AIND Project 3: Research Review: AI Planning and Search

Arun K Viswanathan

3/14/2017

Overview

This paper reviews historical developments in the field of AI planning and search, highlighting the relationships between the developments and their impact on the field of AI as a whole. Much of this material is sourced from AIMA Chapter 10 (3rd Edition).

Representation Languages

Several important developments have taken place in the area of representation languages used for AI planning problems. The "classical" language comes from STRIPS (Fikes and Nilsson, 1971). This was further improved by the Action Description Language, or ADL (Pednault, 1986), which remoevd some of STRPS' restrictions, enabling more realistic problems to be encoded. The next major improvement was the Planning Domain Description Language, or PDDL (Ghallab et al, 1998). PDDL was introduced as computer-parseable and has become the *de facto* standard. The most recent version, PDDL 3.0 (Gerevini and Long, 2005), includes plan constraints and preferences.

Action Sequences

Another area of major developments is in the formulation of action sequences. Early planners in the 1970s considered totally ordered action sequences and decomposed problems into ordered sub plans with sub goals. This approach, known as linear planning, was discovered to be incomplete since it does not allow for actions to be interleaved. This led to a new strategy of goal regression planning (Waldinger, 1975) which reorders steps in a totally ordered plan to avoid conflicts between steps. Further evolution lead to partial-order planning which was is use for over 20 years from the 1970s to the 1990s. This then fell out of use with the emergence of the faster technique of state space planning, beginning with unPOP (McDermott, 1996). These planners use heuristics like the "ignore-delete-list", preprocessing of action schemas to make constraints more explicit, forward and backward searches.

Action Schema Representations

There has been significant evolution in the choice of representation for action schemas. Situation calculus was first introduced to allow the use of first-order logic and overcome some problems with PDDL (McCarthy 1993). An interesting finding was that the most compact forms of propositionalized action schemas did not necessarily lead to the fastest solution times (Kautz et al, 1996). Compact data structures for boolean expressions, known as binary decision diagrams have recently been studied. These have come out of studies in the hardware verification community. Integer programming has also been studied (Vossen et al, 2001).