

Example Problems



School of Electronic and Computer Engineering
Peking University

Wang Wenmin



Contents

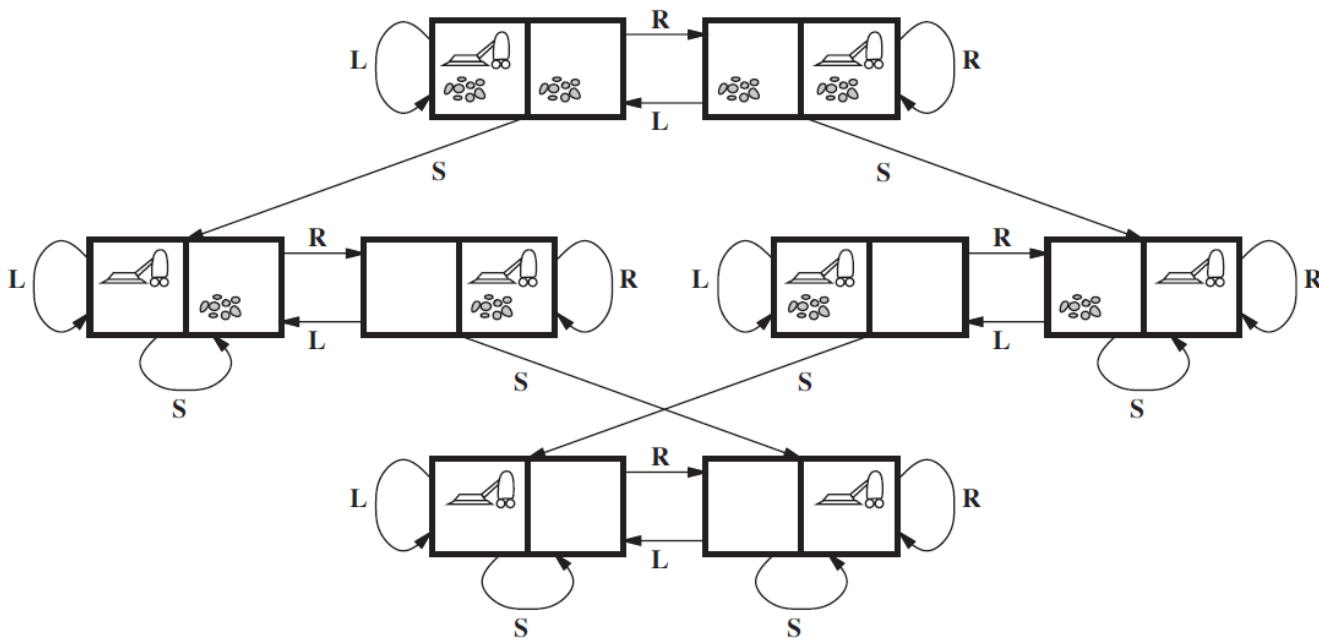
- ❑ 3.2.1 Example1: Vacuum-cleaner world
- ❑ 3.2.2 Example2: 8-puzzle
- ❑ 3.2.3 Example3: 8-queens problem

Example 1: Vacuum-cleaner world 真空吸尘器世界

- Vacuum-cleaner world has been introduced in “2.1.6. Intelligent agent paradigm”.
真空吸尘器世界已经在 “2.1.6. Intelligent agent paradigm” 中讲过。

- The states are determined by both the agent location and dirt location.
其状态是由智能体的位置和灰尘的位置决定的。

- Links denote actions:
 $L = \textit{Left}$, $R = \textit{Right}$, $S = \textit{Suck}$.
链接表示动作：
 $L = \text{左移}$, $R = \text{右移}$, $S = \text{吸尘}$ 。



Example 1: Vacuum-cleaner world 真空吸尘器世界

States 状态

- Agent is in one of two locations, each may contain dirt or not.

智能体在两个地点中的一个，每个也许有灰尘或者没有。

- Possible states, 2 locations: $2 \times 2^2 = 8$ ($n \times 2^n$).

可能的状态，2个地点： $2 \times 2^2 = 8$ ($n \times 2^n$)。

1) Initial state 初始状态

Any state can be as the initial state.

