GUIA DE EJERCICIOS 1.
INTELIGENCIA ARTIFICIAL

SERGIO HERNÁNDEZ
SHERNANDEZ@UCM.CL

1. 8 Puzzle

Escriba un programa para resolver el problema del 8-puzzle usando el algoritmo A*.

El problema del 8-puzzle fue inventado en 1870 y se juega en una grilla de 3x3 con 9 bloques cuadrados. Cada bloque tiene una etiqueta con un número del 1 al 8 y existe un espacio en blanco en el cual el jugador puede mover un bloque adyacente.

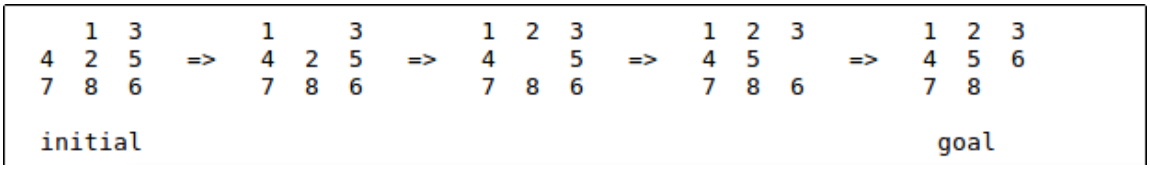

Figure 1. Solución del problema 8-puzzle

1.1. **Búsqueda No Informada y Heurística.** Cada estado del puzzle puede ser visto como un vértice de un grafo tipo árbol, cuyo nodo raiz es el estado inicial del puzzle.

Las estrategias de búsqueda no-informada no utilizan ningún criterio para ponderar las soluciones y por ende dependen en gran medida de la complejidad del problema.

Ejercicio 1

1.1 Determinar el factor de ramificación del problema y la profundidad máxima del arbol de búsqueda.
1.2 Calcular en forma teórica la complejidad computacional y los requerimientos de memoria de los algoritmos de búsqueda no informada por anchura y profundidad.
1.3 Programar en Matlab una función que devuelva los estados hijo de cualquier estado.

En el caso de las estrategias de búsqueda informada, dos heurísticas admisibles para implementar una estregia $f(n) = h(n)$ tipo best-first son:

- Distancia Hamming : Número de bloques en la posición incorrecta.

- Distancia Manhattan : Suma de las distancias (suma de la distancia horizontal y vertical) desde el bloque hasta la posición objetivo ($d_M = |X_n - X_g| + |Y_n - Y_g|$).

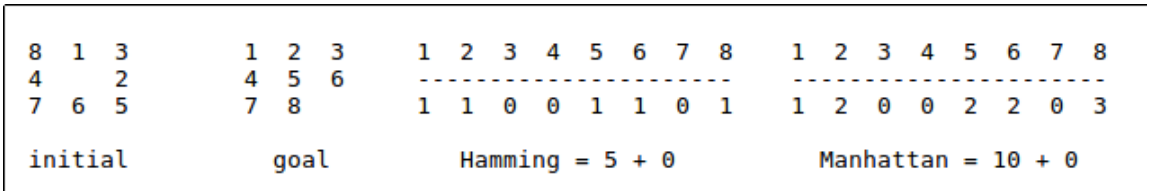

FIGURE 2. Solución del problema 8-puzzle utilizando heurísticas

El siguiente código calcula la distancia de Hamming.

Program 1 Programa Matlab para calcular distancia Hamming

```
function cost = Hamming( state , goal )
% state=[6,4,5;1,2,3;7,0,8];
% goal=[1,2,3;4,5,6;7,8,0];
    cost = sum (sum (~(state == goal)));
end
```

Ejercicio 2

2.1 Programe en Matlab la distancia Manhattan.
2.2 Utilice ambas distancias para implementar el algoritmo A (tomando en cuenta $f(n) = h(n) + g(n)$, con $g(n)$ el costo de llegar a n).*
2.3 Calcule los costos de A (en tiempo y número de pasos) de resolver el problema usando las distancias de Hamming y Manhattan. Utilice 1000 estados iniciales aleatorios y promedie los resultados.*

Ejercicio 3

3.1 Discuta y programe IDA para el problema 8-puzzle.*
3.2 Compare IDA con A* en términos de performance (tiempo e iteraciones) en encontrar un resultado.*

REFERENCES

[1] Alexander Reinefeld, Complete solution of the eight-puzzle and the benefit of node ordering in IDA. Proceedings of the 13th international joint conference on Artifical intelligence - Volume 1, 1993. http://dl.acm.org/citation.cfm?id=1624025.1624060