

第一次作业答案

3.7 给出下列问题的初始状态、目标测试、后继函数和耗散函数。选择精确得足以实现的形式化。

- a. 只用四种颜色对平面地图染色,要求每两个相邻的地区不能染成相同的颜色。
- b. 一间屋子里有一只3英尺高的猴子,屋子的房顶上挂着一串香蕉,离地面8英尺。屋子里有两个可叠放起来、可移动、可攀登的3英尺高的箱子。猴子很想得到香蕉。
- d. 有三个水壶,容量分别为12加仑、8加仑和3加仑,还有一个水龙头。可以把壶装满或者倒空,从一个壶倒进另一个壶或者倒在地上。要求量出刚好1加仑水。

a. Initial state: No regions colored.

Goal test: All regions colored, and no two adjacent regions have the same color.

Successor function: Assign a color to a region.

Cost function: Number of assignments.

b. Initial state: As described in the text.

Goal test: Monkey has bananas.

Successor function: Hop on crate; Hop off crate; Push crate from one spot to another; Walk from one spot to another; grab bananas (if standing on crate).

Cost function: Number of actions.

c. Initial state: considering all input records.

Goal test: considering a single record, and it gives “illegal input” message.

Successor function: run again on the first half of the records; run again on the second half of the records.

Cost function: Number of runs.

Note: This is a **contingency problem**; you need to see whether a run gives an error message or not to decide what to do next.

d. Initial state: jugs have values $[0, 0, 0]$.

Successor function: given values $[x, y, z]$, generate $[12, y, z]$, $[x, 8, z]$, $[x, y, 3]$ (by filling); $[0, y, z]$, $[x, 0, z]$, $[x, y, 0]$ (by emptying); or for any two jugs with current values x and y , pour y into x ; this changes the jug with x to the minimum of $x + y$ and the capacity of the jug, and decrements the jug with y by the amount gained by the first jug.

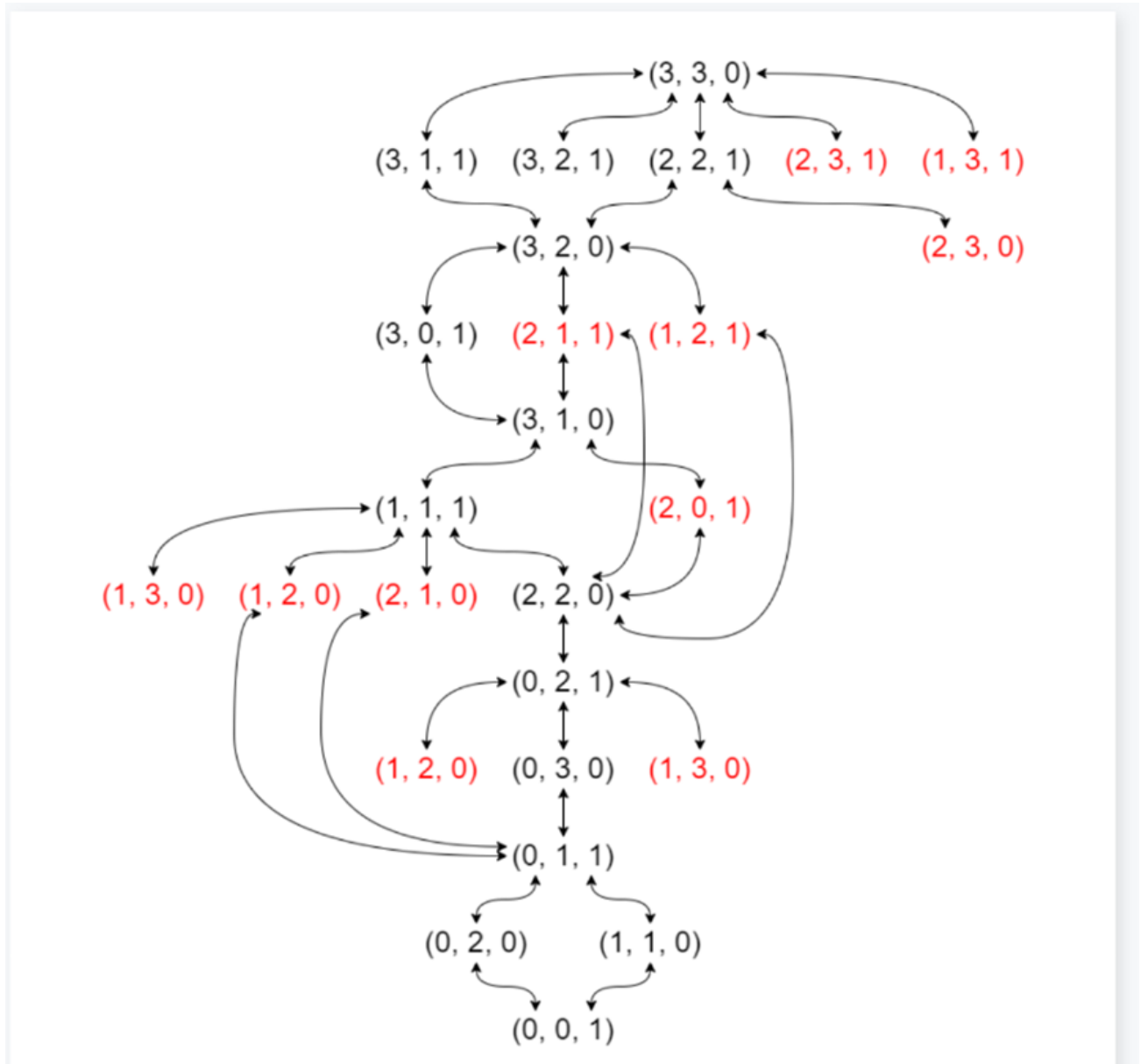
Cost function: Number of actions.

3.9 传教士和野人问题通常描述如下:三个传教士和三个野人在河的一边,还有一条能载一个人或者两个人的船。找到一个办法让所有的人都渡到河的另一岸,要求在任何地方野人数都不能多于传教士的人数(可以只有野人没有传教士)。这个问题在AI领域中很著名,因为它是第一篇从分析的观点探讨问题形式化的论文的主题 (Amarel, 1968)

- 精确地形式化该问题，只描述确保该问题有解所必需的特性。画出该问题的完全状态空间图。
- 用一个合适的搜索算法实现和最优地求解该问题。检查重复状态是个好主意吗？
- 这个问题的状态空间如此简单，你认为为什么人们求解它却很困难？

a.

- 初始状态：3个传教士和3个野人以及一条船在岸的这一边，另一岸为空
- 目标测试：传教士和野人都到达另一岸
- 后继函数：移动1或2个人以及船到另一岸
- 代价函数：船移动的次数



b.

- 由于问题规模有限，几乎所有的最优搜索算法都可以
- 最优解之一： $(3, 3, 0) \rightarrow (2, 2, 1) \rightarrow \dots \rightarrow (0, 2, 0) \rightarrow (0, 0, 1)$

- 问题规模有限，可以不必检查重复状态

c.

几乎所有的移动要么是非法的，要么就需要返回到上一状态。