搜索概述和状态空间法

万永权

目录

CONTENTS

搜索技术

- **问题求解过程是搜索答案（目标）的过程 。**
- 因此问题求解技术也叫搜索技术
- 依靠经验，利用已有知识，根据问题的实际情况，不断寻找可利用知识，从而构造一条代价最小的推理路线，使问题得以解决的过程称为搜索。

搜索的定义

- 求解问题的第一步，就是把问题描述清楚，也就是目标的表示，它涉及到一种知识表示策略——状态空间表示法。第二步就是搜索策略。这里的“搜索”是指，智能系统尝试性地找到目标解的动作序列。

- 搜索问题可分解为两个关键的子问题：
  - （1）搜索什么；
  - （2）在哪里搜索。
  - 前者是指搜索的解，即目标为何。后者是指搜索空间。搜索空间就是由一系列状态构成。

搜索的类型

按是否使用启发式信息：

盲目搜索

也称为无信息搜索，即只按预定的控制策略进行搜索，在搜索过程中获得的中间信息不用来改进控制策略。

启发式搜索

在搜索中加入了与问题有关的启发性信息，用于指导搜索朝着最有希望的方向前进，加速问题的求解过程并找到最优解。

适用情况

1. 不良结构或非结构化问题；

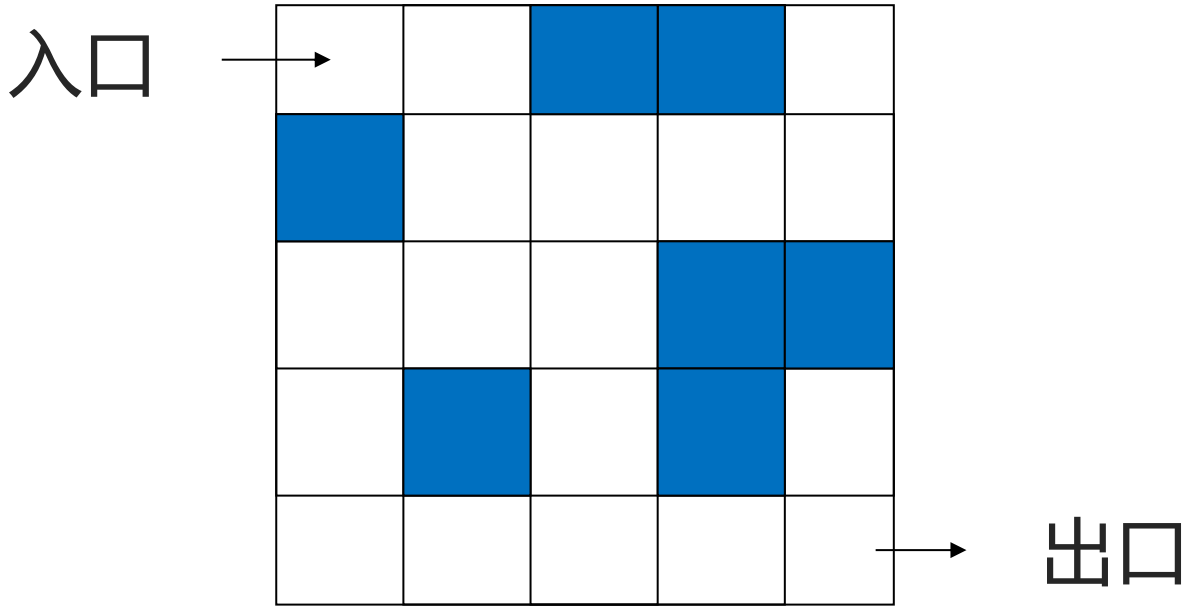
2. 难以获得求解所需的全部信息；

3. 没有现成的算法可供求解使用。

搜索实例

搜索实例

迷宫



搜索实例

八数码难题

2	8	3
1		6
7	4	5

初始状态

1	2	3
8		4
7	6	5

目标状态

搜索的挑战

组合爆炸

- 魔方问题
- 博弈问题
- 皇后问题
- 行商问题
- 排课问题
- 背包问题

状态空间表示法

- **状态空间**（state space）表示法是人工智能中最基本的**形式化方法**，是其他形式化方法和问题求解技术的出发点。
- 通过在某个可能的解空间内寻找一个解来求解问题。这种基于解空间的问题表示和求解方法就是状态空间法。状态空间搜索的研究焦点在于设计高效的搜索算法，以降低搜索代价并解决组合爆炸问题。

