

Tarea 1 - Taller de Programación

Sebastián Cassone

*Departamento de Ingeniería Informática
Universidad de Santiago de Chile*

I. SOLUCIÓN PROPUESTA

Para abordar el problema planteado, se emplea el algoritmo A* como estrategia de resolución, que consiste en explorar diferentes estados del tablero de manera inteligente. En este proceso, se utilizó un árbol AVL (Georgii Adelson-Velskii y Yevgeniy Landis) para almacenar tanto los tableros visitados como los pendientes de visitar, manteniendo el equilibrio del árbol cada vez que se agregaba un nuevo nodo generado por el algoritmo A*.

Para optimizar el almacenamiento en el árbol, se decidió conservar únicamente la información relevante en el árbol de todos los tableros visitados, eliminando el tablero en sí y manteniendo solo la distancia calculada hasta ese momento. La distancia se determinó mediante una heurística cuidadosamente seleccionada, en este caso, distancia euclidiana, que demostró ser la más eficiente entre la distancia euclidiana, la distancia de Manhattan y el recuento de casillas mal colocadas, en términos de tiempo de resolución.

Por otro lado, el árbol de tableros abiertos registra todos los cambios en los nodos durante el proceso de búsqueda. Esto permite un seguimiento detallado de la evolución del algoritmo y proporciona información valiosa para su mejora continua.

La solución propuesta se caracteriza por su eficiencia, gracias a la combinación de una heurística bien diseñada y el uso de un árbol AVL. La complejidad de búsqueda, inserción y eliminación de nodos en el árbol es de $O(\log(n))$, lo que garantiza tiempos de ejecución óptimos. Por lo tanto, bajo este enfoque, el problema se resuelve en el menor tiempo posible.

II. EXPERIMENTOS REALIZADOS

Se realizó una exhaustiva comparación entre distintas heurísticas para la resolución de problemas en tableros de dimensiones variadas (desde 2x2 hasta 5x5) mediante algoritmos de búsqueda. Las heurísticas evaluadas incluyeron la distancia euclidiana, la distancia de Manhattan y el recuento de casillas mal colocadas, así como combinaciones de estas heurísticas.

En términos de tiempo de resolución, se observaron los siguientes resultados:

La combinación de la distancia euclidiana y el recuento de casillas mal colocadas arrojó un tiempo mínimo de resolución de 0.000748 segundos y un máximo de 0.003521 segundos. Utilizando únicamente el recuento de casillas mal colocadas, se obtuvo un tiempo mínimo de 0.000557 segundos y un máximo de 0.003611 segundos. La heurística basada en la distancia de Manhattan registró un tiempo

mínimo de 0.000724 segundos y un máximo de 0.003944 segundos. Empleando únicamente la distancia euclidiana como heurística, se logró un tiempo mínimo de 0.000592 segundos y un máximo de 0.003303 segundos.

Por lo que se termina eligiendo la heurística de distancia euclidiana.

Los experimentos se llevaron a cabo en tableros que variaban desde 2x2 hasta 5x5. En todos los casos, los tiempos de ejecución variaban considerablemente debido a los múltiples procesos gestionados por el planificador del kernel del Sistema Operativo. No es factible asegurar un tiempo constante debido a esta variabilidad inherente. No obstante, se ha registrado exhaustivamente cada uno de los experimentos realizados.

Tablero 2x2 se hizo en un tiempo de 6.9e-05 segundos y en 2 pasos.

Tablero 3x3 se hizo en un tiempo de 0.000116 segundos y en 8 pasos.

Tablero 4x4 se hizo en un tiempo de 0.000287 segundos y en 15 pasos.

Tablero 5x5 se hizo en un tiempo de 0.000461 segundos y en 24 pasos.

III. CONCLUSIONES

En conclusión, se implementó con éxito el algoritmo A* junto con un árbol AVL para resolver el problema planteado de manera eficiente. La estrategia de explorar estados del tablero de manera inteligente, utilizando una heurística cuidadosamente seleccionada, permitió encontrar soluciones óptimas en un tiempo razonable.

Los resultados obtenidos muestran que el algoritmo A* combinado con el árbol AVL es altamente efectivo, con tiempos de ejecución y cantidad de pasos razonables para cada tamaño de tablero evaluado. Este enfoque ofrece una solución escalable y eficiente para problemas similares, demostrando su utilidad en la resolución de problemas complejos en diversos contextos.