P3 Research Review
by Levan Iordanishvili

In the field of AI planning and search a number of milestone projects have built on the successes of their predecessors. STRIPS was the “first major planning system” (Russell & Norvig, 2016, p. 400), a problem solver which begins at some initial state and with *operators* “transforms the world model” to another state, continuing to search until a goal state is found, until a world model “satisfies some stated goal condition” (Fikes & Nilsson, 1971, p. 190).

STRIPS was developed with the primary goal of helping robots “re-arrange objects and navigate” (Fikes & Nilsson, 1971, p. 190), a task which required more complex representations of the world than the more simple “games” previous algorithms had been working on solving.

The STRIPS representation language provided a way of defining the initial state with a set of “available operators,” “their effects on the world models” and the goal state (Fikes & Nilsson, 1971, p. 191). The operators were defined in an operator schema, providing the operator’s effects and the “conditions under which the operator is applicable” (Fikes & Nilsson, 1971, p. 192). The most influential portion of STRIPS was not the algorithm itself but the representation language (Russell & Norvig, 2016, p. 400).

STRIPS-like representation was later used by Avrim Blum and Merrick Furst in their Graphplan algorithm, a system which performed well and “revitalized the field of planning” (Russell & Norvig, 2016, p. 401). The algorithm does an extensive analysis of a given state rather than “immediately embarking upon a search” (Blum & Furst, 1997, p. 1). It creates a Planning Graph, a structure which encodes a useful information about the state.

The useful state information is then used to reduce the searching by only exploring more promising paths. The Planning Graph construction process identifies “mutual exclusion relationships” that “can be of enormous help in reducing the search for a subgraph of a Planning Graph that might correspond to a valid plan” (Blum & Furst, 1997, p. 5). The authors of the paper cite the mutual exclusion information as the factor that accounts for most of the algorithm’s efficiency. (Blum & Furst, 1997, p. 15).

In 1999 Peter van Beek and Xinguang Chen took on Graphplan inspired algorithms such as Blackbox with CPlan by applying Constraint Satisfaction to planning and presented their algorithm’s advantage in “terms of time and space efficiency over the current state-of-the-art planners.” (Van Beek & Chen, 1999, p. 1)

They cite the need to develop a “robust CSP model” for each new domain as a disadvantage to their approach: each new domain requires a model. That disadvantage is alleviated by the fact that once the model is built, existing constraint solvers can be used. Therefore, the algorithm itself needs less work (Van Beek & Chen, 1999, p. 1). Using constraints can also encode types of information about the world that would be easy to encode in CSP but “difficult for traditional planners” such as “capacity restrictions” (e.g. amount of fuel available) (Van Beek & Chen, 1999, p. 3).

Experiments showed that CPlan can be “one to two orders of magnitude more efficient in both time and space on problems which the other systems can solve and can scale to harder problems which the other systems cannot solve” (Van Beek & Chen, 1999, p. 6).

References:

Blum, A. L., & Furst, M. L. (1997). Fast planning through planning graph analysis. *Artificial Intelligence*, *90*(1-2), 281-300. doi:10.1016/s0004-3702(96)00047-1

Fikes, R. E., & Nilsson, N. J. (1971). Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, *2*(3-4), 189-208. doi:10.1016/0004-3702(71)90010-5

Russell, S. J., & Norvig, P. (2016). *Artificial intelligence: A modern approach*. Boston: Pearson.

Van Beek, P., & Chen, X. (1999). CPlan: A Constraint Programming Approach to Planning. Retrieved from http://aaai.org/Papers/AAAI/1999/AAAI99-083.pdf