Research Review

Nhat Bui - AIND Project 3

Sypnosis

This report goes over three major developments in the field of AI planning and search and discuss their impact and relative importance to progress of the field as a whole.

Discussion

The first foundational development was the introduction of the Stanford Research Institute Problem Solver, or STRIPS (Fikes and Nilsson, 1971) that formed the backbone of all later frameworks for planning problem formulation. STRIPS was an automated planner that served as the planning component for the Shakey robot developed in the late 1960s who could navigate and satisfy goals in a physical but artificial environment (Russell and Norvig, 2009). STRIPS brought together different aspects of AI research at the time, namely resolution theorem proving and heuristic state space search (Fikes and Nilsson, 1993). However, it was the language STRIPS used to encode problem states and its transitions effected by the planning agent, rather than its algorithmic approach, that remains its most influential contribution. The language, being the first of its kind, lacked rigor in describing logical effects of actions, but nevertheless became the base from which all later refinements and extensions were adapted to solve more complex and realistic problems. Most notable examples of STRIPS successors include the Action Description Language, or ADL (Pednault, 1986), which set out to improve STRIPS expressiveness and make it applicable to more realistic scenarios, and the Planning Domain Description Language, or PDDL (Ghallab et al., 1998) which was an attempt to unify planning domain description languages under a well defined and understood formal framework.

The next development that had a long lasting effect on the field was the advent of Partial Order Planning, or POP. POP arose to address a shortcoming of the most popular method for planners then: Linear Planning, in which a plan is concatenated from plans that achieve individual subgoals, without allowing steps in the sub plans to interleave (Sacerdoti, 1975). STRIPS is an example of this type of planner. Unfortunately, this strategy had some serious flaws as it was not able to solve some very simple problems, such as the Sussman anomaly, where each subgoal cannot be achieved without undoing another using linear planning (Sussman, 1975). POP, pioneered by the NOAH planner (Sacerdoti, 1975, 1977) and the NONLIN planner (Tate, 1975b, 1997), avoids this by leaving the order of the steps as open as possible and only commits to an order in the form of a temporal constraint (one step must always come before another) when absolutely necessary i.e there is an order actions that will result in a conflict as one action deletes another action’s precondition. Thanks to this behaviour POP can detect conflicts and rearrange steps as needed to create a merged plan that satisfy all subgoals, making POP a complete planner that successfully avoids Sussman anomaly. POP since its introduction has remained the most popular paradigm in the planning literature for 20 years until the resurgence of state-space search total order planner starting with UnPOP (McDermott, 1996), Heuristic Search Planner, or HSP (Bonet and Geffner, 1999) and their subsequent improvements.

A more recent development that had a big impact on the field was the GRAPHPLAN algorithm which introduced the planning graph, a data structure that can be constructed quickly from descriptions of a problem domain. GRAPHPLAN extracts a plan directly from the planning graph, achieving orders of magnitude performance gain over partial order planners of the time. However, the planning graph itself is versatile and can be adapted to work with different systems, lending vast improvements to other approaches. Examples of systems that employ the planning graph are IPP (Koehler et al., 1997), STAN (Fox and Long, 1998), SGP (Weld et al., 1998) and LPG (Gerevini and Serina, 2002, 2003) - winner of the 2002 AIPS planning competition.

Bibliography

Bonet, B. and Geffner, H. (1999). Planning as heuristic search: New results. In ECP-99, pp. 360–372.

Fikes, R. E. and Nilsson, N. J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. AIJ, 2(3–4), 189–208.

Fikes, R. E. and Nilsson, N. J. (1993). STRIPS, a retrospective. AIJ, 59(1–2), 227–232.

Fox, M. S. and Long, D. (1998). The automatic inference of state invariants in TIM. JAIR, 9, 367–421.

Gerevini, A. and Serina, I. (2002). LPG: A planner based on planning graphs with action costs. In ICAPS-02, pp. 281–290.

Gerevini, A. and Serina, I. (2003). Planning as propositional CSP: from walksat to local search for action graphs. Constraints, 8, 389–413.

Ghallab, M., Howe, A., Knoblock, C. A., and Mcdermott, D. (1998). PDDL—The planning domain definition language. Tech. rep. DCS TR-1165, Yale Center for Computational Vision and Control.

Koehler, J., Nebel, B., Hoffmann, J., and Dimopoulos, Y. (1997). Extending planning graphs to an ADL subset. In ECP-97, pp. 273–285.

McDermott, D. (1996). A heuristic estimator for means-ends analysis in planning. In ICAPS-96, pp. 142–149.

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.

Russell, S. J. and Norvig, P. (2009). Classical planning. In Artificial intelligence: a modern approach 3rd edition, pp. 393-396.

Sacerdoti, E. D. (1975). The nonlinear nature of plans. In IJCAI-75, pp. 206–214.

Sacerdoti, E. D. (1977). A Structure for Plans and Behavior. Elsevier/North-Holland.

Sussman, G. J. (1975). A Computer Model of Skill Acquisition. Elsevier/North-Holland.

Tate, A. (1975b). Using Goal Structure to Direct Search in a Problem Solver. Ph.D. thesis, University of Edinburgh.

Tate, A. (1977). Generating project networks. In IJCAI-77, pp. 888–893.

Weld, D. S., Anderson, C. R., and Smith, D. E. (1998). Extending graphplan to handle uncertainty and sensing actions. In AAAI-98, pp. 897–904.