



Sistemas de Inteligencia Artificial

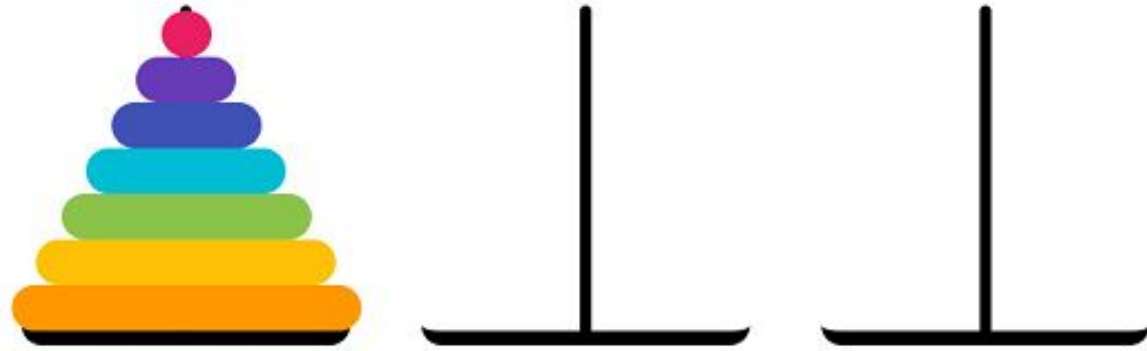
Trabajo práctico 1 - Métodos de búsqueda

De Simone, Franco - 61100

Dizenhaus, Manuel - 61101

Cornídez, Milagros - 61432

Problema elegido: Torres de Hanoi



3 postes, 7 discos



Reglas de juego:

- 1- Se colocan 7 discos en forma decreciente y con el mayor en la base en el poste de la izquierda.
- 2- El juego consiste en trasladar todos los discos al poste ubicado a al derecha.
- 3- Solamente se puede mover un disco por vez.
- 4- No se puede colocar un disco de mayor diámetro sobre uno de menor diámetro.



Definición del problema

Estado Inicial: Torre de la izquierda con 7 discos apilados de mayor a menor.
Torre del centro y de la derecha vacías.

Conjunto de posibles acciones: Dado un estado, se puede mover un disco a otra torre siempre y cuando el tope de la torre sea un disco de mayor diámetro que él.

Modelo de transición: Retorna las tres torres con sus respectivos discos cada una.

Condición de solución: Los 7 discos se encuentran apilados de mayor a menor en la torre de la derecha.



Estados

Cada estado se compone por un arreglo de 3 posiciones donde cada posición representa una torre.

Cada torre es de tipo “long” que representa los discos que están en ella actualmente.

Por ejemplo:

En el estado inicial, `torres[0] = 87654321` ya que contiene los 7 discos (el número 8 representa que la torre está vacía)

Si `torres[0] = 87654`, `torres[1] = 832` y `torres[2] = 81`, la primera torre contiene los 4 mayores discos, la tercera el menor y la segunda los restantes.

