Logic and Planning: Research Review

John Carpenter, PEng.

I. Introduction

Logic and planning within AI is the process of determining a course of action given some constraints and requisites. While the concepts of logic and planning existed well before 1970s, there was a concerted effort and that time to arrive at a language to adaquately describe and model different scenarios. The languages had to be flexible enough to describe a variety of situations yet still maintain a structure that can be coded within a computer model. The following articles describe some of the early work that went into determining those languages.

II. STRIPS

In 1971,Richard Fikes and Nils Nilsson[1] began to describe a language and process to handle automatic plan generation. This language, STRIPS became the cornerstone for generalizing the "planning" problem in algorithmic terms that are easily manipulated and understood by researchers and machines. What was novel about the STRIPS algorithm was the inclusion of both addition and deletion steps for each operator. With those additional steps some algorithmic simplifying can occur[2]. This reduced the steps in goal finding and made it easier to build plans for much more complex interactions.

But the STRIPS framework was not without problems. It was limited to a single action executing at one time and the complexity of problems was limited[2]. Even with those shortcomings the STRIPS framework became an intuitive way to handle automatic planning in almost any scenario. Quite a few changes to the framework have occurred over the years but the language introduced by Fikes and Nilsson was a viable foundation to continue research in the field.

III. Fast Planning Through Planning Graph Analysis

One attempt to reduce the complexity of the planning problem was proposed by Avrim Blum and Merrick Furst[4]. Their Graphplan introduced a concept to handle complex planning operations that were overly complex for the STRIPS framework. The approach centered around two(2) concepts. First, the state changes in the STRIPS framework were exchanged for a time-series approach. STRIPS framework assumes actions take place instantaneously and independent of other actions[2]. Graphplan creates a flow through time presenting a series of actions and dependencies. This flow provides significant opportunities to reduce the search space, similar to tree pruning.[4] Secondly, the approach begins by creating a Planning Graph. This planning graph provides a compact view of the whole problem and exposes some constraints that can be simplified before searching[4].

Graphplan brought together many different methods STRIPS, partial-order planning and graph searches into a single process. This introduced many key advantages to planning that weren't previously avaible. This process was shown to find the shortest path to a goal if a solution exists[4] which can be valuable in planning a problem but also in reducing the state actions and features to only those sets that are required for the solution. Also, given the flow nature of the actions it is possible to pass information dynamically within the graph. This can provide a "learning" algorithm allowing faster paths to the solution.

IV. Stochastic Search

The Graphplan algorithm made significant contributions in simplifying the search and goal finding for a large number of planning problems. But inherent to the design the "planning graph" was slow at solving arbitrarily large and complex problems. Problems where the search space is combinatorial (traveling salesman problem) Graphplan doesn't benefit from reducing the problem size as in other types of problems. Kautz and Selman[3] discuss a number of different stochastic methods that can be used that make the problems much more efficient to solve. Using a stochastic methods such as WalkSAT[4], Selman showed that they could produce orders of magnitude performance improvement over traditional systematic approaches in Graphplan. While this wasn't the first time stochastic approaches were used for solving problems they were assumed to be unstable or unsuited for planning problems. The approach Selman et al, took in their paper[3] was to convert the Graphplan into a series of clauses which can then be fed into the appropriate stochastic model, WalkSAT in this example.

The results were that the stochastic approach was able to drastically reduce the solution time compared to a Graphplan. However, while the stochastic approach was computationally more effective it isn't always guaranteed to arrive at the optimal solution. Selman proposes that more work can be achieved by combining the strengths of both approaches for solving a general problem.

V. Conclusions

This paper touches on the historical approaches to general problem solving and presents some of the core frameworks and thinking behind the research. From STRIPS to the Graphplan and further it can be shown that using a general framework you can reduce complicated problems into much simpler components.

References

[1] Richard E. Fikes and Nils J. Nilsson. 1971. *STRIPS: a new approach to the application of theorem proving to problem solving.* In Proceedings of the 2nd international joint conference on Artificial intelligence (IJCAI'71). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 608-620.

[2] Richard E. Fikes, Nils J. Nilsson 1995. *STRIPS, a retrospective*. Computer Science Department, Stanford University, Stanford, CA

[3] H. Kautz, B. Selman 1996. *Pushing the Envelope: Planning: Propositional Logic, and Stochastic Search*. In Proceedings of AAAI-96, Portland, OR.

[4] Avrim L. Blum and Merrick L. Furst 1995. *Fast Planning Through Planning Graph Analysis*. ARTIFICIAL INTELLIGENCE 1 Vol 90, pp 1636-1642

[4] B. Selman, H. Kautz, B. Cohen 1994. *Noise Strategies for Local Search*. Proceedings AAAI-94, Seattle, WA, pp 337-343