



Métodos de búsqueda

Sistemas de Inteligencia Artificial

Ariadna Fernandez Truglia
Faustino Maggioni
Florencia Chao

Rompecabezas de 8 números

Este juego presenta un tablero con 9 posiciones y 8 números (1 a 8) y tiene como objetivo dejarlos ordenados de menor a mayor, con la última celda libre, moviendo los números de a 1 por vez al lugar vacío del tablero.



Métodos de Búsqueda No Informados





① BPA

Óptima y completa

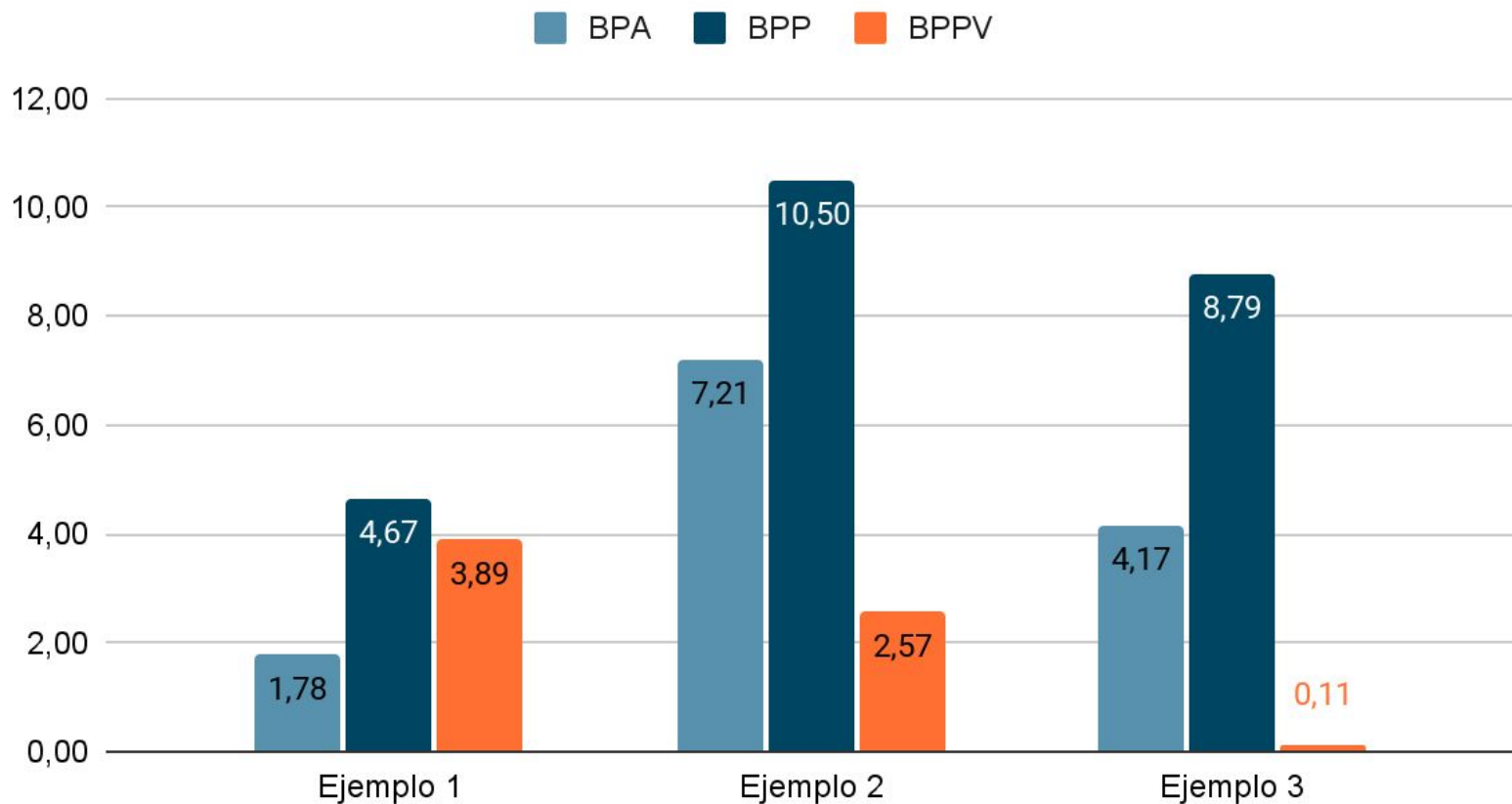
② BPP

No óptima y completa

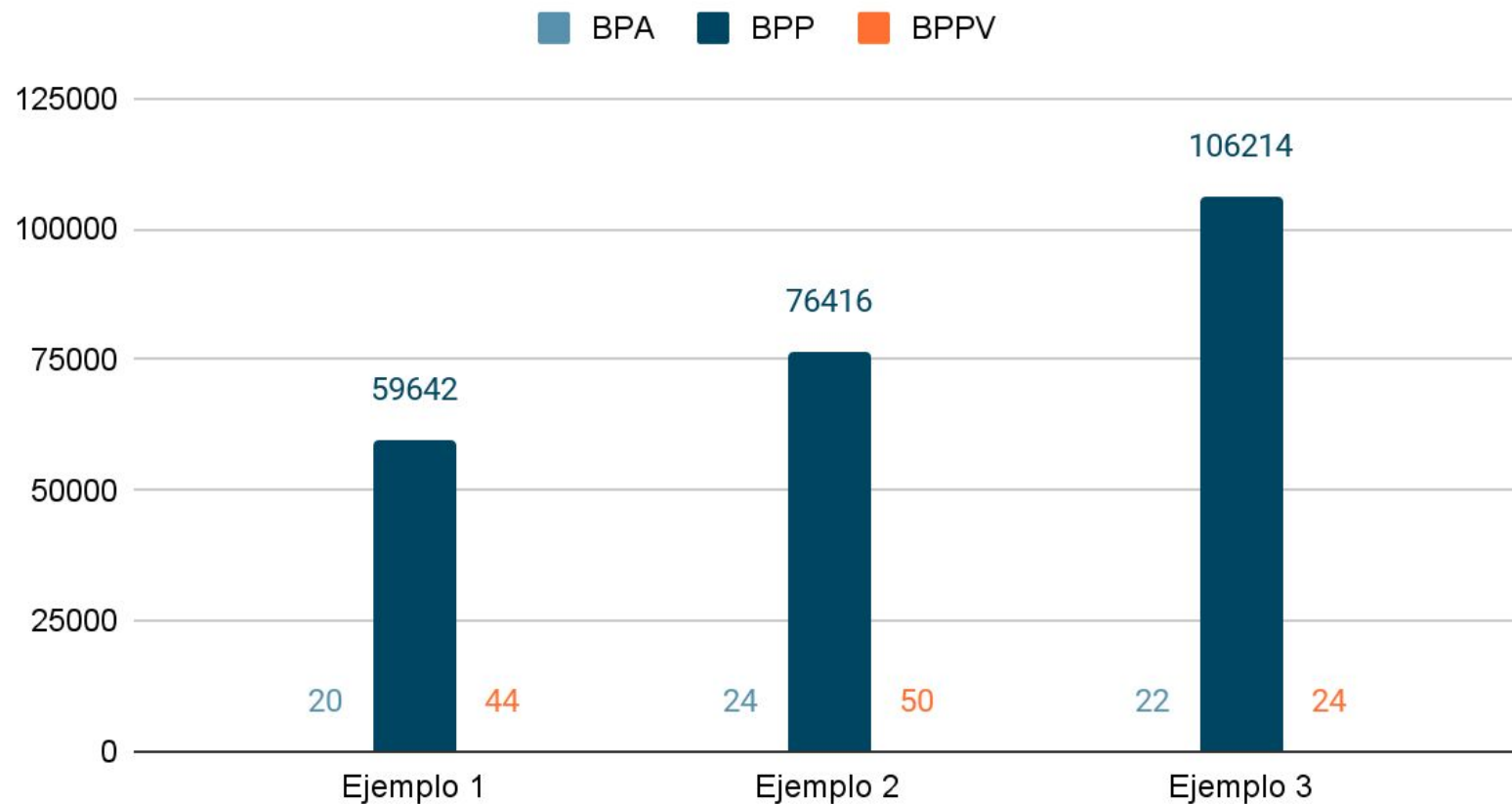
③ BPPV

No óptima y completa

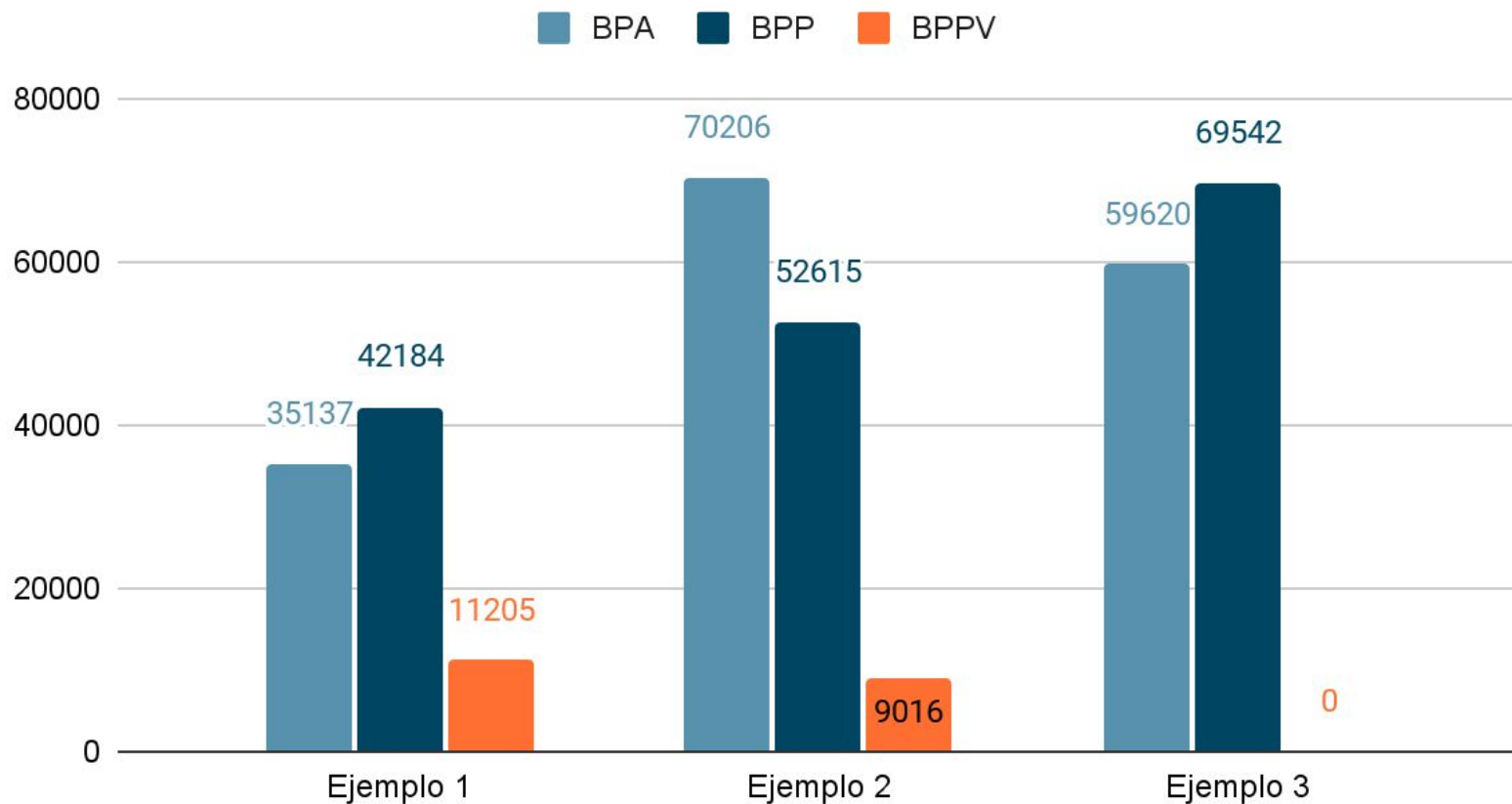
Tiempo



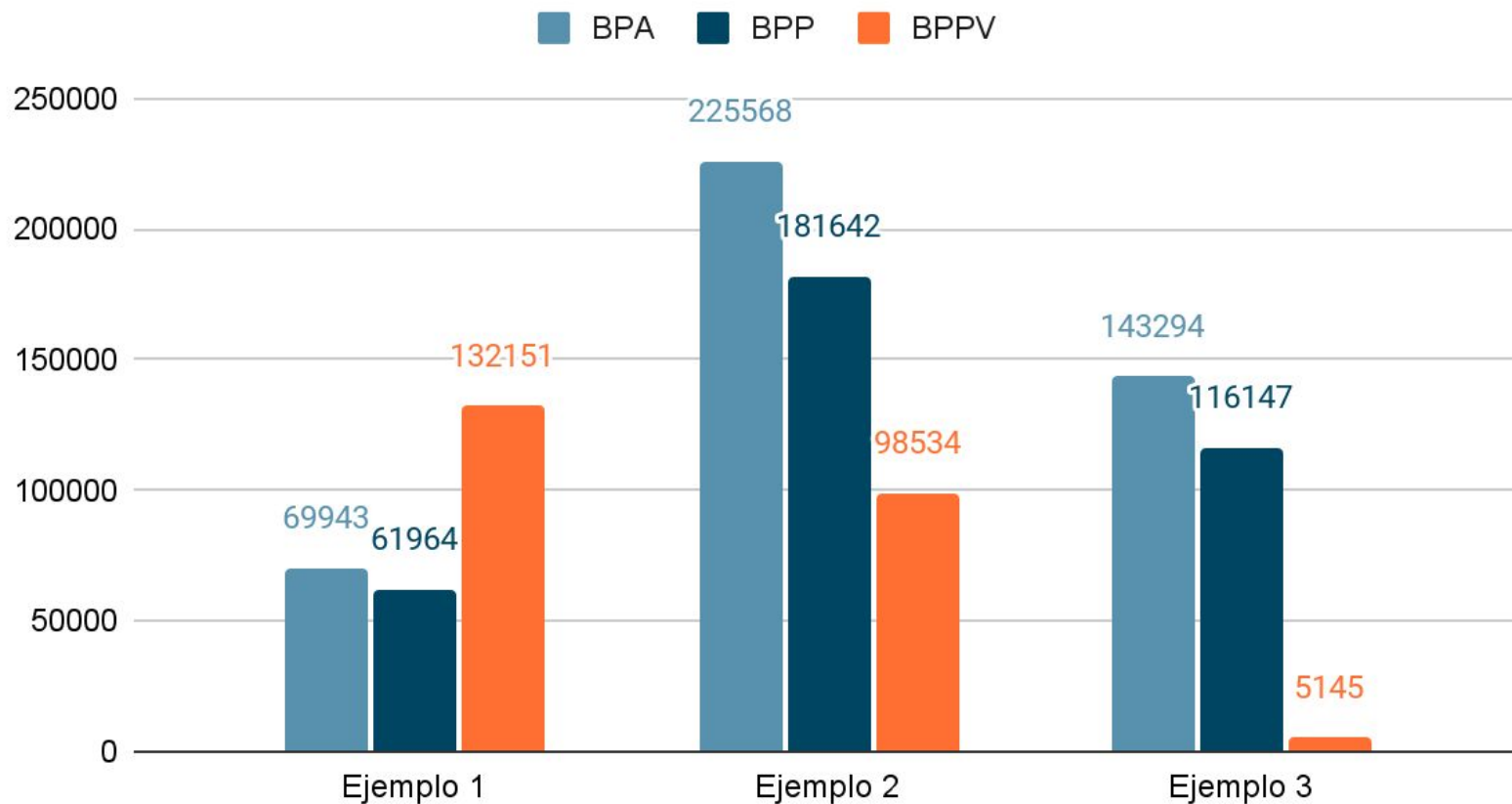
Depth/Costo



Frontera



Nodos Expandidos



Métodos de Búsqueda Informados





Heurística 1: Cantidad de números en el lugar incorrecto

Admisible
Notación: Tile

En esta heurística si un número no se encuentra en el estado final, se suma 1

1	2	5
4	6	3
8		7

$$h(n.e) = 6$$



Heurística 2: Cantidad de números en la fila incorrecta y en la columna incorrecta

Admisible
Notación: Row Col

$$h(n.e) = r(n.e) + c(n.e)$$

- $r(n.e)$ es la cantidad de números en filas incorrectas
- $c(n.e)$ es la cantidad de números en columnas incorrectas

Filas

1	2	5
4	6	3
8		7

$$\rightarrow r(n.e) = 2$$

Columnas

1	2	5
4	6	3
8		7

$$\rightarrow c(n.e) = 5$$

Heurística 3: Distancia lineal

No admisible
Notación: Linear

1	2	3
4	5	6
7	8	9



1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---

- Trata la grilla del juego como una sola fila de 9 posiciones, ordenando las celdas de izquierda a derecha y de abajo hacia arriba
- Retorna los pasos que tiene que hacer para llegar a su posición correcta en esta lista (para la izquierda o para la derecha).



