

Búsqueda: Senku

Sistemas de Inteligencia Artificial

Grupo 2

Barruffaldi, Carla

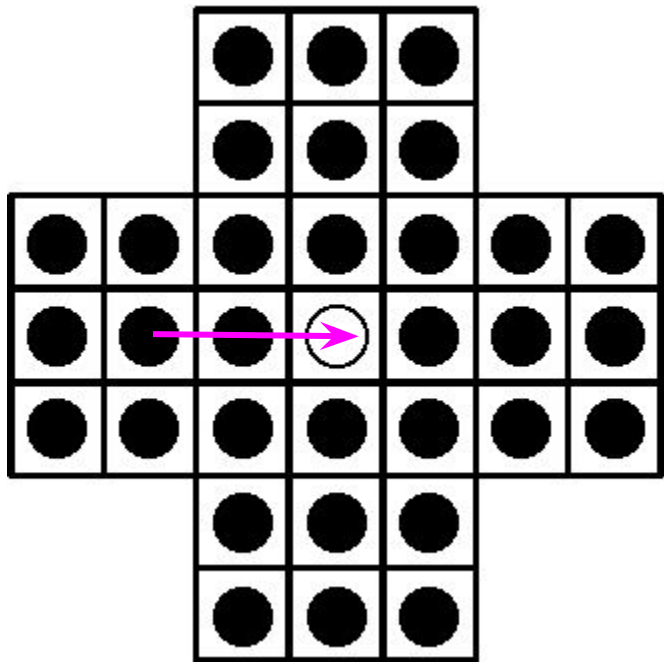
Bianchi, Luciano

Cerdá, Tomás

Lynch, Marcelo

Acerca del Senku

Tablero Inicial

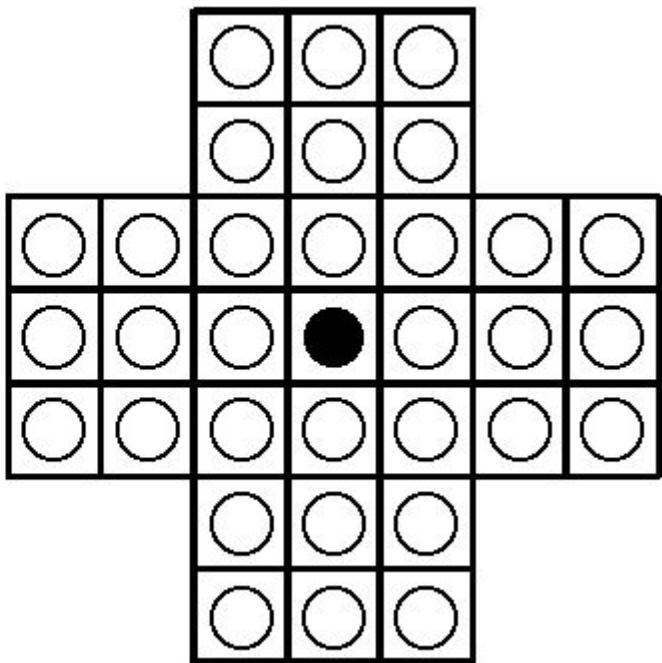


Movimiento

- En línea recta, dos posiciones, saltando un casillero con una ficha.
- Al mover, se elimina la ficha saltada.

Acerca del Senku

Tablero Final



- **Siempre** se realizan *#fichas - 1* movimientos.
- Si la ficha final no queda en el centro, no es una solución exitosa.

