AIND Research Review: History of Planning Search

Stefan Heidekrüger

August 13, 2017

In this document we'll explore some of the milestones in the domain of Planning Search as well as the general evolution of the field.

The beginning: STRIPS

In a retroperspective relased some 25 years after the fact, Fikes and Nilsson (1993)—the original authors of the STRIPS sytem—lay out inception of the field during the 1960s and 1970s. They not that a series of robotics experiments sparked the development of many of the early planning and searching tools, including the A^* algorithm and STRIPS, which the authors describe as the *"seminal framework for attacking the 'classical planning problem'"*, i.e. involving state-space search or modeling the entire system by a well defined set of possible states and actions. STRIPS first introduced the notion of state space modeled planning that we used in class, also called the "STRIPS-assumption":

- State is represented by a set of *sentences* that are true about the system
- Each action profile is reperesented by sets of *preconditions*, *deletions* and *additions* of sentences.
- An action can only be executed if its preconditions are fulfilled. In that case, the action's additions will be added to the system state and its deletions will be removed.

Planning was then performed using search in this state space, using heuristic algorithms such as A^*. In its applications, STRIPS also employed an *execution monitor* system: Often the actions and state-model coudln't adequately represent reality: The resulting state after an action didn't correspond to reality. In this case the system would infer the real state from sensors and replan.

STRIPS mainly influenced the field by its modeling language, which has become a staple of planning research (Russell and Norvig, 2009, Chapter 10.6).

Beyond State Space Search

Starting in the late 70s, direct state space search started to fall out of fashion, with more attention being given to approaches such as *linear planning*, *interleaving* and graph-based models such as *task networks* (Russell and Norvig, 2009, Chapter 10.3). Weld (1999) gives a write up of this period in AI planning, which can be exemplified by the GRAPHPLAN system, an approach very similar to that of the Planning Graph seen in the AIND lectures. As opposed to direct State Space search, a planning graph is built up level-by-level before the actual search, alternating between nodes of literals (describing the possible state at that level) and possible *action nodes*. Of utmost importance for the efficiency of these methods are the inclusions of *mutex* relations, which specify that at each level, several actions or literals will be mutually exclusive. This in turn significantly reduces the dimensionality of the search space and leads to performance improvements.

Another approach popular in this period is reformulating the planning problem as a conjunctive normal state SAT formula and then using SAT solvers to come up with a plan. (Weld, 1999, p. 111f)

Resurgence of State Space Search

Since the late 1990s, there has been a resurgence in state space search algorithms, fueled by the first use of the "ignore-delete-list" heuristic. (Russell and Norvig, 2009, Ch. 10.3). This heuristic, that considers distances to the goal by ignoring "negative" effects of actions proved superior to those heuristics used in previous decades and lead to state space solving algorithms that were much faster than earlier versions. At the same time, there was more research being conducted on the theoretical properties and complexities of several forms of planning problems, many of which were proven to be NP-hard. Hoffmann (2005) conducts an in-depth survey showing that depending on domain and thus state-space and planning-graph topology, either graph-planning or state-space search approaches can be seen as superior.

Current Research

As in many other fields of AI, there have been recent advances and new approaches made possible by the introduction of Deep Learning. DeepMind (2017) introduces two recent papers that use Neural Networks to give a planning agent ”imagination” for its internal model representation: In one of these papers, Pascanu et al. (2017), the authors in a sense consider planning as a bandit-problem with a trade-off between exploration and exploitation: The specification of the state-space and results of possible actions are not explicitly given to the agent. Instead, these are inferred through executing actions and observing their results. The ”memory” built in this way can then be used to construct an internal state space model of the agent, which in turn the agent can use to ”imagine” the results of possible future actions and plan ahead. In the paper, the agent has been trained using reinforcement learning on an arcade-style video game where the agent is tasked with piloting a space ship to a certain destination, having to use special actions such as weapons enroute.

References

DeepMind. Agents that imagine and plan, `https://deepmind.com/blog/agents-imagine-and-plan/`. *Blogpost*, (July 20), 2017.

R. E. Fikes and N. J. Nilsson. Strips, a retrospective. *Artificial Intelligence*, 59(1):227–232, 1993.

J. Hoffmann. Where’ignoring delete lists’ works: Local search topology in planning benchmarks. *Journal of Artificial Intelligence Research*, 24:685–758, 2005.

R. Pascanu, Y. Li, O. Vinyals, N. Heess, L. Buesing, S. Racanière, D. Reichert, T. Weber, D. Wierstra, and P. Battaglia. Learning model-based planning from scratch. *arXiv preprint arXiv:1707.06170*, 2017.

S. J. Russell and P. Norvig. Artificial intelligence: a modern approach (3rd edition), 2009.

T. Weber, S. Racanière, D. P. Reichert, L. Buesing, A. Guez, D. J. Rezende, A. P. Badia, O. Vinyals, N. Heess, Y. Li, et al. Imagination-augmented agents for deep reinforcement learning. *arXiv preprint arXiv:1707.06203*, 2017.

D. S. Weld. Recent advances in ai planning. *AI magazine*, 20(2):93, 1999.