Research Review

Marco Zorzi

AI planning grow up thanks to the study and development of state-space search, theorem proving, and control theory from the real needs of robotics, scheduling, and other domains.

In 1971 with “STRIPS - STanford Research Institute Problem Solver” by Fikes and Nilsson was the first major planning system for a robot project and the representation language used by STRIPS called LISP has been far more important than its algorithmic approach: in fact, what we call the classical language is like what STRIPS used. In 1986 Pednault with Action Description Language relaxed some of the STRIPS restriction to made possible to encode more realistic problems. In 1992 Bylander shows that simple STRIPS planning is PSPACE-complete and then Nebel, in 2000, explores schemes for compiling Action description language (ADL) into STRIPS. ADL is an automated planning and scheduling system used mostly in robotics.

Planners in the early 1970s generally considered totally ordered action sequences and problem decomposition was achieved by computing a subplan for each subgoal and then stringing the subplans together in some in order. This approach is called linear planning by Sacerdoti in 1975 but was soon discovered to be incomplete. In the same year Richard Waldinger introduced a technique in which steps in a totally ordered plan are reordered to avoid conflict between subgoals with goal-regression planning.

So, Partial-order planning dominated the next 20 years of research and the first clear formal exposition was TWEAK in 1987 by David Chapman. TWEAK is a planner that was simple enough to allow proofs of completeness and intractability with NP-hardness and undecidability of various planning problems.

Chapman’s work led to describe a complete partial-order planner in 1991 by McAllester and Rosenblitt but type of planning declined in the late 1990s in favor of faster emergent methods.

In 1998 the Problem Domain Description Language, or PDDL by Ghallab, was introduced as a computer-parsable, with standardized syntax for representing planning problems and has been used as the standard language for the International Planning Competition. With a Gerevini and Long work’s, in 2005, Problem Domain Description Language evolves in 3.0 version and it includes plan constraints and preferences. In 2008 and 2011 it evolves in 3.1 version: It introduced object-fluents (i.e. functions' range now could be not only numerical (integer or real), but it could be any object-type also). Thus PDDL 3.1 adapted the language even more to modern expectations with a syntactically seemingly small, but semantically quite significant change in expressiveness.

In 2001 Helmert analyzes several classes of planning problems and the results were:

- constraint-based approaches such as GRAPHPLAN is best for NP hard domains but not in domains with many objects because that means they must create many actions
- search-based approaches do well in domains where feasible solutions can be found without backtracking.

It’s clear that planning research has been central to AI since its beginning and continue to be important nowadays.

***Reference**:*

Book:

Artificial Intelligence a Modern Approach (3rd Edition), Chapter 10, “BIBLIOGRAPHICAL AND HISTORICAL NOTES”.

Site:

https://en.wikipedia.org/wiki/STRIPS
https://en.wikipedia.org/wiki/Action_description_language
https://en.wikipedia.org/wiki/Planning_Domain_Definition_Language