- 基础概念：
  - 状态
  - 操作符

状态空间表示法

1）状态（state）

状态是用来描述在问题求解过程中某一个时刻**进展情况**等**陈述性知识**的数据结构。

它可形式地表示为：$S_k = \{S_{k0}, S_{k1}, \dots\}$

当对每一个分量都给以确定的值时，就得到了一个具体的状态。

- 其中，每个元素S_{ki}称为一个状态变量。
- 状态的表示还可以根据具体应用，采取合适的数据结构，如**符号、字符串、多维数组、树**和**图**等。

状态空间表示法

- **2）操作(Operator)**
- **操作**也称为**运算**，表示引起状态变化的**过程性知识**的一组关系或函数，它会引起状态中的某些分量发生改变，从而使问题由一个具体状态转换到另一个具体状态。
- 操作可以是一个动作（如棋子的移动）、过程、规则、数学算子等，**表示状态之间存在的关系**。

状态空间表示法

◆ 3）状态空间（State Space）

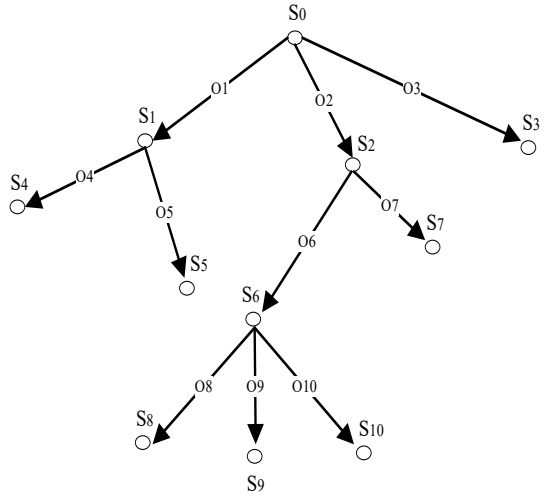
状态空间用来表示一个问题的全部状态以及这些状态之间的相互关系。

常用一个**四元组**（S,O,S_0,G）来表示，其中:

- S 为问题的**状态集合**;
- O 为**操作符的集合**;
- S_0 是问题的**初始状态**，是S的一个非空真子集，即$S_0 \subset S$;
- G 为问题的**目标状态**，它既可以是若干具体状态，也可以是满足某些性质的路径信息描述，$G \subset S$。

状态空间表示法

- 状态空间也可以用**有向图**来表示，
  - **节点**表示问题的**状态**，
  - 节点之间的**有向边**表示引起状态变换的**操作**，有时边还赋有**权值**，表示变换所需的**代价**。
- **问题的解可能是**图中的**一个状态**，**也可能是**从初始状态到某个目标状态的**一条路径**，**还可能是**达到目标所花费的**代价**。
- 图中，**问题的解**便是一条从节点 S_0 到节点 S_8 的路径，它是一个从初始状态到目标状态的有限的操作算子序列 $\{O_1, O_2, \cdots O_k\}$，称为**求解路径**。**问题的解往往并不唯一。**



问题状态描述

- 要完成某个问题的状态描述，需要确定的三件事：
  1. 该状态描述方式，特别是初始状态描述；
  2. 算符集合及其对状态描述的作用；
  3. 目标状态描述的特性；

例： 猴子摘香蕉问题

- 在讨论谓词逻辑知识表示时，我们曾提到过这一问题，现在用状态空间法来解决这一问题。

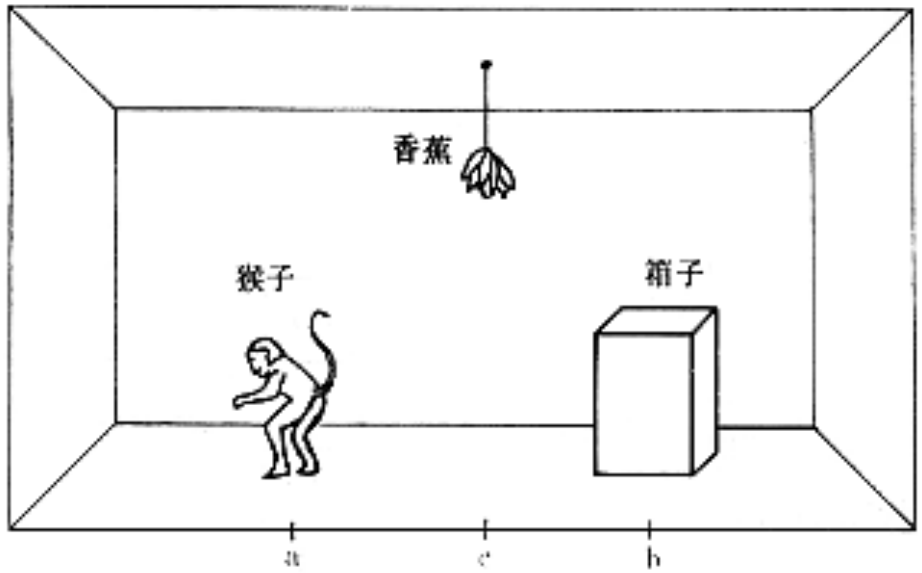


图 2.5 猴子和香蕉问题

状态定义

- 用一个四元组（W，x, Y, z）来表示问题的状态
  - W：猴子的水平位置
  - x：当猴子爬到箱子顶上取1，否则取0
  - Y：箱子的水平位置
  - z：当猴子摘到香蕉时取1，否则取0