① Heurística Global



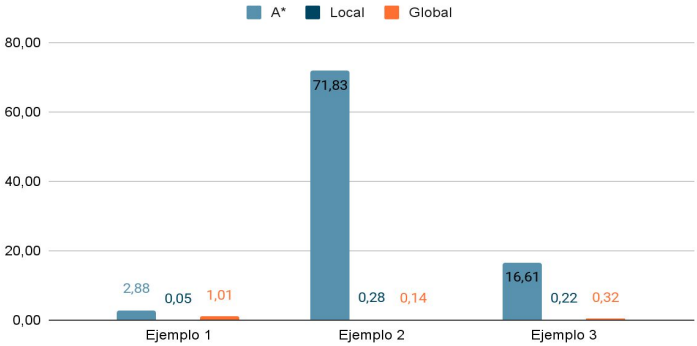
② Heurística Local

③ A^*

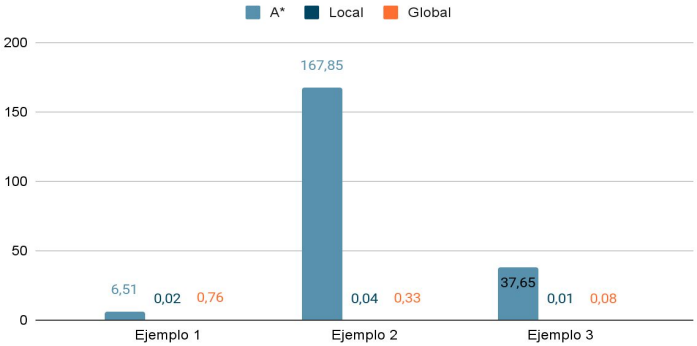


Tiempo

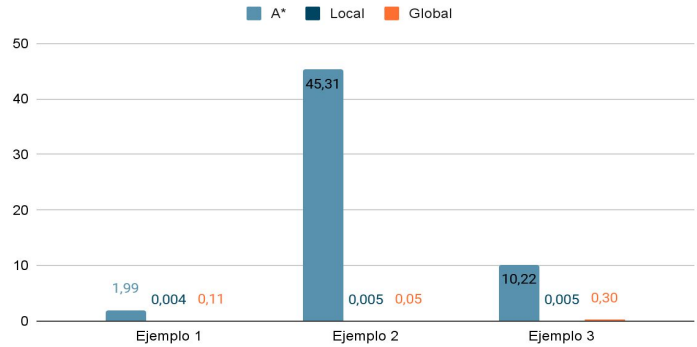
Tiempo



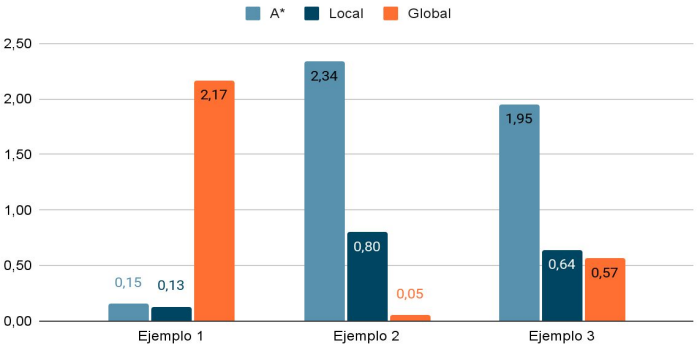
Tiempo - Tile



Tiempo - Row Col

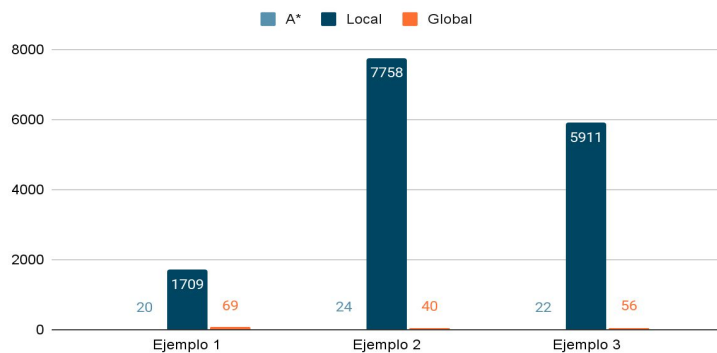


Tiempo - Linear

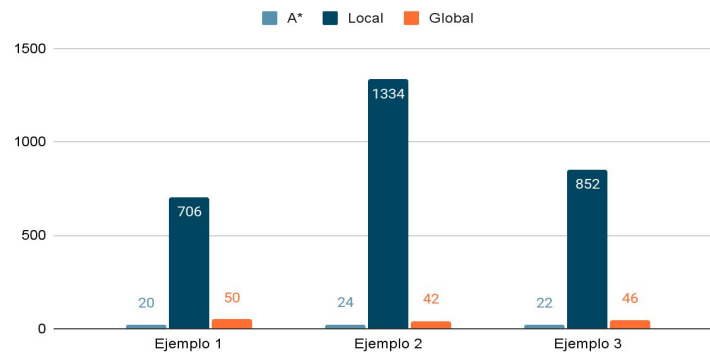


Profundidad/Costo

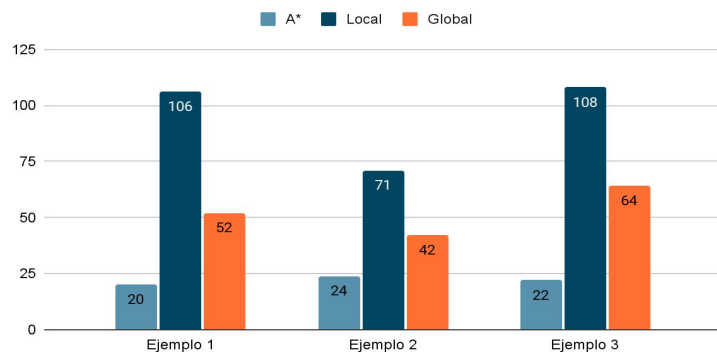
Depth/Costo



