AIND - Research Review

Robson Cardoso

April 09, 2017

This quick report tries to connect the development of three well-known formal languages used for planning systems which were very important to the development of Artificial Intelligence.

According to Norvig and Russel (2009), *Stanford Research Institute Problem Solver*, STRIPS, was the first major planning system. It was an automated planner and a formal language. "The representation language used by STRIPS has been far more influential than its algorithmic approach" (Norvig and Russel, 2009) and it is the base for most languages for expressing automated planning problem instances in use today.

According to Fikes and Nilsson (1971), "STRIPS is a member of the class of problem solvers that search a space of "word models" to find one in which a given goal is achieved". STRIPS separates the processes of theorem proving from those of searching through a space of world models, allowing the representation of objects (world models) much more complex and general than any of those used in GPS, the General Problem Solver (Newell and Simon, 1961), and provides more powerful search heuristics than those found in theorem-proving programs. The problem space of STRIPS is defined by initial world model, the set of available operators and their effects on world models, and the goal statement (Fikes and Nilsson, 1971).

As an extension of STRIPS, the *Action Description Language* (Pednault, 1986), ADL, relaxed some of the STRIPS restrictions and made it possible to encode more realistic problems (Norvig and Russel, 2009). Proposed by Pednault, ADL is an automated planning and scheduling system which allows the effects of an operator to be a conditional. Contrary to STRIPS, ADL applies the principle of the open world — which states that everything not occurring in the conditions is unknown instead of being assumed false — and allows negative literals and disjunctions.

Inspired by STRIPS and ADL, the *Planning Domain Description Language*, PDDL, was released in 1998 by Drew McDermott aiming to become a community standard for the representation and exchange of planning domain models (Fox and Long, 2003). According to Norvig and Russel (2009), it has been used as the standard language for the International Planning Competition since 1998. Although the core of PDDL is a STRIPS formalism, the language incorporates extensions beyond that. Fox and Long (2003) says that these extensions include the ability to express a type structure of the objects in a domain, typing the parameters that appear in actions and constraining the types of arguments to predicates, actions with negative preconditions and conditional effects, and the use of quantification in expressing both pre- and post-conditions. According to Fox and Long (2003), these extensions are essentially those proposed by ADL.

Finally, PDDL has enabled considerable progress to be made in planning research because of the ease with which systems sharing the standard can be compared and the enormous increase in availability of shared planning resources (Fox and Long, 2003). It provides the foundation on which an expressive standard can be constructed, enabling the domain models of the applications-driven community to be shared and motivating the development of the planning field towards realistic applications.

REFERENCES

Fikes, R. E., Hart, P. E., and Nilsson, N. J. (1972). Learning and executing generalized robot plans. AIJ, 3(4), 251–288.

Fox, M., Long, D. (2003). PDDL2.1: An extension to PDDL for expressing temporal planning domains. Journal of Artificial Intelligence Research, 20, 61–124.

Newell, A. and Simon, H. A. (1961). GPS, a pro- gram that simulates human thought. In Billing, H. (Ed.), Lernende Automaten, pp. 109–124. R. Olden- bourg.

Norvig, P., Russell, S., & (2003). Artificial Intelligence — A Modern Approach. Prentice Hall

Pednault, E. P. D. (1986). Formulating multiagent, dynamic-world problems in the classical planning framework. In Reasoning about Actions and Plans: Proc. 1986 Workshop, pp. 47–82