- 所有可能的状态为

  S_0: (a, b, 0, 0)　初始状态
  S_1: (b, b, 0, 0)
  S_2: (c, c, 0, 0)
  S_3: (c, c, 1, 0)
  S_4: (c, c, 1, 1)　目标状态

操作符：

① goto(U)： 猴子当前位置W走到水平位置U

（W, 0 , Y, z）→(U, 0 , Y, z)

注：猴子必须不在箱子上

② pushbox(V)： 猴子将箱子从W位置推到水平位置V

（W, 0, W, z）→ (V, 0, V, z)

注：猴子与箱子必须在同一位置

操作符：

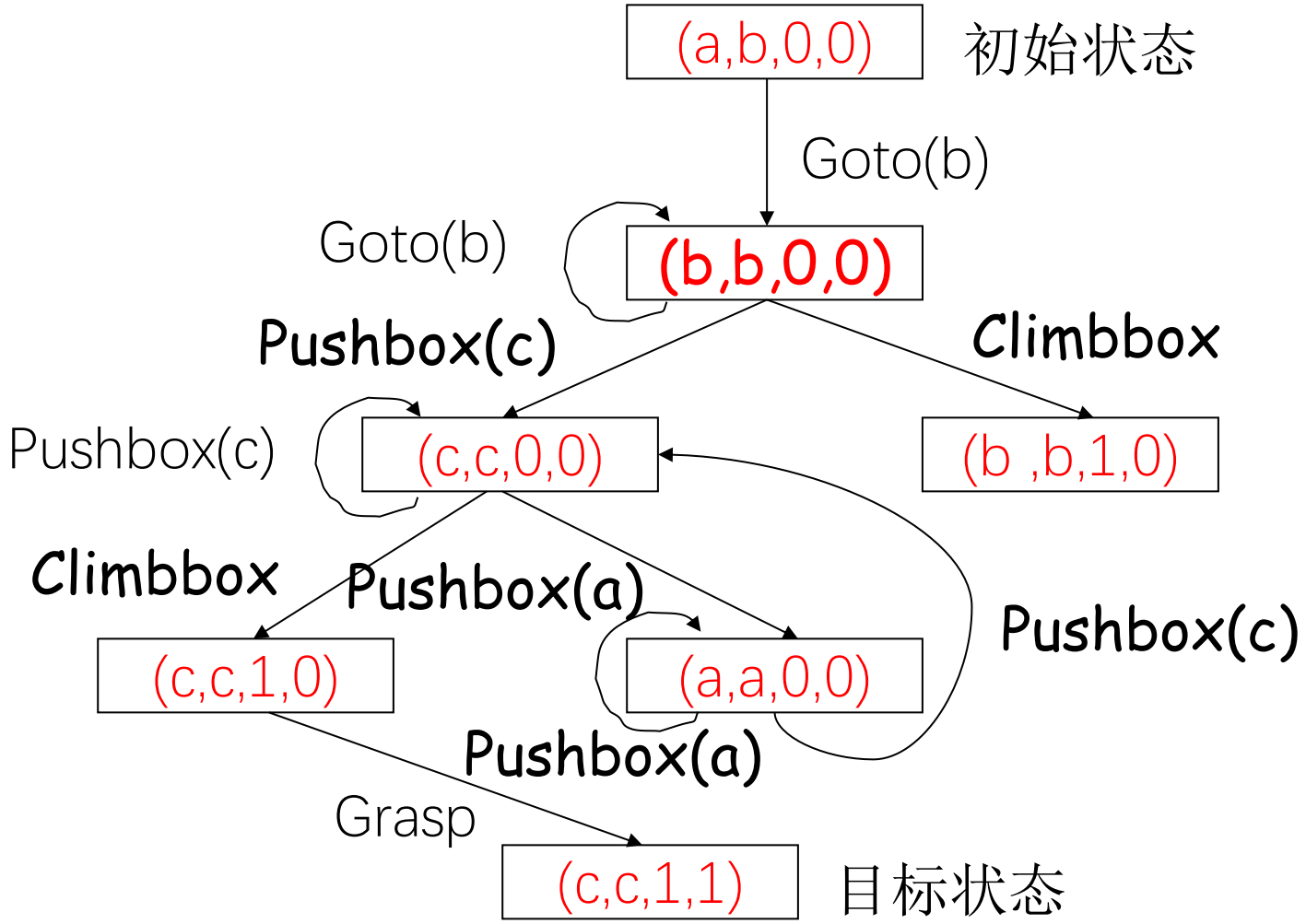
③ **climbbox**：　猴子爬到箱子上：

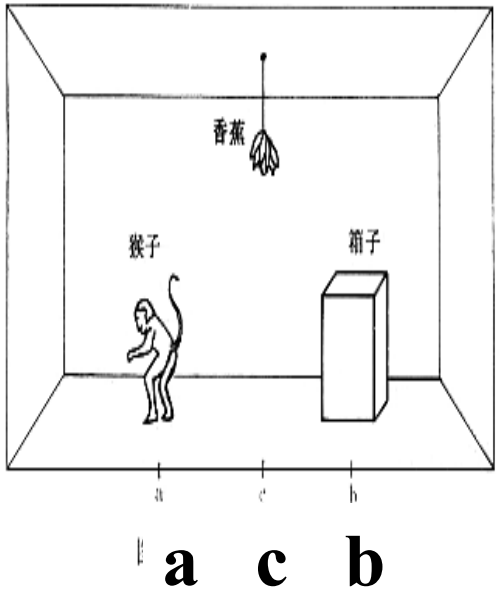
（W, 0, W, z）→ (W，1, W, z)

④ **grasp**：　猴子摘到香蕉：

（c, 1, c, 0）→ (c, 1, c, 1)

操作序列为：
