Martin Přílučík – Artificial Intelligence nanodegree

Planning Search - research review

Purpose

Purpose of this document is to present summary of three important developments in planning and search and highlight its impact on AI as whole.

STRIPS

STRIPS (STanford Research Institute Problem Solver) was presented on 1971 by Fikes and Nilsson [1] as a problem solving program implemented in LISP. STRIPS is searching space of "world models" to to find one with goal achieved (198). A world model is represented by as set of well-formed formulas (wffs) of the first order predicate calculus . The process of theorem proving and searching a space of world models was separated (190). STRIPS defined an operator having precondition, add list and delete list (197).

Impact:
The language used by STRIPS influenced the AI planning domain and was close to what is called "classical" language (Russel, Norvig, 393).

Heuristic Search Planner

The idea of planning as heuristic search as was presented by Bonet and Geffner [2]. Heuristic Search Planner (HSP) is a planner based on the ideas of heuristic search. Heuristic search algorithms perform forward search from an initial state to a goal state using an heuristic function that provides an estimate of the distance to the goal.

Heuristic search planners transform planning problems into problems of heuristic search by automatically extracting heuristics from STRIPS encodings. They differ from specialized problem solvers such as those developed for the 24-Puzzle and Rubik's Cube in that they use a general declarative language for stating problems and a general mechanism for extracting heuristics from these representations. (Bonet, Geffner, 1)

The heuristic function allowed to deal with any STRIPS planning problem as a problem of heuristic search. This means algorithm such as A* could be used for planning.

HSP: A hill-climbing planner
HSP uses the additive heuristic h_{add} to guide a hill-climbing search from the initial state to the goal. The hillclimbing search is very simple: at every step, one of the best children is selected for expansion and the same process is repeated until the goal is reached. (Bonet, Geffner, 10)

HSP2: A best-first search planner
HSP is competitive with the best current planners over many domains. However, HSP is not an optimal planner, and what's worse - considering that optimality has not been a traditional concern in planning - the search algorithm in HSP is not complete. The last problem can be overcome by switching from hill-climbing to a best-first search (BFS). The BFS planner is superior in performance to HSP, and it appears to be superior to some of the best planners over a large class of problems. HSP2 uses the WA^* algorithm with the non-admissible heuristic h_{add} which is evaluated from scratch in every new state generated (Bonet, Geffner, 10).

Impact :
In the AIPS98 Planning Contest, the HSP planner showed that heuristic search planners can be competitive with state-of-the-art Graphplan and SAT planners (Bonet, Geffner, 1).
HSP was first to make state-space search practical for large planning problems (Russel, Norvig, 395).

GRAPHPLAN

The idea of planning through graph analysis was presented by Blum and Furst [4]. Graphplan is a planner in STRIPS-like domain and the algorithm is based on planning graph analysis (Blum, Furst, 281). It is not immediately starting with a search but instead begins with constructing a structure called planning graph. The algorithm guarantees it will find the shortest plan among those in which independent actions may take place at the same time. (Blum, Furst, 282).

Planning graph consist of two kind of nodes and three kind of edges.The levels alternate between proposition and action nodes (Blum, Furst, 284)

Another important concept that graphplan uses is mutual exclusion among nodes. It identifies and records mutual exclusion relationships and propagates them using simple rules like *inference* and *competing needs* (Blum, Furst, 286).

During experimentation Graphplan outperformed other popular planners on natural and artificial problems. The four major factors were mutual exclusion, parallel plans, memorizing and low level costs (Blum, Furst, 295).

One main limitation of Graphplan is that it applies only to STRIPS-like domains (Blum, Furst, 299).

Impact:
Graphplan was orders of magnitude faster than the partial-order planners of that time and initiated development of other graph-planning system as IPP, STAN (Russell, Norvig, 395).

References

[1] Richard E. Fikes, Nils J. Nilsson : *STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving*, Artificial Intelligence 2 (1971), 189-208

[2] Blai Bonet, Héctor Geffner: *Planning as heuristic search*, Artificial Intelligence 129 (2001), 5–33

[3] Russell, S. and Norvig, P. (2010). *Artificial intelligence*. New Jersey: Pearson.
[4] Avrim L. Blum , Merrick L. Furst: *Fast planning through planning graph analysis*, Artificial Intelligence 90 (1997) 28 I-300