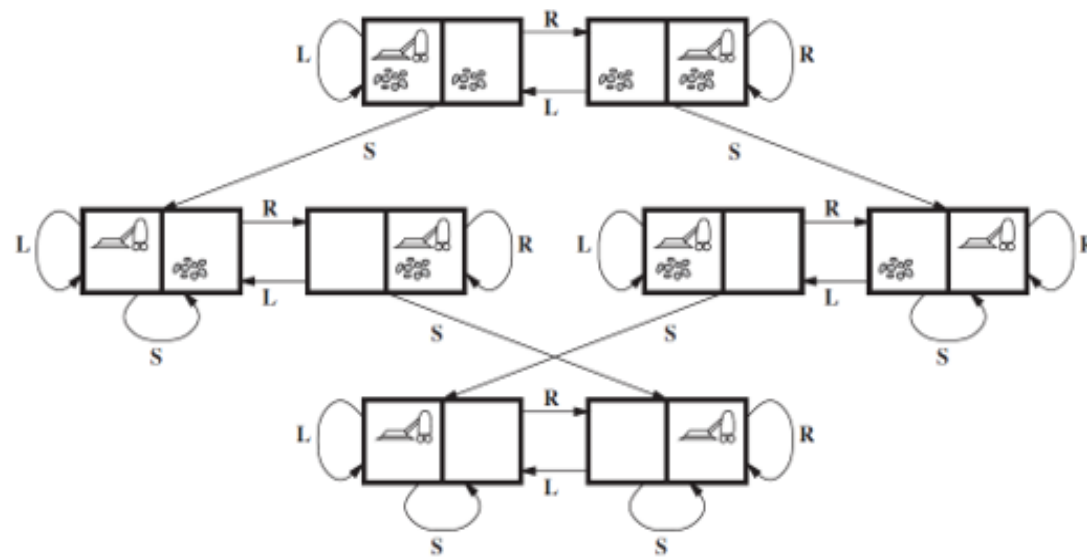
任何状态都可以作为初始状态。

2) Actions 动作

Each state has just three actions:

Left, Right, and Suck.

每个状态仅有三个动作：左移，右移，以及吸尘。



Example 1: Vacuum-cleaner world 真空吸尘器世界

3) Transition model 转换模型

The actions have their expected effects, except that moving:

该动作应有的预期效果，下述动作除外：

- *Left* in the leftmost, 在最左边进行左移
- *Right* in the rightmost, 在最右边进行右移
- *Suck* in a clean square. 在清洁区域进行吸尘

4) Goal test 目标测试

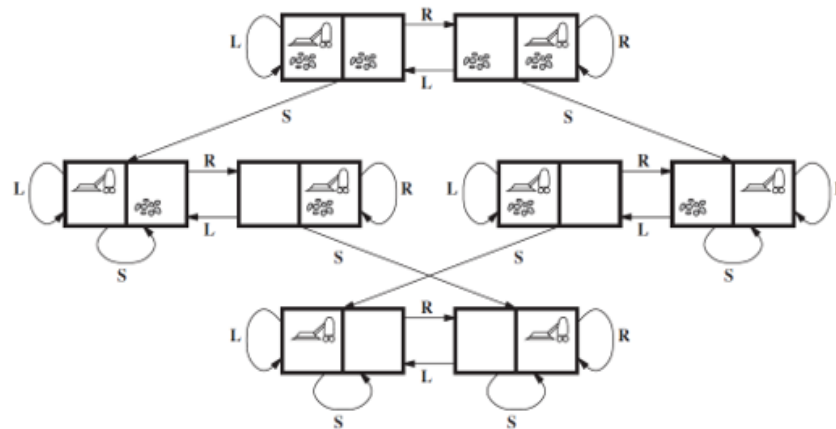
Whether all the squares are clean.

是否所有的区域内都干净。

5) Path cost 路径代价

The number of steps in the path (each step costs 1).

等于路径的步数（每一步的代价）。



Example 2: 8-puzzle 8数码难题

□ 8-puzzle: 3×3 board with 8 numbered tiles and a blank space.

8数码难题： 3×3 棋盘上有8个数字棋子和一个空格。

7	2	4
5		6
8	3	1

Start state

	1	2
3	4	5
6	7	8

Goal state

A tile adjacent to the blank space can slide into the space. The object is to reach a specified goal state.

与空格相邻的滑块可以移向该空格，目的是达到一个指定的目标状态。

Example 2: 8-puzzle 8数码难题

□ States 状态

- Each of 8 numbered tiles in one of the 9 squares, and blank in the last square.

8个数字滑块每个占据一个方格，而空格则位于最后一个方格。

□ 1) Initial state 初始状态

- Any state can be the initial state. 任意一个状态都可以成为初始状态。

□ 2) Actions 动作

- Simplest formulation defines the actions as movements of the blank space: *Left*, *Right*, *Up*, or *Down*.

