Three important developments in AI search and planning

AI search and planning has its roots in theorem proving, control theory, state-space search, robotics, scheduling and other such fields. These roots manifested themselves in STRIPS (STanford Research Institute Problem Solver), which is often regarded as the first automated planner. The name STRIPS subsequently became synonymous with the formal language of the planner inputs, and serves as the basis for most “action languages” that express automated planning today. Any particular STRIPS[1] instance consists of specifications of: (i) a set of conditions provided by propositional variables; (ii) a set of actions, each of which itself comes with a specification of a set of conditions/propositional variables which (ii-a) must be true for the action to be executable, (ii-b) must be false for the action to be executable, (ii-c) are made true by the action, and (ii-d) are made false by the action; (iii) the initial state, specified by a set of conditions that are initially true; (iv) the goal state, specified a set of of true and false conditions.

The approach used in planning up until the early 1970s consisted of “linear planning”, which operated according to sequences of totally ordered sequence. This approach consisted of dividing goals into individual subgoals, identifying subsequences of actions to achieve each of these subgoals, then finally concatenating the subsequences of actions to identify the goal plan. This was found to have problems, such as the Sussman anomaly, whereby achieving one subgoal meant undoing another already achieve subgoal. This led to the birth of partial order planning, which consists of producing actions that can meet some goal(s) but with specifying a partial order instead of an exact order for those actions. This introduces some flexibility to the order in which the actions can be carried out, and solves problems with linear planning such as the Sussman anomaly. Historically, TWEAK (Chapman, 1987) was the first concrete planner based on these principles.

More recently, an “Imagination-based Planner”[2] (IBP) has been proposed by researchers at DeepMind. This represents an augmentation of existing model-based planners, by learning all aspects of the planning process from experience, including the construction, evaluation and execution of a plan tailored to the target problem. Applicable to both continuous and discrete problems, the IBP employs its model of the environment for imagination-based planning as well as gradient-based policy optimization. It consists of four major components: (i) the manager, which at each iteration decides whether to imagine or act, and is a discrete policy map from a history to a route which determines whether the agent will will execute an action in the world, or imagine the consequences of a proposed action; (ii) the controller, a policy map from a history as well as a state to an action, the state being provided as input by the manager, and determining whether the controller’s output action would be executed in the world or used for imagining; (iii) the imagination, a map from states and actions to consequent states and scalar rewards; (iv) the memory, which recurrently updates external and internal data from the previous history to the current. This model was applied with success to solving a continuous control task of docking a (simulated) spaceship to its mothership, as well as solving a discrete 2d maze problem.

1. http://ai.stanford.edu/~nilsson/OnlinePubs-Nils/PublishedPapers/strips.pdf
2. https://arxiv.org/pdf/1707.06170.pdf