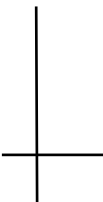




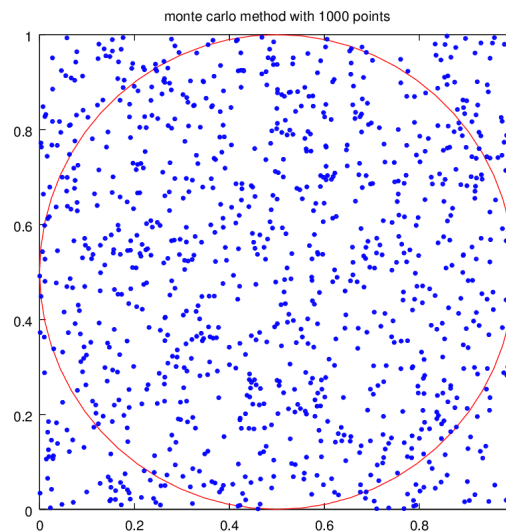
A Survey of Monte Carlo Tree Search Methods

雷炳杰



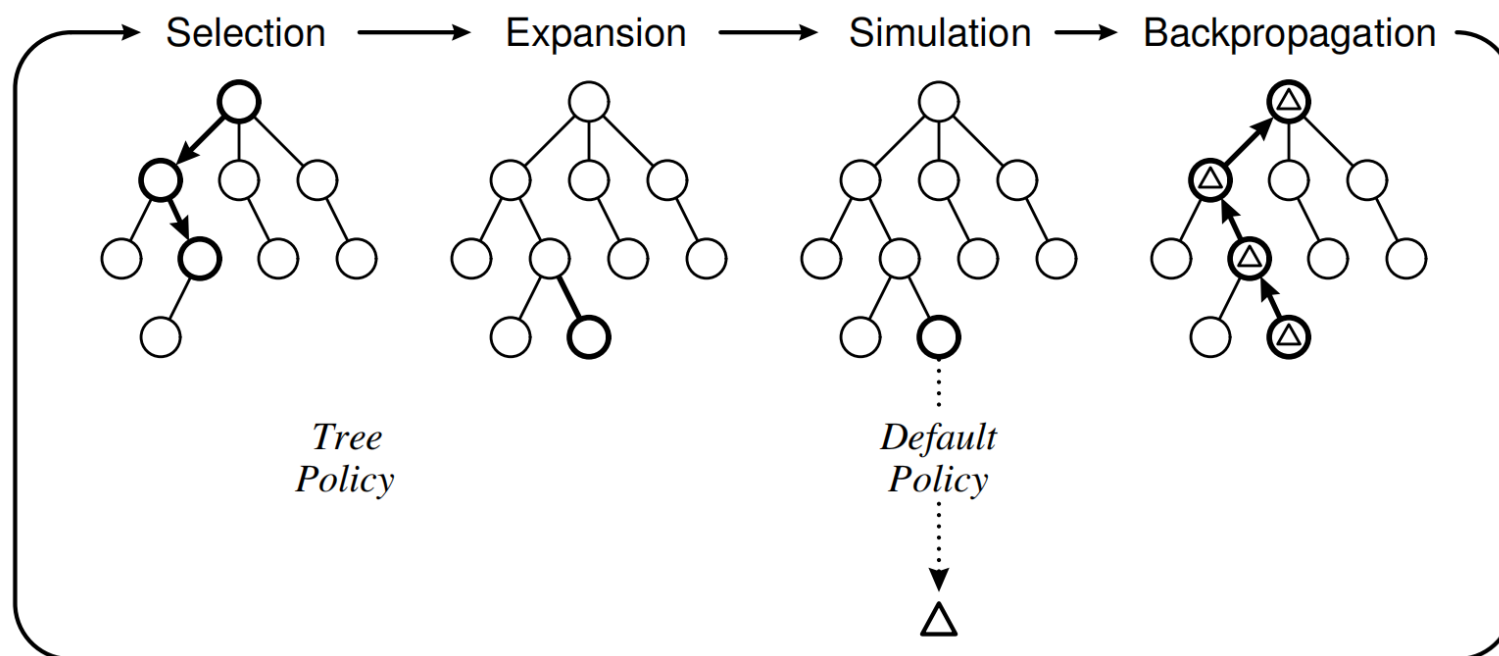
蒙特卡洛方法

- 统计模拟方法
- 一类是所求解的问题本身具有**内在的随机性**，借助计算机的运算能力可以直接模拟这种随机的过程。
- 另一类是所求解问题可以转化为**某种随机分布的特征数**



