AI Planning

“Automated planning studies the problem of reasoning about actions to achieve goals or to maximise a reward. Actions are usually expressed in terms of preconditions and effects. Preconditions indicate the requirements that must hold to apply the action, while effects are the consequence (including the cost) of applying the action to the state of the world.

Automated planning has been applied to diverse, real-world application areas such as space exploration, manufacturing, machine tool calibration, and road traffic management.”

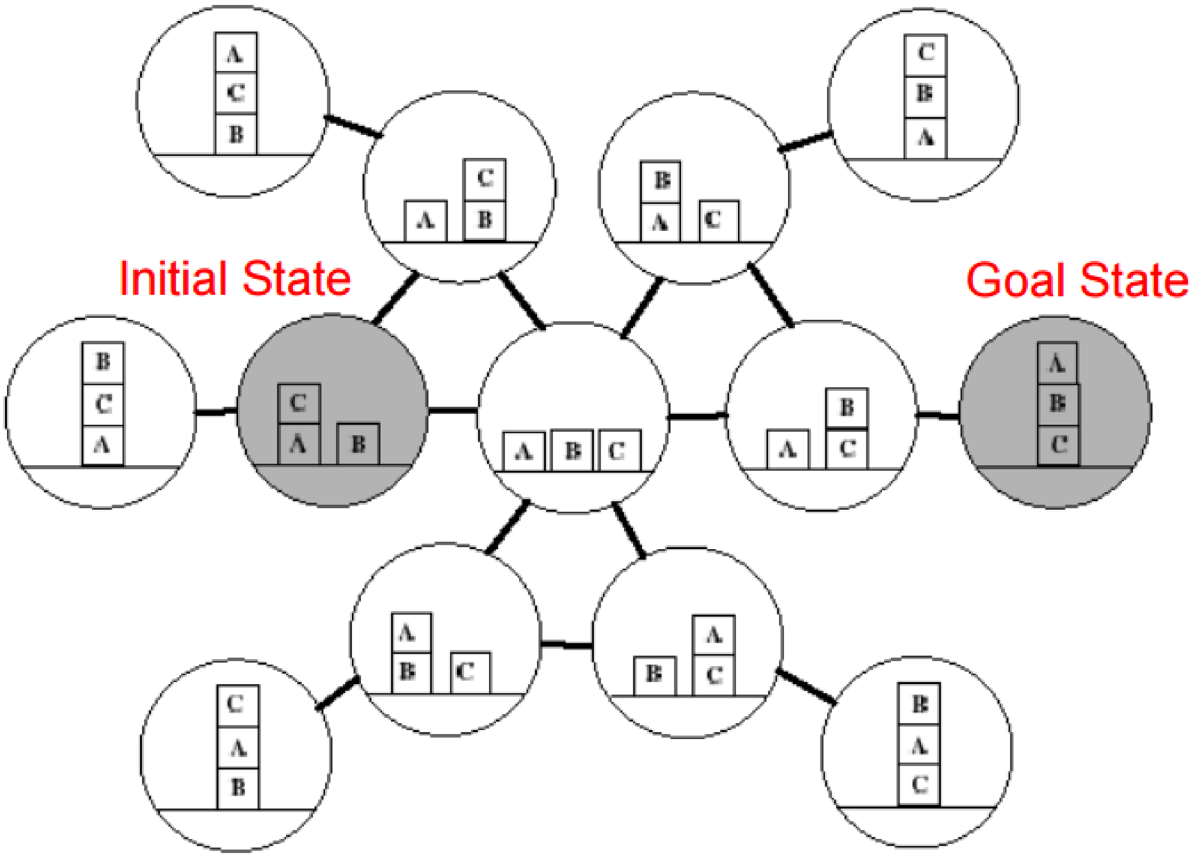
Vallati et al., The 2014 International Planning Competition: Progress and Trends, AI Magazine, Sept. 2015

Deterministic planning, e.g. STRIPS (1971)

```
Initial state: At(A), Level(low), BoxAt(C), BananasAt(B)
Goal state:    Have(Bananas)
```

```
Actions:
               // move from X to Y
               _Move(X, Y)_
               Preconditions:  At(X), Level(low)
               Postconditions: not At(X), At(Y)

               // climb up on the box
               _ClimbUp(Location)_
               Preconditions:  At(Location), BoxAt(Location), Level(low)
               Postconditions: Level(high), not Level(low)

               // climb down from the box
               _ClimbDown(Location)_
               Preconditions:  At(Location), BoxAt(Location), Level(high)
               Postconditions: Level(low), not Level(high)

               // move monkey and box from X to Y
               _MoveBox(X, Y)_
               Preconditions:  At(X), BoxAt(X), Level(low)
               Postconditions: BoxAt(Y), not BoxAt(X), At(Y), not At(X)

               // take the bananas
               _TakeBananas(Location)_
               Preconditions:  At(Location), BananasAt(Location), Level(high)
               Postconditions: Have(bananas)
```

A search problem



Solution: a set of actions with ordering constraints (or a sequence)

Diagram: https://web.engr.oregonstate.edu/~afern/classes/cs533/notes/strips-intro.pdf

