

# 一 八皇后问题

解法共 9 种，找出所有解的复杂度较低方法为状态空间法

问题分析 可以使用状态空间法进行求解 ( $N$  皇后)

定义解的形式为八元组  $V = \{x_1, \dots, x_N\}$ ，向量中的元素  $x_i$  代表放置在第  $i$  行的皇后的列位置

解对于整数  $i, j$  ( $1 \leq i < j \leq N$ ) 应满足约束：

$$\textcircled{1} \quad x_i \neq x_j$$

$$\textcircled{2} \quad |i-j| \neq |x_i - x_j|$$

求解过程

1 创造一个解集合  $S$  用于存放所有解，起初为空集

2 初始化空棋盘

3 假设正在处理第  $k$  列 ( $1 \leq k \leq N$ )，依次尝试将皇后放在  $(k, c)$  位置。检查其是否已经和前  $k-1$  行放置的皇后冲突。如果冲突，放弃该尝试  $c+1$ ；若安全，暂时认为其有效，返回的对第  $k+1$  行进行分析。

4 如果  $k$  行的全部  $N$  列都找不到安全位置，回退到  $k-1$  行，撤销这一行的皇后位置并尝试下一行。

5 当处理完全部  $N$  行且没有产生冲突，将找到的解  $\{x_1, x_2, \dots, x_N\}$  保存到  $S$  中，回溯寻找其他可行解。

6 当回溯退回第一行且这一行所有列都尝试完毕，返回  $S$ ，算法结束。

## A\* 算法

概念：一种典型启发式算法，用于在静态路网中求解最短路径

思想：采用估值函数  $f(n) = g(n) + h(n)$ ，使用了启发式算法思想，

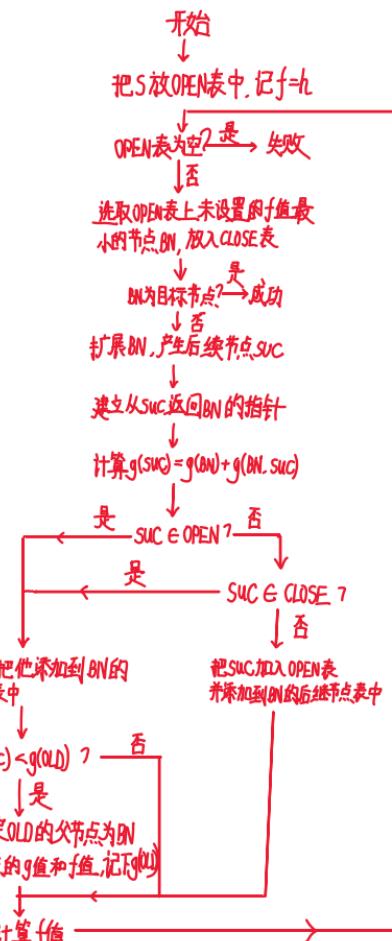
能够比 UCS 算法更高效地找到一个较好的解或者最优解

关系：当  $g(n)=0$  或  $h(n) \gg g(n)$ ，退化为贪婪最佳优先搜索。当  $h(n)=0$ ，退化为 UCS（一致代价搜索）

A\* 算法保证最优的要求： $h(n)$  满足可采纳性、一致性

# 求解流程

- (1) 把 S 放入 Open 表中, 记  $f=h$ , 令 Close 为 空集
- (2) 若 OPEN 为空表, 宣告失败
- (3) 选取 Open 表中未设置过的具有最小  $f$  值的节点, 为最佳节点 BN
- (4) 若 BN 为目标节点, 成功求解
- (5) 若 BN 不是目标节点, 扩展之, 产生后续节点 SUC
- (6) 对所有后续节点 SUC, 重复下列过程
  - (a) 建立从 SUC 返回 BN 的指针
  - (b) 计算  $g(SUC) = g(BN) + g(BN, SUC)$
  - (c) 如果  $SUC \in open$  表或  $SUC \in close$  表,  
称节点为 OLD, 将其添加到 BN 的后继节点表
  - (d) 比较新旧路径代价, 若  $g(SUC) < g(OLD)$ , 重新确定 OLD 的父节点为 BN, 记下较小代价  $g(OLD)$ , 修正父节点  $g$  和  $f$  的值, 否则停止扩展节点
  - (e) 若 SUC 既不在 OPEN 表中, 也不在 CLOSE 表中, 将其加入 OPEN 表, 并加入 BN 后继节点表, 然后转向(1)
- (7) 计算  $f$  值, 回到(2)



图搜索要维护一个 OPEN 表和 CLOSE 表, 前者保存已被发现但是未尚扩展的点, 后者保存已经扩展完的点。

# 图搜索算法

状态空间图大小  $M+E$

( $d$ 为目标节点所在最浅深度)