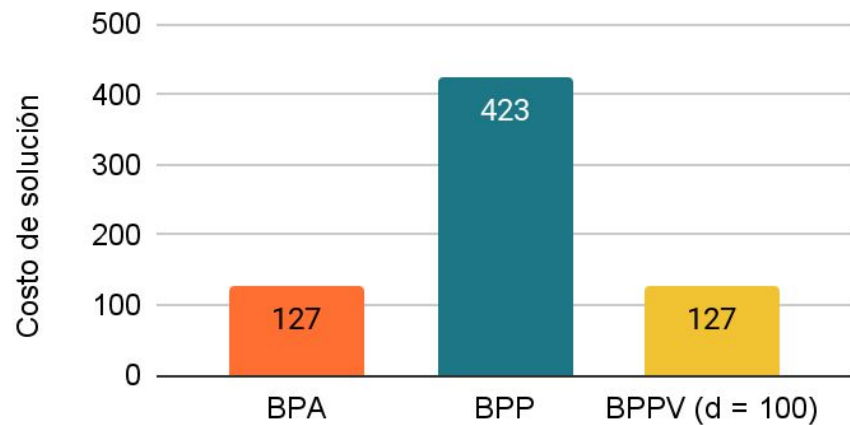




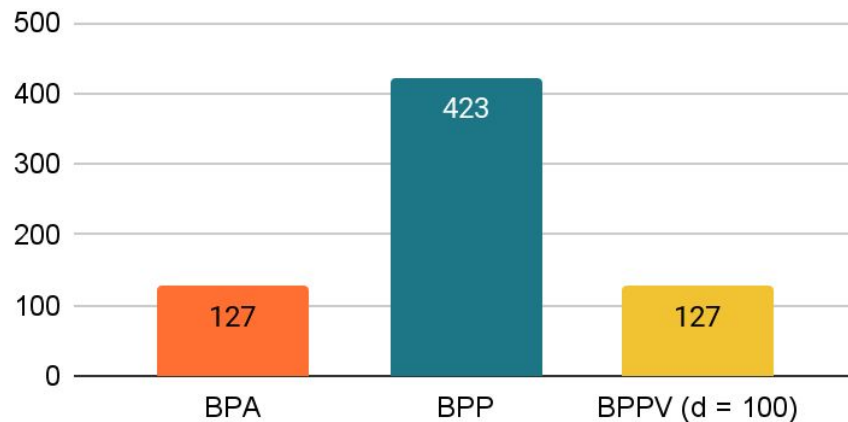
Métodos de búsqueda desinformados

Comparación de costo (y profundidad)

Costo de solución (Desinformados)

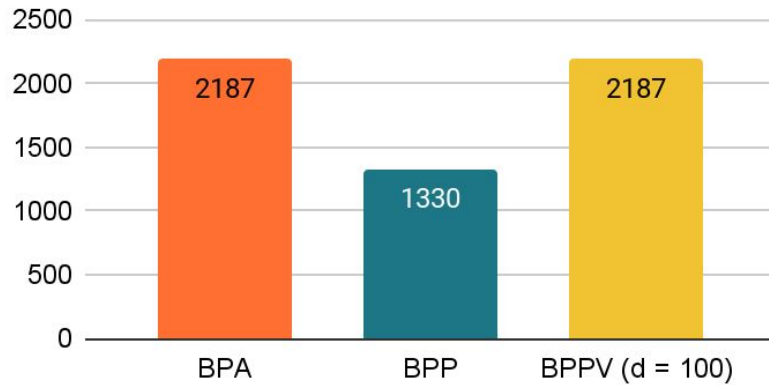


Profundidad de solución

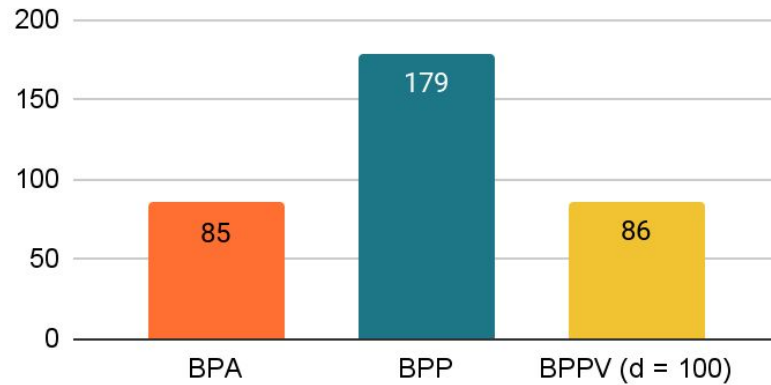


Comparación de nodos expandidos y nodos frontera

Nodos expandidos (Desinformados)

