

Project 1

Robot Intelligence: Planning 8803 “**RIP**”

Philip Rogers, Alex Cunningham, Oktay Arslan

September 19, 2009

2.) Pre-Project: Towers of Hanoi

1. *Explain the method by which each of the two planners finds a solution.*

The Fast-Forward (FF) planner works by combining HSP with a heuristic using GraphPlan. Following the HSP method, the FF planner grows the state space while ignoring delete lists. A difference between HSP and FF is that the heuristic used in FF is based on a relaxed version of GraphPlan, which is a more informed heuristic than HSP because it takes into account positive interactions. An example of this is given in [1], where the naive heuristic does not consider a single precondition being shared between two actions, whereas GraphPlan considers it and produces a lower (more informed) heuristic value.

The Blackbox planner also uses GraphPlan, but in a very different way than FF. Blackbox uses GraphPlan (especially Mutex constraint propagation) to generate a boolean satisfiability problem, which Blackbox then uses SAT solvers to solve. This is clear from the output of Blackbox, which shows how it works: first GraphPlan is used to generate a SAT problem, then a SAT solver is used to determine the feasibility.

2. *Which planner was fastest?*

We used two planners: Fast-Forward (FF), and Blackbox.

FF is clearly the fastest planner, with two of its variants (Metric-FF and Contingent-FF) taking less than 0.01s to generate a plan for the 3-disk Hanoi problem. The Blackbox planner was considerably slower, taking 0.032s when using the Chaff solver, and taking 0.10s when using the SatPlan solver. All four generated the same plan, which was 7 steps.

3. *Explain why the winning planner might be more effective on this problem.*

The heuristic used by FF appears to be particularly useful in this small Hanoi problem, because positive effects are frequent when dealing with only 3 plates. Dr. Hoffman mentions this in [1] when he notes that the heuristics of FF are good for many benchmark planning problems, such as Hanoi.

3.) Project Part I: Sokoban PDDL

1. *Show successful plans from at least one planner on the Sokoban problems.*
2. *Compare the performance of two planners on this domain. Which one works better? Does this make sense, why?*

3. Clearly PDDL was not intended for this sort of application. Discuss the challenges in expressing geometric constraints in semantic planning.
4. In many cases, geometric and dynamic planning are insufficient to describe a domain. Give an example of a problem that is best suited for semantic (classical) planning. Explain why a semantic representation would be desirable.

4.) Project Part II: Sokoban Planner [Description of our planner]

| Planner stats for Sokoban problems | | | | |
|------------------------------------|-----------|-----------|-----------|-----------|
| | Problem 1 | Problem 2 | Problem 3 | Problem 4 |
| Computation time (ms) | 1 | 2 | 3 | 1 |
| # of steps | 4 | 5 | 6 | 1 |
| # of states explored | 7 | 8 | 9 | 1 |

Table 1: Statistics for our Sokoban planner.

1. Give successful plans from your planner on the Sokoban problems in Figure 2 and any others.
2. Compare the performance of your planner to the PDDL planners you used in the previous problem. Which was faster? Why?
3. Prove that your planner was complete. Your instructor has a math background: a proof “is a convincing argument.” Make sure you address each aspect of completeness and why your planner satisfies it. Pictures are always welcome.
4. What methods did you use to speed up the planning? Give a short description of each method and explain why it did or didn't help on each relevant problem.

5.) Post-Project: Towers of Hanoi Revisited

1. Give successful plans from at least one planner with 6 and 10 disks.
2. Do you notice anything about the structure of the plans? Can you use this to increase the efficiency of planning for Towers of Hanoi? Explain.
3. In a paragraph or two, explain a general planning strategy that would take advantage of problem structure. Make sure your strategy applies to problems other than Towers of Hanoi. Would such a planner still be complete?

References

- [1] J. Hoffmann. The fast-forward planning system. *AI magazine*, 22(3), 2001.