{Goto(b), Pushbox(c), Climbbox, Grasp}

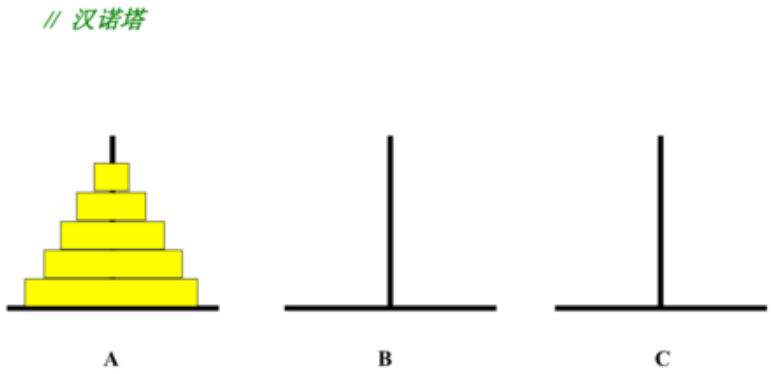




状态空间问题的例子

- 例2 二阶汉诺塔问题。设有三根柱子，它们的编号分别是1号、2号和3号。在初始情况下，1号柱子上穿有A和B两个圆盘，A比B小，A位于B的上面。要求把这两个圆盘全部移到另一根柱子上，而且规定每次只能移动一个圆盘，任何时刻都不能使大圆盘位于小圆盘的上面。

汉诺塔游戏：
https://www.saolei.games/h

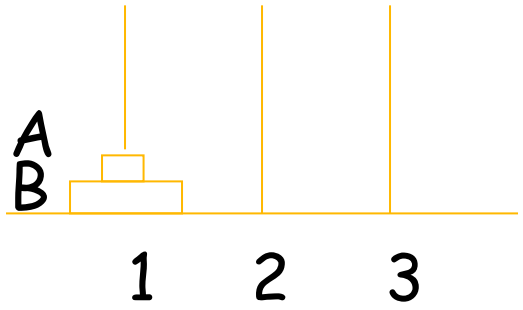


解：设用$S_k=\{S_{k0}, S_{k1}\}$表示问题的状态，其中，S_{k0}表示圆盘A所在的柱子号，S_{k1}表示圆盘B所在的柱子号。全部可能的问题状态共有以下9种：

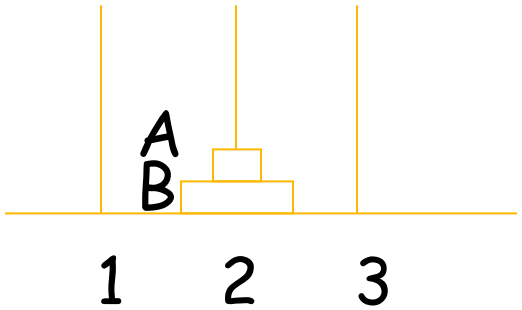
$S_0=(1, 1)$ $S_1=(1, 2)$ $S_2=(1, 3)$ $S_3=(2, 1)$ $S_4=(2, 2)$

