WIKIPEDIA

State space search

State space search is a process used in the field of computer science, including artificial intelligence (AI), in which successive configurations or *states* of an instance are considered, with the intention of finding a *goal state* with a desired property.

Problems are often modelled as a state space, a set of *states* that a problem can be in. The set of states forms a graph where two states are connected if there is an *operation* that can be performed to transform the first state into the second.

State space search often differs from traditional computer science search methods because the state space is *implicit*: the typical state space graph is much too large to generate and store in memory. Instead, nodes are generated as they are explored, and typically discarded thereafter. A solution to a combinatorial search instance may consist of the goal state itself, or of a path from some *initial state* to the goal state.

Contents

Representation

In state space search a state space is formally represented as a tuple $S :< S, A, Action(s), Result(s,a), Cost(s,a) >$, in which:

- S is the set of all possible states;
- A is the set of possible action, not related to a particular state but regarding all the state space;
- $Action(s)$ is the function that establish which action is possible to perform in a certain state;
- $Result(s,a)$ is the function that return the state reached performing action a in state s
- $Cost(s,a)$ is the cost of performing an action a in state s. In many state spaces is a constant, but this is not true in general.

Examples of State-space search algorithms

Uninformed Search

According to Poole and Mackworth, the following are *uninformed* state-space search methods, meaning that they do not know information about the goal's location.[1]

- Depth-first search
- Breadth-first search
- Lowest-cost-first search

Informed Search

Some algorithms take into account information about the goal node's location in the form of a heuristic function[2]. Poole and Mackworth cite the following examples as informed search algorithms:

- Heuristic depth-first search
- Greedy best-first search
- A* search

See also

- State space
- State space planning

References

1. Poole, David; Mackworth, Alan. "3.5 Uninformed Search Strategies‣ Chapter 3 Searching for Solutions ‣ Artificial Intelligence: Foundations of Computational Agents, 2nd Edition" (http://artint.info/2e/html/ArtInt2e.Ch3.S5.html). *artint.info*. Retrieved 7 December 2017.
2. Poole, David; Mackworth, Alan. "3.6 Heuristic Search‣ Chapter 3 Searching for Solutions ‣ Artificial Intelligence: Foundations of Computational Agents, 2nd Edition" (http://artint.info/2e/html/ArtInt2e.Ch3.S6.html). *artint.info*. Retrieved 7 December 2017.

- Stuart J. Russell and Peter Norvig (1995). *Artificial Intelligence: A Modern Approach*. Prentice Hall.

Search Problem Example: water jug

Water Jug Problem

- You are given two jugs: a 4-litre and a 3-litre. Neither has any measuring markers on them.
- There is an endless supply of water to fill the jugs.
- How can you get exactly 2 litres of water into the 4-litre jug?