Research Review: AI Planning and Search Historical Developments and Impact

STRIPS (Stanford Research Institute Problem Solver) was the first major automated planner developed by Richard Fikes and Nils Nilsson in 1971. Instead of its algorithmic approach, its legacy was the representation language of the inputs to the planner, which is used as the *base for most planning languages today*. A STRIPS instance is composed of an initial state, a goal state, and a set of actions with preconditions and postconditions. A plan for such an instance is a sequence of actions executed from the initial state to lead it to the goal state.

Edwin Pednault proposed Action Description Language (**ADL**) in 1987, which improved upon STRIPS by allowing the effects of an operator to be conditional. While STRIPS assumes everything not occurring in the conditions to be false (closed-world assumption), ADL assumes these to be unknown (open-world assumption) and also allows negative literals and disjunctions. This enables ADL to *model more realistic problems*; for example, in a block world where a block A is twice as big as blocks B and C, the action MoveOnto(B, A) might only have the effect of negating Clear(A) if On(A, C) is already true. This conditional effect would be hard to express in STRIPS.

The Planning Domain Definition Language (**PDDL**) was developed in 1998 as an attempt to *standardize planning languages* and has been used for the International Planning Competition (IPC) since, evolving with each competition. Developments in PDDL since its inception have introduced, for example, *numeric fluents* for non-binary resouces, *plan metrics*, and *continuous actions* in PDDL2.1 and *state-trajectory constraints* and *preferences* in PDDL3.0.

Planners in the 1970s generally used **total-order planning**, where actions for a task are completely sequenced. A drawback to this *noninterleaved* approach appears when it fails to solve some very simple problems, such as the *Sussman Anomaly*, where in the block world with initial state Clear(B) and On(C, A) with a goal state of On(A, B) and On(B, C). If the planner tries to move C to clear A and move A atop B, then it cannot pursue goal On(B, C) without first clearing B. **Partial-order planning** is an approach that specifies all necessary actions, but *leaves the ordering as open as possible*, which makes it more efficient and adept at finding the quickest path.

UCPOP is an extension of partial-order planning that handles action schemata with increased expressivity through variables, conditional effects, disjunctive preconditions, and universal quantification. Sensory Graphplan (**SGP**) handles a superset of UCPOP and is much faster.

STRIPS. In *Wikipedia*. Retrieved August 1, 2017, from https://en.wikipedia.org/wiki/STRIPS
Action description language. In *Wikipedia*. Retrieved August 1, 2017, from https://en.wikipedia.org/wiki/Action_description_language
Planning Domain Definition Language. In *Wikipedia*. Retrieved August 1, 2017, from https://en.wikipedia.org/wiki/Planning_Domain_Definition_Language
Russell and Norvig, Artificial Intelligence: A Modern Approach (Third Edition)