

Programación de Juegos

Tema 4: Técnicas de Inteligencia Artificial para Videojuegos



Universidad
de Huelva

Departamento de Tecnologías
de la Información

Área de Ciencias de la Computación e
Inteligencia Artificial



Objetivos:

- Tener una visión global de la Inteligencia Artificial “*tradicional*” o clásica en videojuegos, y familiarizarse con sus diferentes técnicas.





Índice:

1. Introducción a la IA para Videojuegos

2. Modelos de IA en Videojuegos

2.1 Juegos con adversario

2.2 Búsqueda de caminos

2.3 Mecanismos de toma de decisión

2.3.1 Máquinas de estados finitos

2.3.2 Árboles de decisión

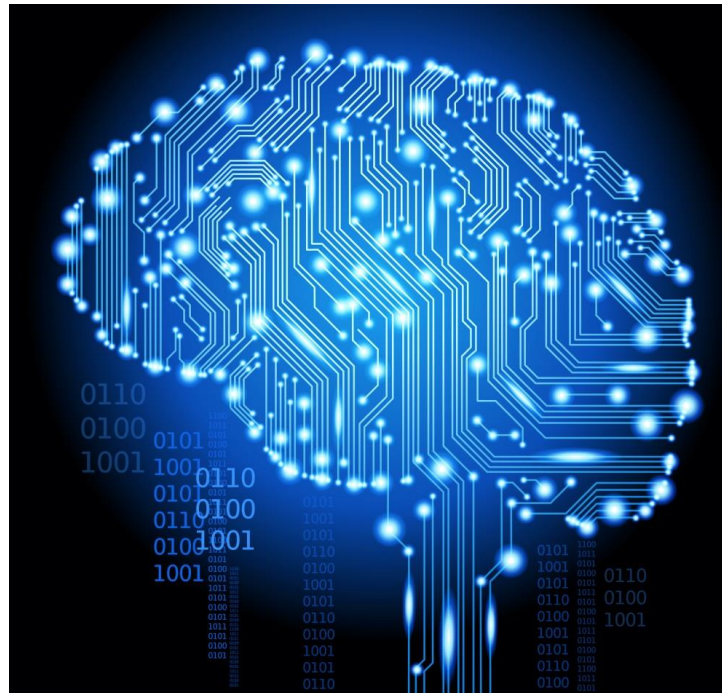
2.4 Árboles de comportamiento

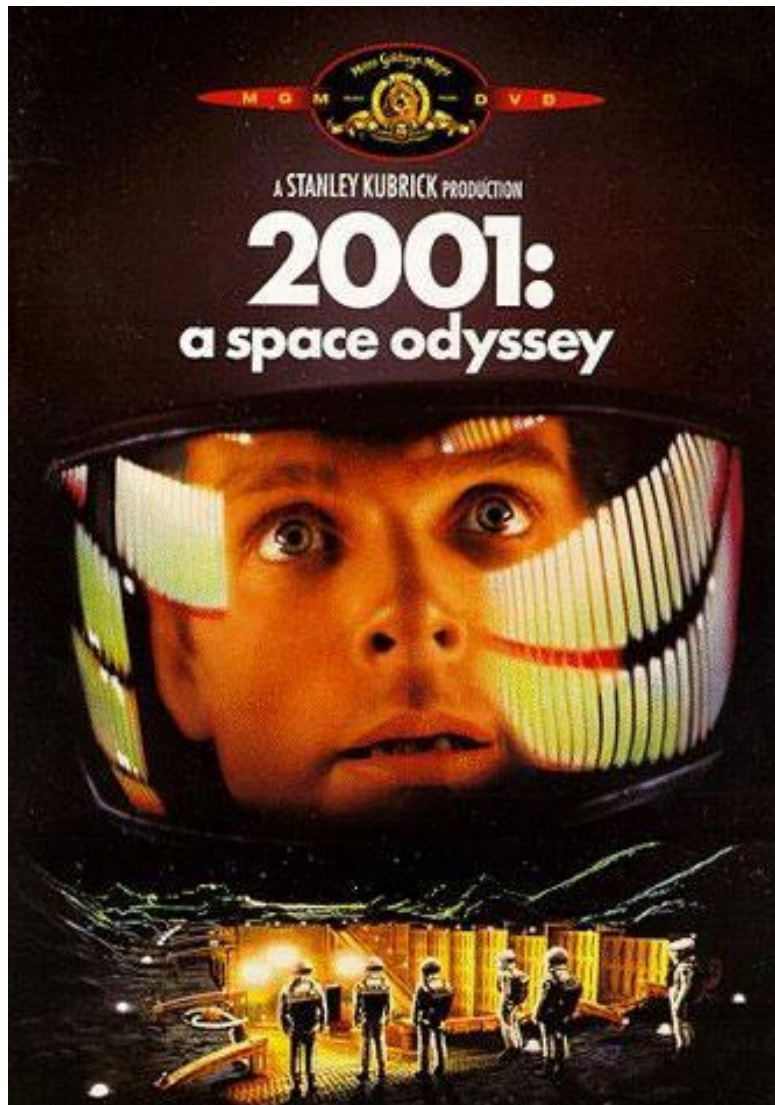
3. Bibliografía y Lecturas Complementarias



1. Introducción a la Inteligencia Artificial para Videojuegos

¿Qué entendemos por Inteligencia Artificial?



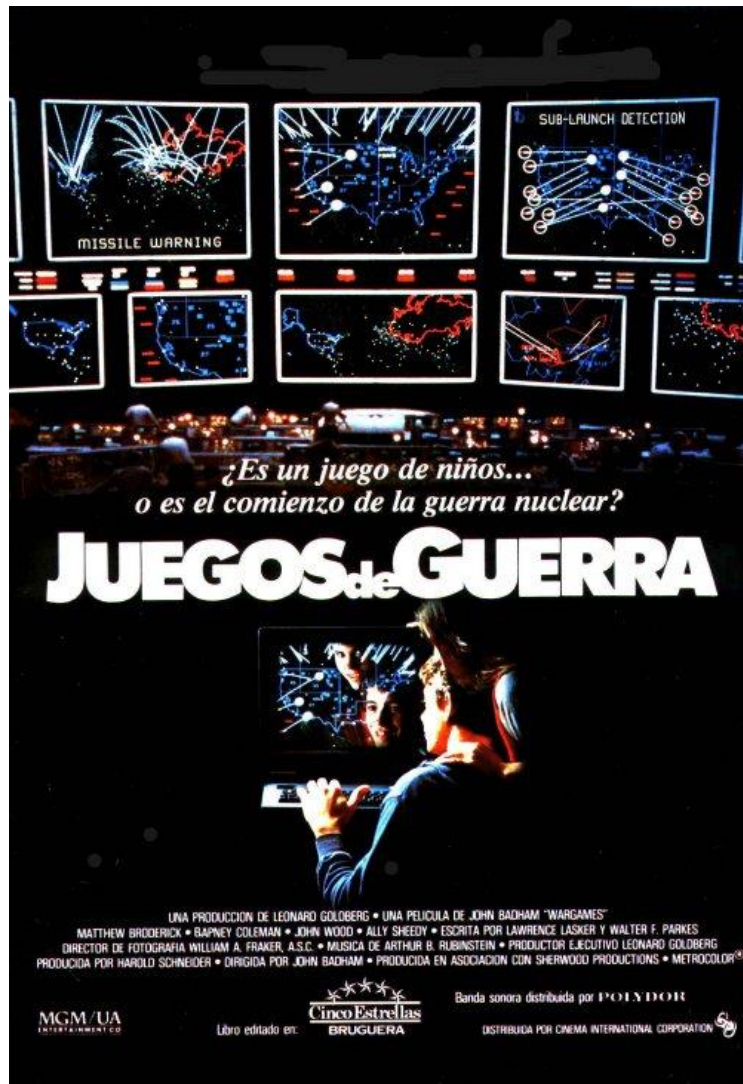


Estreno: Abril de 1968

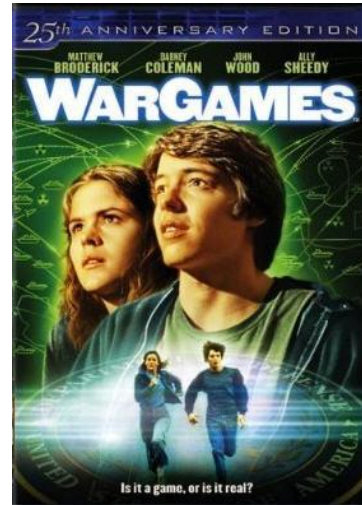
Director: Stanley Kubrick



[HTTP://GTO.E5.KZ/](http://GTO.E5.KZ/)



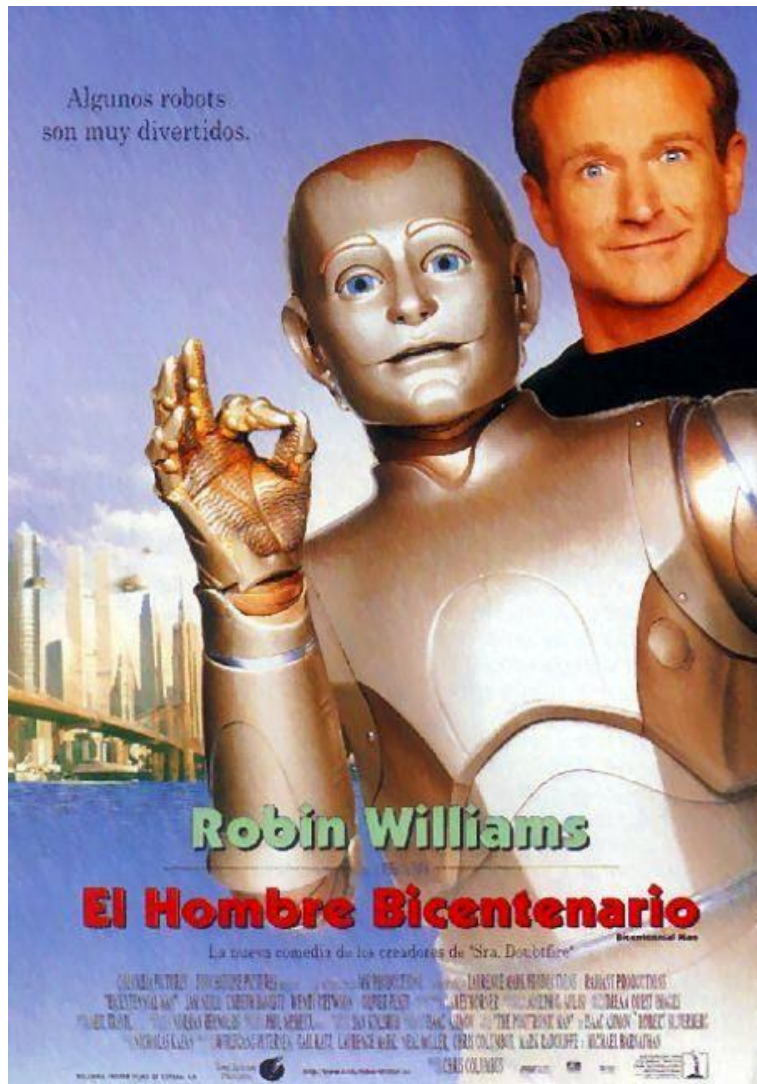
Estreno: Junio de 1983
Director: John Badham



