

Planning: Research Review

Raphael Ballet

This work presents a short report on three relevant developments in the field of artificial intelligence planning.

STRIPS Acronym for "Stanford Research Institute Problem Solver", it was a problem solver designed to solve planning tasks for a mobile robot at SRI [Fikes and Nilsson, 1993]. It attempted to create a representation of a planning problem using the definition of world models (formal description of the states), goal states, and operators, that is, actions that transform a given world model to another world model. The algorithm then used theorem-proving methods together with a state-space search algorithms in order to find a sequence of actions (operators) needed to reach a goal state given an initial world model [Fikes and Nilsson, 1971]. The theorem-proving methods were used to obtain applicable operators at a given world model, as well as to test if it has reached the goal state.

Furthermore, even though the STRIPS algorithm had an important role in planning problems, it was its formal language representation to the planning problem that is the base for most actual planning algorithms [Russell and Norvig, 2009]. One practical example is the *Problem Domain Description Language* (PDDL) [McDermott et al., 1998], a problem-specification language used as a standard for most actual planning search problems which uses STRIPS, ADL and others.

Partial-Order Planning Compared with the so-called "total-order planning", that is, an algorithm responsible to find an unique and ordered sequence of actions to achieve all the subgoals [Russell and Norvig, 2009]; the partial-order planning tries to obtain a set of actions needed to achieve each subgoal independent of its final ordering [Sacerdoti, 1975]. The final action order is then obtained by combining all the independent subgoal plans.

Partial-order planning enabled more flexibility and efficiency in planning problems. It could obtain all the sequence of actions needed to solve a given subproblem, thus avoiding the need to search through all state space. Examples of application of such

concept include the pioneer NOAH [Sacerdoti, 1975] and the TWEAK [Chapman, 1987] problem solvers.

Graphplan The Graphplan was developed by Blum and Furst [1997] and reduced considerably the speed and the amount of search needed compared with other traditional total-order or partial-order planning methods. At first, it uses planning domains represented in STRIPS language to construct a graph structure, called planning graph, in such a way that it can identify and remove possible action and state constraints, thus pruning impossible nodes [Blum and Furst, 1997]. The planning graph is organized in levels such that each level correspond to all the possible actions and states reachable at a given time-step [Russell and Norvig, 2009]. Using the planning graph, it then uses a search algorithm, namely Graphplan, to return the path with minimum length.

References

- Avrim L Blum and Merrick L Furst. Fast planning through planning graph analysis. *Artificial intelligence*, 90(1):281–300, 1997.
- David Chapman. Planning for conjunctive goals. *Artificial intelligence*, 32(3):333–377, 1987.
- Richard E Fikes and Nils J Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial intelligence*, 2(3-4):189–208, 1971.
- Richard E Fikes and Nils J Nilsson. Strips, a retrospective. *Artificial Intelligence*, 59(1):227–232, 1993.
- Drew McDermott, Malik Ghallab, Adele Howe, Craig Knoblock, Ashwin Ram, Manuela Veloso, Daniel Weld, and David Wilkins. Pddl-the planning domain definition language. 1998.
- Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009. ISBN 0136042597, 9780136042594.
- Earl D Sacerdoti. The nonlinear nature of plans. Technical report, DTIC Document, 1975.