$S_5=(2, 3)$ $S_6=(3, 1)$ $S_7=(3, 2)$ $S_8=(3, 3)$

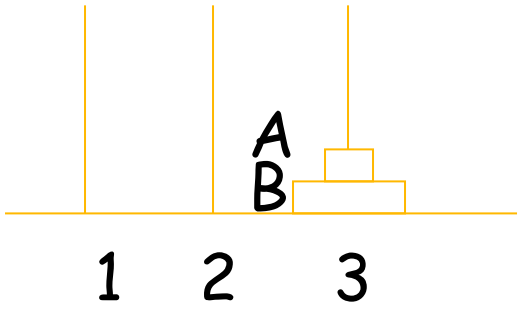
其中，初始状态集合为S_0 和目标状态集合S_4、S_8 如下图示。



(a)初始状态S_0



(b)目标状态S_4



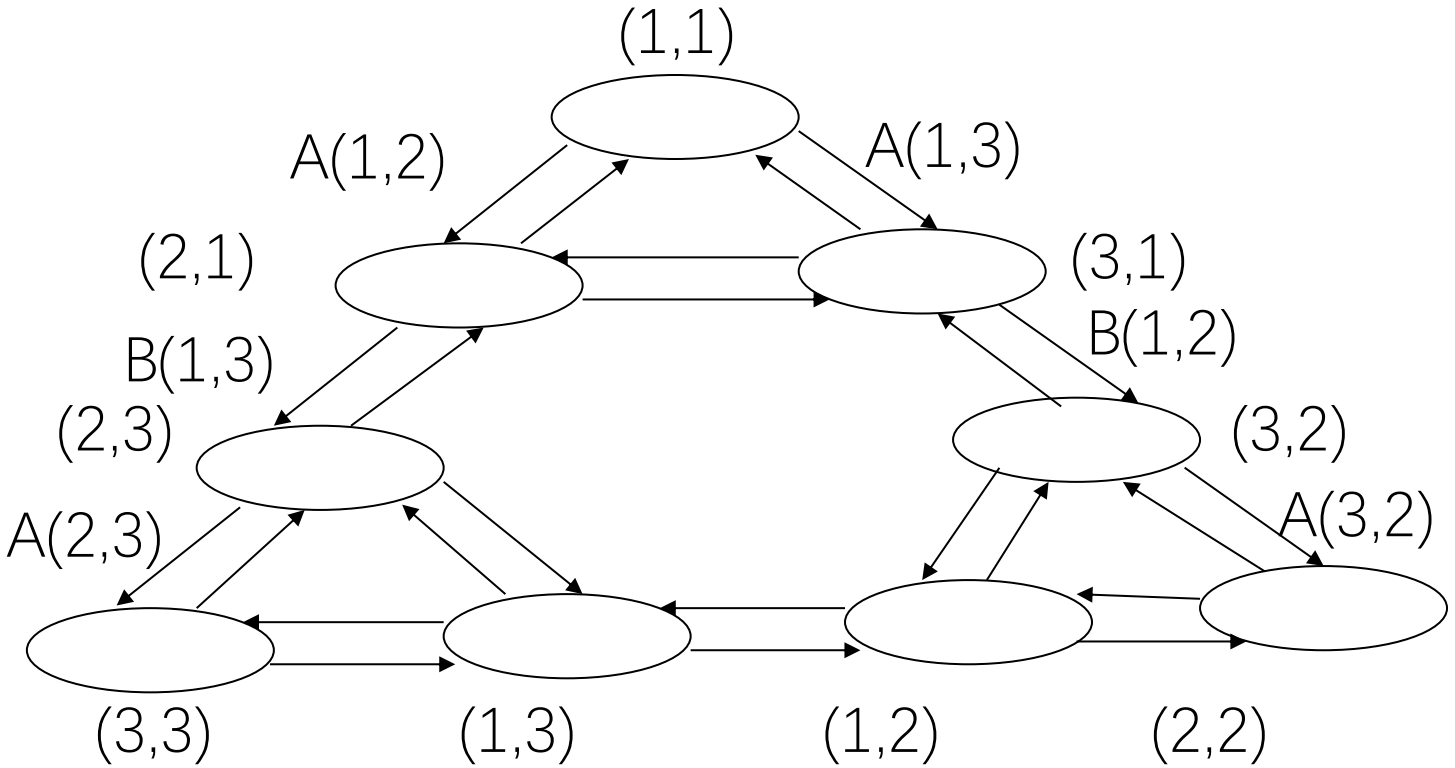
(c)目标状态S_8

问题的初始状态集合为S={S_0}，目标状态集合为G={S_4, S_8}。操作分别用A(i, j)和B(i, j)表示。其中A(i, j)表示把圆盘A从第i号柱子移到j号柱子上；B(i, j)表示把圆盘B从第i号柱子移到第j号柱子上。共有12种操作，它们分别是：

A(1, 2)　A(1, 3)　A(2, 1)　A(2, 3)　A(3, 1)　A(3, 2)

B(1, 2)　B(1, 3)　B(2, 1)　B(2, 3)　B(3, 1)　B(3, 2)