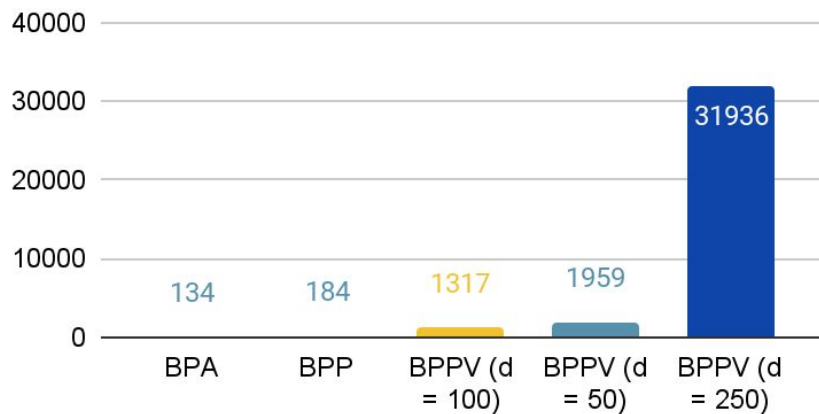


Nodos frontera al finalizar

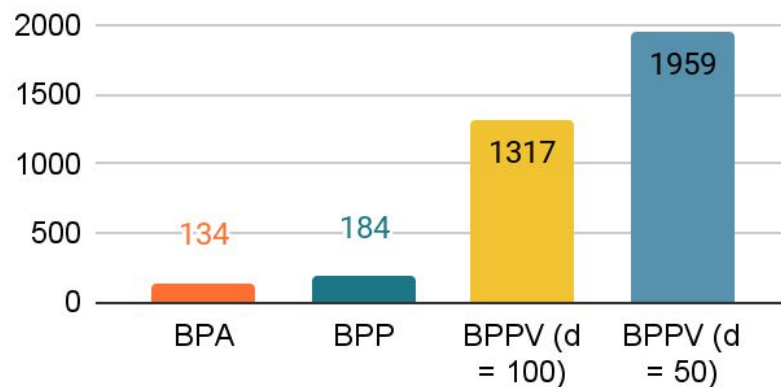


Comparación de tiempos de ejecución (ms)

Tiempo de ejecución (ms)



Tiempo de ejecución (ms)





Métodos de búsqueda informados

Definición de Heurísticas

- Admisibles:
 - Heurística 1: Discos restantes por colocar
 - Heurística 2: Discos restantes por colocar + $2 \times$ Discos ubicados a derecha pero mal ubicados
- No admisibles:
 - Heurística 3: Armado de torre de n discos (para todo n menor o igual a 7) en la columna solución



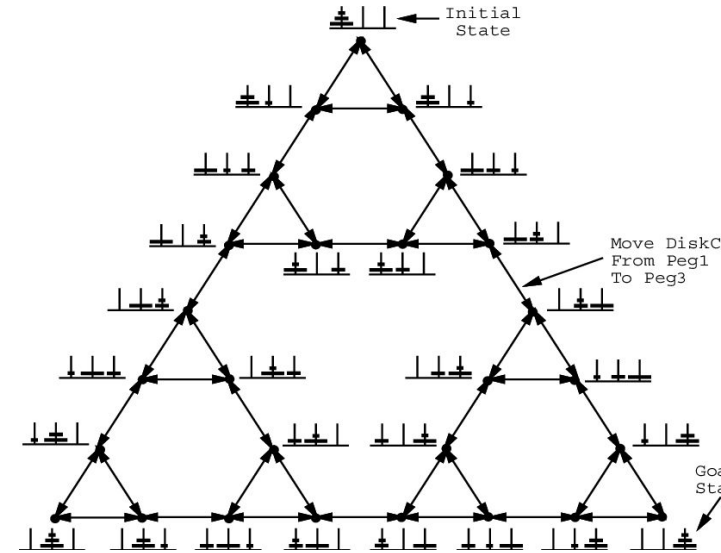
Heurística óptima

- Se puede probar, mediante inducción matemática, que la heurística para el camino óptimo está dada por:
 - **$2^{(\text{discos que faltan ubicar en último poste correctamente}) - 1}$**
 - Demostración:
<https://math.stackexchange.com/questions/2650/how-to-prove-the-optimal-towers-of-hanoi-strategy>



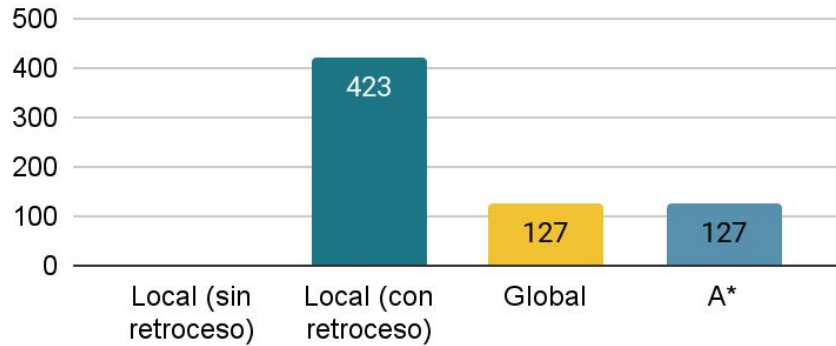
Heurística 1: Discos restantes

- Función heurística definida de manera simple:
 $h(\text{estado}) = 7 - \text{número de discos bien ubicados}$
- $h(n.s) \leq h_{\text{Optima}}(n.s)$, para todo estado