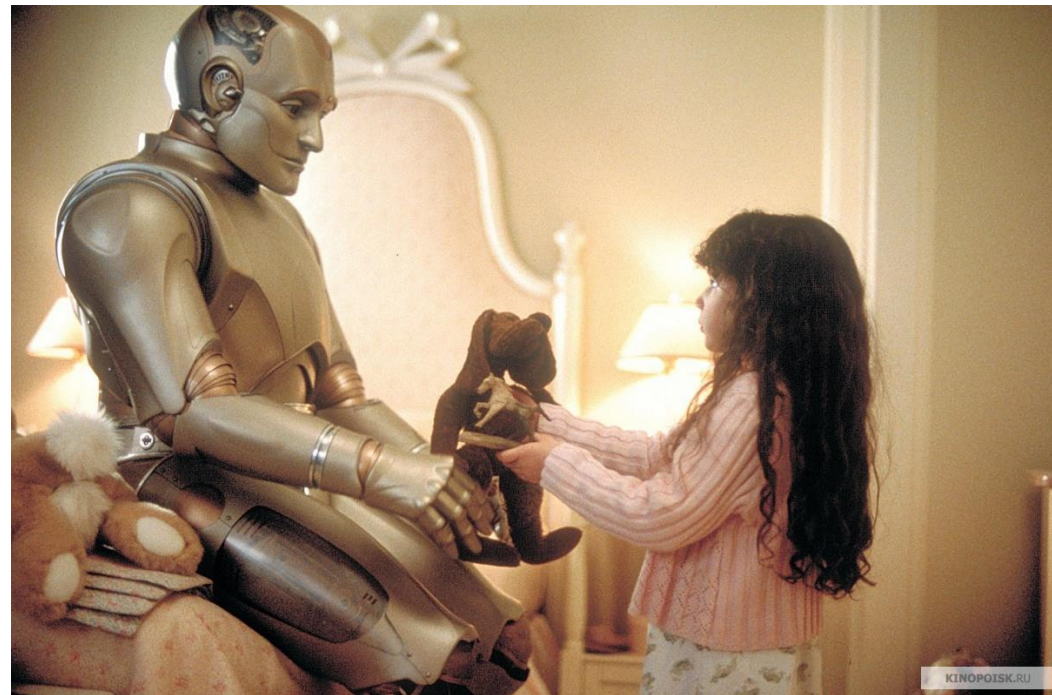


Estreno: Octubre de 1984
Director: James Cameron





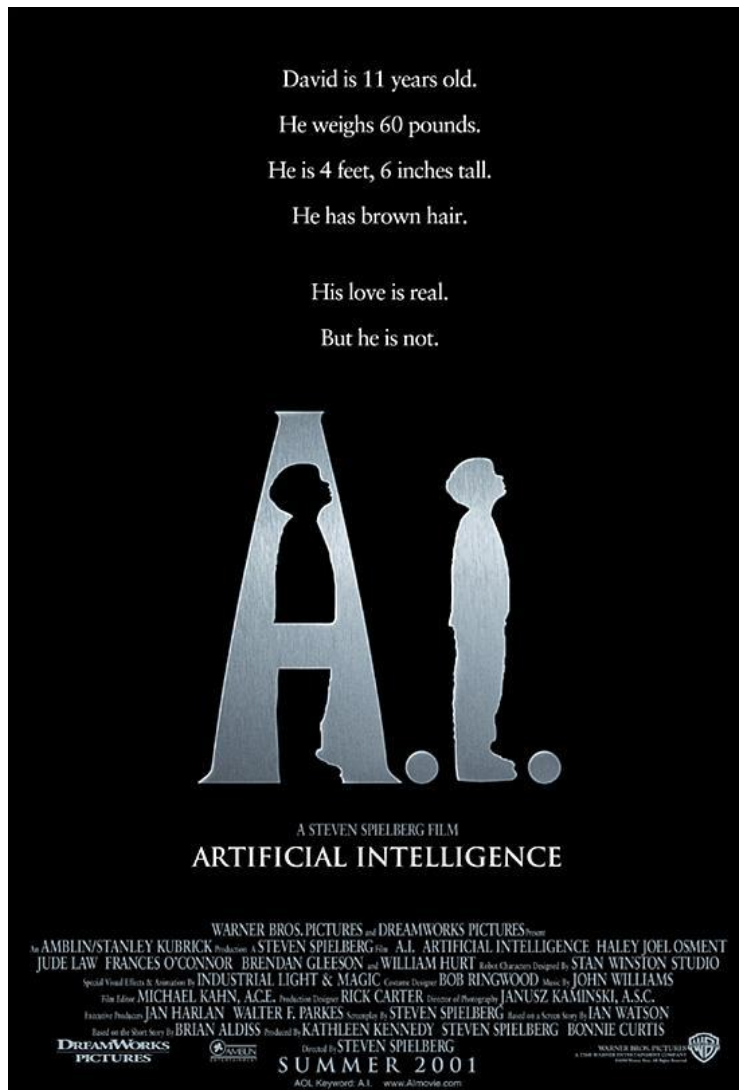
Estreno: Diciembre de 1999
Director: Chris Columbus

