

T01 Search and Game Tree Search

17341015 陈鸿峥

2019 年 9 月 24 日

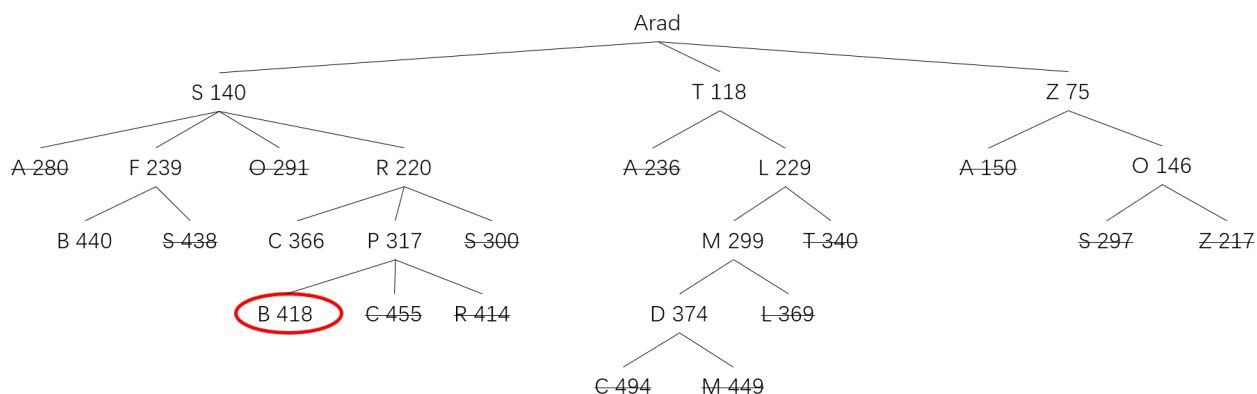
目录

1	Q1 - 环检测算法	2
2	Q2 - A^* 算法	2
3	Q3 - $\alpha - \beta$ 剪枝	4

1 Q1 - 环检测算法

问题 1. 使用带环检测的一致代价搜索寻找从 *Arad* 出发去 *Bucharest* 的路径，并作出搜索树。

解答. 如下图所示，环检测部分已经用删除线标出。注意每次添加进边界集的顺序以字母为序，但是扩展结点以路径代价为序。



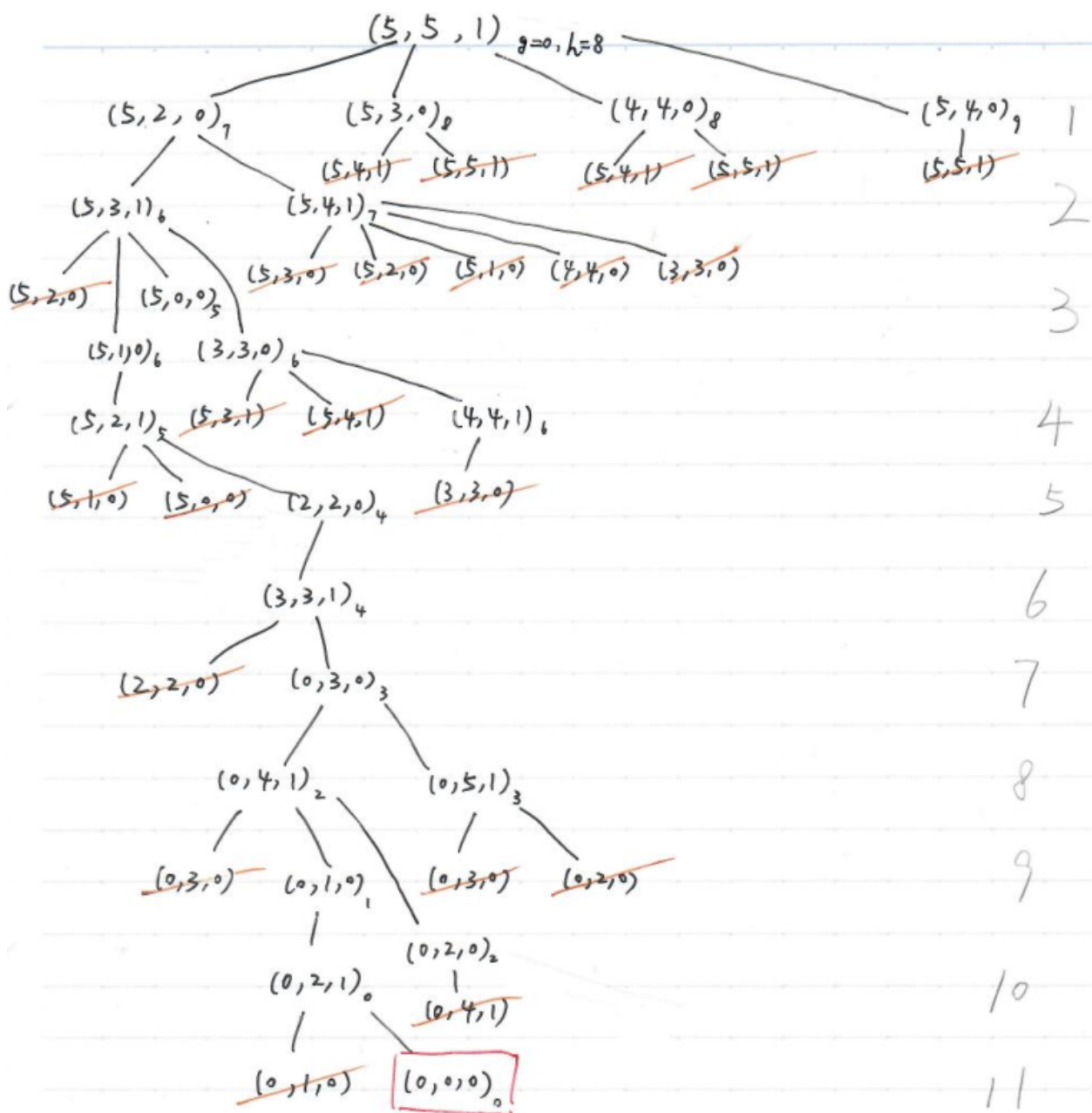
2 Q2 - A^* 算法

问题 2 (传教士与野人问题). 考虑 $M = 5$ 且 $K = 3$ 的情况，采用启发式函数 $h(n) = M + C - 2B$ 。记录带环检测的 A^* 算法的操作，作出搜索树；对于每一个结点，记录它的 g 和 h 值。

解答. 如下图所示， g 值用铅笔写在右侧， h 值写在每个元组的旁边。

$M=5 \quad K=3$

Date



由于整个过程非常繁复，因此我也编程进行实现验证结果，如下图所示，其中括号内内容为当前启发式函数 $f = g + h$ 的值， d 则表示被环检测检测出来需要被剪枝的节点。

```

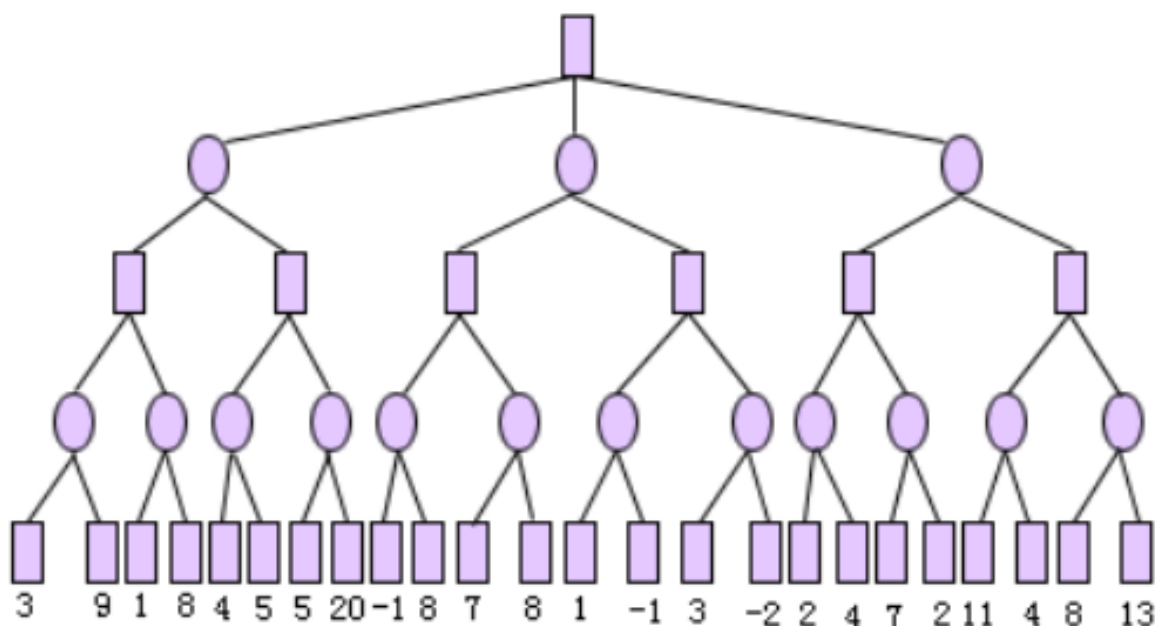
chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/ArtificialIntelligence/T01
chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/ArtificialIntelligence/T01$ python sol.py
5 5 1 (8)
  5 2 0 (8)
    5 3 1 (8)
      5 2 0 (d)
      5 0 0 (8)
    5 3 0 (9)
      5 4 1 (d)
