Research Review

In this review we walk through three important development in the history of AI planning and search: STRIPS (Fike and Nilsson, 1997), TWEAK (Chapman, 1987) and GRAPHPLAN system (Blum and Furst, 1995, 1997).

STRIPS (*"STRIPS: A new approach to the application of theorem proving to problem solving,"* Fike and Nilsson, 1971) was the first major AI planning system. The most influential aspect is the representation language used by STRIPS. The same name was later used to refer to the formal language of inputs of this planner. Specifically, an instance of STRIPS is composed of: an initial state; goal states; a set of actions consisting of preconditions and effects. This language is the base of most of the languages for expressing automated planning problem instances in use today. The Problem Domain Description Language, or PDDL (Ghallab et al., 1998), for example, was introduced ast a computer-parsable, standardized syntax for representing planning problems and shas been used as the standard language for the International Planning Competition since 1998.

TWEAK (*"Planning for conjunctive goal,"* Chapman, 1987) is a planner that was simple enough to allow proof of completeness and intractability (NP-hardness and undecidability) of various of planning problems. It was the first clear formal exposition of **partial-order planning**, which is a solution to the interleaving problem of linear planning. Since 1970s, partial-order planning had been dominated for 20 years of research. Chapman shown that all conjunctive planners, linear and nonlinear, work the same way. The efficiency and correctness of these planners depends on the traditional add/delete-list representation for actions, which drastically limits their usefulness. Partial-order planning fell out of favor in the late 1990s as faster methods emerged.

Avrim Blum and Merrick Furst (1995, 1997) revitalized the field of planning with their GRAPHPLAN system (*"Fast planning through planning graph analysis,"* Blum and Furst, 1995, 1997), which was orders of magnitude faster than the partial-order planners. They constructed and analyzed Planning graph to problems in STRIPS-like domain. GRAPHPLAN always returns a shortest possible partial-order plans. A planning graph encodes the planning problem in such a way that many useful constraints inherent in the problem become explicitly available to reduce the amount of search needed. Furthermore, planning graphs have polynomial size and can be built in polynomial time.