16-782, Fall '20, Planning Techniques for Robotics
Homework III: Symbolic Planning

Name:Radhika Mohanan
Andrew ID-rmohanan

1) **Code Execution Instruction:**
   In Windows for executing the code use the following commands:

   g++ planner.cpp -o planner.out
   planner.out environment.txt

   where environment.txt is the environment file (eg. Blocks.txt, BlocksTriangle.txt, FireExtinguisher.txt)

2) **Planner has following IDs**.
   0 -> Base Planner without heuristics
   1-> Planner with inadmissible heuristics
   2-> Planner with admissible Heuristics
   Different planners can be run by changing the heuristic values (to 0 or 1 or 2) during its initialization in int heur_val=0; in the code (line 31)

3) **Environment:**
   The code has been tested on all the three environments given: Blocks, BlocksTriangle and FireExtinguisher. All the three environment have different symbols, actions, initial and goal conditions. The planner was able to generate optimal results for each of the environments. Thus the planner is domain independent and generates a plan for any environment that follows the strips representation

4) **Planner Details**
   A) Base Planner without heuristics: The heuristics Value is set to zero. The planner is a basic Dijkstra search where the open list prioritizes based on the g values of the nodes.
   B) Planner with inadmissible heuristics: The Edge cost here is assumed to be 1. The heuristics is the difference between the total number of conditions in the goal state and the number of conditions in the given state that are found in the goal state. These heuristic overestimates as multiple conditions can be satisfied by a single action. Search algorithm used is Astar.
   C) Planner with admissible heuristics: The heuristic is calculated for each state using a forward Dijkstra search. It does not overestimate since it accounts for actions satisfying multiple conditions. Search algorithm used is weighted A star with epsilon =2, which is expected to reduce the planning time significantly.

5) Results-Plan

A) Blocks Plan:
MoveToTable(A,B)
Move(C,Table,A)
Move(B,Table,C)

B) BlocksTriangle Plan:
MoveToTable(T1,B3)
MoveToTable(T0,B0)
MoveToTable(B0,B1)
Move(B1,B4,B3)
Move(B0,Table,B1)
Move(T1,Table,B0)

C) FireExtinguisher Plan:
MoveToLoc(A,B)
LandOnRob(B)
MoveTogether(B,W)
FillWater(Q)
MoveTogether(W,F)
TakeOffFromRob(F)
PourOnce(F)
LandOnRob(F)
MoveTogether(F,W)
FillWater(Q)
Charge(Q)
MoveTogether(W,F)
TakeOffFromRob(F)
PourTwice(F)
LandOnRob(F)
Charge(Q)
MoveTogether(F,W)
FillWater(Q)
MoveTogether(W,F)
TakeOffFromRob(F)
PourThrice(F)

6) Performance Summary

Table 1: Results obtained for different planners and environments

Environment	Planner Type	Time(secs)	States Expanded	Number of Moves
Blocks	Base	0.061	13	3
	Inadmissible	0.01	4	3
	Admissible	0.152	3	3
Blocks Triangle	Base	10.01	362	6
	Inadmissible	2.491	93	6
	Admissible	147.417	6	6
Fire Extinguisher	Base	4.738	410	21
	Inadmissible	4.611	410	21
	Admissible	269.163	21	21

The above table provides summary of performances of 3 planners for 3 different environments given. Time taken, number of states expanded, and number of plan moves are compared. All the 3 planners take optimal number of steps for each environment, therefore the number of moves are same for different planners in each environment. In terms of time taken the inadmissible heuristic planner takes shortest time, but the number of states expanded are lowest for admissible heuristics. Except for **Blocks** environment, the time taken by admissible heuristics planner is greater than 30secs, rest all the planners take lesser than 30 secs for all the mentioned environments.

7) Conclusion

The inadmissible heuristics planner is faster than the rest of the planners for the given environments and it can be used to get quicker results. But the states expanded by it is still higher than the admissible heuristics and the results can be sub-optimal. Admissible heuristics provide lesser states expansion, but the computational complexity is high, taking longer time. Using heuristics is better than without heuristics as it provides more scope for improvement in terms of time taken, number of states expanded, and optimality of the results obtained by the planners. The base planner is neither faster than the inadmissible heuristics planner nor expands lesser nodes as compared to admissible heuristics planner. It provides average results. For the use of admissible or inadmissible heuristics there is a trade-off between time taken and optimality required by the planners for the environment.