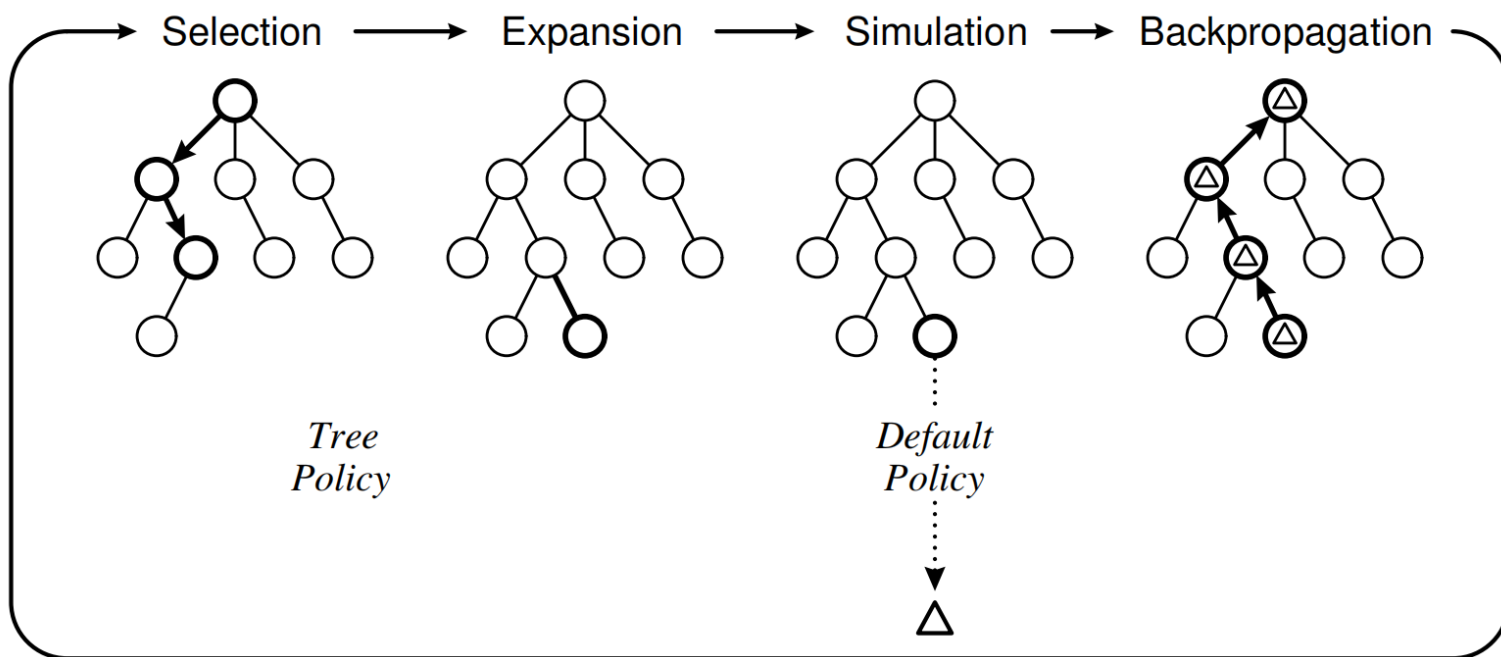
蒙特卡洛树搜索

- 蒙特卡洛树搜索是一种基于随机抽样的用于某些决策过程的启发式搜索算法。
- 蒙特卡洛树搜索的每个循环包含四个步骤：**选举**(selection)、**扩展**(expansion)、**模拟**(simulation)、**回溯**(BackPropagation)



