Homework 1 Report

R04922034 吳軒衡

Compilation:

Execute “make” will compile the codes that generate the database as well as that solves a given puzzle.

Project Layout:

```
Hw1-----*5x5------------ |-------bin/             ----- Where binaries are located
          |              |--------db/             ----- Where precomputed db is located
          |              |--------testcase/       ----- Where testcases are located
          |              |--------log/            ---- Whter logs are located
          |              |--------Makefile
          |              |--------new_db.cpp  ----- Pattern Database Generator
          |              |---------run.cpp        ----- Problem Solver
        4x4------------ |--------bin/            ----- Where binaries are located
                        |--------db/           ----- Where precomputed db is located
                        |--------testcase/test.in          ----- Where testcases are located
                        |--------log/              ---- Where logs are located
                        |--------Makefile
                        |--------new_db_sub.cpp ----- 2by2 pattern generator
                        |--------run_sub.cpp      ----problem solver using 2by2 pattern
                        |--------new_db.cpp        ----- 4by4 pattern db generator
                        |--------run.cpp          ----problem solver using 4by4 pattern
```

Execution Notice:

The precomputed database will be cleared whenever the generator is run
Make clean will also clear the database
Testcase is loaded from the firstline in testcase/test.in

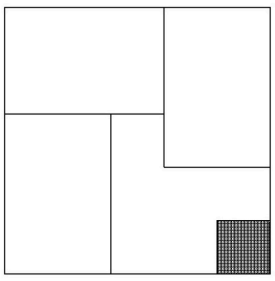
Implementation Detail:

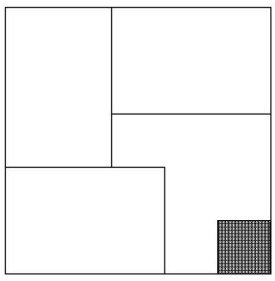
A. Precomputing the Database Patterns

A* with Manhattan distance as heuristic is used as the way to generate the precomputed database.

1. 5x5

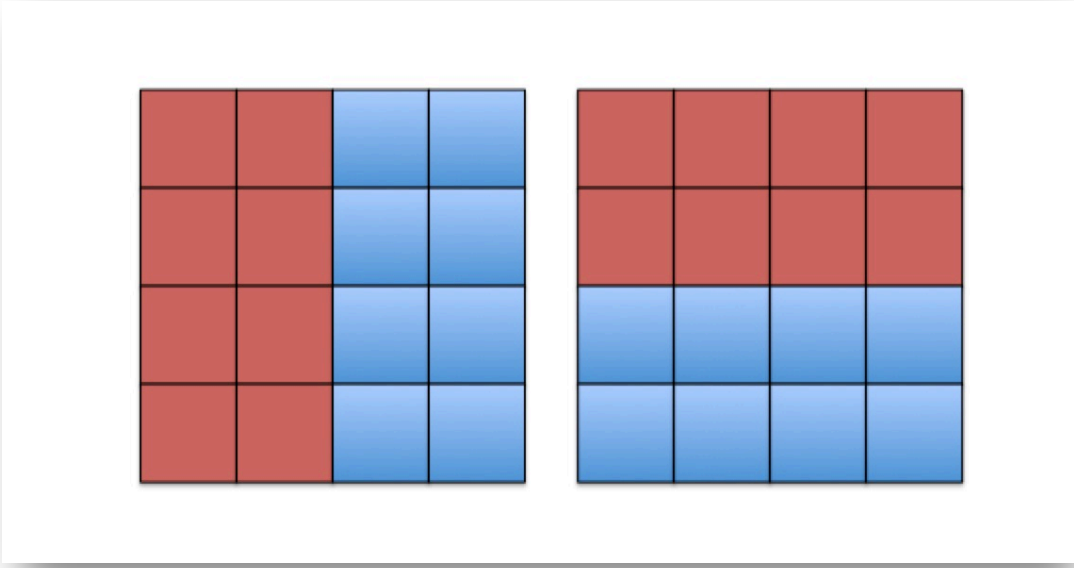
This implementation basically follows “Disjoint pattern database heuristics “ (Korf 2002) , with the patterns flipped horizontally and vertically according to the normal definition of the problem.





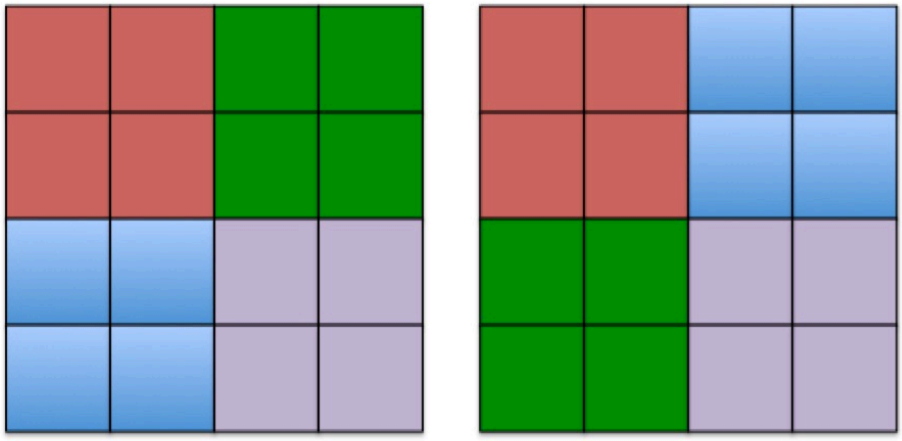
2. 4x4 using 2x4 patterns

In Korf 2002. , disjoint patterns of size 8 and 7 are used, but for the simplicity of implementation, disjoing patterns of size 8 and 8 are used.



3. 4x4 using 2x2 patterns

The use of 2x2 patterns further reduce the number of patterns needed to be generated.



B. Problem Solving:

Perform Iterative Deepening A* using the larger of the sum of two sets of precomputed pattern as heuristic.

Experiment:

BFS has been tried to generate precomputed database, however, the runtime of the BFS is intolerable. A single run of BFS also fails to generate all additive patterns.

This report lacks the comparison of performance between different methods since the time is not enough.