Advancements in AI Planning & Search

The field of AI planning & search has seen many changes and developments over the past 50 or so years which have led us to where we are today with applications of this technology becoming so accurate and convenient that we are using them on a day to day basis. These applications range from consumer oriented such as Google Maps to more large scale use cases such as airplane route planning.

One of the more interesting projects was a robot that was being developed in the late 1960s and early 1970s by a group of researchers at the SRI AI laboratory. The robot, dubbed “Shakey the Robot”, was designed to be able to push objects around in a multi room environment. This meant that it needed to be able to efficiently navigate in this environment. This problem, which might be considered simple by today’s standards, provided the context and motivation for the development of the A* search algorithm, the STRIPS planning system, “triangle tables” for plan execution, and more which have all been used to develop more advanced systems and techniques.

STRIPS is often cited as providing a seminal framework for attacking the “classical planning problem” which involved finding a sequence of actions that will transform any given world state into any of a set of goal states in a static state world that can only be acted on by one agent.

Frustrated by the technical difficulties of describing in formal logic the effects of an action, and particularly the difficulty of specifying those aspects of a situation that are not affected by an action (i.e. the “frame problem”), the team at SRI began considering “ad hoc” representations for robot actions and algorithms for modeling their effects, while still maintaining their logic-based representation of individual states. These considerations produced the STRIPS operator representation and the algorithm for modeling the effects of an operator based on the “STRIPS assumption” that a plan operator affects only those aspects of the world explicitly mentioned in the operator’s deletions and additions lists. This assumption is still used today as is done so by us in the third project of the Nanodegree.

The scope of problems that could be solved with STRIPS was limited due to its narrow definition of a problem in which only one action could occur at any time, nothing changed except as a result of planned actions, and that actions were effectively instantaneous. Despite this and the vagueness of the STRIPS solution to the frame problem, the STRIPS representation and reasoning framework was used as the basis for most automatic planning research for many years. It had sufficient intuitive appeal to most researchers for them to believe that it was a viable foundation on which to develop techniques that would be effective in more realistic models.

The Planning Domain Definition Language was introduced as a computer-parsable, standardized syntax for representing planning problems. It was developed with hope that it will encourage empirical evaluation of planner performance, and that it will foster the development of standard sets of problems all in comparable notations. It has roughly the expressiveness of UMCP for actions, and roughly the expressiveness of ADL for propositions. ADL (Action Description Language) relaxed some of the restrictions of STRIPS and made it possible to encode more realistic problems. There have been several extensions to PDDL. It has been used as the standard language for the International Planning Competition since 1998.

A* came about as a result of attempts made by the team of researchers to improve the path planning done by the robot. Nils Nilsson developed an algorithm, which he called A1, that was faster than the then best known method, which was Dijkstra’s algorithm, for finding shortest paths in graphs. Bertram

Raphael suggested some significant improvements upon this algorithm, calling the revised version A2. Then Peter E. Hart introduced an argument that established A2, with only minor changes, to be the best possible algorithm for finding shortest paths. Hart, Nilsson and Raphael then jointly developed a proof that the revised A2 algorithm was optimal for finding shortest paths under certain well-defined conditions.

The basic principle of A* search was that at every step it determines which nodes to expand further based on a simple function which was the sum of the cost to get to the current node from the start and a heuristic that estimates the cost of the cheapest path from the current node to the goal. This way instead of expanding all nodes for no reason it prioritized expanding the nodes which brought it closer to the goal. This is what is known as an informed search algorithm.

A* search is still one of the most commonly used algorithms for the common pathfinding problem.

References:

1. **STRIPS:**
    - STRIPS, a retrospective (Richard E. Fikes and Nils J. Nilsson, 1993)
2. **The Planning Domain Definition Language:**
    - The Planning Domain Definition Language, version 1.2, (Malik Ghallab, 1998)
    - Artificial Intelligence, A Modern Approach, Third Edition (Stuart Russell and Peter Norvig)
3. **A*:**
    - A* Search Algorithm (Wikipedia)