蒙特卡洛树搜索:选择

- 从根结点R开始，选择连续的子结点向下至叶子结点L。下面的结点有更多选择子结点的方法，使游戏树向最优点扩展移动，这是蒙特卡洛树搜索的本质。



蒙特卡洛树搜索:选择 -- UCT算法

$$\frac{w_i}{n_i} + c * \sqrt{\frac{\ln t}{n_i}}$$

w_i : 第 i 次移动后取胜的次数;

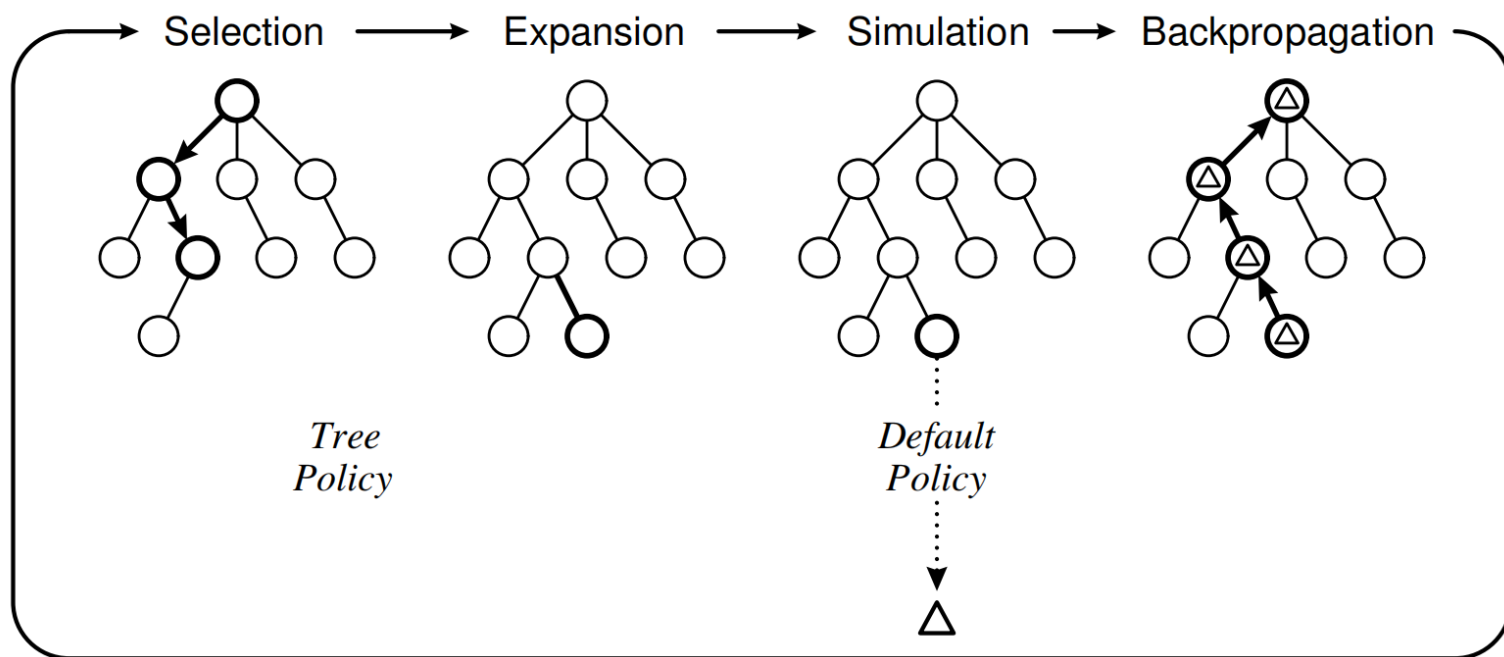
n_i : 第 i 次移动后仿真的次数;

c : 为探索参数, 理论上等于 $\sqrt{2}$; 在实际中通常可凭经验选择;

t : 代表仿真总次数, 等于所有 n_i 的和。

蒙特卡洛树搜索:扩展/模拟/回溯

- 除非任意一方的输赢导致游戏结束，否则L会创建一个或多个子结点或从结点C中选择。
- 在结点C中进行随机布局。
- 使用布局结果更新从C到R的路径上的结点信息。



