Research Review of Planning Algorithms

Shreyash Gotee

September 16, 2017

Introduction

This report discusses 3 different planning algorithms used for solving planning problems. There are a number of techniques that can be used to solve the planning problem like total-order planning, GraphPlan, ADL etc. However, this report discusses partial-order planning and talks about different algorithms developed over this technique. The three algorithms discussed in this report are UCPOP[1], UNPOP[2] and RePOP[3].

Partial order Planning

Partial-order planning (POP) is an approach to automated planning that leaves decisions about the ordering of actions as open as possible. For any given problem that requires a sequence of actions to achieve the goal, POP specifies all the actions needed to achieve the goal state but specifies their order only when necessary. It is based on the 'Principle of Least Commitment'. This results in a smaller search space, thereby making it efficient. However, this might not result in an accurate plan all the time.

UCPOP

Penberthy and Weld came up with UCPOP in 1992. It handles a subset of Pednault's ADL action representation. It begins with an initial plan that has only a 'start' step (whose effects have the initial conditions) and a 'goal' step (whose preconds have the goals). The algorithm adds an action to the plan if it either supports an 'open' precond or resolves a 'threat' or both. A solution is returned when it finds a causal link for every subgoal.

UCPOP has all the advantages of the general POP but since POP is not accurate all of the time, backtracking might be required in some cases.

UNPOP

UNPOP is not a POP technique. It applies Greedy Regression-match graph over means-end analysis to solve the automated planning problem. McDermott felt that POP was not a good approach for automated planning and other techniques weren't receiving enough credit. Hence he named his method UN-POP.

In 1996, McDermott suggested that while creating a plan to reach a goal, every sub-goal created must be explored immediately by back-chaining until we can verify that there is an action A which is feasible and leads to the goal. In other words, we recompute the entire sub-goal hierarchy after each action.

This technique has two benefits: a heuristic estimator of the number of actions required to get to a problem solution from the current state; and a set of actions that are good candidates for the next step to take. The resulting situation-space search algorithm searches many fewer states than traditional planners on a large class of problems, although it takes longer than usual per state.

RePOP

Partial-order planning fell out of favor in the late 1990s as faster methods emerged. Nguyen and Kambhampati (2001) suggest that a reconsideration is merited: with accurate heuristics derived from a planning graph, their RePOP (Revival for POP) planner scales up much better than GRAPHPLAN in parallelizable domains and is competitive with the fastest state-space planners.

RePOP is implemented on top of UCPOP. It ranks the partial plans by using an effective distance-based heuristic estimator, exploits reachability analysis by using invariants to discover implicit conflicts in the plan and resolves unsafe links by posting disjunctive ordering constraints into the partial plan.

RePOP generates partially ordered plans that are typically the shortest and most flexible. Graphlplan produces optimal number of time steps but RePOP comes close. Ultimately, RePOP was proved to be more effective than POP and retains the flexibility of POP without any efficiency penalty.

Conclusion

UCPOP was one of the earliest POP technique to be considered efficient, but it was not accurate enough. This led to the development of RePOP which finally brought POP techniques in the realm of truly effective planning algorithms. UNPOP, though developed to prove POP technique completely useless, gave rise to a number of state-space search planning algorithms, some of which are still considered the most efficient algorithms. Thus, all three algorithms have played a crucial part in the overall development of Artificial Intelligence.

References

[1] J. Scott Penberthy and Daniel S. Weld, *UCPOP: A Sound, Complete, Partial Order Planner for ADL.* In KR-92

[2] Drew McDermott, *A Heuristic Estimator for Means-Ends Analysis in Planning.* In AIPS-1996

[3] XuanLong Nguyen and Subbarao Kambhampati, *Reviving Partial Order Planning.* In IJCAI-01