Implementación

Búsqueda NO Informada*

- Depth First
- Breadth First
- Uniform Cost
- Profundización Iterativa

Búsqueda Informada*

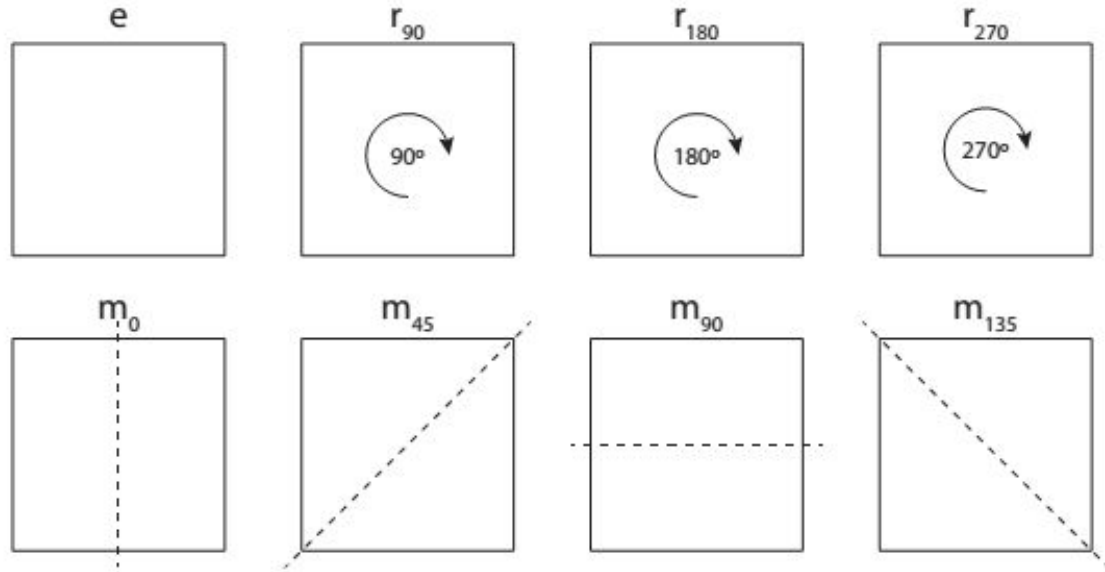
- A*
- Greedy Search

***Tree Search y Graph Search**

Análisis del Problema

Simetría

Son equivalentes todos los tableros simétricos, siguiendo las 8 simetrías del cuadrado.



Se eliminan el 75% de los estados sólo en el primer movimiento.

Estados Incorregibles

- Son estados que se sabe que nunca llegarán a la solución

¿Cómo se identifican?

Sean 3 posiciones contiguas en el tablero x, y, z se debe cumplir que:

$$v(\tilde{x}) + v(y) \geq v(z).$$

		-1	0	-1		
		1	1	1		
-1	1	0	1	0	1	-1
0	1	1	0	1	1	0
-1	1	0	1	0	1	-1
		1	1	1		
		-1	0	-1		

		●		●		
●	●	●	●	●		
●		●	●			
●			●	●		
			●			
		●		●		

Heurísticas

para A^* y Greedy Search

Heurísticas

1. Cantidad de fichas restantes

- Poco efectiva porque depende exclusivamente de la cantidad de movimientos realizados.
- Termina funcionando como un DFS.

2. Cantidad de fichas aisladas

- Una ficha se considera aislada si no hay ninguna ficha en un casillero adyacente.
- **$h = 2N + M$** // N : cantidad de fichas, M : cantidad de aisladas

Heurísticas

3. Dificultad de remoción

- Según el nivel de conveniencia de la ubicación de las fichas en el tablero.
- $h = 2N + S$ // N : cantidad de fichas, S : suma de valores de las posiciones ocupadas según la tabla.

		4	1	4		
		2	1	2		
4	2	3	1	3	2	4
1	1	1	0	1	1	1
4	2	3	1	3	2	4
		2	1	2		
		4	1	4		

Heurísticas

4. Distancia entre fichas

- Suma de la distancia Manhattan entre todos los pares de fichas del tablero.
- Similar a la de fichas aisladas, con mayor generalidad.

5. Distancia Manhattan al centro

- $h = 2S$ // S : suma de la distancia Manhattan de cada ficha al centro del tablero (se usa una tabla para evitar recalcular cada vez).