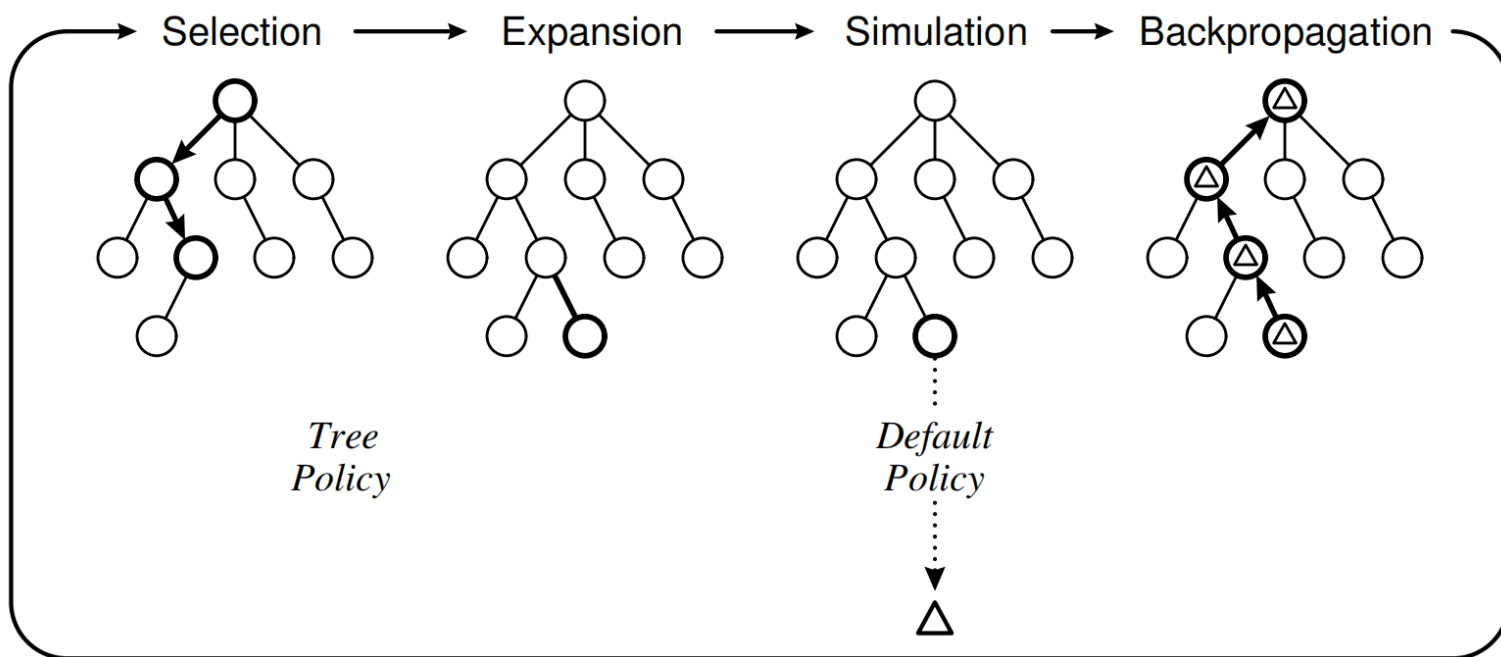
蒙特卡洛树搜索:两种策略学习机制

- 搜索树策略:

