

Tarea 2

Fecha de entrega: Lunes 28 de Mayo 2023

ALUMNOS: BEATRIZ ERRÁZURIZ CAMUS NRO.19638906

Parte 2

Actividad 3: Estudio Analítico Para analizar y reflejar el algoritmo *minimax* implementado utilizaremos en el primer caso un tablero con 7 columnas de manera de que el jugador blanco logré predecir mejor debido a anticipar una mayor cantidad de estados del juego.

Para el segundo casos la variaciones de columnas fueron 3, 5, 7, 9, 11, 13, 15, 17, 19, 21.

	Negro = Random, Blanco = Minimax	Negro = Minimax , Blanco = Minimax
1	Blanco: 106, Negro: 96	Blanco: 97, Negro: 101
2	Blanco: 100, Negro: 66	Blanco: 105, Negro: 76
3	Blanco: 103, Negro: 30	Blanco: 104, Negro: 90
4	Blanco: 101, Negro: 51	Blanco: 102, Negro: 82
5	Blanco: 91, Negro: 100	Blanco: 102, Negro: 83
6	Blanco: 88, Negro: 104	Blanco: 104, Negro: 92
7	Blanco: 100, Negro: 66	Blanco: 106, Negro: 86
8	Blanco: 106, Negro: 88	Blanco: 100, Negro: 87
9	Blanco: 104, Negro: 73	Blanco: 104, Negro: 63
10	Blanco: 52, Negro: 106	Blanco: 106, Negro: 74

Cuadro 4: Simulación de 10 juegos para analizar algoritmo *minimax*

Caso 1: Negro: Random, Blanco: Minimax

En un caso ideal, el algoritmo *minimax* claramente es mejor que la estrategia *random*. Si el jugador conoce todos los estados posibles del juego y logra eficientemente desarrollar una estrategia que minimice su máxima pérdida, entonces puede derrotar a un jugador *random* y probablemente lo logrará con una diferencia considerable, considerando que el oponente no realiza jugadas estratégicas. Sin embargo, realizar una búsqueda completa de este árbol de estados a menudo requiere una cantidad excesiva de tiempo y memoria, a pesar de las técnicas de *poda alfa y beta*.

En este caso particular, observamos que el jugador blanco gana la mayoría de los juegos, pero la diferencia de puntuaciones no es realmente significativa en los datos. El jugador blanco logra obtener una ventaja debido a que el algoritmo le permite predecir la siguiente jugada de su oponente, ya que conoce de antemano la mejor jugada que éste puede hacer. De esta manera, el jugador blanco busca minimizar su máxima pérdida.

Sin embargo, este enfoque no se ha logrado de manera completamente eficiente, ya que en muchas ocasiones la diferencia de puntuaciones no fue considerable o hubo jugadas en las que, a pesar del uso del algoritmo, el jugador negro resultaba ganador. Una de las posibles razones podría ser el tamaño o la dimensión del tablero y la profundidad del árbol, ya que esto dificulta que el algoritmo *minimax* examine todas las posibles jugadas

hasta la profundidad deseada. Dado que la profundidad del árbol está limitada a 3 y el tamaño del tablero es grande, es posible que el algoritmo *minimax* no disponga de suficiente información para tomar decisiones óptimas, lo que podría llevar a pasar por alto estrategias o movimientos importantes, ya que no puede explorar todos los estados posibles del juego. La cantidad en la que el enfoque *minimax* supera al enfoque *random* dependerá precisamente del tamaño del tablero y la profundidad del árbol, es decir, de la cantidad de información que tenga el jugador.

Otras razones podrían ser que el jugador negro utilice estrategias aleatorias (*random*) que, por coincidencia, le beneficien, o simplemente que el jugador blanco no haya tenido en cuenta ciertos estados de juego en su enfoque estratégico.

Caso 2: *Negro: Minimax, Blanco: Minimax*

En este caso, observamos que el jugador blanco tuvo una ventaja considerable sobre el negro, a pesar de que ambos utilizaron el algoritmo *minimax*. Una razón posible puede ser que el orden de juego (quién comienza) tiene un impacto significativo, ya que el jugador blanco tiene más oportunidades para tomar decisiones estratégicas antes que el jugador negro.

Por otro lado, la dimensión del tablero también afectó el resultado de las jugadas. En los casos donde las dimensiones eran mayores, era más difícil para el jugador negro predecir qué movimiento hacer basándose en las jugadas del jugador blanco debido a la profundidad limitada del árbol. Esto podría haber llevado a que el jugador negro no predijera ciertas jugadas y colocara ovejas en filas donde el jugador blanco ya había colocado, sin considerar cómo esto podría perjudicarlo en el futuro. En el primer caso, donde las columnas eran 3, podríamos deducir que el juego fue más "justo", ya que el jugador negro tenía un porcentaje mayor de información sobre los posibles estados del juego, lo que le permitió predecir de manera más efectiva qué jugada hacer sin perjudicarse a sí mismo.

Los problemas de tiempo y memoria son, efectivamente, limitaciones para la predicción de jugadas. Para mejorar esto, se podría utilizar una limitación en la profundidad del árbol, pero también sería más eficiente considerar el uso de heurísticas.