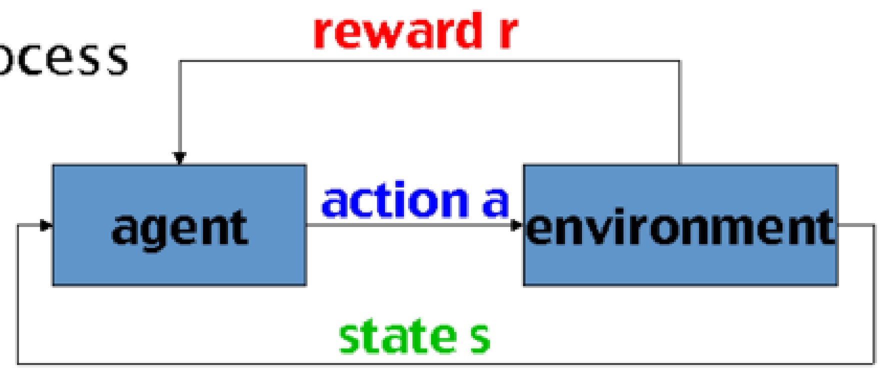
Extensions to deterministic planning

- More expressive action definition languages, e.g. adding continuous change and concurrent actions
- Driven by planning competitions

Probabilistic planning

Markov Decision Process (MDP)

- Markov decision process
  - state $s \in S$
  - action $a \in A$
  - policy $p(a|s)$
  - reward $r(s,a)$
  - dynamics $p(s'|s,a)$



- Optimal policy: maximize cumulative reward
  - finite horizon: $E[\ r(1) + r(2) + r(3) + \ldots + r(T)]$
  - infinite horizon: $E[\ r(1) + \gamma r(2) + \gamma^2 r(3) + \ldots\ ]$

    $0 \leq \gamma \leq 1$: temporal discount factor
  - average reward: $E[\ r(1) + r(2) + \ldots + r(T)]/T,\ T \to \infty$

Slide from http://yosinski.com/mlss12/MLSS-2012-Doya-Neural-Implementation-of-Reinforcement-Learning/

Variations of probabilistic planning

- Model-based (probabilistic state transitions and rewards are known in advance)
- Model free (action effects and rewards must be learned) => Reinforcement learning
- Partial observability: the current state is uncertain – modelled by a probability distribution over states (POMDPs)
- Multiple cooperative decision-makers: Decentralised POMDPs
- Adding game theoretic concepts: partially observable stochastic games (POSGs)

Applications

- "It's one small step in the history of space flight. But it was one giant leap for computer-kind, with a state of the art artificial intelligence system being given primary command of a spacecraft. Known as Remote Agent, the software operated NASA's Deep Space 1 spacecraft and its futuristic ion engine during two experiments that started on Monday, May 17, 1999. For two days Remote Agent ran on the on-board computer of Deep Space 1, more than 60,000,000 miles (96,500,000 kilometers) from Earth. The tests were a step toward robotic explorers of the 21st century that are less costly, more capable and more independent from ground control."

  From NASA's site about Deep Space 1's Remote Agent, according to http://aitopics.org/topic/planning-scheduling .
- "The Hubble Space Telescope uses a short-term [planning and scheduling] system called SPSS and a long-term planning system called Spike"

  https://en.wikipedia.org/wiki/Automated_planning_and_scheduling#Deployment_of_planning_systems
- "POMDPs model many kinds of real-world problems. Notable works include the use of a POMDP in management of patients with ischemic heart disease, assistive technology for persons with dementia, the conservation of the critically endangered and difficult to detect Sumatran tigers and aircraft collision avoidance"

  https://en.wikipedia.org/wiki/Partially_observable_Markov_decision_process
- Robotics …

Planning competition domains

- International Planning Competition 2014
  - Road accident management
  - Road layout generation
  - Aircraft maintenance
  - Wildfire Management (how to allocate firefighting resources to protect assets)
  - Invasive Species Management (how to allocate personnel resources to limit the spread of the Tamarisk invasive plants in stream systems)
  - Academic Advising: an academic advisor’s task of recommending courses for students to help them graduate as quickly as possible.

Some details of problem size

- Monte-Carlo planning in large POMDPs, 2010
  - 10 × 10 battleship (with 5 ships) => 10^{18} states

    "POMCP was able to sink all ships more than 50 moves faster, on average, than random play, and more than 25 moves faster than randomly selecting amongst preferred actions (which corresponds to the simple strategy used by many humans …)"
  - Partially observable PacMan => 10^{56} states

    Demo: http://www0.cs.ucl.ac.uk/staff/D.Silver/web/Applications.html
- http://robots.stanford.edu/papers/thrun.mcpomdp.pdf
  - Simulated mobile robot planning application

    "The optimal solution takes approximately 25 steps; thus, a successful POMDP planner must be capable of looking 25 steps ahead. … A successful policy was consistently found after 17 episodes (or 6,150 backups), in all 13 experiments. In our current implementation, 6,150 backups require approximately 29 minutes on a Pentium PC"