HISTORICAL DEVELOPMENTS IN AI PLANNING – A RESEARCH REVIEW

Planning is an integral part of AI that lies at the heart of several state-of-the-art technologies such as autonomous robots and self-driving cars. However, planning itself has been around for a much longer period of time. One of the biggest pursuits within the domain of AI has been to teach robots and intelligent agents to model an environment and search within it effectively with the use of better plans.

One of the first major planning developments was the conceptualization of STRIPS, a problem-solving program, the language of which uses a sequence of operators in a space of world models to transform a given initial world model into a model in which a given goal formula can be proven to be true. STRIPS requires the problem space to be defined by an initial world model. The input language consists of states, goals and a set of actions. If the current state of the world meets all the preconditions required to perform an action, it can be applied. Applying an action adds literals from the post-condition to the world state if they are positive and removes them from the same if they are negative.

However, STRIPS had several limitations when it came to handling realistic problems since it wasn't an Action Description Language. This is because STRIPS operators represented operations of world models, not on the states of a transitional system. This was overcome by the introduction of PDDL, a planning language definition that was designed to standardize AI planning languages. One of the key aspects of PDDL was to separate the descriptions of parameterized actions that characterized domain behaviors from the description of specific objects, initial conditions and goals that characterized a problem instance. It also fulfilled all the characteristics required for a language to be classified as an ADL by using a STRIPS-like formalism at its core that was extended, thereby enhancing its expressive power including the ability to express a type structure for the objects in a domain, typing parameters that appeared in actions and constraining the types of arguments to predicates. In addition, it allowed actions with negative preconditions and conditional effects as well as the use of quantification in expressing both pre- and post-conditions.

The development of these powerful planning representations was accompanied by the conceptualization of GRAPHPLAN, a general-purpose planner designed for STRIPS-style domains (which also works well with PDDL since it lies in the same domain). It is capable of always returning a shortest-possible partial-order plan, or determining that no valid plan exists. It uses Planning Graphs which encode the planning problem in such a way that many useful constraints inherent in the problem are explicitly available to reduce the amount of search needed overall. They can also be constructed relatively quickly, with polynomial time and space. The Planning Graph itself is not the state-space graph. The use of this compact structure enables searching for a plan using either progression or regression search while offering much higher efficiency than its predecessors including the total-order planner Prodigy, and the partial-order planner UCPOP. GRAPHPLAN utilizes an approach along the lines of dynamic programming problem solvers, rather than greedily searching through the entire search space. The Planning Graph uses its edges to represent relations between actions & propositions. If a valid plan exists in the STRIPS formulation, then that plan exists as a subgraph within the Planning Graph. It is also able to specify mutex relationships - wherein two actions are considered mutex if no valid plan could possibly contain both and two states are considered mutex if no valid plan could have both simultaneously true.

The developments mentioned in this paper are few of the advancements that led the domain of AI planning to where it is today. With the oncoming of machine learning and modern AI, it is exciting to see how Planning continues to evolve – utilizing new methods that couldn't have been anticipated a few years back. It continues to inspire and tackle some of the most complex problems that intelligent systems face today. While classical planning might be rendered obsolete soon, it will continue to be at the core of oncoming developments in this field and inspire better technologies in the coming future.

References

1. Blum, Avrim L., and Merrick L. Furst. "Fast Planning Through Planning Graph Analysis." Jan. 1995, doi:10.21236/ada303260.

2. Fikes, Richard E., and Nils J. Nilsson. "Strips: A New Approach to the Application of Theorem Proving to Problem Solving." Artificial Intelligence, vol. 2, no. 3-4, 1971, pp. 189–208., doi:10.1016/0004-3702(71)90010-5.

3. Russell, Stuart J., and Peter Norvig. *Artificial Intelligence: a Modern Approach*. Pearson, 2016.

4. Ghallab, Malik & Knoblock, Craig & Wilkins, David & Barrett, Anthony & Christianson, Dave & Friedman, Marc & Kwok, Chung & Golden, Keith & Penberthy, Scott & Smith, David & Sun, Ying & Weld, Daniel. (1998). "PDDL - The Planning Domain Definition Language."