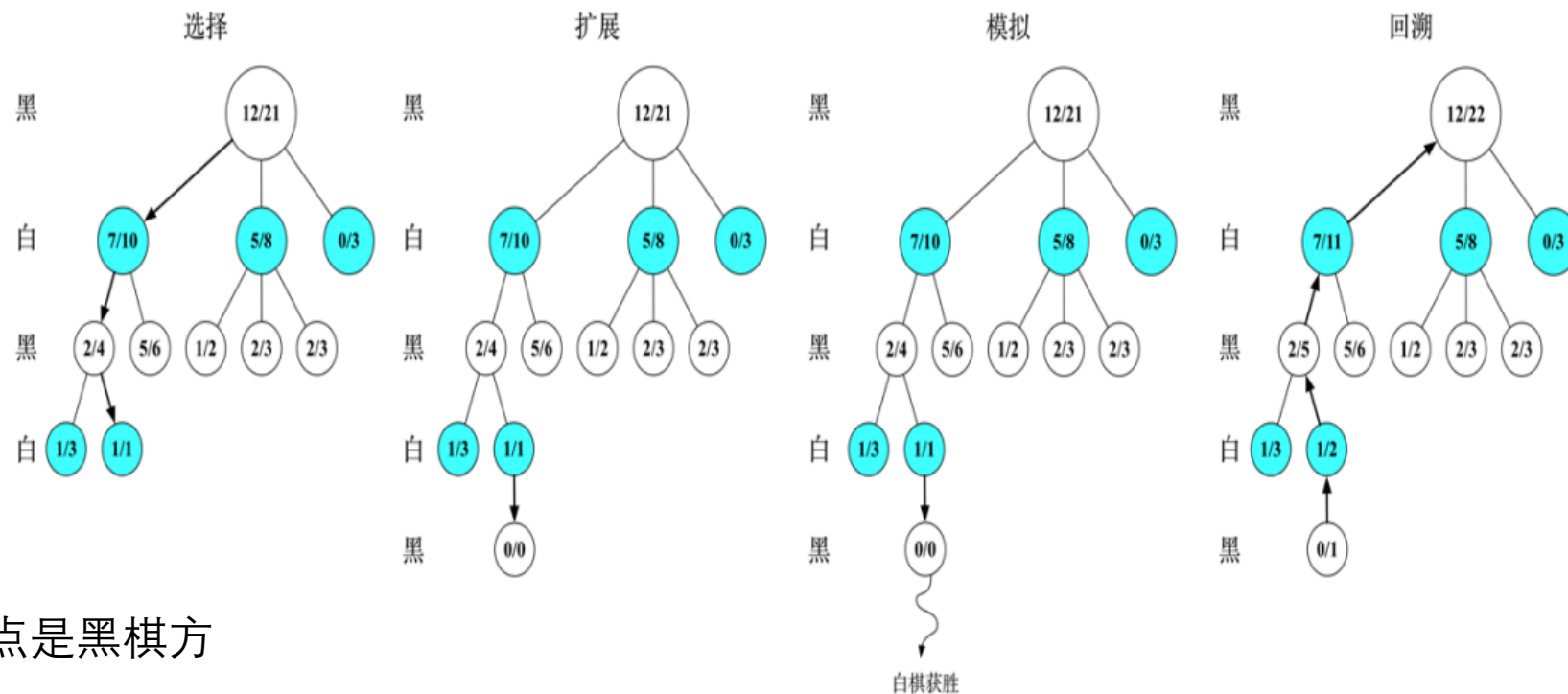
从已有的搜索树中选择或创建一个叶子结点。

搜索树策略需要在利用和探索之间保持平衡。

- 模拟策略：从非叶子结点出发模拟游戏，得到游戏仿真结果。



蒙特卡洛树搜索：实例



- 假设根节点是黑棋方
- 图中每一个节点都代表一个局面
- 每个局面记录两个值 A/B

A : 该局面被访问中黑棋胜利的次数

B : 该局面被访问的总次数

蒙特卡洛树搜索: 选择实例

- 在节点12/21, 黑棋行棋, 评估该节点下面的3个局面

- 左一: 7/10 对应的局面奖赏值

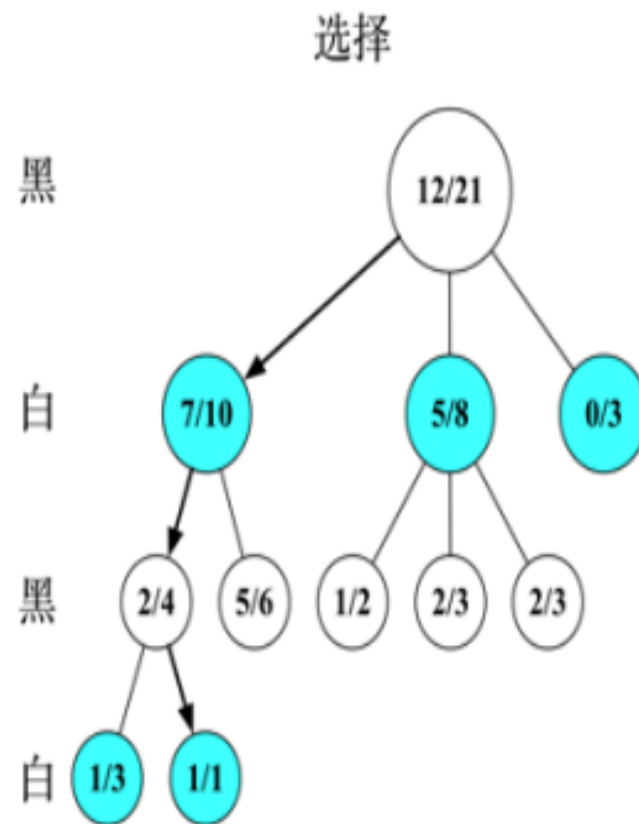
