

Planning Research Review

AI Planning arose from investigations into state-space search, theorem proving, and control theory and from the practical needs of robotics, scheduling, and other domains^[1]. There are many developments in the field of AI planning and I choose STRIP, ADL, and PDDL action language for short summary.

STRIPS(Stanford Research Institute Problem Solver) is language representing the planning problem developed in 1971 by Fikes and Nilsson. It is consisted by 3 parts. Initial State, Goal state, and Operator. All these three parts use condition(i.e. propositional variables)^[2] as a base unit. Initial state is set of true conditions and there are no false conditions because STRIPS assume if there is no condition written at state it's false. Goal state given as pair $\langle T, F \rangle$ which specify true conditions and false condition, respectively. Operator given with 2 parts Precondition, Postcondition. As state has just true condition, precondition has no false condition. Postcondition is consisted with two list ADD list and DELETE list and this transform the state. STRIPS find a sequence of operator that reach a goal state from initial state. STRIPS has limit because it could only be used in closed world, but it is the base to most of the languages for expressing automated planning problem used today.

ADL (Action Description Language) developed at 1986 by Pednault. ADL is based on STRIPS but there is different thing. The notion UNKNOWN added to ADL so compared to STRIPS everything not occurring in the conditions is unknown instead of being assumed false. In addition, whereas in STRIPS only positive literals and conjunctions are permitted, ADL allows negative literals and disjunctions as well^[4]. This relaxed some of the STRIPS restrictions and made it possible to encode more open world. For detail, STRIPS could be Rich \wedge Beautiful but the same sentence could be expressed in ADL as \neg Poor \wedge \neg Ugly. STRIPS only support ground literals in goals but ADL support quantified variables in goals like $\exists x \text{ At}(P1, x) \wedge \text{At}(P2, x)$. STRIPS goal has only conjunction(\wedge) but ADL has disjunction(\vee). STRIPS doesn't support equality(=) but in ADL equality predicate could be used.

PDDL(Planning Domain Definition Language, Ghallab et al., 1998), was introduced as a computer-parsable, standardized syntax for representing planning problems and has been used as the standard language for the International Planning Competition since 1998^[1] and was inspired by STRIPS and ADL^[3]. This was developed to reinforce the lack of reasoning and expressive power of STRIPS. PDDL can provide sufficient expressive power for very complex realistic planning problems.

[1] Artificial Intelligence A Modern Approach (3rd Edition) Chapter 10 Classical Planning – Summary, Bibliographical and Historical Notes, Exercises

[2] <https://en.wikipedia.org/wiki/STRIPS>

[3] https://en.wikipedia.org/wiki/Planning_Domain_Definition_Language

[4] https://en.wikipedia.org/wiki/Action_description_language

[5] https://en.wikipedia.org/wiki/Automated_planning_and_scheduling

[6] https://en.wikipedia.org/wiki/Propositional_variable

[7] <http://www.teach.cs.toronto.edu/~csc384h/summer/Tests/csc384w16-strips-examples.pdf>

[8] http://www.cogsys.wiai.uni-bamberg.de/teaching/ws0405/s_planning/slides/Introduction_AI_Planning_addon.pdf

[9] https://en.wikipedia.org/wiki/Closed-world_assumption

[10] https://en.wikipedia.org/wiki/Open-world_assumption