AIND Planning Project: Research review

Jaco Fourie

August 2017

1 The STRIPS problem solver

The STRIPS problem solver was developed at Stanford Research Centre as a tool for efficiently solving a class of problems that can be described using first-order predicate calculus. It was first described in a paper by Fikes and Nilsson at the International Joint Conference for Artificial Intelligence in 1971 [2]. STRIPS was also later used to refer to the language used to describe inputs to the STRIPS problem solver as it became the basis for many other automated planning systems.

A problem is described in STRIPS as a quadruple $< P, O, I, G >$ where P is a set of conditions, O is a set of operators, I describes the initial condition at the start of the problem, and G describes the goal state. In the description of both the goal and the initial state it is important to note that the *closed world assumption* holds. In other words any unmentioned literals are assumed to be false rather than unknown.

2 The Action Description Language (ADL)

With the introduction of the STRIPS problem solver and language researchers had a tool to build further research on and this also led to a better understanding of the limitations of STRIPS and the need for a more flexible alternative. The Action description language (ADL) was introduced by Edwin Pednault while working at the Data Abstraction Research Group for IBM. He published his description of the system at the Reasoning About Actions and Plans workshop in 1987 [6].

There are several notable differences between STRIPS and ADL. Firstly, the *closed world assumption* that was mentioned earlier does not hold in ADL. Therefore no assumption is made regarding unmentioned literals and they are considered to be unknown. In ADL one can also describe the goal state using more than just ground literals but can include quantified variables to describe a goal (e.g. $\exists x\ At(C1, x) \wedge At(C2, x)$ means that cargo C1 and C2 are at the same location). These improvements allowed researchers to describe more complex problems in ADL that were too difficult to describe in STRIPS. It is possible to translate a problem defined in ADL to STRIPS but this is a worst-case

exponential problem and therefore may be too computationally expensive to do practically.

3 The GRAPHPLAN Algorithm

Graphplan is an automated planning algorithm designed for problems that can be described in STRIPS or STRIPS-like languages. It was introduced by Blum and Furst in 1995 and published in the journal *Artificial Intelligence* [1]. Graphplan finds the shortest possible plan to reach the goal state if such a path exists or states that none exist. Unlike the partial-order planners of that time that searched through the state space using a state-space graph, Graphplan first constructs a *planning graph* which is both smaller and also encodes many useful constraints that are inherent in the problem reducing the amount of search required.

The idea of using planning graph instead of searching through the state-space graph led to many further developments like the IPP [5], STAN [3], and LPG [4] algorithms.

References

[1] Avrim L. Blum and Merrick L. Furst. Fast planning through planning graph analysis. *Artificial Intelligence*, 90(1):281 – 300, 1997.

[2] Richard E. Fikes and Nils J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3):189 – 208, 1971.

[3] M. Fox and D. Long. The automatic inference of state invariants in tim. *Journal of Artificial Intelligence Research*, 9:367–421, 1998.

[4] Alfonso Gerevini, Alessandro Saetti, and Ivan Serina. Planning with numerical expressions in lpg. In *Proceedings of the 16th European Conference on Artificial Intelligence*, ECAI'04, pages 667–671, Amsterdam, The Netherlands, The Netherlands, 2004. IOS Press.

[5] J. Koehler, B. Nebel, J. Hoffmann, and Y. Dimopoulos. Extending planning graphs to an ADL subset. In *Proceedings of the Fourth European Conference in Planning*. Morgan Kaufmann Publishers, 1997.

[6] Edwin P. D. Pednault. Formulating Multi-Agent Dynamic-World Problems in the Classical Planning Framework. In Michael P. Georgeff and Amy L. Lansky, editors, *Reasoning About Actions and Plans: Proceedings of the 1986 Workshop*, pages 47–82, San Mateo, CA, 1987. Morgan Kaufmann Publishers.