

Research Review

By Sunil Thakur

This short review consists of a historical report on different action and planning languages, and how they had an impact on the AI field as a whole.

Introduction

Planning is very important concept in the field of artificial intelligence where the agent can use the structure of the problem to make a plan of actions to achieve some goal. The main idea is to reach to goal state, which consists of state variables and actions make some facts true and others false. Given a description of the possible initial states of the world, a description of the desired goals, and a description of a set of possible actions, the planning problem is to find a plan that is guaranteed (from any of the initial states) to generate a sequence of actions that leads to one of the goal states.

Historical developments

STRIPS (1971)

Stanford Research Institute Problem Solver, a first major planning system created by Richard Fikes and Nils Nilsson from the Stanford Research Institute (SRI). It introduced notions like actions having preconditions and postconditions, and is the primary inspiration for later planning languages. This language described a set of applicable operators that allowed transforming one state into a different state. This definition of a framework to solve complex planning problems has been a central to much of the research in artificial intelligence.

STRIPS was the main planning component for SRI's Shakey robot project.

ADL(1986)

Action Description Language (ADL) is another automated planning language, considered an improvement over STRIPS. Formulated in 1986 by Edwin Pednault, it extends the syntax of STRIPS action schemata, dropping some of STRIP's restrictions and allowing more complex problems to be encoded (Russell and Norvig, 2010). It allows disjunctive, quantified, conjunctive and negative

preconditions, besides conditional effects. Additional to increasing the domain encoding convenience, ADL reduces the size of domain descriptions needed (Coles and Smith, 2007). Having such extended language was advantageous during the time since it was possible to compile it to STRIPS, making it backwards compatible.

PDDL(1998)

On the path of standardization of planning domain and problem description languages, the **Planning Domain Definition Language (PDDL)** was created by Drew McDermott. PDDL descended from several forebears, which is why it supports STRIPS, ADL and others, though not all planners fully implement PDDL. Many variations and extensions of PDDL has been created since its origin having several extensions, including features like non-propositional state-variables (NDDL) and notation to create ontologies (MAPL).

WARPLAN

Early research in the planning domain normally used linear programming, which considers totally ordered action sequences. This approach was soon learned to be incomplete and the notion of interleaving; being able to interleave actions from different sub-plans within a single sequence was introduced. WARPLAN is a planner written by David Warren which implements a solution known as goalregression planning to the interleaving problem. WARPLAN was the first planner to be written in a logic programming language (Prolog). The implementation of a planner using such language was able to showcase the great benefits in terms of reduced complexity that can be achieved by using logic-programming languages. WARPLAN is only 100 lines of code.

References

1. Richard E. Fikes, Nils J. Nilsson (Winter 1971). "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving".
2. Stuart J. Russell, Peter Norvig (2010), *Artificial Intelligence: A Modern Approach* (3rd Edition).
3. Fikes, R. and Nilsson, N. (1993). STRIPS, a retrospective. *Artificial Intelligence*, 59(1-2), pp.227-232