根据上述9种可能的状态和12种操作，可构成二阶梵塔问题的状态空间图，如下图所示。



从初始节点(1, 1)到目标节点(2, 2)及(3, 3)的任何一条路径都是问题的一个解。其中，最短的路径长度是3，它由3个操作组成。例如，从 (1, 1)开始，通过使用操作A(1, 3)、B(1, 2)及A(3, 2)，可到达 (2, 2)。

传教士和野人过河问题

- 有3个传教士和3个野人来到河边渡河，河岸有一条船，每次至多可供2人乘渡。任何时刻，在河两岸或者船上，野人数目如果超过传教士的数目，传教士有可能被野人吃掉。能不能找出一种安全的过河方法？

在线游戏：　https://novelgames.com/en/missionaries

传教士和野人

- 设定状态变量和值域
  - 设左岸的传教士人数为m，则 m = {0,1,2,3}；对应的右岸的传教士人数为3-m。
  - 设左岸的野人数为c,则c = {0,1,2,3}； 右岸的野人数为3-c 。
  - 左岸的船数为b ， b={0,1}， 右岸的船数为1-b。

- 左岸的状态，用三元组s = (m,c,b)表示， 右岸可以不标出。
- 初始状态：s0 = (3, 3, 1)
- 目标状态：sg = (0,0,0)

分析

- 根据要求，共得出以下5种可能的渡河方案：
  （1）渡2传教士
  （2）渡2野人
  （3）渡1野人1传教士
  （4）渡1传教士
  （5）渡1野人

操作符

- Pmc：从左岸划到右岸;
- Qmc：从右岸划到左岸;
- 船上人数组合（m，c）共5种（1，0），（1，1），（2，0），（0，1），（0，2）

∵每一种船上的人数组合可与P，Q二种操作对应，

∴系统共有5×2＝10种操作（规则）：

- F={P10,P20, P11, P01, P02, Q 10, Q 20, Q 11, Q 01, Q 02}

如：P10：if （M，C，B=1） then （M-1，C，B-1）

如果船在左岸，那么一个传教士划船到右岸

Q01：if （M，C，B=0） then （M，C+1，B+1）

如果船在右岸，那么一个野人划船回到左岸

分析

1. 首先可以划去左岸边野人数目超过传教士的情况，即S4、S8、S9、S20、S24、S25等6种状态是不合法的；
2. 应划去右岸边野人数目超过修道士的情况，即S6、S7、S11、S22、S23、S27等情况；
3. 应划去4种不可能出现状态：划去S15和S16——船不可能停靠在无人的岸边；划去S3——传教士不可能在数量占优势的野人眼皮底下把船安全地划回来；划去S28——传教士也不可能在数量占优势的野人眼皮底下把船安全地划向对岸。

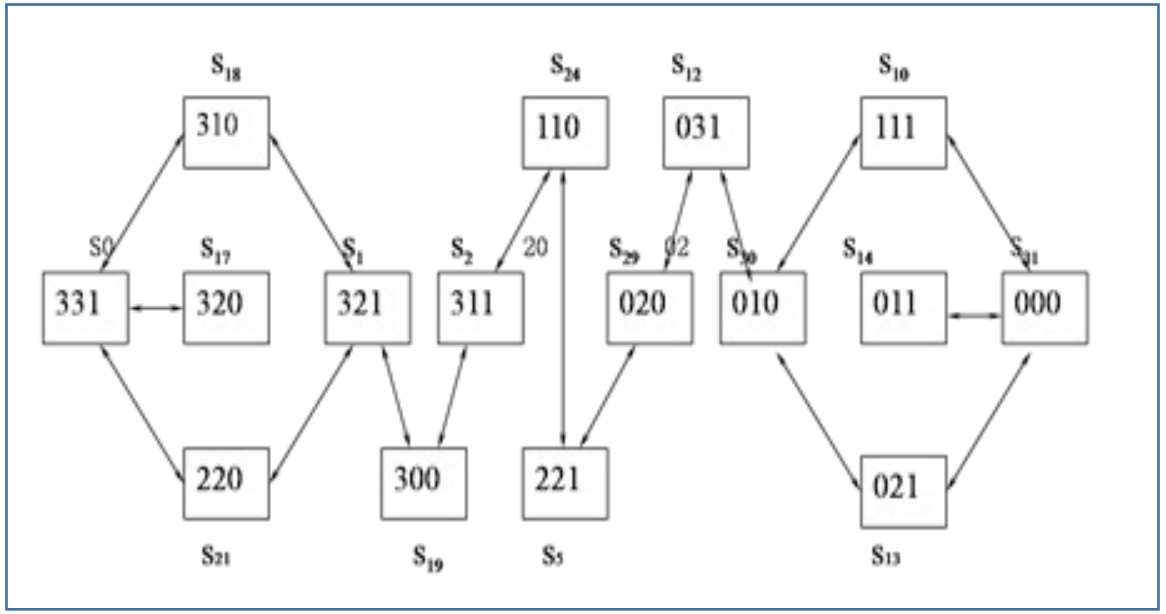
状态	m,c,b	状态	m,c,b	状态	m,c,b	状态	m,c,b
S_0	3 3 1	S8	1 3 1	S16	3 3 0	S24	1 3 0
S_1	3 2 1	S9	1 2 1	S17	3 2 0	S25	1 2 0
S2	3 1 1	S10	1 1 1	S18	3 1 0	S26	1 1 0
S3	3 0 1	S11	1 0 1	S19	3 0 0	S27	1 0 0
S4	2 3 1	S12	0 3 1	S20	2 3 0	S28	0 3 0
S5	2 2 1	S13	0 2 1	S21	2 2 0	S29	0 2 0
S6	2 1 1	S14	0 1 1	S22	2 1 0	S_{30}	0 1 0
S7	2 0 1	S15	0 0 1	S23	2 0 0	S_{31}	0 0 0

