

# Foundations of Artificial Intelligence

## 7. State-Space Search: Examples of State Spaces

Malte Helmert

University of Basel

February 27, 2019

# State-Space Search: Overview

## Chapter overview: state-space search

- 5.–7. Foundations
  - 5. State Spaces
  - 6. Representation of State Spaces
  - 7. Examples of State Spaces
- 8.–12. Basic Algorithms
- 13.–19. Heuristic Algorithms

# Three Examples

In this chapter we introduce three state spaces that we will use as illustrating examples:

- 1 blocks world
- 2 route planning in Romania
- 3 missionaries and cannibals

# Blocks World

# Blocks World

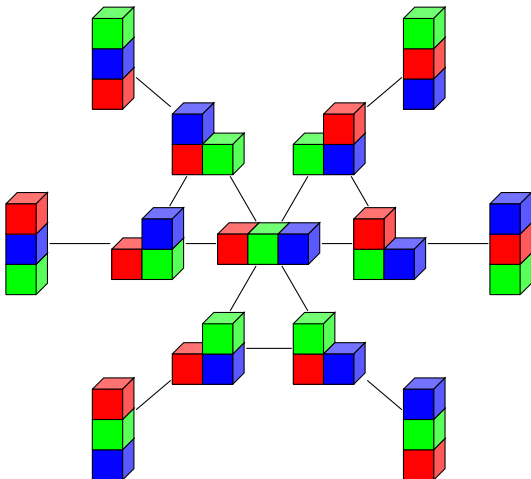
**Blocks world** is a traditional example problem in AI.

## Setting: Blocks World

- Colored blocks lie on a table.
- They can be stacked into towers, moving one block at a time.
- Our task is to create a given goal configuration.

# Example: Blocks World with Three Blocks

(action names omitted for readability;  
initial state and goal can be arbitrary)



# Blocks World: Formal Definition

state space  $\langle S, A, cost, T, s_0, S_* \rangle$  for blocks world with  $n$  blocks

## State Space Blocks World

states  $S$ :

partitions of  $\{1, 2, \dots, n\}$  into nonempty ordered lists

example  $n = 3$ :

- $\{\langle 1, 2, 3 \rangle\}, \{\langle 1, 3, 2 \rangle\}, \{\langle 2, 1, 3 \rangle\},$   
 $\{\langle 2, 3, 1 \rangle\}, \{\langle 3, 1, 2 \rangle\}, \{\langle 3, 2, 1 \rangle\}$
- $\{\langle 1, 2 \rangle, \langle 3 \rangle\}, \{\langle 2, 1 \rangle, \langle 3 \rangle\}, \{\langle 1, 3 \rangle, \langle 2 \rangle\},$   
 $\{\langle 3, 1 \rangle, \langle 2 \rangle\}, \{\langle 2, 3 \rangle, \langle 1 \rangle\}, \{\langle 3, 2 \rangle, \langle 1 \rangle\}$
- $\{\langle 1 \rangle, \langle 2 \rangle, \langle 3 \rangle\}$

# Blocks World: Formal Definition

state space  $\langle S, A, cost, T, s_0, S_* \rangle$  for blocks world with  $n$  blocks

## State Space Blocks World

actions  $A$ :

- $\{move_{b,b'} \mid b, b' \in \{1, \dots, n\} \text{ with } b \neq b'\}$ 
  - move block  $b$  onto block  $b'$ .
  - both must be uppermost blocks in their towers
- $\{tactable_b \mid b \in \{1, \dots, n\}\}$ 
  - move block  $b$  onto the table ( $\rightsquigarrow$  forming a new tower)
  - must be uppermost block in its tower

action costs  $cost$ :

$cost(a) = 1$  for all actions  $a$



# Blocks World: Formal Definition

state space  $\langle S, A, cost, T, s_0, S_* \rangle$  for blocks world with  $n$  blocks

## State Space Blocks World

transitions:

example for  $a = move_{2,3}$ :

transition  $s \xrightarrow{a} s'$  exists iff

- $s = \{ \langle b_1, \dots, b_k, 2 \rangle, \langle c_1, \dots, c_m, 3 \rangle \} \cup X$  and
- if  $k > 0$ :  $s' = \{ \langle b_1, \dots, b_k \rangle, \langle c_1, \dots, c_m, 3, 2 \rangle \} \cup X$
- if  $k = 0$ :  $s' = \{ \langle c_1, \dots, c_m, 3, 2 \rangle \} \cup X$

