

Métodos de Búsqueda

Trabajo Práctico 1

Grupo 10

Sistemas de Inteligencia Artificial

2023 - 2Q

65913 - Clara Marto

65929 - Nuno Braz

65997 - Gabriel Polania

Contenido

01 8-PUZZLE

02 GRID WORLD

03 CONCLUSIONES

01

8-PUZZLE

Estructura de Estado

- Tablero 3x3 con números del 1 al 8, más un espacio en blanco
- Número total de estados posibles es 9!
- Para solucionarlo es necesario mover los números adyacentes al espacio en blanco
- Los movimientos válidos son arriba, abajo, izquierda y derecha
- Es posible representar este problema con un grafo dirigido

5	7	3
8	2	
1	6	4

Estado Inicial



1	2	3
8		4
7	6	5

Estado Final

Heurísticas admisibles no-triviales

01

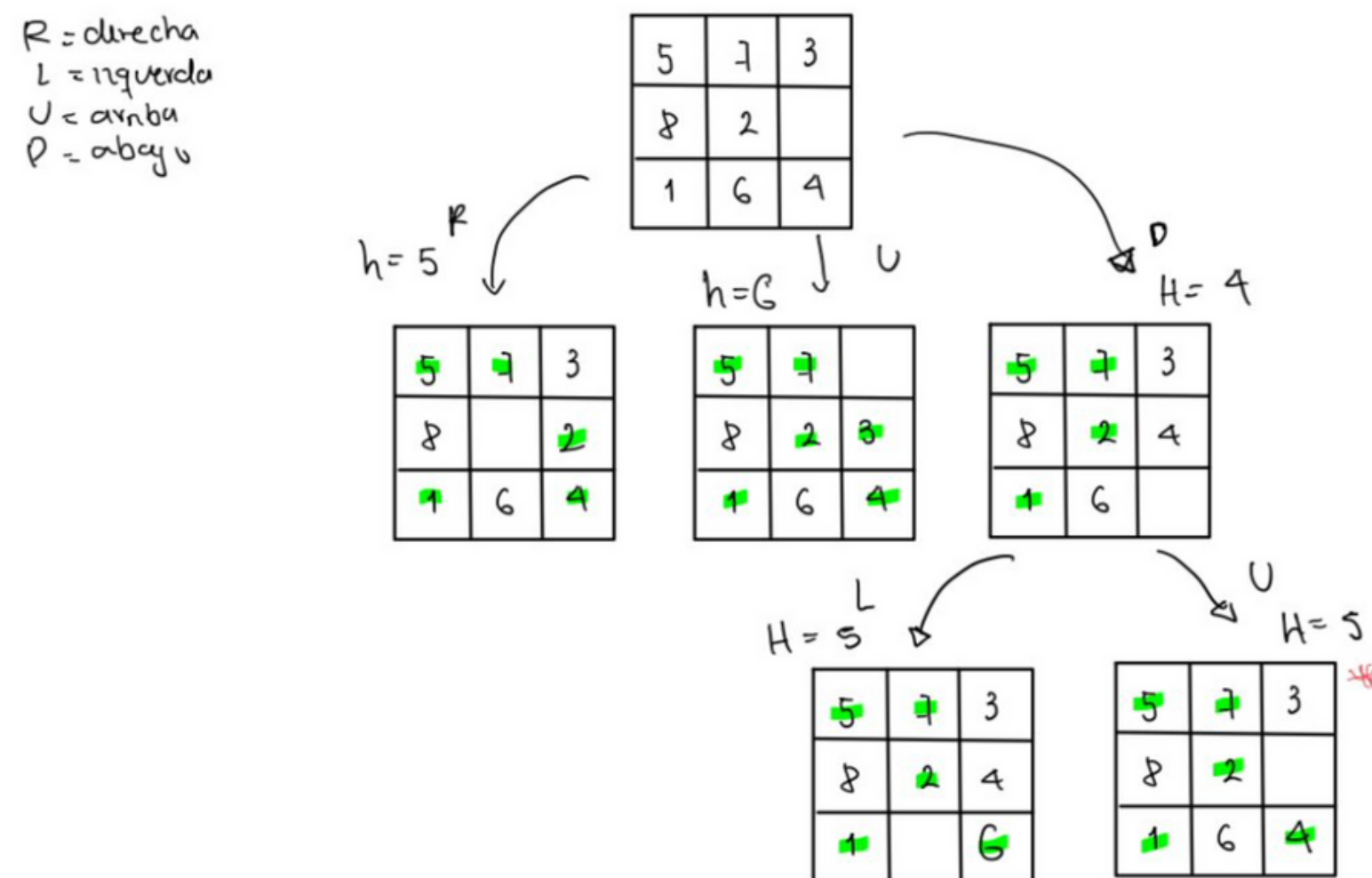
Número de piezas fuera de lugar

02

Distancia de Manhattan

Número de piezas fuera de lugar

- Implica sumar los números en posiciones incorrectas en un tablero
- El objetivo principal es minimizar la cantidad de movimientos necesarios para resolver el problema
- Busca optimizar la búsqueda de la solución al considerar el costo asociado a los movimientos



Número de piezas fuera de lugar

Método de Búsqueda

- **Greedy Best-First**
- Este algoritmo se clasifica como informado porque utiliza una función heurística para guiar la búsqueda
- Se centra en explorar eficientemente estados, optando por el estado con la menor heurística en cada paso
- Es importante destacar que no garantiza la optimalidad, lo que significa que no siempre encontrará la solución óptima.