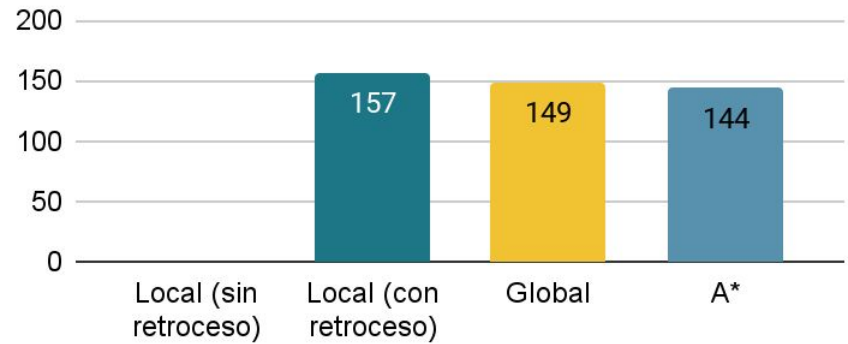
Resultados con Heurística 1

Costo de solución (Informadas)



Heurística 1

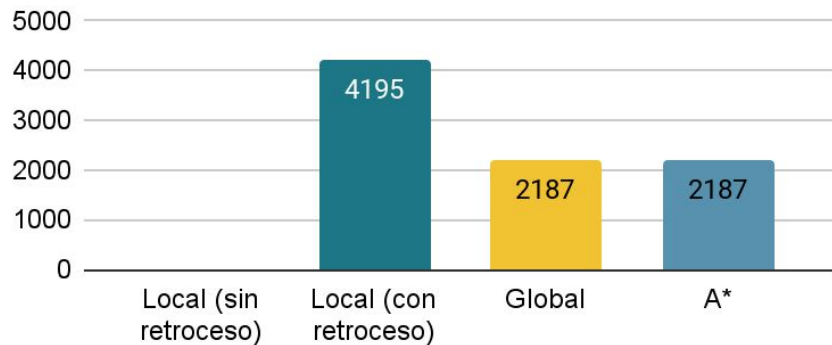
Tiempo de ejecución (ms) (Informadas)



Heurística 1

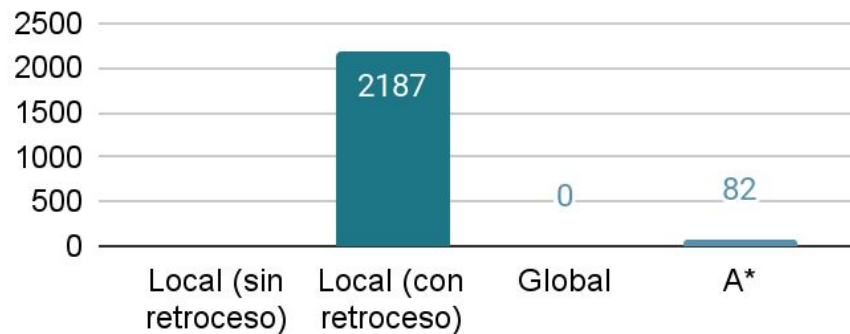
Resultados con Heurística 1

Nodos expandidos (Informadas)



Heurística 1

Nodos frontera al finalizar



Heurística 1

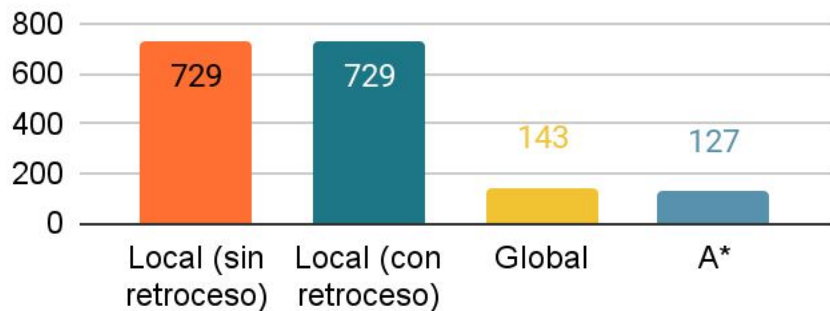
Heurística 2: Discos restantes + Mal ubicados

- Función heurística basada en movimientos restantes.
- Se considera que, para cada disco que aun no esta ubicado a derecha, se debe realizar al menos un movimiento. Y que para disco mal ubicado a derecha, se necesita al menos un movimiento para sacarlo de aquel poste, y colocar el correcto.
- $h(\text{estado}) = \text{discosAunNoADerecha} + 2 * (\text{discosADerechaMalUbicados})$



