Artificial Intelligence Nanodegree

Implement a Planning Search – Research Review

Ollie Graham
22nd October 2017

AI planning arose from investigations into state-space search, theorem proving, and control theory and from the practical needs of robotics, scheduling, and other domains.[1]

Planning Languages

The first major planning system was known as the **St**anford **R**esearch **I**nstitute **P**roblem **S**olver (STRIPS), which was developed by Richard Fikes and Nils Nilsson in 1971 at SRI International.[2] The same name was later used to refer to the formal language of inputs to this planner, which became the base for most of the languages for expressing automated planning problem instances in use today. These languages are commonly known as action languages.

The Problem Domain Description Language, or PDDL (inspired by STRIPS and the action decription language (ADL)), was introduced in 1998 as a computer-parsable standardised syntax for representing planning problems.[3] There have been several extensions since then, PDDL3.1 being the latest version in use today.[4]

Partial Order Planning

In the 1970s planning generally considered totally ordered action sequences. A subplan for each decomposed part of a problem was generated and then the subplans were strung together in some manner to generate an overall plan. This was termed linear planning by Sacerdoti[5]. This approach was soon discovered to be incomplete as it cannot be used to solve some very simple problems as shown by Allen Brown during experimentation with the HACKER system.

A complete planner must allow for interleaving of actions from different subplans within a single sequence.

A solution to the interleaving problem was goal-regression planning, a technique in which steps in a totally ordered plan are re-ordered so as to avoid conflict between subgoals. This was introduced by Waldinger and also used by Warren's WARPLAN.

WARPLAN was notable because it was the first planner to be written in the logic programming language PROLOG, taking only 100 lines of code which was a small fraction of comparable planners of the time and showed the efficiency of such logic programming languages.

Research in partial order planning concentrated on the detection of conflicts and the protection of achieved conditions from interference.[6] The construction of partially ordered plans was pioneered by the NOAH planner and by Tate's NONLIN system.[7]

Partial order planning dominated the next 20 years of research, but fell out of favour in the late 1990s as faster methods emerged.

Graphplan

Avrim Blum and Merrick Furst revitalised the field of planning with the GRAPHPLAN system which performed several orders of magnitude faster than the partial order planners of the time.[8]

Given a problem statement Graphplan explicitly constructs and annotates a compact structure called a Planning Graph, in which a plan is a flow of truth-values through the graph. Useful information for constraining search can quickly be propagated through the graph as it is being built.

Helmert (2001) analysed several classes of planning problems and shows that constraint based approaches such as GRAPHPLAN and SATPLAN are best for NP hard domains, whilst search based approaches do better in domains where feasible solutions can be found without backtracking.

Graphplan and Satplan have trouble in domains with many objects because that means they must create many actions. This can be addressed in some cases by generating propositionalised actions dynamically, only as needed, rather than instantiating them all before the search begins.

Planning research has been central to AI since its inception and will continue to be so as the domain evolves.

References:

1. *Artificial Intelligence: A Modern Approach by Norvig and Russell p. 393 Ch 10.6*
2. *https://en.wikipedia.org/wiki/STRIPS*
3. *McDermott, Drew; Ghallab, Malik; Howe, Adele; Knoblock, Craig; Ram, Ashwin; Veloso, Manuela; Weld, Daniel; Wilkins, David (1998). "PDDL---The Planning Domain Definition Language" (PDF). Technical Report CVC TR98003/DCS TR1165. New Haven, CT: Yale Center for Computational Vision and Control. CiteSeerX 10.1.1.51.9941*
4. *https://en.wikipedia.org/wiki/Planning_Domain_Definition_Language*
5. Earl D. Sacerdoti – The Non Linear Nature of Plans (1975-09-03) IJCAI'75 Proceedings of the 4th international joint conference on Artificial intelligence - Volume 1 pp 206 - 214
6. G.J. Sussman (1975) *A Computer Model of Skill Acquisition* Elsevier Science Inc. New York, NY, USA. Book version of his PhD thesis.
7. http://www.aiai.ed.ac.uk/project/early-planners/
8. A. Blum and M. Furst, "Fast Planning Through Planning Graph Analysis" [pdf], Artificial Intelligence, 90:281--300 (1997).