
Práctica 3

Búsqueda con Adversario

Inteligencia Artificial



ÍNDICE

1. Mejoras Poda Alfa-Beta desarrolladas.....	3
2. Heurísticas propuestas.....	4
3. Comparativa Podas+Heurísticas vs Ninjas.....	6
4. Análisis de las Comparativas.....	7
5. Reflexión Personal.....	9

1. Mejoras Poda Alfa-Beta desarrolladas

1. **Poda alfa-beta:** Se trata de la poda alfa-beta básica. La implementación es necesaria para llevar a cabo las mejoras posteriores.
2. **Poda Probabilística:** Tal y como menciona el guión, para reducir el número de nodos evaluados se tienen que llevar a cabo podas más agresivas. En el caso de esta mejora, las podas se realizan en el momento en el que el resultado actual no da expectativas de superar al mejor resultado hasta el momento.

Por ello, relajamos la condición de poda definiendo un umbral de tolerancia que, en mi caso, se trata de una constante. Si la diferencia entre alfa y beta es menor que el valor de dicha constante, se asume que no merece la pena seguir buscando y se poda.

3. **Ordenación de movimientos:** Al igual que en la mejora anterior, el objetivo es reducir el número de nodos evaluados, aumentando la eficacia del algoritmo. Para ello, ordenamos los movimientos antes de seguir expandiendo, es decir, una vez generados los nodos hijo, los ordenamos según el valor de su heurística (mayor valor para nodos MAX, menor para nodos MIN). Una vez ordenados, comenzamos a expandir los nodos hijos en el orden en el que han sido ordenados.

De esta forma, se exploran los movimientos más prometedores primero, llegando a podar una gran cantidad de ramas con nodos que probablemente no conduzcan a una solución óptima.

2. Heurísticas propuestas

1. **ValoracionTest:** La heurística tiene las siguientes características:
 - a. Si el jugador gana, se devuelve la constante positiva “gana”. Si ocurre lo contrario, se devuelve “pierde”.
 - b. Se consideran dos factores:
 - i. Ficha en casilla segura suma 1 punto.
 - ii. Ficha en casilla meta suma 5 puntos.
 - c. Se calcula y devuelve la diferencia entre los puntos del jugador que llama a la heurística y el oponente.

2. **miValoracion1:** Para realizar esta heurística, se han tenido en cuenta los siguientes puntos clave a lo largo del desarrollo de la partida. Las ponderaciones dadas para estas situaciones son las siguientes:
 - a. Por cada ficha en meta se suma la constante definida como “PESO_META”.
 - b. Se calcula la distancia de la ficha a la meta y se resta dicha distancia a la puntuación.
 - c. Si el jugador tiene una ficha en una casilla segura sumamos “PESO_SEGURA”.
 - d. Sumo “PESO_BARRERA” por tener una barrera con fichas del mismo color.
 - e. Consultamos el valor de “goal_move”. Dependiendo de su valor, se suma “PESO_BONUS_GOAL” o no.
 - f. Consultamos el valor de “eating_move”. Dependiendo de su valor, se suma “PESO_BONUS_EAT” o no.
 - g. Finalmente se devuelve la diferencia entre la puntuación del jugador que llama a la heurística y el oponente

3. **miValoracion2:** Una segunda versión de la heurística “miValoración1” , que tiene nuevas características esenciales adicionales a las de la primera versión. Las ponderaciones añadidas son:
 - a. Si una ficha permanece en casa, se suma la constante negativa “PESO_CASA”.
 - b. Si se saca un rebote, se suma la constante negativa “PESO_REBOTES”.
 - c. Se suma la constante “PESO_MOVILIDAD” por cada dado disponible.
 - d. Si resulta que el jugador tiene turno doble, se suma “PESO_TURNO_DOBLE”.
 - e. Para cada ficha, si se tiene la posibilidad de comer una ficha enemiga (se considera que hay posibilidad cuando la distancia de la ficha a la ficha del oponente es pequeña y esta no está en una casilla segura) se suma “PESO_COMER_POSIBLE”.
 - f. Siguiendo una lógica inversa a la del punto anterior, si una de nuestras fichas es vulnerable sumamos la constante negativa “PESO_VULNERABLE”.
 - g. Para aumentar la agresividad una vez se llevan varios turnos jugados, se cambian los valores de “PESO_DISTANCIA” y “PODER_COMER_POSIBLE”
 - h. Finalmente se devuelve la diferencia entre la puntuación del jugador que llama a la heurística y el oponente.