Resultados

Comparación de algoritmos de búsqueda

	Iterativo	DFS	BFS	Uniform Cost	A*	Greedy
Tiempo [ms]	x	79393	x	x	55	50
# nodos expandidos	x	2910002	x	x	475	475
# nodos frontera	x	116	x	x	166	166
# nodos generados	x	2910118	x	x	641	641

x - BFS y Uniform no terminaron por falta de memoria (4 Gb). Iterativo fue detenido luego de 10 horas de procesamiento.

*A** y Greedy usando la heurística de dificultad de remoción.

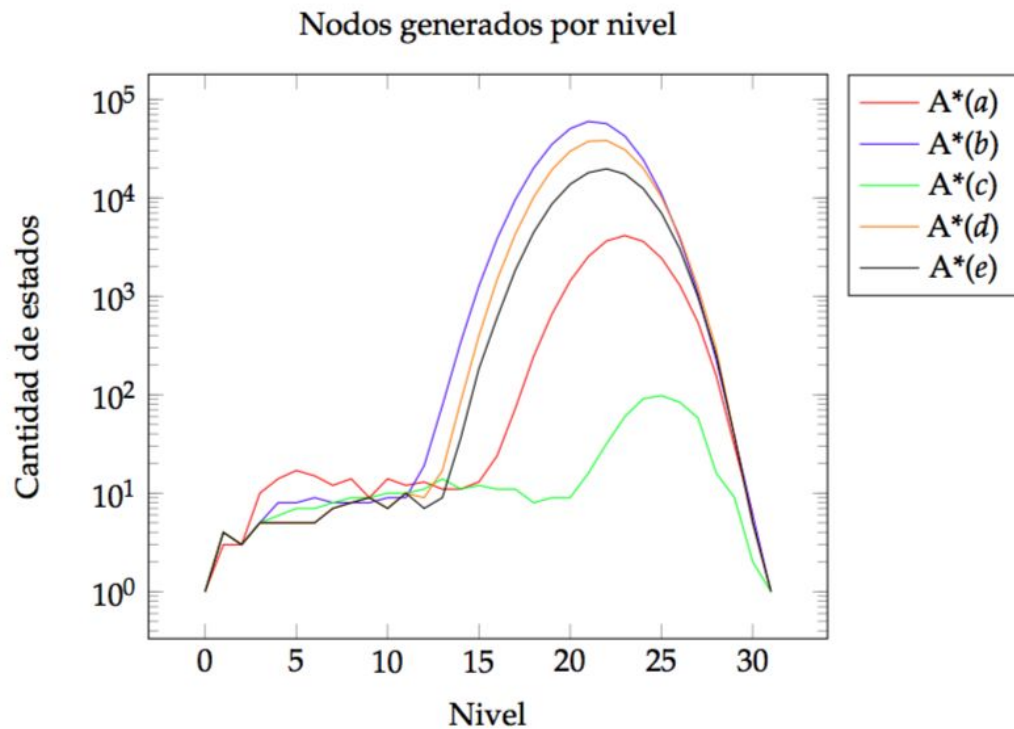
Comparación de heurísticas en A*

	# fichas restantes	# fichas aisladas	Dificultad de remoción	Distancia entre fichas	Distancia al centro
Tiempo [ms]	714	7309	57	3476	1625
# nodos expandidos	20835	319106	475	111782	59064
# nodos frontera	180	692	166	96049	49326
# nodos generados	21015	319798	641	207831	108390

Comparación de heurísticas en Greedy Search

	# fichas restantes	# fichas aisladas	Dificultad de remoción	Distancia entre fichas	Distancia al centro
Tiempo [ms]	18390	8185	50	6031	1587
# nodos expandidos	827873	372811	475	111782	56674
# nodos frontera	108	129	166	96049	47807
# nodos generados	827981	372940	641	207831	104481

Nodos generados por nivel



Conclusiones

Conclusiones

- El problema consiste en encontrar el camino a la solución y no el camino óptimo
- La capacidad de memoria y poder de procesamiento es un limitante para la resolución de este problema utilizando ciertas estrategias
- El uso de las optimizaciones como análisis de simetría y eliminación de estados incorregibles disminuye notablemente la cantidad de nodos generados y expandidos