# Blocks World: Formal Definition

state space  $\langle S, A, cost, T, s_0, S_* \rangle$  for blocks world with  $n$  blocks

## State Space Blocks World

initial state  $s_0$  and goal states  $S_*$ :

one possible definition for  $n = 3$ :

- $s_0 = \{\langle 1, 3 \rangle, \langle 2 \rangle\}$
- $S_* = \{\{\langle 3, 2, 1 \rangle\}\}$

(in general arbitrarily choosable)

# Blocks World: Properties

blocks	states	blocks	states
1	1	10	58941091
2	3	11	824073141
3	13	12	12470162233
4	73	13	202976401213
5	501	14	3535017524403
6	4051	15	65573803186921
7	37633	16	1290434218669921
8	394353	17	26846616451246353
9	4596553	18	588633468315403843

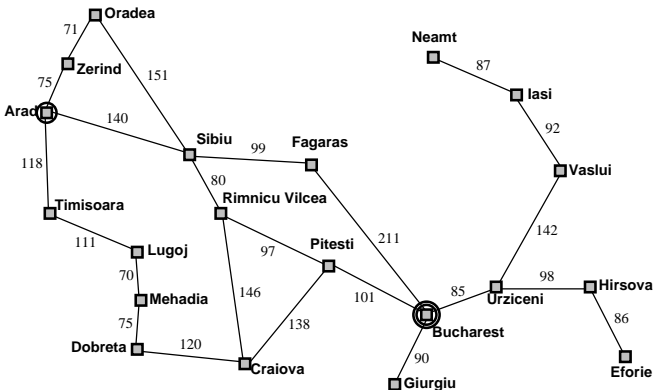
- For every given initial and goal state with  $n$  blocks, simple algorithms find a **solution** in time  $O(n)$ . (How?)
- Finding **optimal solutions** is **NP-complete** (with a compact problem description).

# Route Planning in Romania

# Route Planning in Romania

## Setting: Route Planning in Romania

We are on holiday in Romania and are currently located in Arad. Our flight home leaves from Bucharest. How to get there?



# Romania Formally

## State Space Route Planning in Romania

- **states**  $S$ : {arad, bucharest, craiova, . . . , zerind}
- **actions**  $A$ :  $move_{c,c'}$  for any two cities  $c$  and  $c'$  connected by a single road segment
- **action costs**  $cost$ : see figure, e.g.,  $cost(move_{iasi,vaslui}) = 92$
- **transitions**:  $s \xrightarrow{a} s'$  iff  $a = move_{s,s'}$
- **initial state**:  $s_0 = arad$
- **goal states**:  $S_* = \{bucharest\}$

# Missionaries and Cannibals

# Missionaries and Cannibals

## Setting: Missionaries and Cannibals

- Six people must cross a river.
- Their rowing boat can carry one or two people across the river at a time (it is too small for three).
- Three people are missionaries, three are cannibals.
- Missionaries may never stay with a majority of cannibals.



# Missionaries and Cannibals Formally

## State Space Missionaries and Cannibals

states  $S$ :

triples of numbers  $\langle m, c, b \rangle \in \{0, 1, 2, 3\} \times \{0, 1, 2, 3\} \times \{0, 1\}$ :

- number of missionaries  $m$ ,
- cannibals  $c$  and
- boats  $b$

on the **left** river bank

initial state:  $s_0 = \langle 3, 3, 1 \rangle$

goal:  $S_* = \{ \langle 0, 0, 0 \rangle, \langle 0, 0, 1 \rangle \}$

actions, action costs, transitions: ?

# Summary

# Summary

illustrating examples for state spaces:

- **blocks world:**
  - family of tasks where  $n$  blocks on a table must be rearranged
  - traditional example problem in AI
  - number of states explodes quickly as  $n$  grows
- **route planning in Romania:**
  - small example of explicitly representable state space
- **missionaries and cannibals:**
  - traditional brain teaser with small state space (32 states, of which many unreachable)