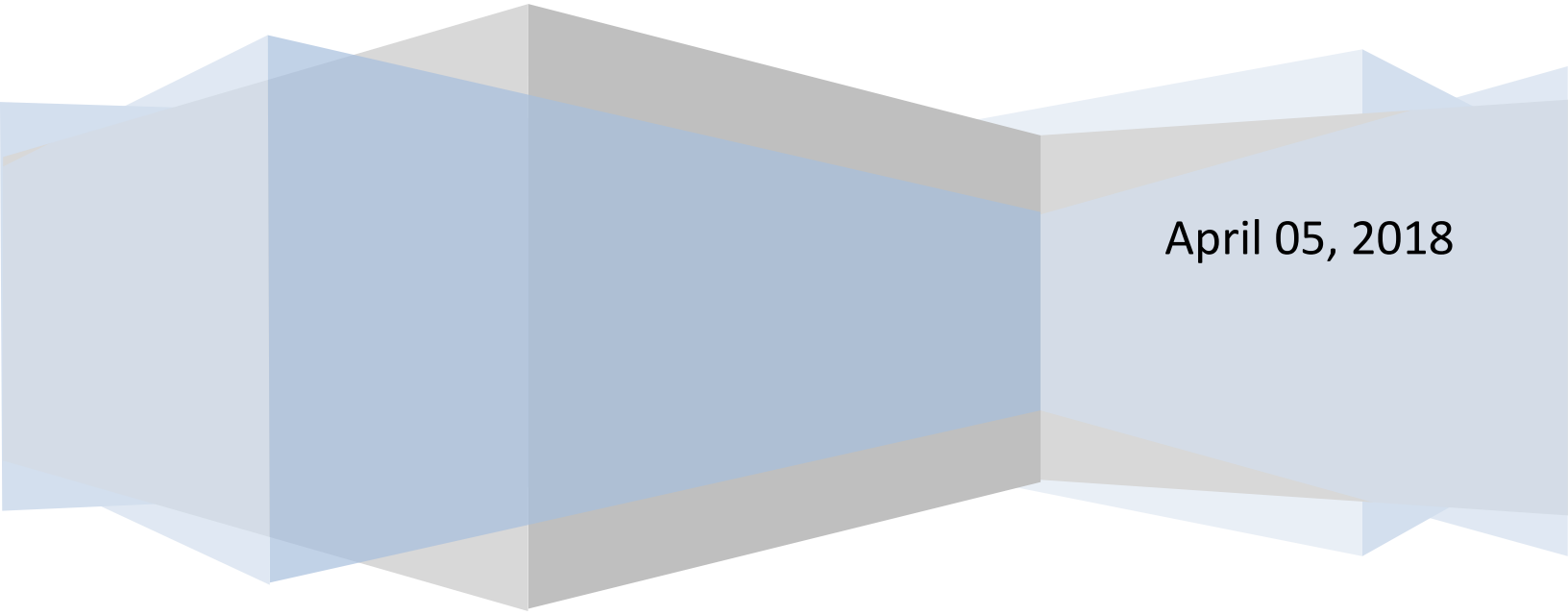
Udacity AIND December 2017 Cohort

Research Review

Developments in AI Planning and Search

Rohit Jain



April 05, 2018

STRIPS

STRIPS (Stanford Research Institute Problem Solver) attempts to find a sequence of operators in a space of world models to transform a given initial world model into a model in which a given goal formula can be proven to be true. STRIPS represent a world model as an arbitrary collection of first-order predicate calculus formulas and is designed to work with .models consisting of large numbers of formulas. It employs a resolution theorem prover to answer questions of particular models and uses means-ends analysis to guide it to the desired goal-satisfying model. A-star algorithm was used with STRIPS to navigate forward/backwards between initial state and goal state. STRIPS constructed a problem-solving tree whose nodes represent sub-problems. It searched the space of "world models" to find one in which a given goal is achieved. STRIPS separated entirely the processes of theorem proving from those of searching through a space of world models. This separation allowed employing separate strategies for these two activities and thereby improving the overall performance of the system.

Graph Plan

Graphplan is a planner which plans in STRIPS-like domains. The algorithm is based on a paradigm known as Planning Graph Analysis. In this approach, rather than immediately embarking upon a search as in standard planning methods, the algorithm instead begins by explicitly constructing a compact structure we call a Planning Graph. A Planning Graph encodes the planning problem in such a way that many useful constraints inherent in the problem become explicitly available to reduce the amount of search needed. Furthermore, Planning Graphs can be constructed quickly. Planning Graph Analysis appears to have significant practical value in solving planning problems even though the inherent complexity of STRIPS planning, which is at least PSPACE-hard is much greater than the complexity of standard Dynamic Programming problems. Planning Graph Analysis is able to provide a quite substantial improvement in running time. Graphplan always returns a shortest possible partial-order plan, or states that no valid plan exists.

Heuristic Search Planner

Heuristic search planners like HSP transform planning problems into problems of heuristic search by automatically extracting heuristics from STRIPS encodings. They differ from specialized problem solvers as they use a general declarative language for stating problems and a general mechanism for extracting heuristics from these representations and use the heuristic in the context of forward and backward state planners to guide the search for the goal. Heuristic search planners, like all planners, are *general problem solvers* in which the same code must be able to process problems from different domains. Heuristic search planners can be competitive with state-of-the-art Graphplan planners over many domains.

Bibliography

- Richard E. Fikes and Nils J. Nilsson. "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving". (Winter 1971)
- Avrim L Blum and Merrick L. Furst. "Fast Planning Through Planning Graph Analysis", Artificial Intelligence, 90:281--300 (1997).
- Blai Bonet and Héctor Geffner. "Planning as heuristic search". Artificial Intelligence 129 (2001)
- Stuart Russell and Peter Norvig. "Artificial Intelligence: A Modern Approach" – Third edition.