Research of historical developments in AI Search an Planning

Throughout the time there were a number of important developments in AI which led to a breakthrough of more intelligent systems and algorithms. In this review we took a look at few of these developments which helped to advance one of the core pieces of AI – planning.

Early planning systems were linear and they only considered ordered action sequences. HACKER is a linear planning system developed in 1973. The way that HACKER works is that it optimises heuristic early commitment and it guesses its way to a solution by debugging errors which were made while searching the path (Stefik M., 1980). So if there is a subgoal interaction problem, HAKER reorders the initial goals so that the solution could be eventually found. According to Stefik M. (1980), HACKER has quite a lot of downsides in terms of efficiency and performance - it's not complete and that solutions are not always optimal. In addition to that, the way HACKER works, makes it very inefficient as it takes time to research the paths which are wrong and then debugging in till the system finally reaches the right solution.

The issues regarding HACKER system were addressed when developing NOAH, partial-order planning system. According to Doyle P. (1997) the main advantage of NOAH is its hierarchical system, which can specify abstract plans for trained experts and give more details for a novice. As Stefik M. (1980) compared, in contrast to HACKER system, NOAH works in a way that it optimizes heuristic late commitment. Even though NOAH is superior to HACKER due to its nonlinear, hierarchical structure, it still has quite a lot of limitations. According to Wilkins D.E., (1988), NOAH could still produce incorrect plans due to its complexity. What is more, NOAH doesn't backtrack so it can only find a solution if it made a correct choice at the very beginning of the search (Wilkins D.E., 1988).

With time, state-space planning became the subject of research. Heuristic Search Planner (HSP), created by Bonet and Geffner, was the first application of state-space search in practice for large planning problems. As described by Bonet and Geffner (2001), to search the state space, HSP uses heuristic, extracted from problem encoding and combined with other search algorithms. HSP is designed to find the estimate of the distance to the goal by performing forward search from its starting state to the goal state. As long as HSP is combined with other search algorithms, it can be computationally expensive, unless we chose algorithms which are effective enough to find the best paths efficiently. According to Long D. (2000), this issue can be solved using hill climbing technique which usually allows to find optimal solution without high computational cost. HSP system was tested in AIPS-98 Planning Competition, hosted by the Artificial Intelligence Planning Systems in 1998. As a result, it was discovered that HSP is competitive with planners like Grphplan and SAT (Long D., 2000). There is still a lot of work being done to optimise HSP so that more search algorithms could be used alongside HSP.

References:

Stefik M. (1980). Planning and Meta – Planning (MOLGEN Part 2). *Artificial Intelligence an International Journal, pp. 161.*

Doyle P. (1997). *Planning. AI Qual Summary.*

Wilkins D.E. (1988). *Practical Planning: Extend the Classical AI Planning Paradigm, pp. 14.*

Bonet B., Geffer H. (2001). *Heuristic Search Planner 2.0*. *AI Magazine Volume 22 Number 3*, pp. 77-78.

Bonet B., Geffner H. (2000). *Planning as Heuristic Search, pp. 1.*

Long D. (2000). *The AIPS-98 Planning Competition. AI Magazine Volume 21 Number 2, pp. 22-23.*