$b$ 为分支因子

$M$ 状态空间中任意路径最大长度

BFS: 时间复杂度  $O(b^d)$ 、空间复杂度  $O(b^d)$

完备性 ✓ 最优性 ✓

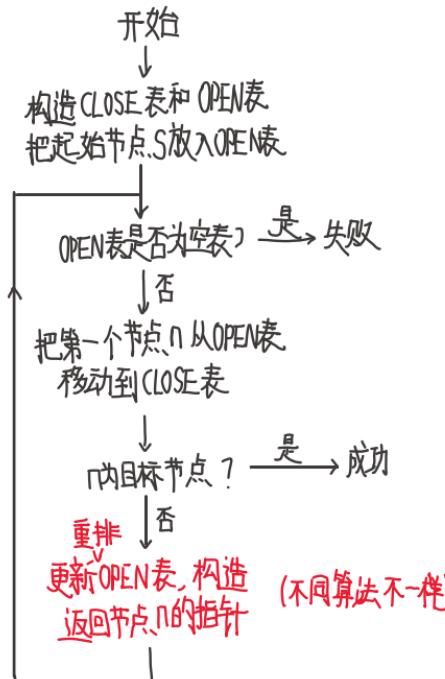
DFS 时间复杂度  $O(b^m)$ 、空间复杂度  $O(b^m)$

完备性: 图完备、树搜索不完备  
不是最优

有界DFS 引入深度探索界限 ( $d$ ) 时间复杂度  $O(b^d)$ 、空间复杂度  $O(b^d)$

↳ ( $d < d$ ) 不完备    ( $d > d$ ) 非最优

## 图搜索通用流程图



UCS算法 时间、空间复杂度  $O(b^{d+|C^*/E|})$

( $d$ 起点到终点最短路径)

最优解代价 ↓ 每一步代价

最优性 ✓  
完备性 没有负权代价时完备

Dijkstra 求一个点到其它所有点最短路径

## 启发式算法

$f(n)$ : 评价函数 / 估价函数，从起始点经过节点  $n$  到目标点的代价估计

$g(n)$ : 从起始点到节点  $n$  的实际代价       $h(n)$ : 从节点  $n$  到目标点的代价估计值  
↓ 越小越接近目标

$$f(n) = g(n) \rightarrow \text{VCS}$$

$f(n) = h(n) \rightarrow$  贪婪最佳优先搜索    对于  $n$  和其后续节点  $n'$  满足  
 $h(n) \leq c(n, n') + h(n')$

$$f(n) = g(n) + h(n) \rightarrow A\text{-算法}$$

$$f(n) = g(n) + h(n) \rightarrow A^*\text{-算法}$$

↓  
对启发函数  
无限制

↓  
启发函数必须满足“可采纳性”、“一致性”

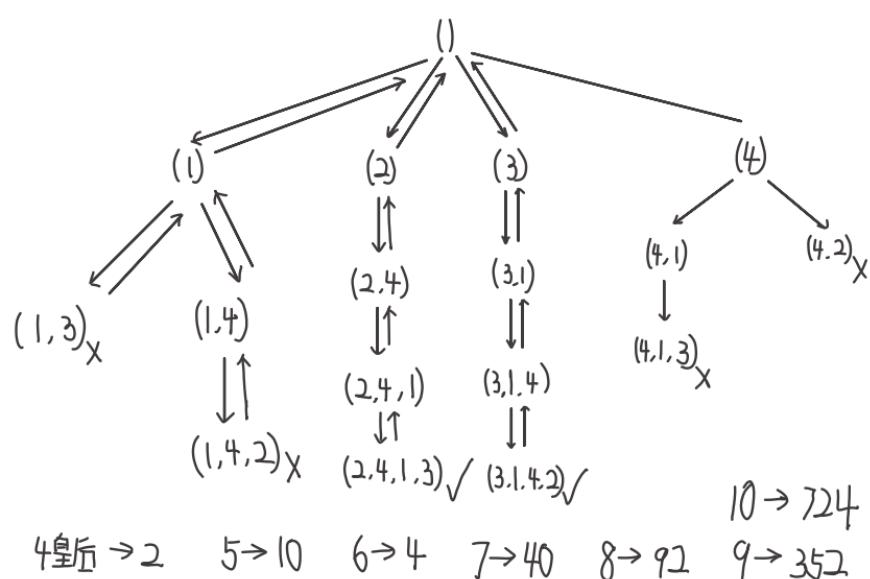
↓  
 $h(n)$  不大于实际代价  $h^*(n)$

贪婪算法：最差时退化为 DFS    不完备、不总最优    时间空间复杂度  $O(b^m)$

A 算法：不完备、不最优

A\* 算法    要求  $h(n)$  满足可采纳性    最优、完备    时间复杂度  $O(b^{ed})$     空间复杂度 指数级

本质上是基于优先级的 BFS



4皇后  $\rightarrow 2$

5  $\rightarrow 10$

6  $\rightarrow 4$

7  $\rightarrow 40$

8  $\rightarrow 91$

9  $\rightarrow 352$

$10 \rightarrow 724$

