In completing this project, Udacity's instructors asked us to read about historical developments in the field of AI planning and search, and present our findings in a one-page report. To do so, I first checked Chapter 10 of the Norvig and Russell text as suggested, and pulled out STRIPS, GRAPHPLAN, and SATPlan to write about. All three of these algorithms are closely related. The first of these, STRIPS, is described as "the first major panning system," which we will be exploring for mainly historical reasons [1]. GRAPHPLAN and SATPlan are more recent developments and were written about as recently as 2001 [1].

STRIPS (STandford Research Institute Problem Solver) was developed in 1971 for the Shakey robot project around the idea of representing the world "by a set of well-formed formulas (wffs) of the first-order predicate calculus" [2]. STRIPS employed "a resolution theorem prover to answer questions of particular models and uses means-end analysis to guide it to the desired goal-satisfying model" [2]. STRIPS is composed of an initial state, a specification of goal states, and a set of actions with pre- and post-conditions in a manner very similar to the problems in the Domain-Independent Planner project for class [3]. STRIPS has been shown "to be PSPACE-complete" by Bylander in 1992 meaning that it can solve any decision problem that can be solved by a Turing machine in polynomial space. [2].

GRAPHPLAN was developed in 1997 as a general-purpose planner for "STRIPS-like domains" [4]. GRAPHPLAN works by constructing a "Planning Graph," which encodes the planning problem such that "constraints inherent in the problem become explicitly available to reduce the amount of search needed" [4]. GRAPHPLAN will find the shortest plan among those in which independent action may happen at the same time. In a Planning Graph, nodes are actions and atomic facts [5]. Edges are either from an atomic fact to its actions for which it is a condition, or from an action to the atomic facts it makes true or false [5]. GRAPHPLAN looks for valid plans using a "backward-chaining strategy" in which it (given a set of goals at time t) will look for a set of actions at t – 1 having the goals as an add effect and iteratively extending the graph until it either succeeds or has proven the goals unreachable [4].

SATPlan is a formal model of planning based on satisfiability rather than deduction [6]. This framework treats a planning problem not as a theorem to be proved, but rather as a set of axioms with the property that any model of the axioms corresponds to a valid plan [6]. This is advantageous as it is easy to specify conditions in any intermediate state, not just the initial and goal states [6]. Its authors also argue that the satisfiability approach is both more flexible, and more accurate in stating different kinds of constraints on plans than the deductive approach.

References

[1] Norvig and Russell. *Artificial Intelligence: A Modern Approach*, 2010, pp. 393 – 396.

[2] Fikes and Nilsson. *STRIPS: A New Approach to The Application of Theorem Proving to Problem Solving*, 1971. http://ai.stanford.edu/~nilsson/OnlinePubs-Nils/PublishedPapers/strips.pdf

[3] https://en.wikipedia.org/wiki/STRIPS

[4] Blum and Furst. *Fast Planning Through Graph Analysis*, 1997. https://www.cs.cmu.edu/~avrim/Papers/graphplan.pdf

[5] https://en.wikipedia.org/wiki/Graphplan

[6] Kautz and Selman. *Planning as Satisfiability*, 1992. http://www.cs.cornell.edu/selman/papers/pdf/92.ecai.satplan.pdf