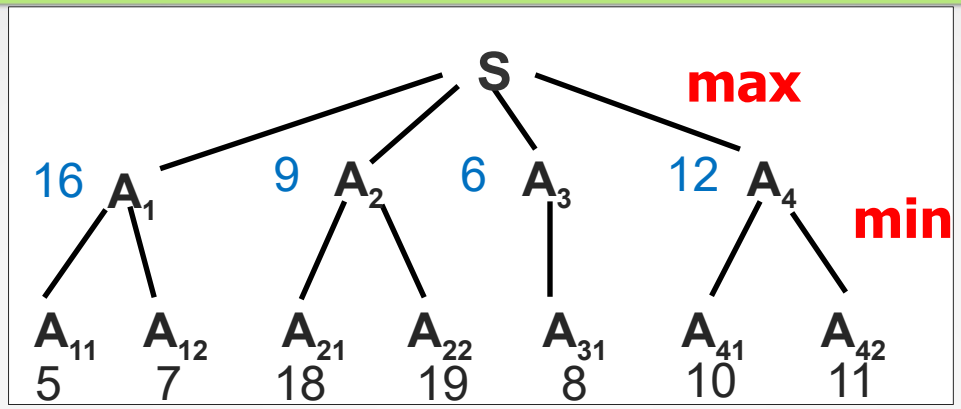
evaluación de los otros nodos mediante MINIMAX

evaluación de nodos hoja según fc de evaluación

Juegos inteligentes

Evaluación Nivel 1

Evaluación Nivel 2

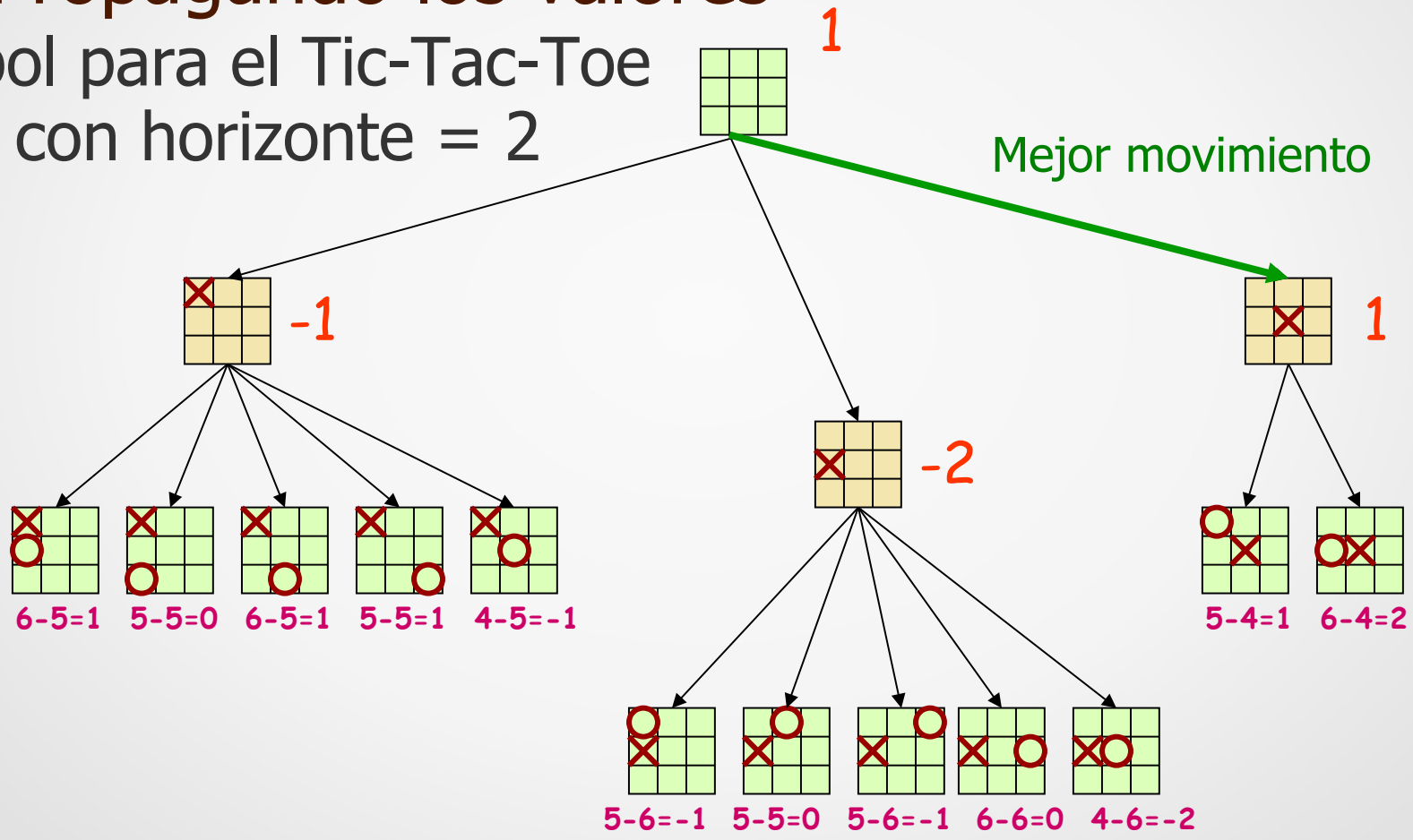


Nivel de dificultad	Estrategia	Descripción	Ejm
0- Principiante	No determinista	Genera los sucesores de S y selecciona aleatoriamente uno de ellos	
1 - Normal	Primero el mejor Goloso/Voraz	Genera los sucesores de S y selecciona el "mejor" de ellos	
2- Experto	Minimax	Para cada posible jugada de la PC se generan las del humano y se selecciona la alternativa que presenta la mayor de las menores utilidades	

Juegos inteligentes

Propagando los valores

Arbol para el Tic-Tac-Toe con horizonte = 2



Juegos inteligentes

Propagando los valores – continuación