表1 传教士和野人问题的全部可能状态

状态空间图

- 当状态数量不是很大时，按问题的有序元组画出状态空间图，依照状态空间图搜索求解。
- 上述分析，共有16个合法状态和允许的操作，可以划出传教士和食人者问题的状态空间图，如图所示。
- 任何一条从S0到达S31的路径都是该问题的解。

控制策略



何一条从S0到达S31的路径都是该问题的解。

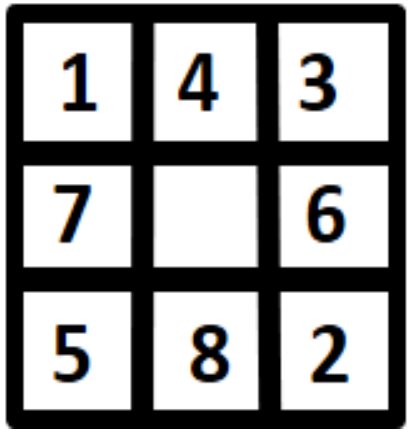
最短路径有4条，由11次操作构成。

- (P11、Q10、P02、Q01、P20、Q11、P20、Q01、P02、Q01、P02)
- (P11、Q10、P02、Q01、P20、Q11、P20、Q01、P02、Q10、P11)
- (P02、Q01、P02、Q01、P20、Q11、P20、Q01、P02、Q01、P02)
- (P02、Q01、P02、Q01、P20、Q11、P20、Q01、P02、Q10、P11)