最简单的形式化是将动作定义为空格的移动：左、右、上、下。

- Different subsets are depending on where the blank is.
不同的子集依赖于空格的位置。

7	2	4
5		6
8	3	1

Start state

Example 2: 8-puzzle 8数码难题

□ 3) Transition model 转换模型

Given a state and action, this returns the resulting state. E.g., if we apply *Left* to the start state, the resulting state has the 5 and the blank switched.

给定状态和动作，其返回结果状态。例如，如果我们对初始状态施加左移动作，由此产生的状态则使5与空格互换。

□ 4) Goal test 目标测试

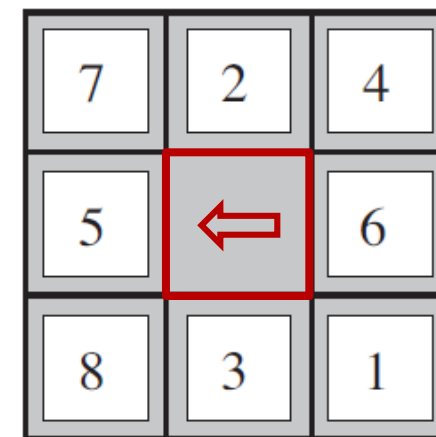
Checks whether the state matches the goal configuration.

即检查状态是否与目标布局相符。

□ 5) Path cost 路经代价

The number of steps in the path (each step costs 1).

等于路径的步数（每一步的代价）。



Transition

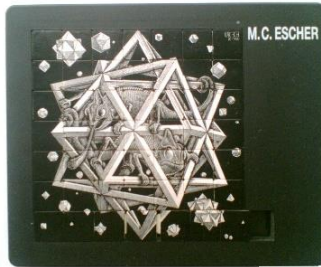
Sliding block puzzles 滑块难题

- The 8-puzzle belongs to the **family** of sliding block puzzles, this family is known to be **NP-complete**.

8数码难题属于滑块难题家族，这个家族被认为是NP完的。



3x3 sliding puzzle.



7x7 sliding block puzzle



15-puzzle

I	H	E	S
H	P	I	O
P	I	S	R
P	O		H

Word puzzle



华容道

Example 3: 8-queens problem 8皇后问题

- The goal is to place 8 queens on a chessboard such that no queen attacks any other. (A queen attacks any piece in the same row, column or diagonal.)

其目标是将8个皇后摆放在国际象棋的棋盘上，使得皇后之间不发生攻击（一个皇后会攻击同一行、同一列或同一斜线上的其他皇后）。

- Two main kinds of formulation:

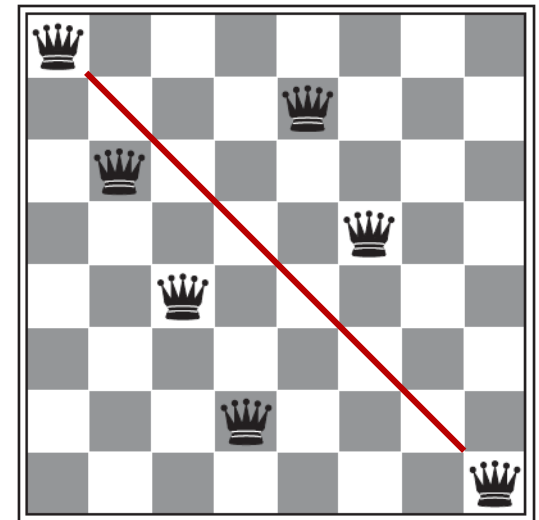
两种主要类型的形式化方法：

- **Incremental formulation**: starts with an empty state, then each action adds a queen to change the state.

增量形式化：从空状态开始，然后每次添加一个皇后改变其状态。

- **Complete-state formulation**: starts with all 8 queens on the board, and moves them around.

全态形式化：初始时8个皇后都放在棋盘上，然后再将她们移开。



A queen attacks another one in the same diagonal.

Example 3: 8-queens problem 8皇后问题

The incremental formulation 增量式形式化

□ **States:** Any arrangement of 1 to 8 queens on the board is a state.

状态：第1至第8个皇后在棋盘上任意摆放，为一个状态。

□ **1) Initial state:** No queens on the board.

初始状态：棋盘上没有皇后。

□ **2) Actions:** Add a queen to any empty square.

动作：添加一个皇后至任意一个空格。

□ **3) Transition model:** Returns the board with a queen added to the specified square.

转换模型：将一个皇后添加到指定空格，再返回该棋局。

□ **4) Goal test:** 8 queens are on the board, none attacked.

目标测试：8个皇后都在棋盘上，并且没有攻击。

□ **5) Path cost:** The number of steps (each step costs 1).

路径代价：等于步数（每步代价为1）。

Thank you for your attention!