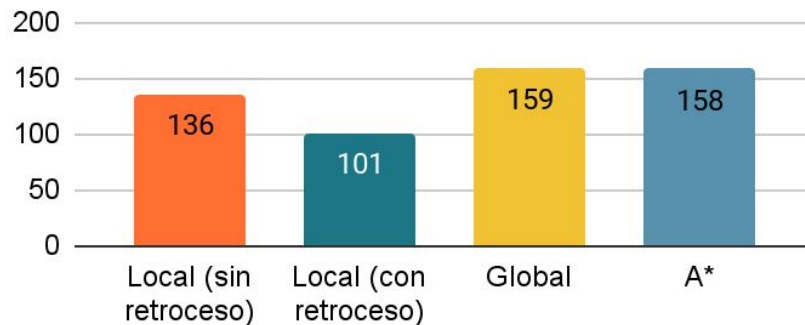
Resultados con Heurística 2

Costo de solución (Informadas)



Heurística 2

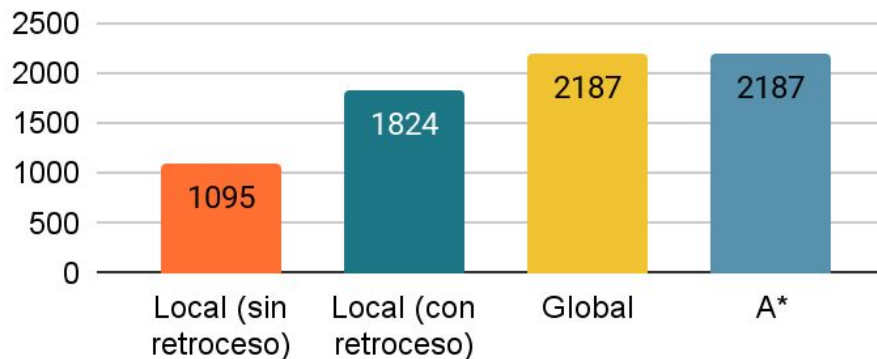
Tiempo de ejecución (ms)



Heurística 2

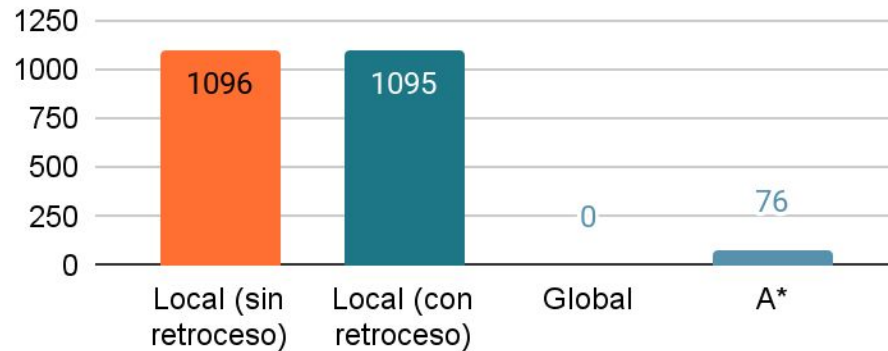
Resultados con Heurística 2

Nodos expandidos (Informadas)



Heurística 2

Nodos frontera al finalizar (Informadas)



Heurística 2

Heurística 3:

La heurística tres se basa en el concepto de “armar cada torre intermedia en la columna solución”, como soluciones intermedias al problema. Ejemplo:

- Se arma una torre de 3 discos en la columna 3, se pone el disco 4 en la columna 2, luego se transfiere la torre de 3 discos a la columna 1, y finalmente se arma la torre de 4 discos en la columna 3. Así sucesivamente.
- En vez de aprovechar que se pueden utilizar las dos columnas para armar torres intermedias, se las arma siempre en la columna solución, hasta que quede la torre de 7 discos en la misma. Esto hace que la heurística sea INADMISIBLE.