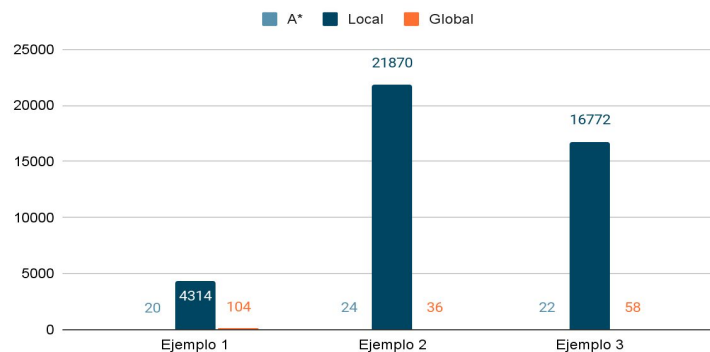
Depth/Costo - Tile



Depth/Costo - Row Col



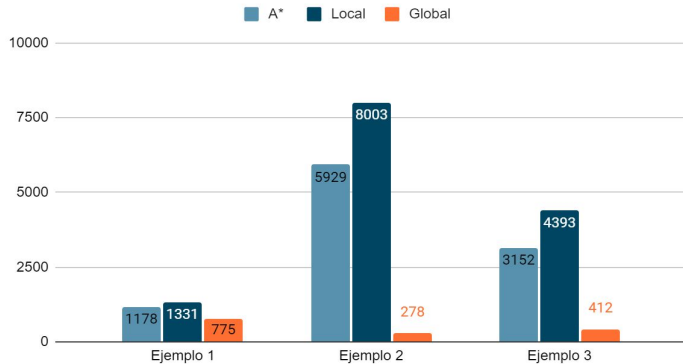
Depth/Costo - Linear



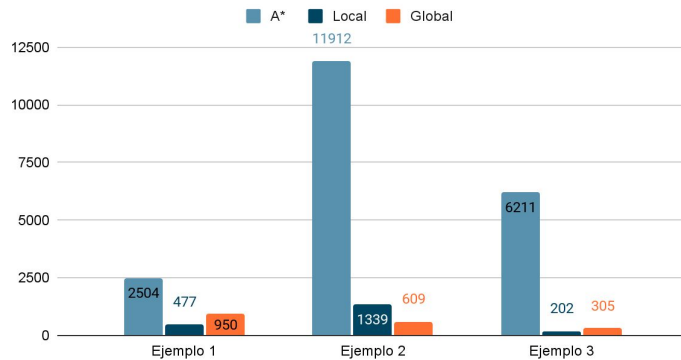


Frontera

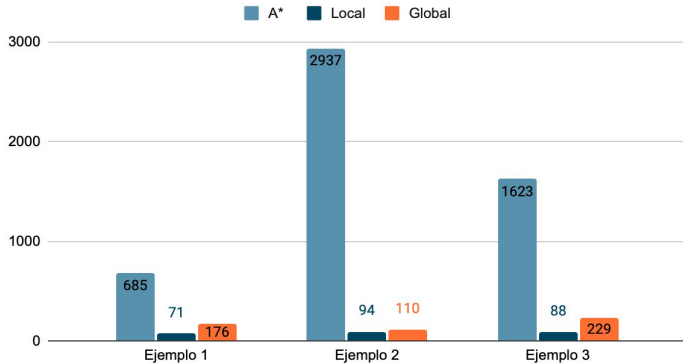
Frontera



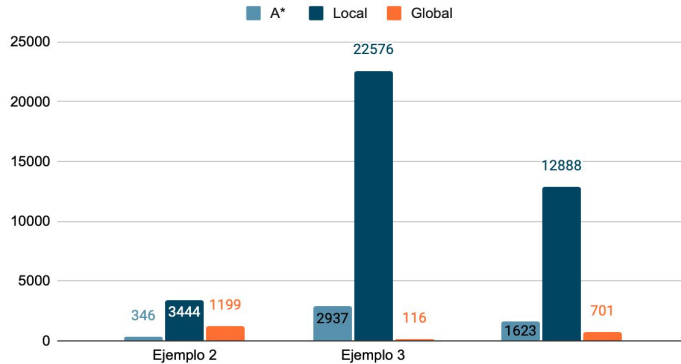
Frontera - Tile



Frontera - Row Col



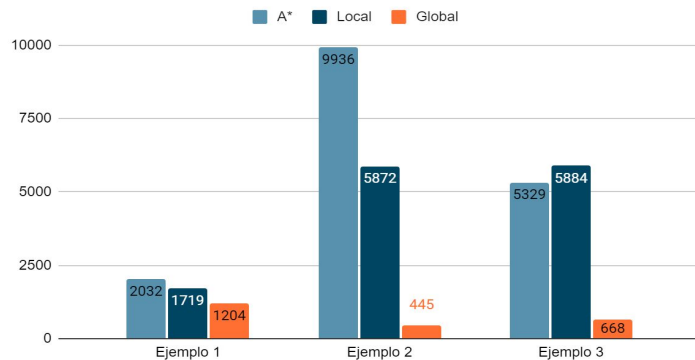
Frontera - Linear



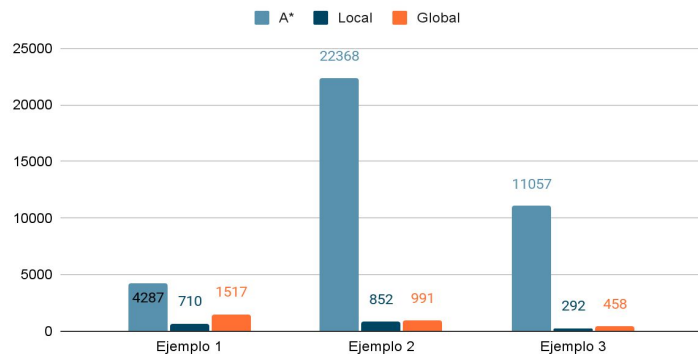