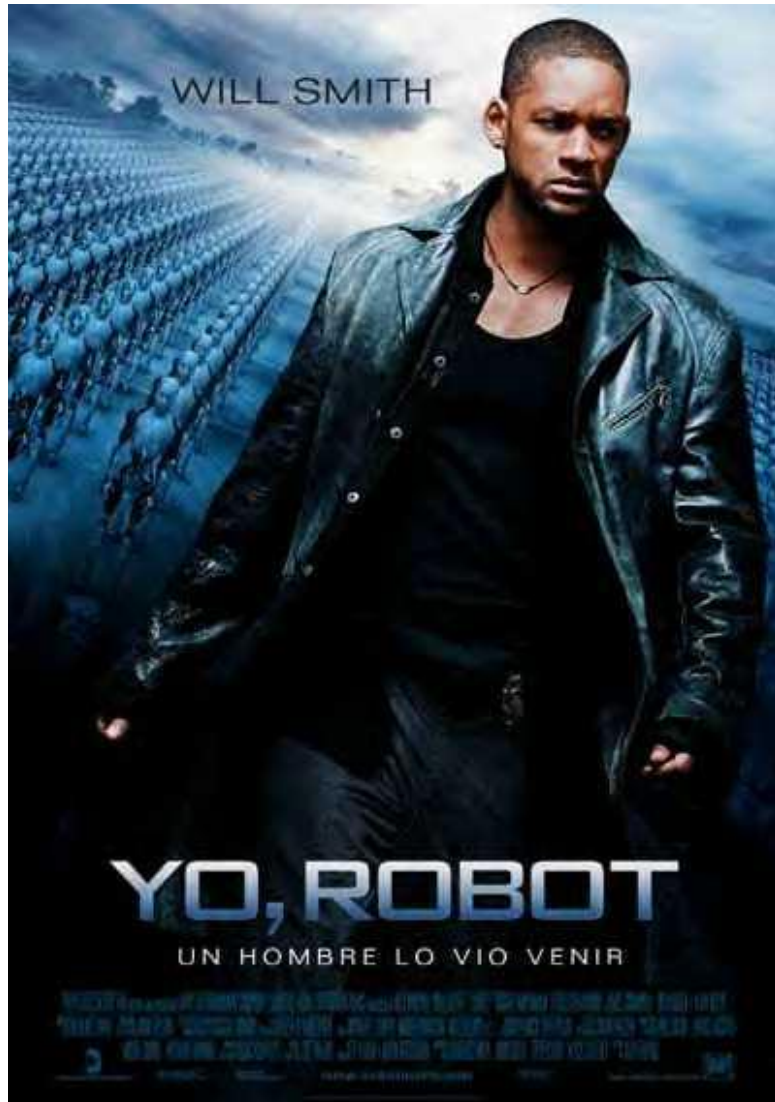




Estreno: Junio de 2001

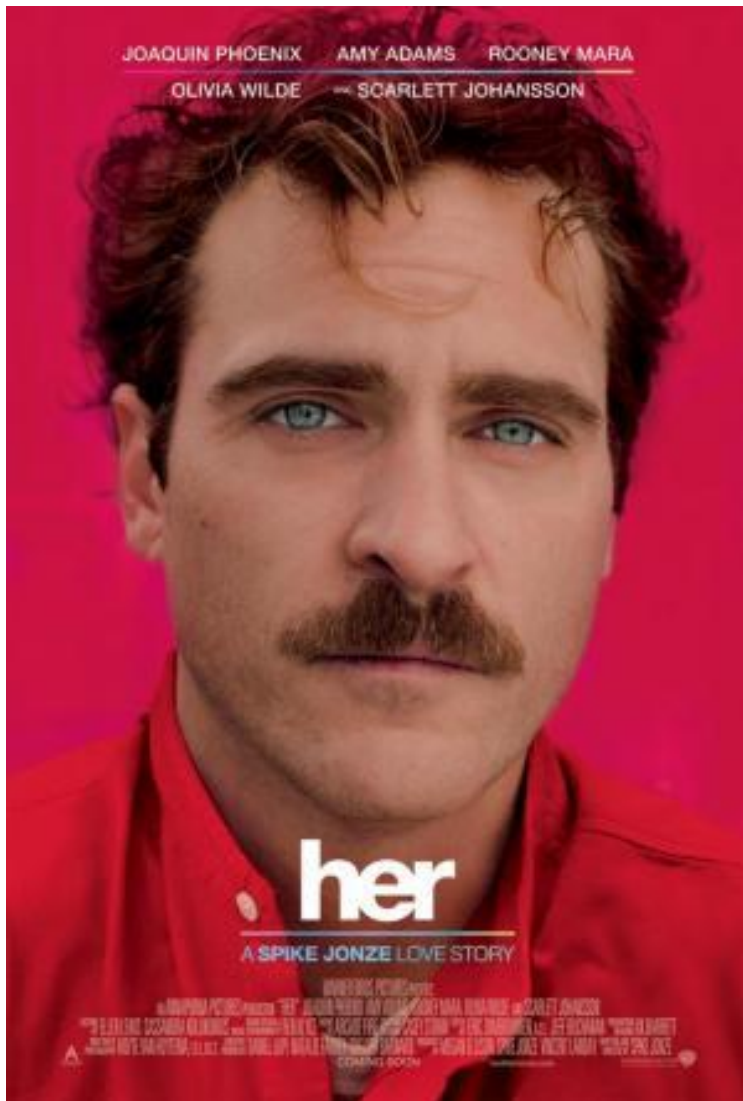
Director: Steven Spielberg





Estreno: Junio de 2004
Director: Alex Proyas





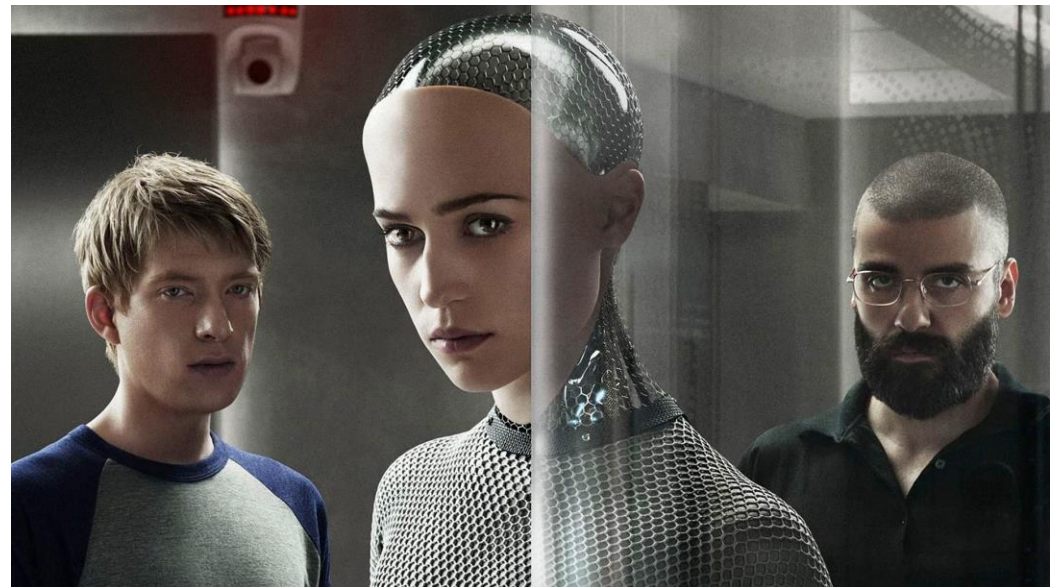
Estreno: 2013

Director: Spike Jonze



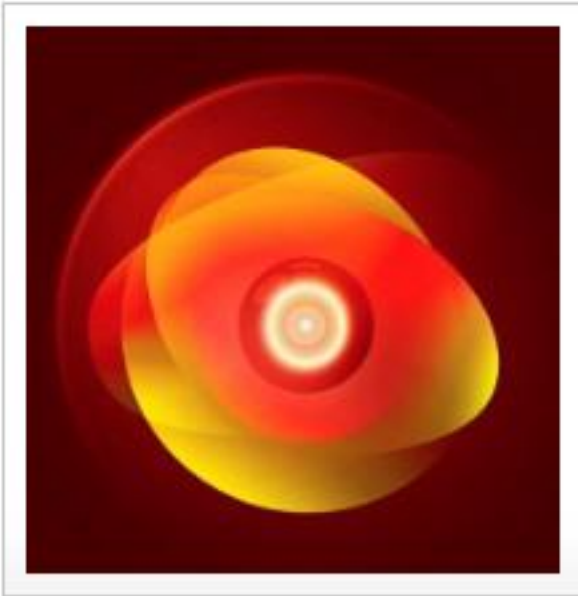


Estreno: Enero de 2015
Director: Alex Garland



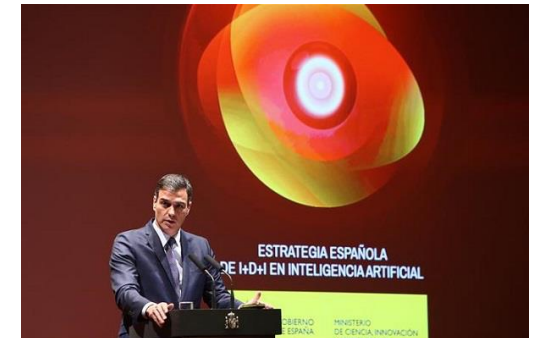


Estrategia Española de I+D+I en Inteligencia Artificial



La Estrategia Española de I+D+I en Inteligencia Artificial se configura como el elemento troncal de una visión de I+D+I propia clave para el desarrollo del marco europeo. Esta Estrategia será, el embrión de la futura Estrategia Nacional para la Inteligencia Artificial, que nos va a permitir coordinar y alinear las inversiones y políticas del Estado, redundando en la mejora de las sinergias y facilitando que las inversiones públicas y privadas estén dirigidas a incentivar el uso estas tecnologías en nuestra sociedad y economía.

https://elpais.com/elpais/2019/03/04/ciencia/1551711395_737695.html



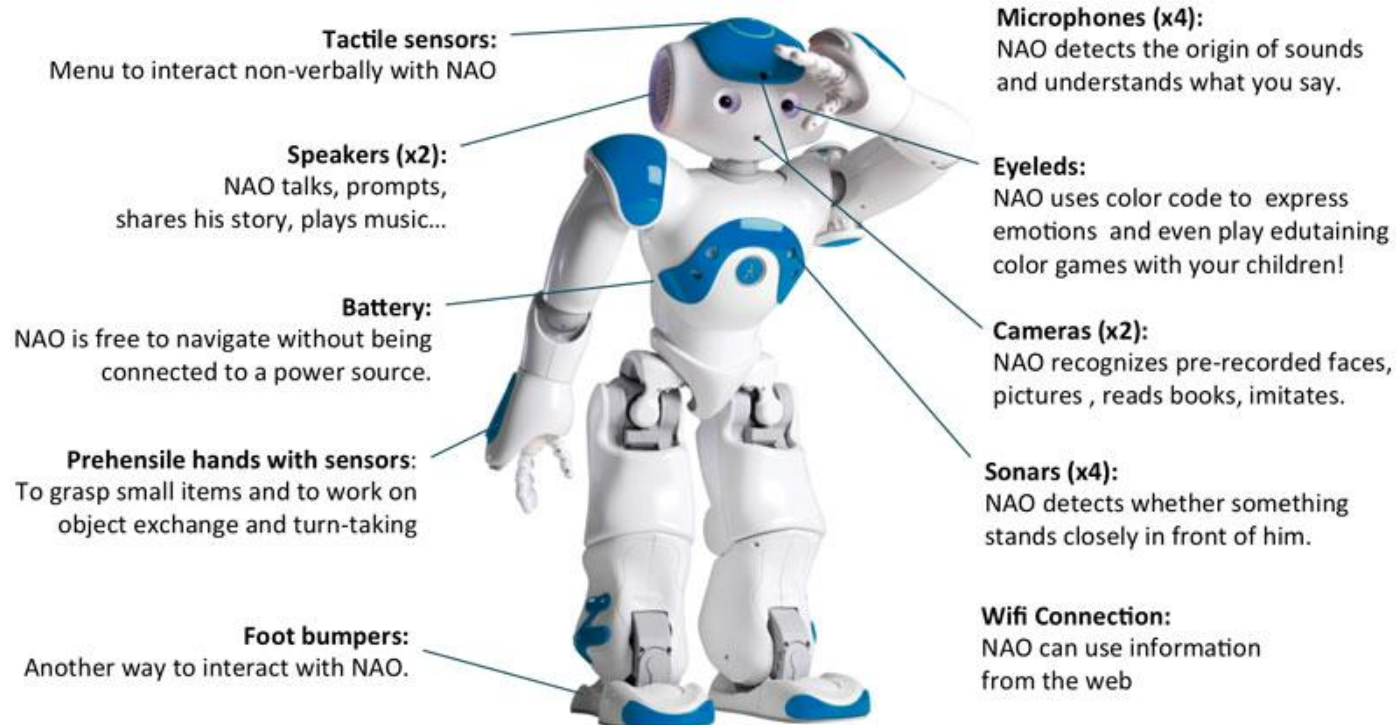
La inteligencia artificial ahorrará "miles de millones" de gasto sanitario

Así lo afirma el Ministerio de Ciencia, que destaca la mejor productividad de los profesionales que la utilizan





... mas allá de la visión del cine...





- Algoritmos de IA Clásica



- Inteligencia Computacional



- Estructuras de datos para almacenar conocimiento
- Algoritmos exploración eficiente de grafos
- Planificación
- Búsqueda de caminos
- Teoría de autómatas, etc.



- Inteligencia Computacional



- Algoritmos de IA clásica



- Simulación del comportamiento humano.
- Algoritmos capaces de razonar como humanos en ambientes restringidos.
- Modelos y algoritmos basados en heurísticas, bioinspirados, no determinísticos, probabilísticos, aproximados, evolutivos, neuronales, *fuzzy*, etc.
- Inteligencia emergente



Algoritmos de IA clásica (*Tema 4*)

+

Inteligencia Computacional (*Tema 5*)

**Gran cantidad de
recursos útiles
para
programar
videojuegos**





No todo son
gráficos,
sonidos,
física,
redes...



