(1997) Fast Planning Through Planning Graph Analysis

Available at https://www.cs.cmu.edu/~avrim/Papers/graphplan.pdf

For planing in STRIPS-Style problems we can, instead of begin the search immediately on the search tree we first build a data structure to represents the problem and their constraints, in such way that we reduce the search space and create a more efficient structure to search on: the Planning Graph.

Searching in graphs has the advantage that we can always find the shortest path in the search space. One disadvantage is that the graph size is polynomial, so depending on the size of the problem we can face some memory issues.

Another limitation of this method is that we can only use in STAMPS-Style domains, that means that actions can’t create new objects (so we need to rebuild the graph and the search) and the effect of actions must be determined statically.

(1994) UMCP: A Sound and Complete Procedure for Hierarchical Task-Network

Planning

Available at http://www.cs.umd.edu/~nau/papers/erol1994umcp.pdf

Hierarchical Task-Network Planning (HTN) is different than the STRIPS that we learn at the course as HTN represents the desireble change that we want in the modeled world, instead of a search to a unique goal, as in STRIPS.

Despite the previous works describing HTN planners before this paper any of them presented a structured a clear and concise algorithm to solve this problems. This clear and concise algorithm is present in this paper as follows:

HTN replaces STRIPS goals to tasks and task networks, with is more powerful, but increases the complexity. There are three types of tasks:

1. *Goal Tasks*, like the Goal in STRIPS, are properties in the model that we want to turn into true, e.g. At(Cargo1, JFK) as shown in the AIND-Planning project
2. *Primitive tasks*, are tasks that we can achieve by executing actions, like Fly() on the AIND-Planning project
3. *Compound tasks*, are tasks that cannot be described as a single task, but a network of *Goal Tasks* or *Primitive Tasks", e.g. “Deliver Cargo 1 at JFK” is the result of a execution of several Load/Unload/Fly/Goal tasks.

Tasks are connected through the Task Network, and a problem is solved following these steps:

1. Input the Planning Problem P (the task network)
2. If P contains only primitive tasks, then resolve the conflicts in P and return the result. If the conflicts cannot be resolved, return failure.
3. Choose a non-primitive task t in P
4. Choose an expansion for t
5. Replace t with the expansion
6. Use critics to find the interactions among the tasks in P, and suggest ways
   to handle them.
7. Apply one of the ways suggested in step 6.
8. Go to step 2.

The paper proposes the semantic and syntax for expressing HTN problems and demonstrates how implement basic expansions and reductions, as future work the researchers plan to study the application of HTN solvers in various domains.

(1971) STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving

Available at http://ai.stanford.edu/~nilsson/OnlinePubs-Nils/PublishedPapers/strips.pdf

Using first order calculus Fikes and Nilsson developed the STRIPS problem solver, that tries to find a finite sequence of actions in a modeled world to reach a pre-determined goal.

For any “world model” they assume that exists a finite sequence of actions that transforms the initial world in a new one with the Goal state is true.

On this paper the authors defined how to define the Problem Space (how to define States, Actions and other elements) and the Search Strategy (using tree search).

A plan for a STRIPS problem is a sequence of operators that transforms the initial world model in a new with the Goal State true.

This is the approach that we learnt at this module at AIND Nanodegree.