Nodos Expandidos

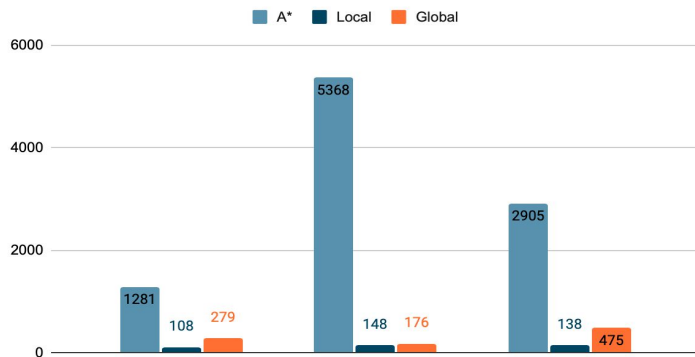
Nodos Expandidos



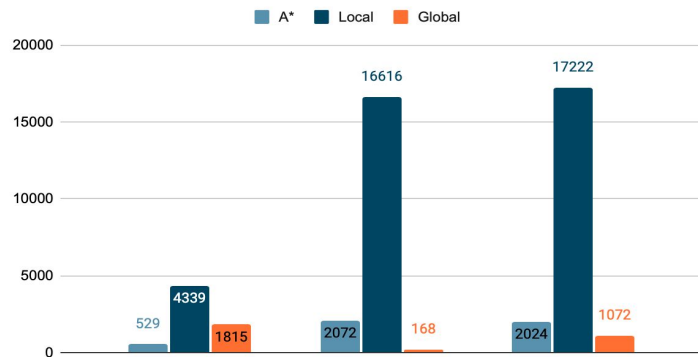
Nodos Expandidos - Tile



Nodos Expandidos - Row Col



Nodos Expandidos - Linear





Conclusiones



Luego de haber analizado los resultados obtenidos de las heurísticas, podemos concluir que:

- Row Col tiene la mejor desempeño en todas las métricas
- Linear y Tile dependen del objetivo que le da el usuario a la búsqueda
- Se necesitan más casos de uso



Problemas



→ De matriz a vector



→ Optimización

F.sort(...) → queue
stack



Bibliografía

- https://cse.iitk.ac.in/users/cs365/2009/ppt/13jan_Aman.pdf
- <https://www.geeksforgeeks.org/check-instance-8-puzzle-solvable/>
- <https://www.cs.princeton.edu/courses/archive/spr10/cos226/assignments/8puzzle.html>
- <https://www.geeksforgeeks.org/8-puzzle-problem-using-branch-and-bound/>