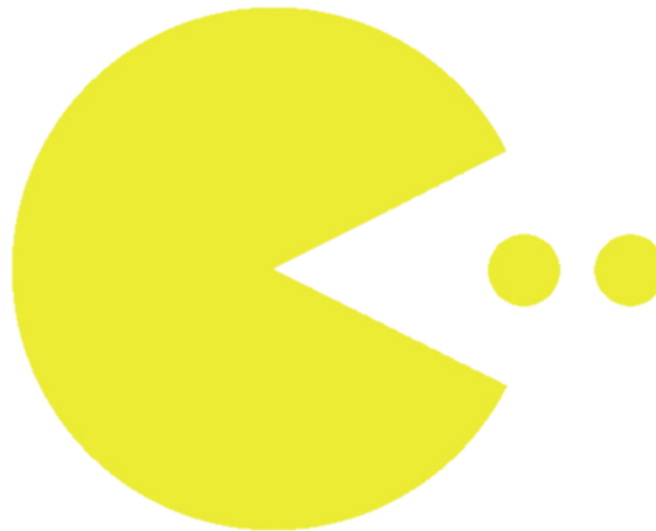
- La IA en videojuegos se usa para *(lista no exhaustiva, abierta)*:
 - Simular comportamientos de los personajes no manejados por el jugador: enemigos, amigos (no manejados por otros jugadores o en red), animales, formaciones (propias o no), etc.
 - Diálogos
 - Enemigos más inteligentes con el tiempo
 - Demos del juego (usando el personaje del jugador)
 - Generación de escenarios, mapas y terrenos
 - Buscar caminos
 - Planificación de movimientos
 - Coordinar el movimiento de grupos
 - ...



- Podríamos hablar de “niveles” de uso de la IA en VJ:
 - **Básico:** Mecanismos de IA clásicos tales como los métodos para la toma de decisión, heurística, máquinas de estados, búsqueda de caminos básica (A^*), búsqueda *minimax*, árboles de tareas/comportamientos (Halo), movimientos autónomos, búsqueda de caminos avanzada, ... casi todos los juegos la usan en alguna medida, a este nivel.
 - **Avanzado:** Juegos sofisticados y técnicas más complejas: *softcomputing*, comportamientos orientados a objetivos, aprendizaje, planificación...
 - **Futuro:** Modelos en estudio (serán tratados al final del Tema 5).



- En definitiva, la IA se utiliza para:
 - Crear (mejores) oponentes
 - Ayudar al jugador
 - Conseguir mayor inmersión
- *Aumentar el realismo y la jugabilidad*





- El reto actual no está en los *frames* por segundo, en más polígonos o en texturas de mayor resolución. Está en el **realismo** de los personajes: en que puedan **aprender y adaptarse**.





- La IA en juegos comprendería dos grandes grupos:
 - **Heurísticas:** Son comportamientos o técnicas sencillas que usaría un humano para atacar o resolver el problema de forma más eficaz en una situación dada. Las introduce el programador generalmente en el propio código. Ej: perseguir al adversario usando diagonales, atacar a los grupos más débiles primero, adelantar por la parte interior de la curva...
 - **Algoritmos:** Método computacional genérico para encontrar una solución o respuesta en diversas situaciones del juego. Ej: Planificación, búsqueda de caminos, etc.



- Otra taxonomía posible se basaría en:
 - Los que emplean búsquedas:
 - Minimax, planificación, ... dependiendo del tamaño del problema, deben limitar la búsqueda para limitar el cómputo.
 - El conocimiento está en los mecanismos de evaluación y valoración de los estados.
 - Los que no:
 - Árboles de decisión, máquinas de estados, etc.
 - Volumen de computación previsible y acotado.
 - El conocimiento está integrado en el código.



- ... pero mirando al futuro, podríamos hablar de:
 - Los que usan modelos *estáticos*:
 - Cuando el jugador descubre un **punto débil**, lo explota... igual que cuando descubre una **táctica vencedora**... y el juego pierde el interés al poco tiempo.
 - Los que usan modelos *con adaptación dinámica*:
 - Es la forma de combatir los inconvenientes citados de las estrategias, comportamientos, etc., no estáticos: poder combatirlos sin necesidad de conocerlos a priori: La **IA se adaptará** reparando debilidades y adaptándose a las estrategias del jugador.

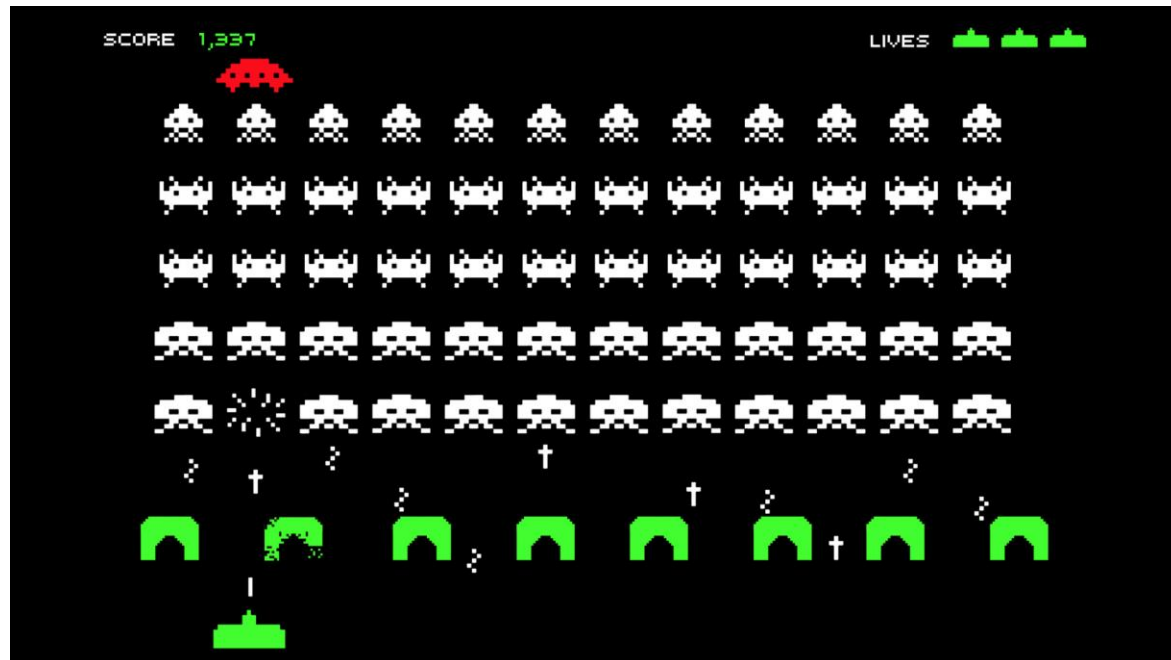


- Orígenes y Evolución:
 - Desde el nacimiento de la IA (1950 aprox), se aplicó a problemas de juegos en los primeros ordenadores. Ej: Ajedrez.
 - Desde los primeros juegos orientados al público (*Spacewar* o *Pong*), ya se incorporaban métodos o algoritmos de IA.



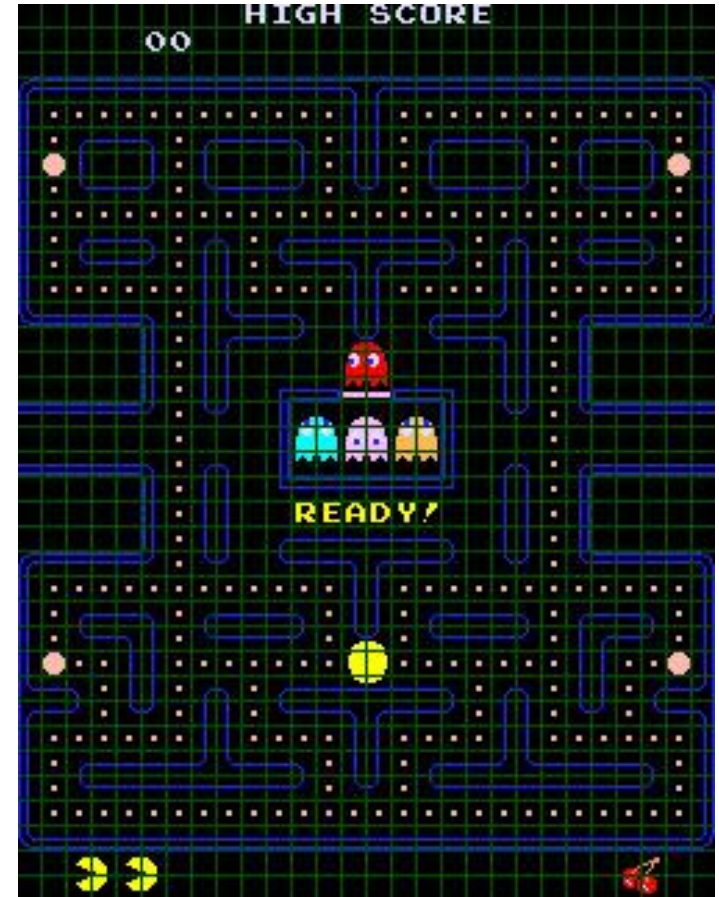


- Orígenes y Evolución:
 - Los movimientos de formaciones aparecieron en los años 70: inicialmente según patrones preprogramados, y posteriormente, reaccionando a los movimientos del jugador.





- Orígenes y Evolución:
 - En la década de los 80, juegos como Pac-Man implementaban ya algoritmos de búsqueda de caminos en laberintos para dotar de *inteligencia* al comportamiento de los fantasmas. (Hoy, Pac-Man es una plataforma clásica para probar algunas técnicas de IA)





- Orígenes y Evolución:

- La **evolución continuó** hasta la explosión de los nuevos géneros de juegos y con ellos nuevas **técnicas de IA aplicadas**.

En la década de **los 90**: Inicio de la incorporación de:

- RRNN artificiales
 - Computación evolutiva
 - Lógica difusa (*fuzzy logic*)
 - Mecanismos de aprendizaje...
 - etc
- ... se puede decir formalmente, que el *game loop* tiene un nuevo módulo desde esta década: el de IA.



- Orígenes y Evolución:
 - Battlecruiser 3000AD (en 1996) incorporaba *presuntamente* una RRNN artificial





- Orígenes y Evolución:
 - Halo en 2001 (Microsoft) reconocía las amenazas y actuaba en consecuencia: juegos menos previsibles.





- Orígenes y Evolución:
 - Far Cry en 2004 usaba IA para reaccionar al estilo del jugador *sin hacer trampas internas*, sino con la última posición conocida.



