Historical Developments in the field of AI Planning

The *planning* problem in Artificial Intelligence is about the decision making performed by intelligent creatures like robots, humans, or computer programs when trying to achieve some goal. It involves choosing a sequence of actions that will (with a high likelihood) transform the state of the world, step by step, so that it will satisfy the goal [1]. Many developments contributed to the growth of AI research, some of them are listed here.

STRIPS: STRIPS is an automated planning system developed by Stanford Research Institute in 1971 by Fikes, Richard E., and Nils J. Nilsson [2]. In STRIPS, we define a world by providing objects, actions, preconditions and effects. Then a problem consisting of initial state and goal condition is defined. STRIPS can then search all possible states, starting from the initial one, executing various actions, until it reaches the goal [3]. In STRIPS planning, there is complete knowledge about the initial state, and the actions are deterministic with only one outcome. Its solution is a linear plan with a sequence of actions. So, it works for a lot of scenarios where the world is completely represented like stacking blocks, rubik's cube, navigating a robot, and much more. But STRIPS will not work for situations where we don't have complete information about initial state, or the actions are non-deterministic, or the environment is dynamic.

Partial-Order Planning: While STRIPS is a forward planner that enforces total ordering on actions at all stages of the planning process, the idea of a partial-order planner is to have a partial ordering between actions and only commit to an ordering between actions when forced. This is sometimes also called a non-linear planner, which is a misnomer because such planners often produce a linear plan [4]. Anthony Barret and Daniel Weld have argued in 1993, that partial-order planning is superior to total-order planning, as it is faster and thus more efficient. They tested this theory using Korf's taxonomy of subgoal collections, in which they found that partial-order planning performs better because it produces more trivial serializability than total-order planning [5]. Examples of partial-order planners are NOAH planner (Sacerdoti, 1975, 1977), NONLIN system (Tate, 1975). One drawback of this type of planning system is that because the algorithm is more complex than others, it requires a lot more computational power for each node.

GRAPHPLAN: Partial-order planners dominated the next 20 years of research. Then Avrim Blum and Merrick Furst (1995, 1997) revitalized the field of planning with their GRAPHPLAN system [6] which was orders of magnitude faster than partial-order planners. Many other graph planning systems soon followed, such as IPP (1997), STAN and SGP (1998). GRAPHPLAN is a general-purpose planner for STRIPS-style domains, based on ideas used in graph algorithms. GRAPHPLAN uses a novel planning graph, to reduce the amount of search needed to find the solution from straightforward exploration of the state space graph. In this approach, both nodes are actions and atomic facts, arranged into alternate levels. This way, a planning graph makes explicit many of the constraints inherent in the problem to reduce the amount of search needed [6]. Also, Planning Graphs are constructed quickly: they have polynomial size and can be built in polynomial time. It's a propositional planner and uses no variables. GRAPHPLAN is still widely used for solving planning problems, though a closely related approach to planning is the Planning as Satisfiability, also called SATPLAN.

References

1. https://users.ics.aalto.fi/rintanen/planning.html
2. Fikes, Richard E., and Nils J. Nilsson. "STRIPS: A new approach to the application of theorem proving to problem solving." *Artificial intelligence* 2.3-4 (1971): 189-208.
3. http://www.primaryobjects.com/2015/11/06/artificial-intelligence-planning-with-strips-a-gentle-introduction/
4. http://artint.info/html/ArtInt_209.html
5. Barrett, Anthony, and Daniel S. Weld. "Partial-order planning: Evaluating possible efficiency gains." Artificial Intelligence 67.1 (1994): 71-112.
6. A. Blum and M. Furst. "Fast planning through planning graph analysis". Artificial intelligence. (1997) 90:281-300.