Distancia de Manhattan

- Utiliza la distancia entre la posición actual y la posición final deseada de cada número en el tablero
- La fórmula utilizada para calcular esta distancia es $h(n) = |x1 - x2| + |y1 - y2|$
- Esto permite calcular las distancias horizontales y verticales entre las posiciones actual y final
- Se calcula la distancia Manhattan para cada número en el tablero y luego se suman todas estas distancias para obtener la heurística total.
- Nunca sobreestima el costo de alcanzar la solución

Distancia de Manhattan

Método de Búsqueda

- **A***
- A* funciona especialmente bien cuando se utiliza una heurística admisible, como la Distancia Manhattan
- Mantiene una lista de estados a explorar y evalúa cada estado utilizando una función de coste combinada
- La función de coste en A* tiene en cuenta el coste actual y una estimación del coste restante para asegurar una búsqueda eficiente

02

GRID WORLD

Implementación

Heurísticas utilizadas

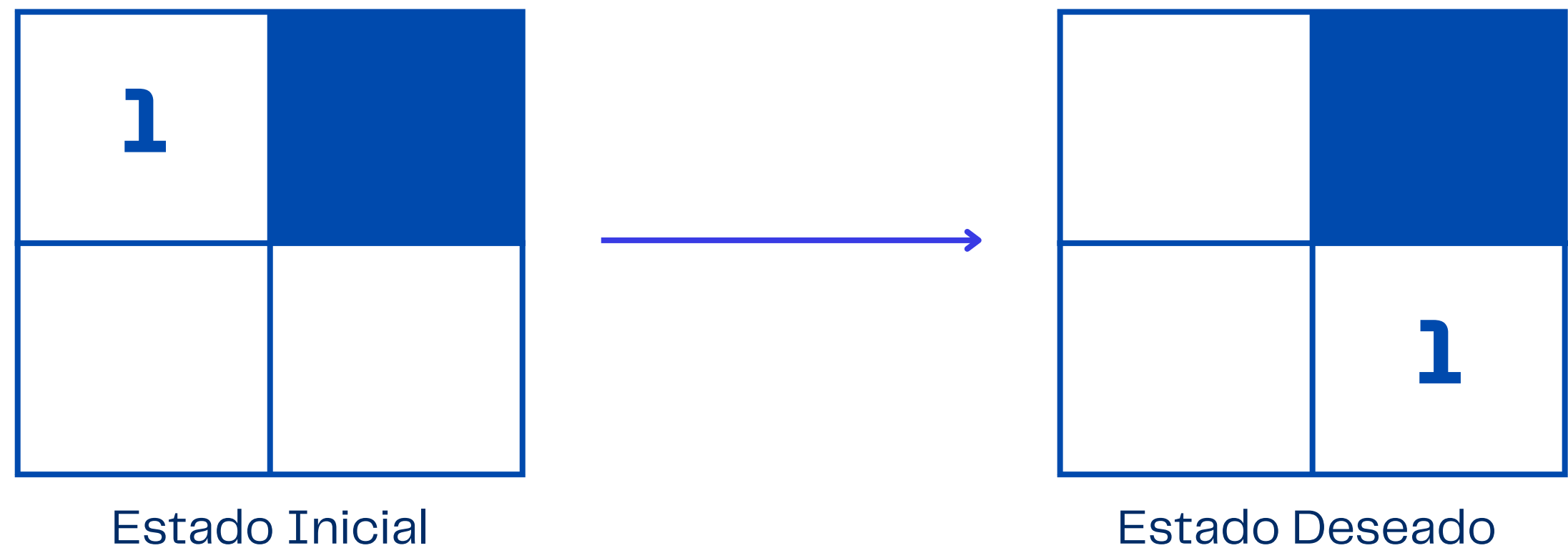
1. Distancia Manhattan del coche más próximo a su meta

- La distancia de Manhattan entre el coche y su objetivo se calcula sumando las diferencias en las coordenadas X y Y entre el coche y la meta, sin considerar diagonales
- Esta métrica no tiene en cuenta obstáculos ni obstáculos en el camino, simplemente mide la distancia en términos de cuadrícula

2. Suma de las distancias manhattan de todos los coches a sus objetivos

- Calcula la suma de las distancias de Manhattan de todos los coches a sus respectivas metas y utiliza esta suma como estimación de la "distancia total" que los coches deben recorrer para llegar a sus metas

Tablero Fácil



Tablero Fácil

Resultados

Algoritmo	Costo solución	Nodos expandidos	Nodos en frontera	Tiempo (s)
BFS	5	4	1	0.014
DFS	5	3	0	0.0
Greedy H1	5	3	1	0.0
Greedy H2	5	3	1	0.0
ASTAR H1	5	3	1	0.0
ASTAR H2	5	3	1	0.0

Tablero Intermedio

1		
2		
	3	

Estado Inicial



3		2
		1

Estado Deseado

Tablero Intermedio

Resultados

Algoritmo	Costo solución	Nodos expandidos	Nodos en frontera	Tiempo (s)
BFS	44	24285	17260	0.541
DFS	945	214	29	0.0
Greedy H1	44	22	12	0.0
Greedy H2	44	106	19	0.011
ASTAR H1	44	54	19	0.001
ASTAR H2	44	231	30	0.007

Tablero Difícil

1		2	
5			
	4		
			3

Estado Inicial



3		4	
		1	
	2		5

Estado Deseado

Tablero Difícil

Resultados

Algoritmo	Costo solución	Nodos expandidos	Nodos en frontera	Tiempo (s)
BFS	---	---	---	---
DFS	---	---	---	---
Greedy H1	---	---	---	---
Greedy H2	---	---	---	---
ASTAR H1	275	17334	5945	80.59

03

CONCLUSIONES

Conclusiones

- La primera heurística resultó ser mejor que la segunda
- El método de búsqueda ASTAR ha demostrado ser más eficaz
- Los métodos de búsqueda BFS y DFS son menos eficientes
- La DFS no es un método óptimo