RESEARCH REVIEW
Daniel Legorreta

Automated planning and scheduling is one of the major fields of AI (among the others like: Machine Learning, Natural Language Processing, Computer Vision and more). Planning and scheduling arose from investigations into state-space search, theorem proving, and control theory and practical needs of robotics, scheduling, and other domains [1].

In classical perspective [2], planning is the process of computing several steps of a problem-solving procedure before executing any of them. This problem can be solved by search and the main difference between search and planning is the representation of states.

In search, states are represented as a single entity (which may be quite a complex object, but its internal structure is not used by the search algorithm) and in planning, states have structured representations (collections of properties) which are used by the planning algorithm [2]. Automated planning systems can be classified into the following categories: domain-specific planners, domain-independent planners, and domain-configurable planners [3].

To represent planning problems is used Artificial Intelligence planning languages that describe environment's conditions which then lead to desired goals by generating chain of actions based on these conditions.

The languages for representing AP tasks are typically based on extensions of first-order logic. They encode tasks using a set of actions that represents the state-transition function of the world (the planning domain) and a set of first-order predicates that represent the initial state together with the goals of the AP task (the planning problem).

In the early days of AP [1], STRIPS were the most popular representation language. STRIPS were designed as the planning component of the software for the Shakey robot project at SRI. The action representation used by STRIPS has been far more influential than its algorithmic approach.

In 1998 the Planning Domain Definition Language (PDDL) was developed for the first International Planning Competition (IPC) and since that date it has become the standard language for the AP community. In PDDL [4], an action in the planning domain is represented by: (1) the action preconditions, a list of predicates indicating the facts that must be true so the action becomes applicable and (2) the action postconditions, typically separated in add and delete lists, which are lists of predicates indicating the changes in the state after the action is applied.

Before the mid '90s, automated planners could only synthesize plans of no more than 10 actions in an acceptable amount of time [5]. During those years, planners strongly depended on speedup techniques for solving AP problems. Therefore, the application of search control became a very popular solution to accelerate planning algorithms.

In the late 90's, a significant scale up in planning took place due to the appearance of the reachability planning graphs [6] and the development of powerful domain independent heuristics [7,8]. Planners using these approaches could often synthesize 100-action plans just in seconds.

Today exist different proposed over between connection on subfields in the artificial Intelligence, this could impact in AP in future years, as Machine Learning, Deep Learning and Deep Reinforced Learning [9,10,11,12].

RESEARCH REVIEW
Daniel Legorreta

References

1. *Artificial Intelligence: A Modern Approach* by P. Norvig and S.Russell 2nd edition Chapter 11.
2. Course G52PAS Planning and Search 2013-2014, School of Computer Science, University of Nottingham, Dr. Natasha Alechina.
3. Dana Nau, Current Trends in Automated Planning, AI Magazine Volume 28 Number 4.
4. M. Fox and D. Long, PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains, Volume 20.
5. Jiménez, S, Turbides, Tde la Rosa; Learning-Based Planning, Encyclopedia of Artificial Intelligence Volume 3.
6. Avrim L. Blum,Merrick L. Furst, Fast Planning Through Planning Graph Analysis, Artificial Intelligence Vol 90.
7. J. Hoffmann and B. Nebel, The FF Planning System: Fast Plan Generation Through Heuristic Search, Volume 14.
8. Blai Bonet,Héctor Geffner, Planning as heuristic search, Artificial Intelligence Vol 129.
9. Austin Tate, Gerhard Wickler, Lee McCluskey, Machine Learning and Adaptation of Domain Models to Support Real Time Planning in Autonomous Systems.
10. J. Andrew Bagnell, Jeff G. Schneider, Autonomous Helicopter Control Using Reinforcement Learning Policy Search Methods.
11. Hongzi Mao, Mohammad Alizadeh, Resource Management with Deep Reinforcement Learning.
12. Masataro Asai, Alex Fukunaga, Classical Planning in Deep Latent Space: Bridging the Subsymbolic-Symbolic Boundary.