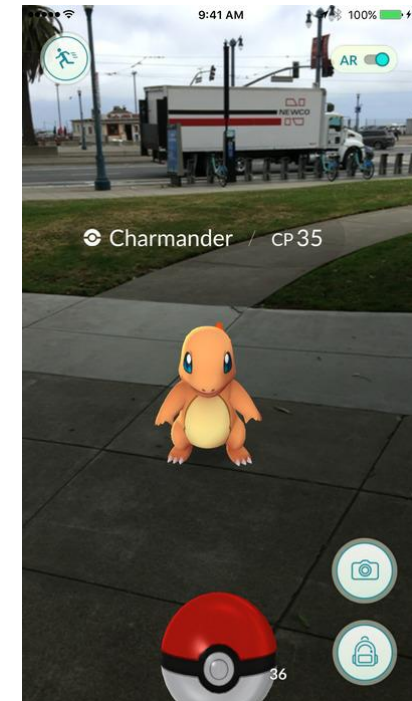


- Orígenes y Evolución:
 - FEAR en 2005 usaba planificación en tiempo real





- Orígenes y Evolución:
 - Actualmente se materializan realidad nuevas ideas como el uso de realidad aumentada, para la cual, es preciso que los caracteres se comporten de forma muy interactiva...





Índice:

1. Introducción a la IA para Videojuegos

2. Modelos de IA en Videojuegos

2.1 Juegos con adversario

2.2 Búsqueda de caminos

2.3 Mecanismos de toma de decisión

2.3.1 Máquinas de estados finitos

2.3.2 Árboles de decisión

2.4 Árboles de comportamiento

3. Bibliografía y Lecturas Complementarias



2. Modelos de IA en Videojuegos

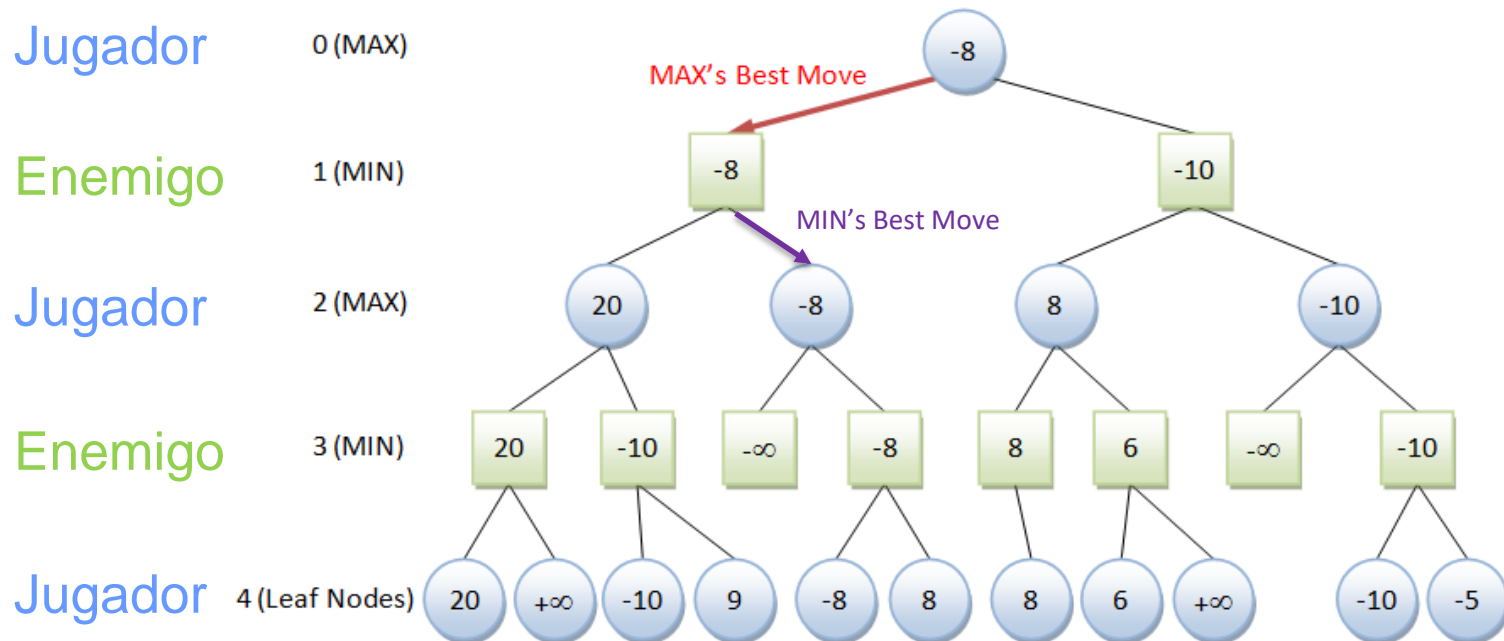
2.1 Juegos con adversario:

- Intereses de jugadores contrarios: Si uno gana, el otro pierde (*“Juegos de suma cero”*).
- Cada jugador conoce las posibles jugadas del contrario.
- No hay azar para elegir la jugada.
- Siempre se responde con la mejor jugada posible, contando con que el rival también hace su mejor jugada.
- Se afrontan generando árboles limitados o finitos (aunque casi siempre serían en realidad infinitos).
- Ej: Juegos de mesa: Ajedrez, Tic-tac-toe, Conecta 4, Damas, Go, etc.



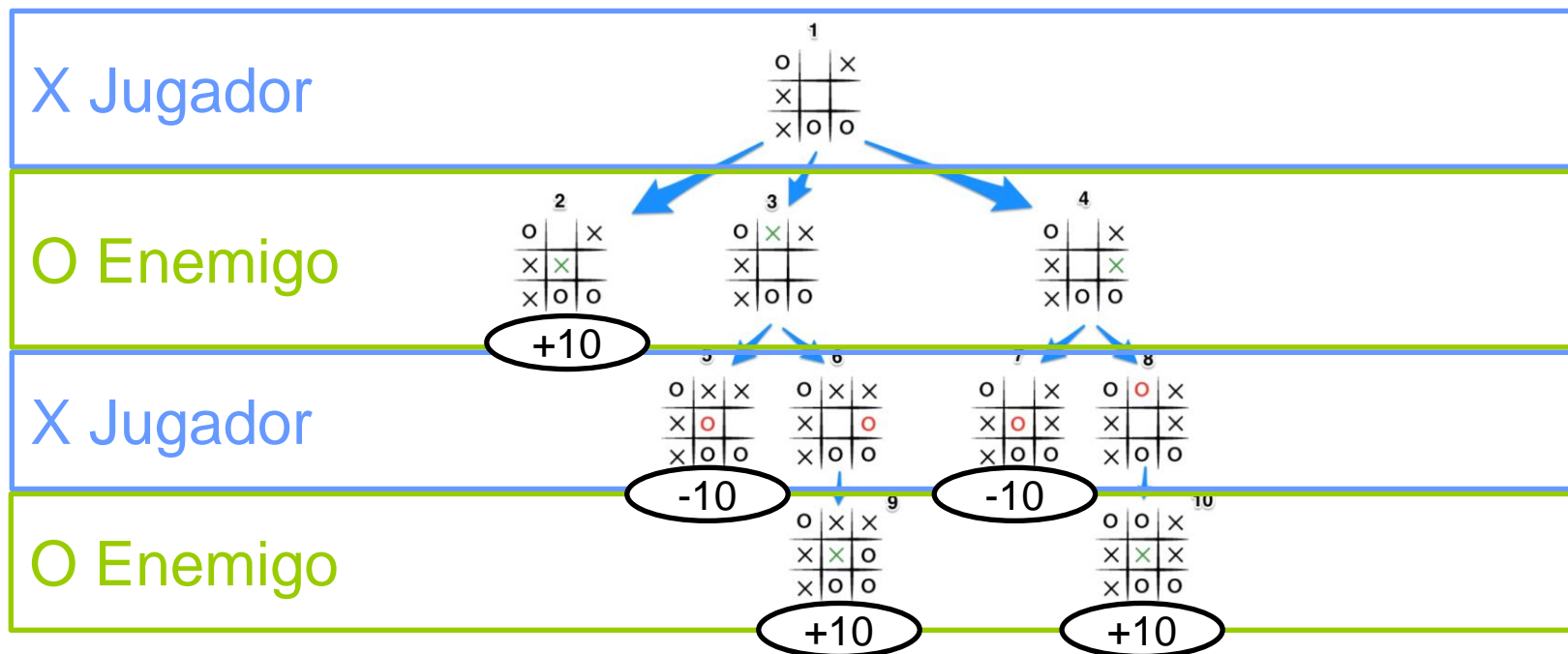


- Juegos con adversario (continuación):
 - Se genera un árbol, hasta cierta profundidad, en el que se calculan varios niveles de jugadas, y se computa la ganancia en cada una de las opciones
 - Se utilizan algoritmos como el clásico y archiconocido *MiniMax*, que hace ese cálculo retropropagando...