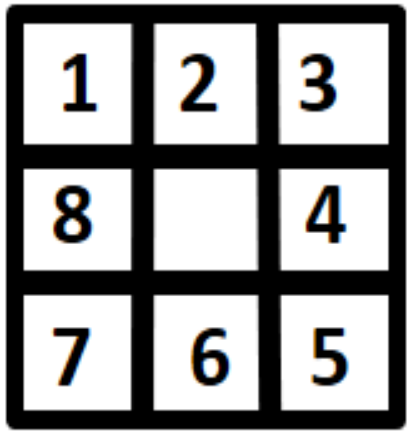
例2.1 八数码问题

八数码问题又称为重排九宫问题、数字华容道。

首先，需要定义八数码问题的状态集合。



（a）初始状态



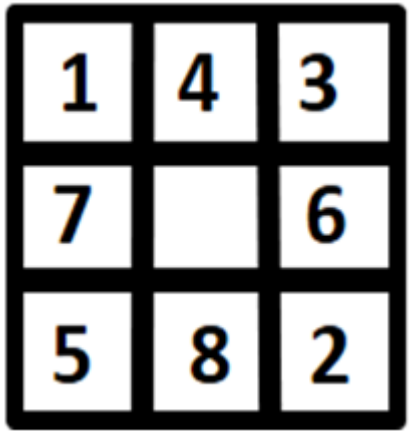
（b）目标状态

- 八个数码的任何一种摆法就是一个**状态**。
- 八数码的所有摆法构成了状态集合S，它们构成了一个**状态空间**.
- 这个状态空间中一共有？个状态。

例2.1 八数码问题

然后，设计操作集合：

将移动空格作为操作，即在方格盘上移动数码等价于移动空格。

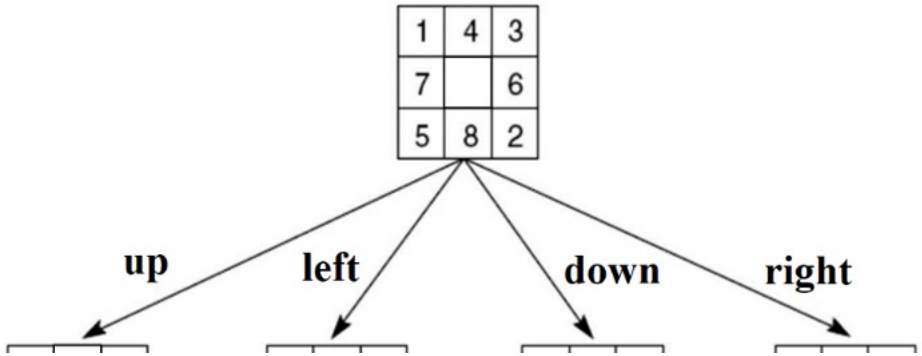


up: 将空格向上移，if 空格不在最上一行
down: 将空格向下移，if 空格不在最下一行
lelt: 将空格向左移，if 空格不在最左一列
right: 将空格向右移，if 空格不在最右一列

- 八数码问题的**解**就是一个使棋盘从初始状态变化到目标状态的数码牌移动序列。
- 显然，八数码问题的**解并不是唯一的**；
- 可以附加一些**约束条件**，例如要求找到一个移动数码牌次数最少的解。

随堂练习-画状态空间图

- 给定下列初始状态和目标状态，算符的移动顺序为
- up, left, down, right, 要求，画出2层状态空间图。



课后练习

给定下列初始状态和目标状态，通过空格移动，找出解的路径。

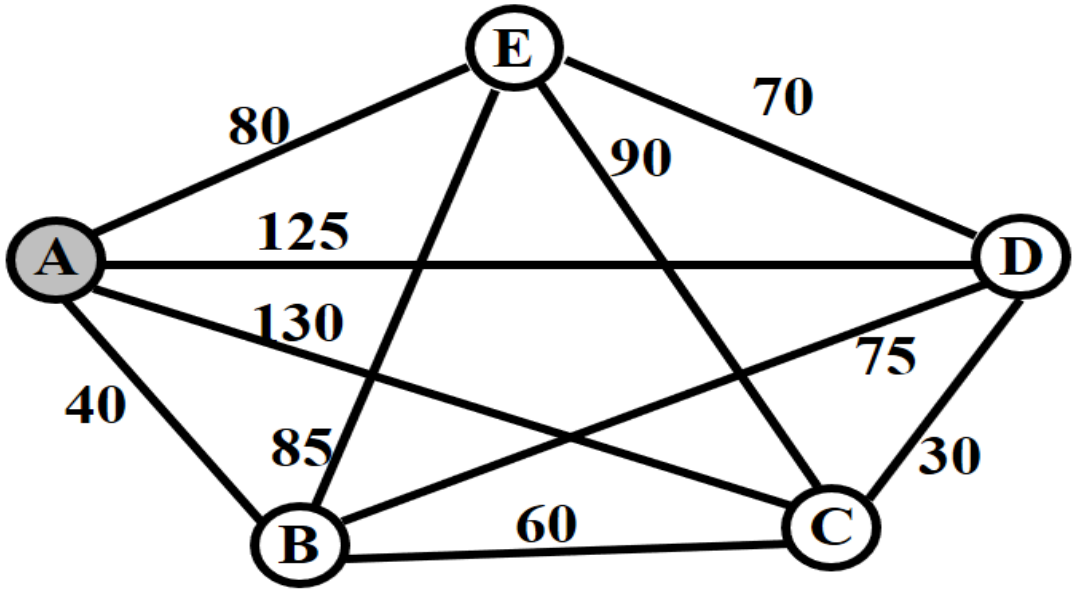
2		3
1	8	4
7	6	5

1	2	3
8		4
7	6	5

例2.2 旅行商问题

- 也称为**旅行推销员问题**。一个推销员要到N个城市去推销产品，已知每对城市之间的距离，他从一个城市出发，访问所有城市后，回到出发地。
- 除了出发地，要求每个城市仅经过一次。
- 所要求解的问题是：应该如何设计一条行进路线，才能使得推销员**访问每座城市所经过的路径最短或者费用最少**？
- 旅行商问题实质是：在一个带有权重的、含有N个节点的完全无向图中，找一个权值最小的**哈密尔顿（Hamilton）回路**。

哈密顿回路的定义： G=(V,E)是一个图，若G中一条路径通过且仅通过每一个顶点一次，称这条路径为哈密顿路径。若G中一个回路通过且仅通过每一个顶点一次，称这个环为哈密顿回路。若一个图存在哈密顿回路，就称为哈密顿图



2.4 状态空间表示法

- 对于大规模的问题，例如旅行商问题中有100个城市，要在有限时间内画出其全部状态空间图，是不可能的。
- 对于**简单问题**，可以采用有向图**直接画出状态空间**。
- 对于大多数**复杂的问题**，根本**无法完全画出其状态空间**，此时**只需清晰地定义状态变换的方式即可**，如图**2.6**。

图2.6 旅行商问题的部分状态空间