

CS5960 Final Project Specification

Sara Bernardini

November 17, 2020

1 Project: Tower of Hanoi

Goal: You will reflect upon the search algorithms learned during the lectures and implement two of them to solve the Tower of Hanoi.

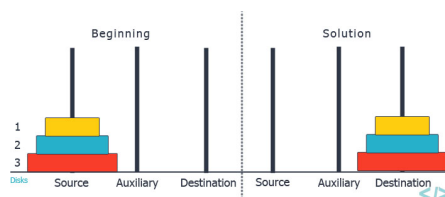


Figure 1: Tower of Hanoi

Setting of the game:

- The Tower of Hanoi is a puzzle game with usually 3 pegs and a fixed number of disks. For this project, we will use **4 pegs**. Consider $n > 0$ number of discs.
- Initial state: At the start, all the disks are in the leftmost peg, with the largest disk on the bottom and the smallest on the top.
- Goal state: All disks are moved over to the rightmost peg.

Rules of the game:

- You can only move one disk to any other peg in each move, and you can only move the top disk on a peg.

- The top disk of any peg can be moved to the top disk of any other peg, with the restriction that you cannot move a larger disk on top of a smaller disk (i.e., disks can only be moved to empty pegs or on top of larger disks).
- This is a unit-cost domain, all moves cost the same. Path cost is its length.

1.1 Basic Project

You are allowed to team-up with another person for the coding part of the project.

Tasks that you have to implement:

1. If you have joined a team for the coding part, name the person.
2. Create an efficient representation of the problem to be used with the search algorithms that you have learned in this course.
3. Implement the A* algorithm and another search algorithm of your choice to solve the problem.
4. Create at least two heuristic functions for this problem to work with A*. At least one of them should be admissible.
5. Run both algorithms with an increasing number n of discs and create a table with a comparison showing runtime vs. n .

[70 marks]

1.2 Extension

1. Devise a Pattern Database (PDB) heuristic for the problem.
2. Let m be the number of disks in your PDB and n be the number of disks in the problem. Find the biggest m such that constructing the PDB takes less than 5 minutes.
3. Take m from the previous question and increase n , starting from $n = m + 1$ until no solution can be found in less than 10 minutes. Show how performance worsens as n becomes bigger (runtime vs. n).

[30 marks]

2 Final Report

[100 marks]

The final report document should be done individually and separately, regardless of whether you have teamed-up for the coding part.

The final report should include the following points:

- Explain the rationale behind the representation you have chosen for the problem.
- Briefly describe the search algorithms that you have chosen to solve the problem.
- Describe the two heuristics that you have devised for the problem and give experimental evidence (graphs or tables) to assess which heuristic makes the algorithm more efficient in solving the problem.
- Indicate the maximum number of discs n that you can handle optimally (you find the optimal solution) and sub-optimally (you find a sub-optimal solution) with a runtime limit of 10 minutes.
- Show tables with the runtime of each algorithm vs. number of discs n .
- Using k as the maximum number of discs that you can handle optimally, show the optimal and sub-optimal length of the solutions for that case.

For the extension:

- Explain the rationale behind the construction of the PDB heuristic.
- Discuss points 2 and 3 of the extension. In particular:
 - Provide the biggest m such that constructing the PDB takes less than 5 minutes.
 - Take m from the previous question and increase n , starting from $n = m + 1$ until no solution can be found in less than 10 minutes. Show how performance worsens as n becomes bigger (runtime vs. n).

3 Deliverables

If you have a team member, only one of you should hand in the code, please coordinate this. You will both share the same score for the coding part.

1. Code appropriately documented - Moodle submission (**December 11, 2020**). During the last week of the term, you will find a tab in Moodle to be able to submit. Only hand in one copy of the code per team.
2. Final report - Moodle submission (**December 11, 2020**). During the last week of the term, you will find a tab in Moodle to be able to submit.

4 Assessment

1. Selected Lab: 15%
2. Final project code: 7.5%

3. Report about final project: 7.5%
4. (Examination: 70%)