Three classical AI planning algorithms

Planning is a key ability for intelligent systems, increasing their autonomy and flexibility through the construction of sequences of actions to achieve their goals. Planning techniques have been applied in a variety of tasks including robotics, process planning, web-based information gathering, autonomous agents and spacecraft mission control. Algorithms are the core techniques behind many of the successful planners. Here I give a short summary of three very popular algorithms that used in planning.

The first is partial-order/non-linear plan algorithm (Ref1 and 2). A partial-order planner is an algorithm or program which will construct a partial-order plan and search for a solution. The input is the problem description, consisting of descriptions of the initial state, the goal and possible actions.
The problem can be interpreted as a search problem where the set of possible partial-order plans is the search space. The initial state would be the plan with the open preconditions equal to the goal conditions. The final state would be any plan with no open preconditions.
One drawback of this type of planning system is that it requires a lot more computational power for each node. This higher per-node cost occurs because the algorithm for partial-order planning is more complex than others. This has important artificial intelligence implications.

The second is GraphPlan algorithm (Ref 2 and 3). GraphPlan is one of the most efficient algorithms for solving the classical AI planning problems. GraphPlan is a general-purpose planner for STRIPS-style domains, based on ideas used in graph algorithms. Given a problem statement, GraphPlan explicitly constructs and annotates a compact structure called a Planning Graph, in which a plan is a kind of "flow" of truth-values through the graph. This graph has the property that useful information for constraining search can quickly be propagated through the graph as it is being built. GraphPlan then exploits this information in the search for a plan. GraphPlan was created by Avrim Blum and Merrick Furst, with subsequent extensions and improvements made by many researchers at many different institutions around the world. GraphPlan's algorithm is sound and complete: any plan the algorithm finds is a legal plan, and if there exists a legal plan then Graphplan will find one. One limitation of GraphPlan is that by guaranteeing to find the shortest possible plan, GraphPlan can make problems more difficult for itself.

The third is HSP algorithm (Ref 1, 3 and 4). HSP is based on the ideas of heuristic search. In HSP the search is assumed to be similar to the search in problems like the 8-Puzzle, the main difference being in the heuristic: while in problems like the 8-Puzzle the heuristic is normally given (e.g., as the sum of Manhattan distances), in planning it has to be extracted automatically from the declarative representation of the problem. HSP thus appeals to a simple scheme for computing the heuristic from Strips encodings and uses the heuristic to guide the search for goal.
Planners based on the ideas of heuristic search are related to specialized solvers such as those developed for domains like the 24-Puzzle, Rubik's Cube, and Sokoban but differ from them daily in the use of a general language for stating problems and a general mechanism for extracting heuristics. Heuristic search planners, like all planners, are general problem solvers in which the same code must be able to process problems from different domains. This generality comes normally at a price: the performance of the best current planners is still well behind the performance of specialized solvers.

There are some interesting comparisons of the various approaches to planning. But it is hard to say which one is the best, and of course the algorithms for AI planning will keep being one of the most important field to push AI move forward.

References

1. Chapter 11, Artificial Intelligence: A Modern Approach by Stuart Russell (2nd Edition), Peter Norvig
2. J. Rintanen and J. Hoffmann, KI, 2001, An overview of recent algorithms for AI planning
3. A. Blum and M. Furst, Fast Planning Through Planning Graph Analysis, Artificial Intelligence, 90:281--300 (1997).
4. B. Bonet, H. Geffner / Artificial Intelligence 129 (2001) 5–33