- 4. miValoracion3:** Nueva versión de “miValoracion2” y la definitiva en esta práctica. Con ella, se intenta ganar a la gran mayoría de ninjas. Los puntos claves de esta, además de los de la versión anterior, son:
- a. Se añade una nueva constante “PESO_PELIGRO” que afina a la constante “PESO_COMER_POSIBLE”. La idea de esta es sumar cuando la ficha del oponente puede ser comida y la distancia de alguna de nuestras fichas a la del oponente es menor a considerada en “miValoracion2”.
 - b. Se añade una funcion lambda para comprobar la condición de vulnerabilidad o posibilidad de comer una ficha.
 - c. Para darle un comportamiento algo más dinámico a la agresividad empleada en la versión anterior, se va actualizando un valor llamado mix_agresiva que condiciona el comportamiento del jugador a lo largo de la partida. Se hace uso de este valor para dar importancia a la acción de comer fichas (del oponente) e ignorar la vulnerabilidad nuestras fichas (lo cual puede ir en nuestra contra).
 - d. Una vez tenemos la ponderación considerada con el valor mix_agresiva y el valor que asignamos en la anterior versión, seleccionamos el máximo entre estos dos.

3. Comparativa Podas+Heurísticas vs Ninjas

En este apartado, se deja un registro de las sucesivas mejoras de las heurísticas con algunas de las mejoras del alfa beta (y he añadido las distintas versiones del algoritmo minimax) con sus respectivos resultados:

ID	Algoritmo	Heurística	Victorias Ninja 1	Victorias Ninja 2	Victorias Ninja 3
0	Poda alfa-beta	valoracionTest	0/2	0/2	0/2
1	Poda Probabilística	miValoracion3	1/2	2/2	2/2
2	Poda Probabilística	miValoracion1	0/2	2/2	0/2
3	Poda Probabilística	miValoracion2	2/2	2/2	1/2
4	Ordenacion	miValoracion2	1/2	1/2	0/2
5	Poda Probabilística	valoracionTest	0/2	0/2	0/2
9	Minimax	miValoracion3	1/2	0/2	0/2
12	Minimax Limitado	miValoracion3	1/2	0/2	0/2

4. Análisis de las Comparativas

Los resultados de la tabla muestran la eficacia de algunas combinaciones algoritmo+heurística que han sido consideradas importantes. En este análisis nos vamos a centrar en las mejoras introducidas versión a versión con el fin de solventar debilidades.

1. Heurísticas: evolución, ventajas e inconvenientes

Heurística	Ventajas	Inconvenientes
valoracionTest	Utilidad como punto de partida.	Básica. No considera factores tales como la vulnerabilidad de las piezas, distancia a casillas significativas, barreras... En resumen, la gran mayoría de puntos que aportan el significado a la heurística.
miValoracion1	Presenta algo de destreza en lo referente a la agresividad o ataque con la consideración de las barreras, metas, comer fichas...	Carece de un factor defensivo tal como considerar la posible pérdida de fichas. Tan solo suma si ponderaciones positivas si tenemos factores relacionados con ataque tales como barreras, comer fichas... Por ello solo gana al Ninja 2 en la prueba que se ha realizado.
miValoracion2	Aporta la capacidad “defensiva” de la que carecía la versión anterior. En este caso se considera un factor determinante en la labor defensiva como es la vulnerabilidad de las propias fichas (si no tenemos estas a salvo y están cercanas a una del oponente).	Se lleva a cabo un intento muy básico de implementación de lo que podemos denominar un comportamiento “agresivo dinámico”.
miValoracion3	Mejorar la versión anterior, aportando un comportamiento que modifica la agresividad del jugador conforme la partida va llegando a su fin.	La mecánica implementada, reduce la capacidad defensiva del jugador en los instantes finales de la partida, lo que puede jugar en nuestra contra. La eficacia no mejora significativamente con respecto a la versión anterior.

2. Algoritmos y mejoras de la poda: ventajas e inconvenientes

Mejora	Ventajas	Inconvenientes
Minimax	Implementación sencilla y buena para ir probando mejoras.	Excesivo coste computacional. Alcanza poca profundidad. Resultados de baja calidad independientemente de la heurística.
Ordenación	Mejora parcial con respecto a la eficiencia de la poda alfa-beta.	Impacto nulo si la heurística no es lo suficientemente buena.
Poda Probabilística	Reducción de la complejidad del espacio de búsqueda sin sacrificar calidad. Mejora con mejor respuesta en las partidas contra los ninjas.	No garantiza encontrar una solución óptima. Si los valores asociados a los nodos (por la heurística) son muy dispersos, el efecto de esta mejora se reduce drásticamente.

5. Reflexión Personal

El desarrollo de los algoritmos ha supuesto un reto significativo debido a las restricciones de tiempo que teníamos para mover una pieza y la limitación en la generación de un número máximo de nodos.

En cuanto a la mejora de las podas, implementarlas no ha sido complicado. El reto ha sido intentar reducir el tiempo de búsqueda sin afectar a la calidad de la solución.

La tarea más compleja ha sido el desarrollo de la heurística. Las versiones iniciales, a pesar de que eran simples y bastantes sencillas de implementar, no eran muy eficaces. Para llegar a unas heurísticas más completas y eficaces ha sido necesario identificar y comprender las reglas del juego y usar las funciones disponibles para consultar la información de este. Además, se requería ser muy preciso en las ponderaciones/puntos aportados en las distintas partes de la heurística, pues una mala elección puede alterar la eficacia de la heurística y, por tanto, el comportamiento del agente.