Research summary on AI planning and search

AI planning is a branch of artificial intelligence that deals with the formation of strategies and actions for solving a certain problem in a certain sequence, or achieving a goal, such that it can be executed by an intelligent agent [2]. Given an initial state of a world, a description of the desired goal, along with possible actions, AI planning is the process through which a plan is generated to get from the initial state to the goal state within boundaries of optimality, space and time efficiency. These techniques and strategies are prevalent in a wide variety of fields and applications in logistics, workflow management, robot task planning.

AI planning require a means of modeling a problem, a set of actions, an initial and goal state in a way that can be applied to a wide range of problem domains. These are known as domain independent planners and can be used to solve a wide ranging set of problems from different domains [2]. Specific languages have been developed to model the planning domains and specific planning problems, and that is the focus of this summary. Three of such modeling languages are STRIPS, ADL and PDDL.

STRIPS, an automated planner built by SRI International in 1971, is the forerunner of modern action languages that are used for expressing automated planning problems [3]. STRIPS consists of an initial state, a description of the goal state that let's the planner know when the search is complete, and a set of actions with pre conditions and post conditions for the actions that tells the planner how to move forward. A plan then, as discovered by the automated planner, is a sequence of actions that transitions from the initial state, through a sequence of intermediary states, eventually to the goal state. In STRIPS, the state variables have the domain {0, 1} and an action consists of three sets of state variables: the PRECONDITION, ADD list, DELETE list. An action is possible in a state where all the variables in the PRECONDITION are 1. A goal is expressed as a set of state variables and the goal is achieved when all the goal state variables have the value of 1 [1].

ADL, is another automated planning and scheduling system used particularly with robots [5]. It is seen as an advancement of STRIPS and was built by Edwin Pednault in 1987. The inadequacies resulting from deficiencies in expressiveness of STRIPS motivated the development of ADL. The expressiveness of ADL and its computation efficiency lies somewhere between STRIPS and Situation Calculus and supports both positive and negative literals in its expressions and unmentioned literals are considered unknown as opposed to false in the case of STRIPS. These are but a couple of differences between STRIPS and ADL.

PDDL was an attempt to standardize action languages, and was developed by Drew McDermott in 1998 [4]. It was inspired by STRIPS, ADL and other action languages at the time and have been actively used in the International Planning Competition and has since evolved with the competition. By serving as a standard, PDDL has fostered research and collaboration through a common formalism that has helped the field of AI planning and progress over the years. It's also seen as a generalization of STRIPS. Some of the key differences compared to STRIPS are that PRECONDITIONs may be an arbitrary boolean combination of atomic facts about the state variables. Instead of unconditional assignments represented by ADD, DELETE lists, the effects can be determined conditionally in the form: IF condition THEN a := v. Goals may be the boolean combination of facts [1].

References

1. https://users.ics.aalto.fi/rintanen/planning.html
2. https://en.wikipedia.org/wiki/Automated_planning_and_scheduling
3. https://en.wikipedia.org/wiki/STRIPS
4. https://en.wikipedia.org/wiki/Planning_Domain_Definition_Language
5. https://en.wikipedia.org/wiki/Action_description_language