Research Review: AI Planning and Search

1. Introduction

Planning in Artificial Intelligence is the ability of making decision intelligently through the construction of sequences of actions to achieve a goal. It involves the representation of actions and world models, reasoning about the effects of actions, and techniques for efficiently searching the space of possible plans. Planning techniques have been applied in a variety of tasks including production line scheduling, construction planning, cargo fleet planning and spacecraft mission control. This report discusses three key developments in the field of AI planning and search.

2. STRIPS

STRIPS (Stanford Research Institute Problem Solver) [1] is an automated planner first introduced by Richard Fikes and Nils Nilsson in 1971 and used in Shakey, the robot developed at the Stanford Research Institute (SRI). It was used to control Shakey to perform tasks like moving objects or navigating around an environment with multiple rooms. It was design with the mind of solving problems in real-time efficiently on the very limited hardware of the time. In STRIPS, a world model is represented by an arbitrary collection of first-order predicate calculus formulas. The actions are represented by operators, which have a set of preconditions and side effects. A STRIPS instance is composed of an initial world model, a set of operators and the goal statement. The role of planning system is to find sequences of operators which lead from initial state to given goal. Both Action Description Language (ADL) [2] and Problem Domain Description Language (PDDL) [3] which are used widely today are considered as advancement of STRIPS.

2. GRAPHPLAN

GRAPHPLAN [4] is a novel algorithm to build a planning graph in STRIPS-like domains. Given a problem statement, GRAPHPLAN first constructs a Planning Graph -- a special data structure that provides a solution to a relaxed planning problem by reducing the branching factor in search algorithms. This structure is annotated with edges indicating sets of pairwise exclusive actions and states. Once the Planning Graph is completed, the algorithm performs a backward search through the graph to find a valid partially ordered plan. Before GRAPHPLAN appeared in 1995, most planning researchers were working under the framework of “plan-space search”. GRAPHPLAN outperformed those prior planners by orders of magnitude. Other graph-planning systems, such as IPP [5], STAN [6] and SGP [7] have been influenced by GRAPHPLAN.

3. SATPLAN

SATPLAN (or better known as Planning as Satisfiability) [8] is a powerful approach to domain-independent planning, first proposed by Henry Kautz and Bart Selman in the 1990s. The basic idea is to consider a bounded-horizon planning problem, to represent the values of state variables at every time point as propositional variables, to represent the relation between two consecutive states as a propositional formula, and then to synthesize a propositional formula that is satisfiable if and only if there is a plan of the given bounded length [9]. Since then, many variations of SATPLAN were introduced and remain among the most effective ways of solving difficult problems by taking advantage of the advances in Boolean Satisfiability Problem (SAT) solvers.

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] Pednault, E. P. (1986). Formulating multiagent, dynamic-world problems in the classical planning framework. *Reasoning about actions and plans*, 47-82.

[3] M., Howe, A., Knoblock, C. A., and McDermott, D. (1998). PDDL—The planning domain definition language. Tech. rep. DCS TR-1165, Yale Center for Computational Vision and Control.

[4] Blum, A. L., & Furst, M. L. (1997). Fast planning through planning graph analysis. *Artificial intelligence*, *90*(1), 281-300.

[5] Koehler, J., Nebel, B., Hoffmann, J., & Dimopoulos, Y. (1997, September). Extending planning graphs to an ADL subset. In *European Conference on Planning* (pp. 273-285). Springer, Berlin, Heidelberg.

[6] Fox, M., & Long, D. (1998). The automatic inference of state invariants in TIM. *Journal of Artificial Intelligence Research*, *9*, 367-421.

[7] Weld, D. S., Anderson, C. R., & Smith, D. E. (1998, July). Extending graphplan to handle uncertainty & sensing actions. In *Aaai/iaai* (pp. 897-904).

[8] Kautz, H. A., & Selman, B. (1992, August). Planning as Satisfiability. In *ECAI* (Vol. 92, pp. 359-363).

[9] Rintanen, J. (2012). Planning as satisfiability: Heuristics. Artificial Intelligence, 193, 45-86.