AIND
Alexandru Oprisan

Important Developments in the History of AI Planning and Search

1. Logic Theorist

Logic Theorist is the name of o program created by Allen Newel, Herbert A. Simon and Cliff Shaw in 1955 -1956. It is regarded as being the first Artificial Intelligence program and has had a huge influence on the field of AI.

The goal of the program was to try to solve logic problems is a way similar to how humans used using reasoning. The program was able to prove 38 of 52 theorems in Principia Mathematica (Alfred North Whitehead and Bertrand Russell), some of the proofs are considered today more elegant than the ones known at the time.

Logic Theorist introduced a number of new concepts that later became fundamental to AI:

- reasoning as search (representing a problem as a search tree with the initial hypothesis as the root, and exploring the tree to find the solution of the problem as a series of deductions)
- heuristics (limiting the exponentially growing number of branches of the search tree based on some “smart”, simple roules)
- list processing (that later led to the development of Lisp)

2. A* Search Algorithm

The A* Search Algorithm is one of the most used search algorithms today for finding the shortest path in a graph. It was first introduced in 1968 by Peter Heart, Nils Nilsson and Bertram Raphael at Stanford Research Institute while working on Shakey The Robot’s navigation.

A* is an improvement of the earlier algorithm by Dijkstra for finding the shortest path in a graph. Just as Dijkstra’s, A* searches among all possible paths to the solution, but unlike Dijkstra’s, it does so in an “informed” manner, expanding at each node the branch considered fittest first. To do the “consideration” A* makes use of heuristics.

For each node explored A* considers the total accumulated cost (from the root to current node) ($g(n)$) and the heuristic estimated cost of reaching the solution from the current node ($h(n)$) and selects the path that minimizes $f(n) = g(n) + h(n)$.

The main reason A* is widely used is that it reduced the complexity of the search problem from exponential to polynomial when the search space is a tree with only one solution and the heuristic function meets certain conditions.

3. Plannig Domain Definition Language (PDDL)

PDDL is a language for representing AI planning domain problems. It was created be team led by Drew McDermott in 1998 and builds on elements from STRIPS and ADL planning languages. It was specifically created for the International Planning Competition (IPC) in 1998 and 2000 and has evolved with IPC ever since.

The latest version of PDDL is PDDL 3.1, released in 2011.

PDDL is regarded as a standard for representing planning problems and has made communication, research and reusing much easier in this domain.

PDDL introduces a separation between “domain description” and “problem description”, thus making it easier to represent more problems in the same domain.