# Planning Research Review

Andrei Savin March 20, 2017

## STRIPS

STRIPS (Stanford Research Institute Problem Solver) is an automated planning system, which came to refer to the formal language in which inputs to the planner were specified. In the introductory paper, Richard E. Fikes and Nils J. Nilson from the Stanford Research Institute describe STRIPS as a "problem solver that searches the space of world models to find one in which a given goal is achieved". The task of STRIPS is to find a sequence of actions that transforms the initial state into the goal state [FN71].

STRIPS represents a state as a conjunction of atoms, either positive literals or first-order literals. For example, a STRIPS state might be $Have(Cake) \land Eat(Cake)$. A goal is a partially specified state. We say that a state $s$ satisfies a goal $g$ if $s$ contains all the atoms in $g$.

Actions are specified as a precondition and an effect. The precondition is a conjunction of positive literals that must be true before the action is executed. The effect is a conjunction of literals describing how the state changes when the action is executed.

The *STRIPS assumption* is that any effect literal not mentioned in the effect of an action remains unchanged [RN09, p. 375].

This action representation introduced by STRIPS is widely regarded as one of the most important contributions in the field of planning. Almost all planning systems since the 1970s have used some variant of STRIPS [RN09, p. 410].

## Partial-Order Planning

Partial-order planning is an approach to exploring the space of plans (planning graph) leaving the ordering of actions as open as possible (as opposed to total-order planning, which produces an exact ordering of actions).

The problem with totally-ordered plan searches (such as the forward and backward state-space search seen in the lectures) is that they cannot take advantage of problem decomposition and use parallel processing and divide-and-conquer algorithms. Partial-order planning allows us to work on several sub-goals independently, solves them with several subplans, and then combines the subplans [RN09, p. 387].

The strategy used is called "least commitment". Plans are represented in a flexible way that enable deferring decisions. For example, if to make a trip to France, we must purchase a travel guide and a plane ticket, it is not necessary to decide (yet) which purchase should be executed first [Wel94, p. 12]

For two unordered actions, the planner can manipulate the two sub-sequences independently, which allows for parallel computation.

Some idea underlying partial-order planning include the detection of conflicts and the protection of achieved conditions from interference [Wel94, p. 410].

Partial-order planning dominated planning research in the 1970s and 1980s, but they have fallen out of fashion since the 1990s.

## Graphplan

Graphpplan is an automated planning algorithm developed by Avrim Blum and Merrick Furst in 1995. It takes a problem expressed in STRIPS and produces a sequence of operations for reaching the goal state [BF97]

Graphpplan extracts a plan directly from the planning graph of a problem (instead of using the planning graph to provide a heuristic as we did in this project). First, Graphplan checks whether all the goal literals are present in the current level and there is no mutual exclusion between any pair of them. If this is the case, a solution might exist within the current graph. Otherwise, it expands the graph by adding the actions for the current level and the state literals for the next levels. The process continues until a solution is found or the algorithm learns that no solution exists [RN09, p. 398].

Graphplan outperformed by orders of magnitude all previous planners when it was released in 1995 (state-space search planners and partial-order planners of the time). Graphplan revitalized the field of planning and inspired other graph planning systems, such as IPP (1997), STAN (1998) and SGP (1999) [RN09, p. 410]

## References


[FN71] Richard E. Fikes and Nils J. Nilsson. “STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving”. In: *Artificial Intelligence* 2 (1971), pp. 189–208. DOI: `http://ai.stanford.edu/~nilsson/OnlinePubs-Nils/PublishedPapers/strips.pdf`.

[Wel94] Daniel S. Weld. “An Introduction to Least Commitment Planning”. In: *AI Magazine* (1994). DOI: `http://homes.cs.washington.edu/~weld/papers/pi.pdf`.

[BF97] Avrim L. Blum and Merrick L. Furst. “Fast Planning Through Planning Graph Analysis”. In: *Artificial Intelligence* 90 (1997), pp. 281–300. DOI: `http://www.cs.cmu.edu/~avrim/Papers/graphplan.pdf`.

[RN09] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach 3rd Edition*. Pearson, 2009. ISBN: 0136042597.