      5 5 1 (d)
  4 4 0 (9)
    5 4 1 (d)
    5 5 1 (d)
    5 4 1 (9)
      5 3 0 (d)
      5 2 0 (d)
      5 1 0 (d)
      4 4 0 (d)
      3 3 0 (d)
      5 1 0 (9)
      3 3 0 (9)
        5 3 1 (d)
        5 4 1 (d)
        5 2 1 (9)
          5 1 0 (d)
          5 0 0 (d)
          2 2 0 (9)
  5 4 0 (10)
    5 5 1 (d)
      4 4 1 (10)
        3 3 0 (d)
          3 3 1 (10)
            2 2 0 (d)
            0 3 0 (10)
              0 4 1 (10)
                0 3 0 (d)
                0 1 0 (10)
                  0 2 1 (10)
                    0 1 0 (d)
              0 5 1 (11)
                0 3 0 (d)
                0 2 0 (d)
                0 2 0 (11)
                  0 4 1 (d)
                    0 0 0 (11)
11
chhzh123@DESKTOP-PV2UBJL: /mnt/d/Assignments/ArtificialIntelligence/T01$

```

可以看出结果确实正确，即使采用A*算法也需要经过11步才能找到最优解。

3 Q3 - $\alpha - \beta$ 剪枝

问题 3. 采用 $\alpha - \beta$ 剪枝对下面的博弈树进行搜索，计算出根节点的效用。



解答. 如下图所示, 剪枝部分已经用红色线标出。