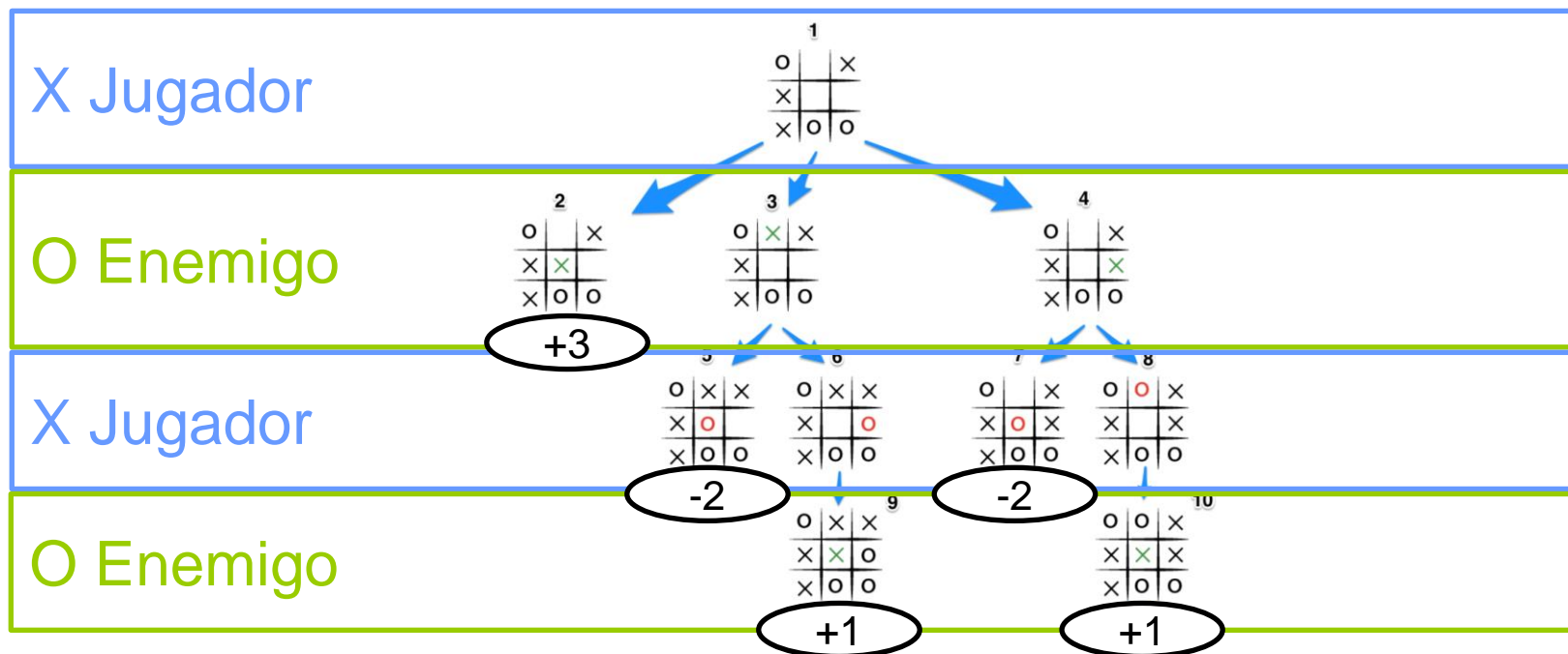


- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...



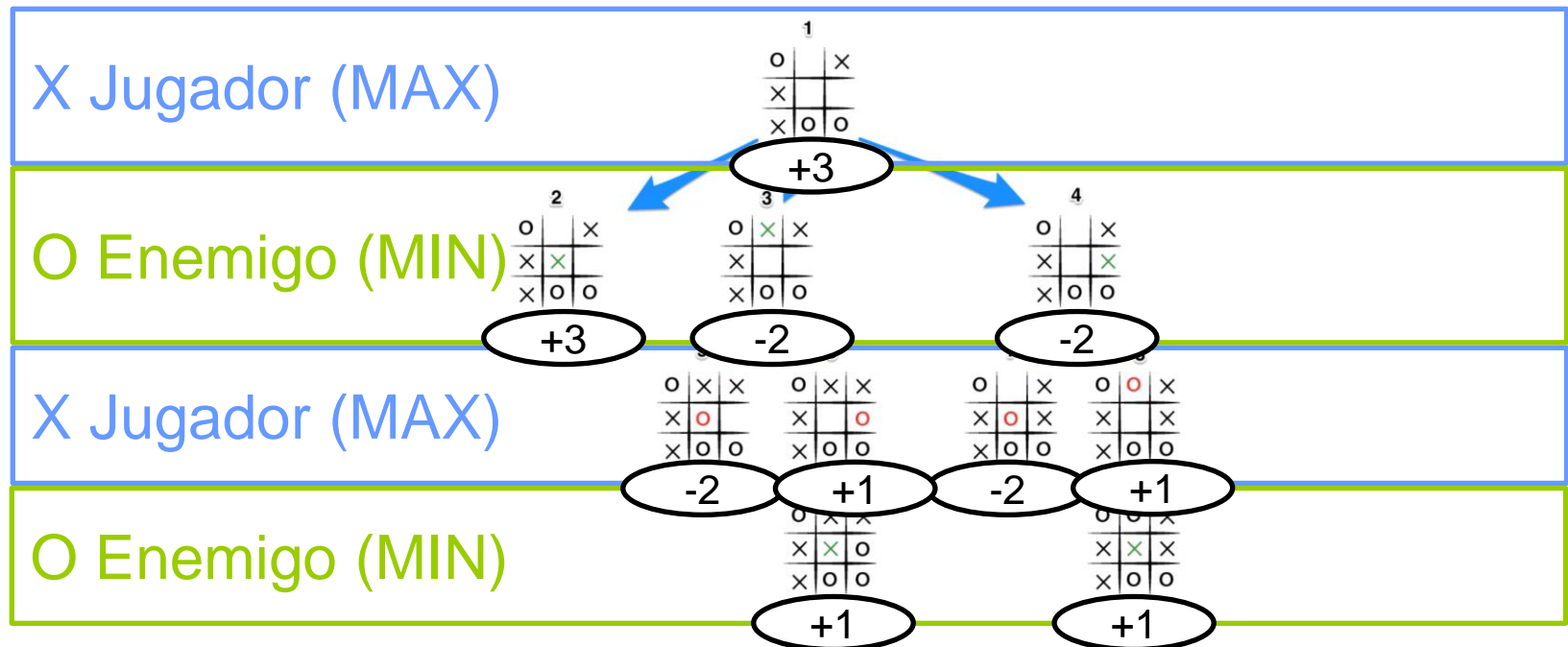


- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...





- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...





