Research review

This review focused on languages for AI Planning systems.

One of the fist main planing system was STRIPS (1971) and it's purpose was to find some composition of operators that transforms the current initial state of the world/model to one that satisfy some goal criteria using lists of adding and deleting clauses. As a result, in STRIPS a world model is represented by a set of well formed formulas of the first order predicate calculus. [1]

Based on STRIP another family of approaches born in 1975 called Hierarchical task network (HTN). The purpose of this systems was to match needs and language of common project planning. Planning problem is specified in the HTN approach by providing a set of tasks, which can be: - primitive tasks, which roughly correspond to the actions of STRIPS; - compound tasks, which can be seen as composed of a set of simpler tasks; It's an analogue for a project major step between milestones, or a project subsystem in classical planning. - goal tasks, which roughly corresponds to the goals of STRIPS, but are more general. Constraints among tasks are expressed in the form of networks, called task networks. A task network is a set of tasks and constraints among them. Such a network can be used as the precondition for another compound or goal task to be feasible. [2] One of the first implementations was Nonlin, which was able to build partially ordered executing plans for HTN formalized problems. [3]

STRIPS representation method made big influence to AI and was adopted by more flexible and general Action Description Language (ADL) in 1987 in IBM. Contrary to STRIPS, the principle of the open world applies with ADL: everything not occurring in the conditions is unknown (Instead of being assumed false). In addition, whereas in STRIPS only positive literals and conjunctions are permitted, ADL allows negative literals and disjunctions as well. ADL has more represented abilities, on the same time, most actual planner implementations translated ADL to STRIPS before executing search. [4]

Planning Domain Definition Language (PDDL) introduced in 1998 in [5] is an attempt to standardize planning languages and make a flexible way to execute plan searching with multiple different solver system based on one description. Interesting, that the first version contained feature like "STRIPS-style actions", which shows how big influence STRIPS had for the whole field and became a standard approach. PDDL became a new standard and as a real standard has different versions with new features and multiple extensions. [6]

References

1. Fikes, R. E., & Nilsson, N. J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. Artificial intelligence, 2(3-4), 189-208.
2. Wikipedia, Hierarchical Task Network
3. Tate, A. (1976) "Project Planning Using a Hierarchic Non-linear Planner", D.A.I. Research Report No. 25, August 1976, Department of Artificial Intelligence, University of Edinburgh.
4. Pednault, E. P. (1989). ADL: Exploring the Middle Ground Between STRIPS and the Situation Calculus. Kr, 89, 324-332.

5. McDermott, D., Ghallab, M., Howe, A., Knoblock, C., Ram, A., Veloso, M., ... & Wilkins, D. (1998). PDDL-the planning domain definition language.
6. Wikipedia, Planning Domain Definition Language