Research Review - AI Planning and Search

STRIPS (Fikes and Nilsson - 1971)

STRIPS (Stanford Research Institute Problem Solver) was the first prominent planning system. It was originally designed as part of SRI AI Labs efforts to build a mobile robot to navigate and push objects around a room. The key outcome of this effort was the framework and notation for planning it provided. Like future advances, STRIPS formally defined a problem space, including an initial state, goals, and the set of actions that can be undertaken – with preconditions and results. The planning problem was broken down to a search through these actions to the goals. It was much more simple than future planning systems, specifically it was a completely controlled environment (no external actions could change the state), only one agent could act at a time, and all actions were instantaneous. Despite this, it created the first lingua franca for automated planning. This formalization of planning logic became the basis for future developments including Action Description Language (ADL), created by Pednault in 1986. Eventually, this led to the development of Problem Domain Descriptive Language (PDDL) published in 1998 by Ghallab et al, which we used in this projects. Many advances in computer science and mathematics are seen after innovation product a descriptive and interpretable notation that enables downstream research. For automated planning, STRIPS was the first.

GRAPHPlan (Blum and Furst - 1997)

One of the core weaknesses of automated planning for the 25 years after STRIPS was published was the inefficiency of solving problems with a state-space graph in which a plan is a path through the graph. GRAPHPlan was a response to this inefficiency. The GRAPHPlan planner uses the Planning Graph that it creates to search in a guided way. In some senses, it is a far simpler method than others, because it includes all possible literals that could be true at each state level. While this seems like a lot of extra information, it reduces the complexity of processing each level dramatically. The result is that GRAPHPlan is guaranteed to find the shortest path, and does so significantly more efficiently than prior methods. Finally, it provides a rich framework for additional extensions to improve performance as well. GRAPHPlan (and the related SATPlan) perform particularly well in NP-hard domains with limited numbers of objects.

Decision Diagrams

In recent years, significant research has focused on using Decision Diagrams (DDs) for automated planning. This line of research has produced substantial efficiencies in search and improvements in planning. This becomes easier because DDs have improved space and time considerations. Binary Decision Trees (BDDs) provide a simpler representation of space and enable binary operations, making it simpler and faster to evaluate each state. In addition to BDDs, there is more research in other Decision Diagrams today, including multi-valued decision diagrams (e.g K-options per level), and multi-terminal BDDs (which combine multiple goal-states together). I read Scott Sanner’s “Decision Diagram in Discrete and Continuous Planning” written in 2012, to understand this.