- Actividad presencial entregable:
Min-Max tictactoe

	x	
		x
o	o	

x	x	
		x
o	o	

	x	
x		x
o	o	

	x	
	x	x
o	o	

	x	x
		x
o	o	

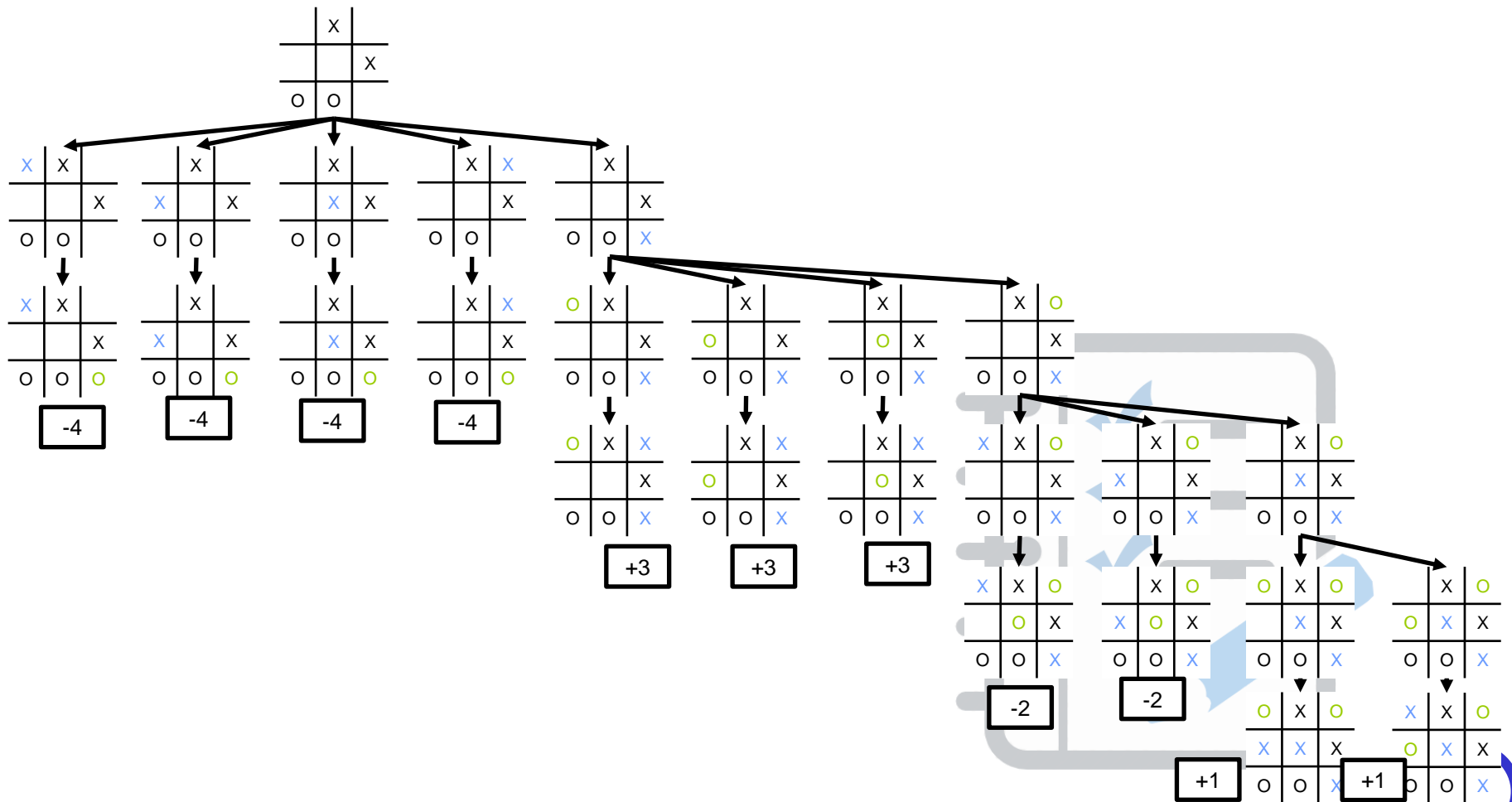
	x	
		x
o	o	x

Haced el árbol Min-Max





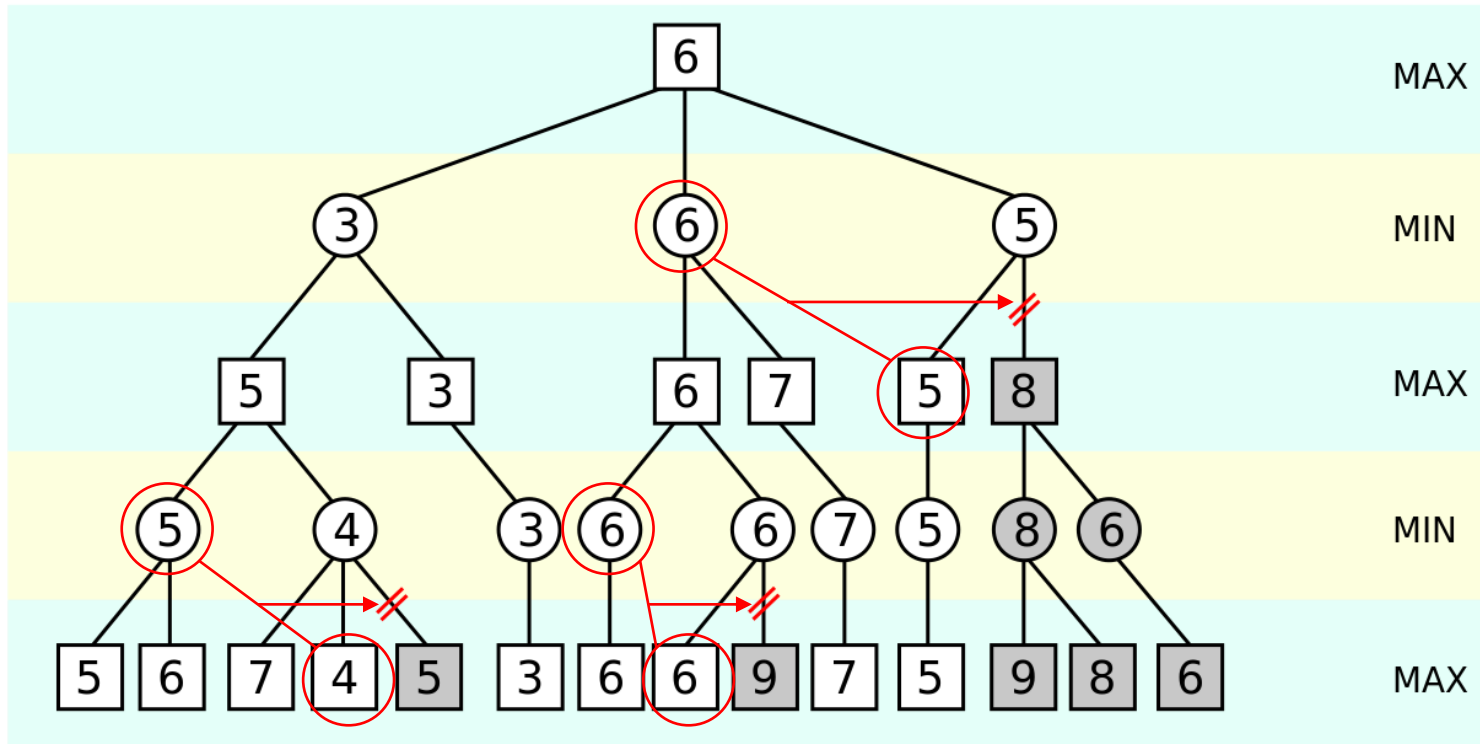
- Actividad presencial entregable:
Min-Max tictactoe







- Juegos con adversario (continuación):
 - ... u optimizaciones también muy estudiadas como la poda *alfa-beta* (*alpha beta pruning*)





- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = -Inf

O	O	X
	X	
O	X	

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)

X Jugador (MAX) O Enemigo (MIN)



- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = -Inf

O	O	X
	X	
O	X	

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = +Inf

O	O	X
X	X	
O	X	

X Jugador (MAX) O Enemigo (MIN)



- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = -Inf

O	O	X
	X	
O	X	

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = +Inf

O	O	X
X	X	
O	X	

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = -Inf

O	O	X
X	X	O
O	X	

O	O	X
X	X	O
O	X	X

Puntos = 0

X Jugador (MAX) O Enemigo (MIN)



- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Alfa = -Inf
Beta = +Inf
¿Alfa >= Beta? False
Puntos = -Inf

O	O	X
	X	
O	X	

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)

Alfa = -Inf
Beta = MIN(+Inf, 0)
¿Alfa >= Beta? Falso
Puntos = 0

O	O	X
X	X	
O	X	

Alfa = MAX(-Inf, 0)
Beta = +Inf
¿Alfa >= Beta? False
Puntos = 0

O	O	X
X	X	O
O	X	

O	O	X
X	X	O
O	X	X

Puntos = 0

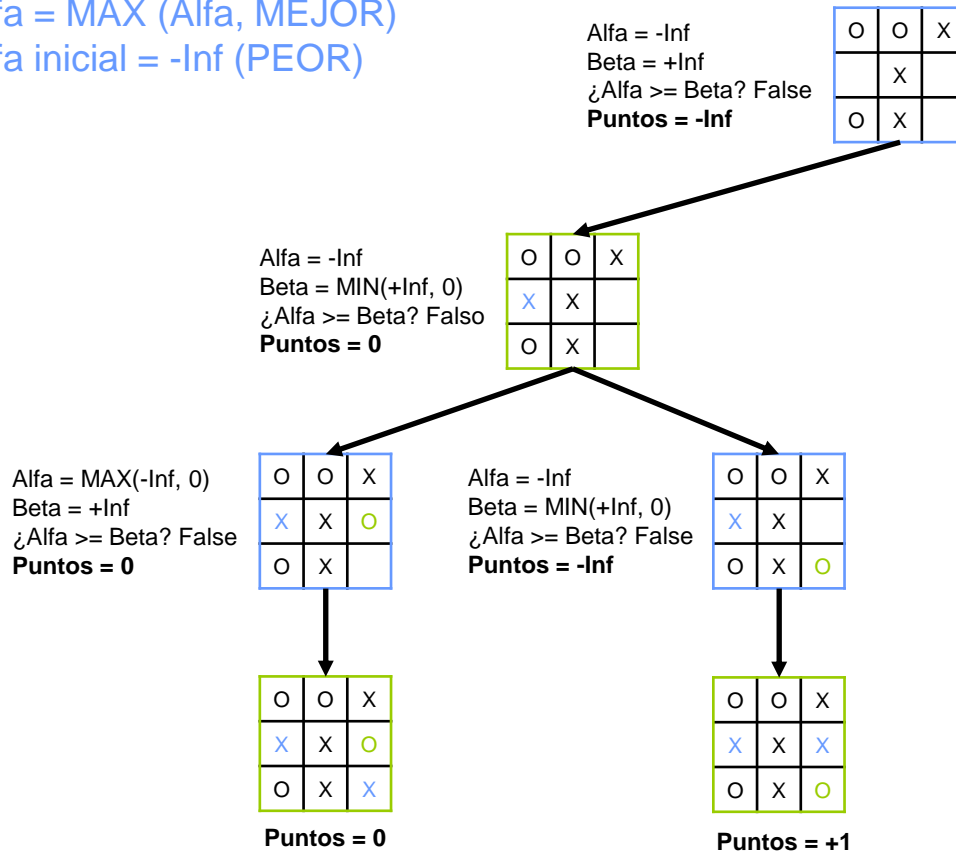
X Jugador (MAX) O Enemigo (MIN)



- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)



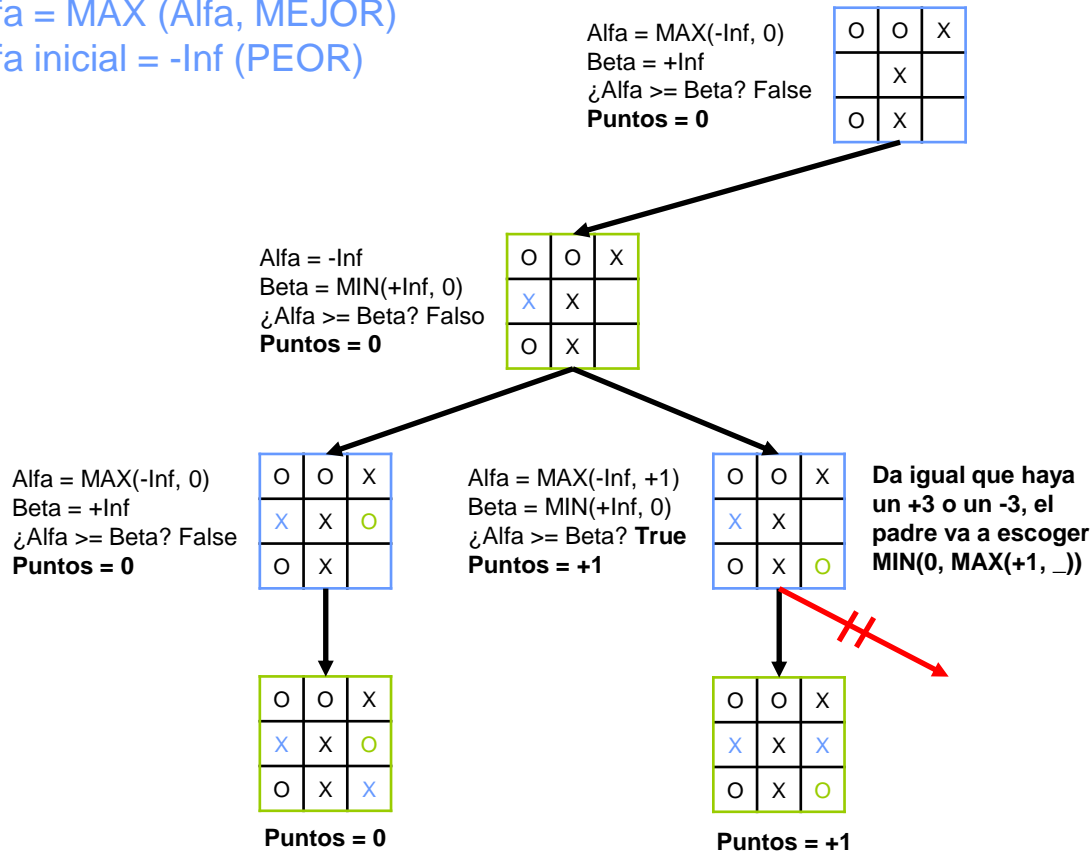
X Jugador (MAX) O Enemigo (MIN)



- Juegos con adversario (continuación):
 - Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)





- Juegos con adversario (continuación):

- Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)

Alfa = MAX(-Inf, 0)
Beta = +Inf
¿Alfa >= Beta? False
Puntos = 0

O	O	X
	X	
O	X	

Alfa = -Inf
Beta = MIN(+Inf, 0)
¿Alfa >= Beta? Falso
Puntos = 0

O	O	X
X	X	
O	X	

Alfa = MAX(-Inf, 0)
Beta = +Inf
¿Alfa >= Beta? False
Puntos = +Inf

O	O	X
	X	X
O	X	

Alfa = MAX(-Inf, 0)
Beta = +Inf
¿Alfa >= Beta? False
Puntos = 0

O	O	X
X	X	O
O	X	

Alfa = MAX(-Inf, +1)
Beta = MIN(+Inf, 0)
¿Alfa >= Beta? True
Puntos = +1

O	O	X
X	X	
O	X	O

Da igual que haya un +3 o un -3, el padre va a escoger MIN(0, MAX(+1, _))