Heurística 3

Para la función h , se toma en cuenta:

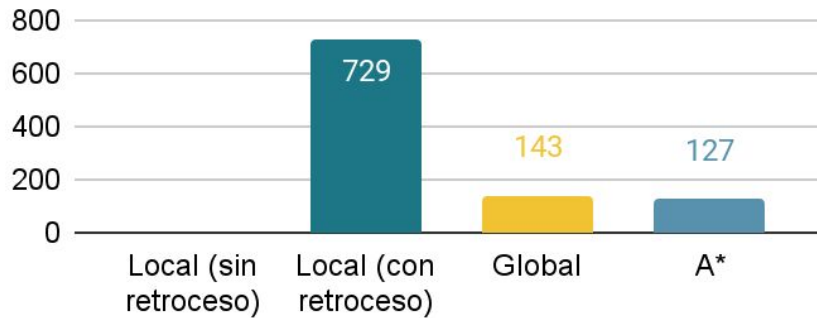
- Sea n el número del disco que se encuentra al tope de la columna solución (se obtiene haciendo módulo 10 del `getTower(2)` del state)
- Se toma $2^{(n-2)}$ movimientos para armar la torre de tamaño n en la cima de la columna solución, dando inicio el proceso
- Luego se mueve la torre de n discos a otra de las dos columnas, se posiciona el disco $n+1$ en la columna solución, y se vuelve a poner a la torre de n discos encima del mismo
- Se repite el proceso incrementando n hasta llegar a 6 (la torre de 7 discos no se mueve) dando la sumatoria:

$$2^{(n-2)} + \sum 2^{(i-1)}, \text{ sumatoria de } n \text{ a } 6$$



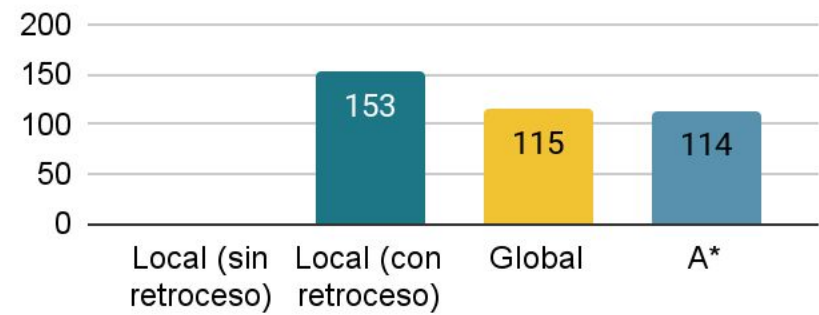
Resultados con Heurística 3

Costo de solución (Informadas)



Heurística 3

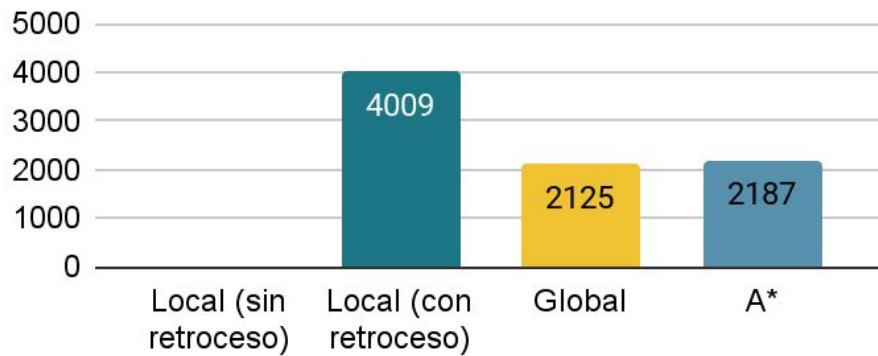
Tiempo de ejecución (ms)



Heurística 3

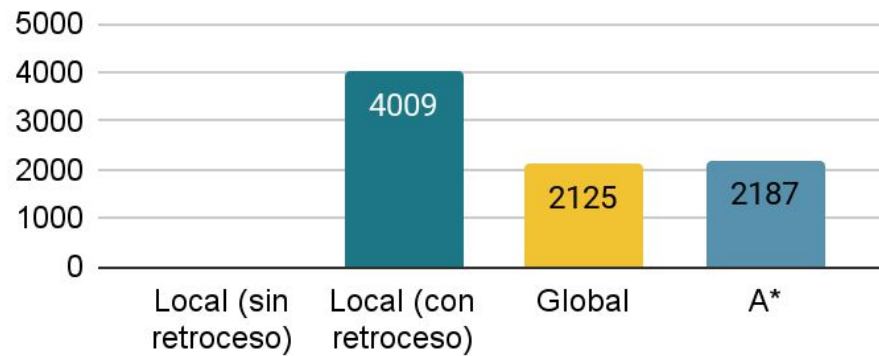
Resultados con Heurística 3

Nodos expandidos (Informadas)



Heurística 3

Nodos expandidos (Informadas)



Heurística 3