In this review, I will cover the STRIPS, GRAPHPLAN, and Heuristic Search Planners as 3 historical developments in the field of AI planning and search.

STRIPS (Stanford **R**esearch **I**nstitute **P**roblem **S**olver)

STRIPS, was the automated planner used in the Shakey, the first mobile intelligent robot. Shakey was a major breakthrough in the artificial intelligence and robotics areas of planning, navigation and computer vision. STRIPS is also used to refer to the formal language for the formal specification of the planning problem. STRIPS provided a seminal framework where the world is regarded as being in a static state and is transformable to another state by performing a set of actions.

STRIPS provided the groundwork for further development in to representation of planning problems. **ADL** (Action Description Language) is extension of STRIPS, relaxed some of the STRIPS restriction and made it possible to encode more realistic problems. ADL allows for both positive and negative literals, assumes unmentioned literals as unknowns, quantified variables in goals, allows conjunctions and disjunctions in the goal, and has built-in support for variable types and equality predicate.

STRIPS and ADL led to standardized syntax for the planning problems in AI called the **PDDL** (Planning Domain Definition Language). This language allows researchers to exchange benchmark problems and compare results. PDDL includes sublanguages for STRIPS, ADL, and the hierarchical task network.

GRAPHPLAN

Graph Plan is an algorithm to search for a solution over the space formed by the planning graph. GRAPHPLAN is a propositional planner that repeatedly adds a level to a planning graph in a way similar to iterative deepening. Once all the goals appear in non-mutually exclusive state in the planning graph, the GRAPHPLAN invokes the search for a plan that solves the problem. If the search fails, another level is added until either a solution is found or no solution is possible.

HSP (HEURISTIC SEARCH PLANNERS)

Heuristic search planners can be characterized along three main dimensions: the state space that is searched (either progression or regression space), the search algorithm used, and the heuristics functions extracted from the problem representation. Heuristic search planners search state space with a heuristic that can be extracted from the problem encoding and then combined with standard search algorithms. Some HSP published are AltAlt (A little of this and a little of that), FF (Fast forward), GRT (Greedy Regression Tables), HSP (Heuristic Search Planner), HSP 2.0, and VHPOP (Versatile Heuristic Partial Order Planner). AltAlt extracts admissible heuristics from the planning graph constructed by a GraphPlan type of procedure and uses this heuristics to drive a regression search from the goal. FF searches the forward progression space using a novel local search strategy that combines Hill-climbing with systematic search. GRT uses STRIPS and forward search. The original version of HSP searches the progression space with a hill climbing algorithm, and a non-admissible heuristics. In HSP2.0, the search can be done either forward or backward, and the heuristics can be non-admissible as used in original

HSP, the admissible heuristics and the more informed admissible heuristics. HSP2.0 can run different options concurrently as threads.

References:

https://www.youtube.com/watch?v=7bsEN8mwUB8&feature=youtu.be

http://artint.info/html/ArtInt_204.html#id5

https://pdfs.semanticscholar.org/076a/e14bfc68acdbaf2ab24913e152d49540e988.pdf

https://bonetblai.github.io/reports/hsp2-aimag.pdf

http://rakaposhi.eas.asu.edu/kbcs-sent.pdf

Russell, S, and P Norvig. Artificial Intelligence: A Modern Approach