O	O	X
X	X	O
O	X	X

Puntos = 0

O	O	X
X	X	X
O	X	O

Puntos = +1

X Jugador (MAX) O Enemigo (MIN)

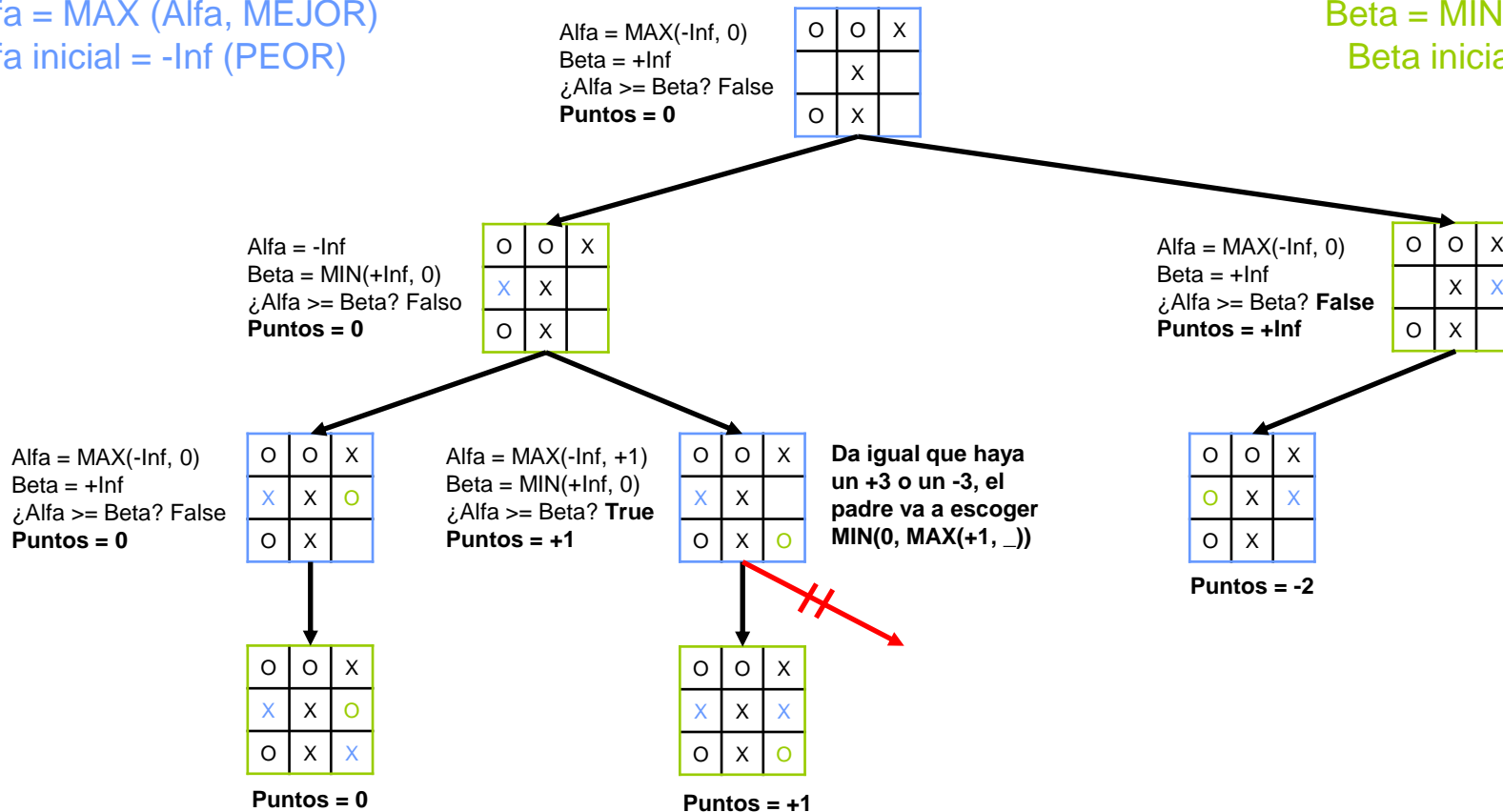


- Juegos con adversario (continuación):

- Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)





- Juegos con adversario (continuación):

- Supongamos el 3 en raya...

Alfa = MAX (Alfa, MEJOR)
Alfa inicial = -Inf (PEOR)

Alfa = MAX(-Inf, 0)
Beta = +Inf
¿Alfa >= Beta? False
Puntos = 0

O	O	X
	X	
O	X	

Beta = MIN (Beta, MEJOR)
Beta inicial = +Inf (PEOR)

Alfa = -Inf
Beta = MIN(+Inf, 0)
¿Alfa >= Beta? Falso
Puntos = 0

O	O	X
X	X	
O	X	

Alfa = MAX(-Inf, 0)
Beta = MIN(+Inf, -2)
¿Alfa >= Beta? True
Puntos = -2

O	O	X
	X	X
O	X	

Da igual que haya un +3 o un -3, el padre va a escoger MAX(0, MIN(-2, _))

Alfa = MAX(-Inf, 0)
Beta = +Inf
¿Alfa >= Beta? False
Puntos = 0

O	O	X
X	X	O
O	X	

Alfa = MAX(-Inf, +1)
Beta = MIN(+Inf, 0)
¿Alfa >= Beta? True
Puntos = +1

O	O	X
X	X	
O	X	O

Da igual que haya un +3 o un -3, el padre va a escoger MIN(0, MAX(+1, _))

O	O	X
X	X	O
O	X	X

Puntos = 0

O	O	X
X	X	X
O	X	O

Puntos = +1

O	O	X
O	X	X
O	X	

Puntos = -2

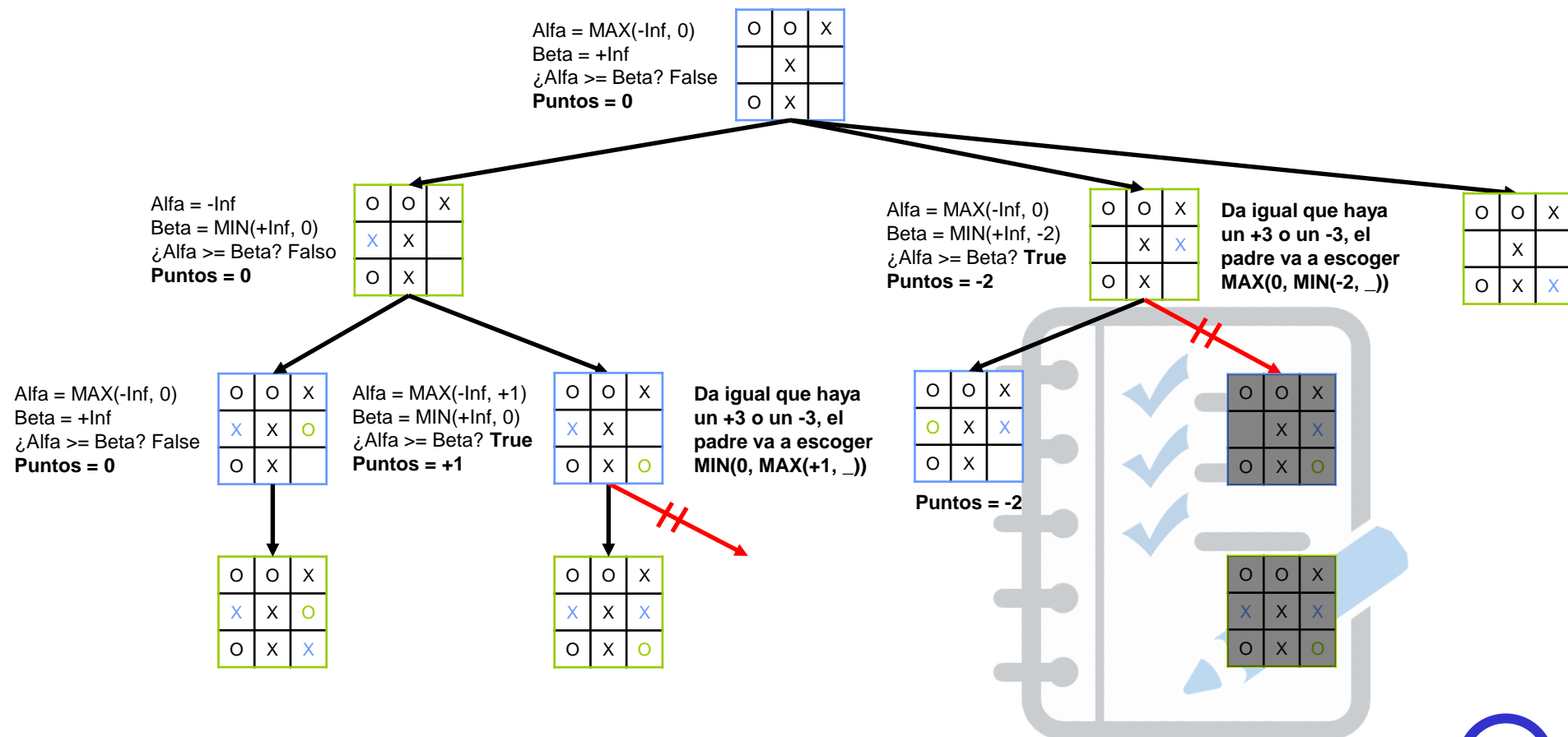
O	O	X
	X	X
O	X	O

O	O	X
X	X	X
O	X	O

X Jugador (MAX) O Enemigo (MIN)



- Actividad presencial entregable:
Min-Max Poda Alfa-Beta





Índice:

1. Introducción a la IA para Videojuegos

2. Modelos de IA en Videojuegos

2.1 Juegos con adversario

2.2 Búsqueda de caminos

2.3 Mecanismos de toma de decisión

2.3.1 Máquinas de estados finitos

2.3.2 Árboles de decisión

2.4 Árboles de comportamiento

3. Bibliografía y Lecturas Complementarias