Historical Developments in AI planning and Search

Planning is a branch of artificial intelligence that concerns the realization of strategies or action sequences to achieve the desired goals. It is an important activity for autonomous agents. Planning involves the representation of actions and world models, reasoning about the effects of actions, and techniques for efficiently searching the space of possible plans. It has been an area of research in artificial intelligence for over three decades.

Development 1: STRIPS (1971)

In 1971, Richard Fikes and Nils Nilsson at Stanford Research Institute developed a new approach to the application of theorem proving in problem solving. The model attempts to find a sequence of operators in a space of world models to transform the initial world model into a model in which the goal state exists. It attempts to model the world as a set of first-order predicate formulas and is designed to work with models consisting of many formulas.

It is a classical planning language is composed from states, goals and set of actions which specifies preconditions and postconditions. We can use progression search algorithms to form optimal plans for this example problem.

Development 2: Planning Graphs (1997)

In 1997, Avrium Blum and Merrick Furst at Carnegie Mellon developed a new approach to planning in STRIPS-like domains. It involved constructing and analyzing a brand-new object called a Planning Graph. They developed a routine called GraphPlan which obtains the solution to the planning problem using a Planning Graph construct.

Graphplan searches for a plan in two stages. The first stage is the construction of a data structure, the plan graph, that efficiently represents information about what the executive could possibly achieve by executing actions from the initial state. The second stage searches, backwards from the goals, for a sub-structure within the plan graph that represents a subset of actions that will achieve the goals.
The idea is that rather than greedily searching, we first create a Planning Graph object. The Planning Graph is useful because it inherently encodes useful constraints explicitly, thereby reducing the search overhead in the future.

Development 3: Heuristic Search Planner (HSP) (1998)

A very influential direction was initiated by work of McDermott in the planning system UNPOP [McDermott 1996] and Geffner and Bonet in HSP [Bonet et al. 1997]. The idea behind this work is to use a classic heuristic guided search. This is the search strategy in which the choice between alternatives is made by evaluating each alternative using a heuristic evaluation function and then selecting the best valued option.

The HSP algorithm instead estimates the optimal value of the relaxed problem. There are several ways in which the heuristic guidance can be applied, such as the well-known A* search, hill-climbing searches, best-first searches and so on.

Conclusion

Over the past decade several new techniques have emerged as promising for future development of classical planning towards application. In particular, these include STRIPS which serves as the base for most of the languages for expressing automated planning problem, planning graphs which has reduced the amount of search needed to find the solution from straight forward exploration of state space graph, heuristic forward search based on informative heuristics that can be automatically generated and that exploit the structure of the problem.