$$\frac{w_i}{n_i} + c^* \sqrt{\frac{\ln t}{n_i}} = \frac{7}{10} + \sqrt{\frac{\ln 21}{10}} = 1.252$$

- 左二: 5/8 对应的局面奖赏值

$$\frac{5}{8} + \sqrt{\frac{\ln 21}{8}} = 1.243$$

- 左三: 0/3 对应的局面奖赏值

$$\frac{0}{3} + \sqrt{\frac{\ln 21}{3}} = 1.007$$



蒙特卡洛树搜索: 选择实例

- 在节点 7/10, 白棋行棋, 评估该节点下面的2个局面

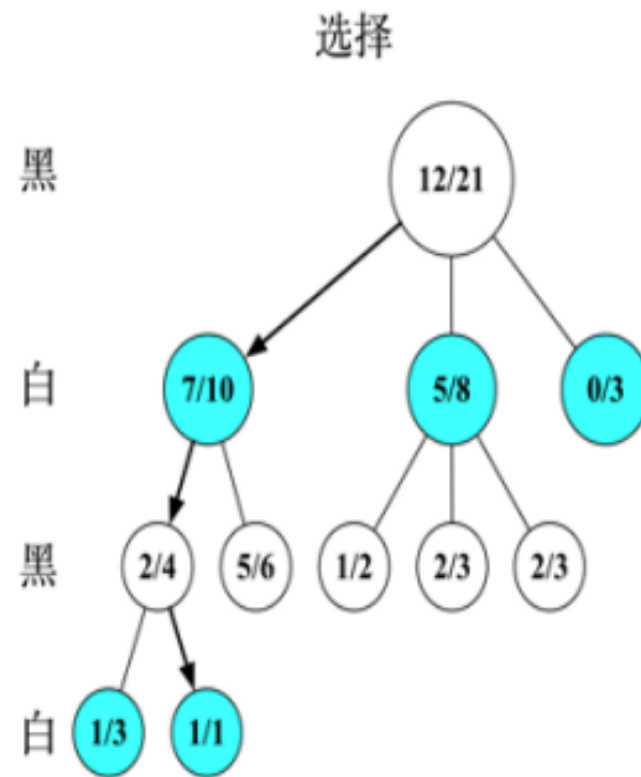
- 左一: 2/4 对应的局面奖赏值

$$\frac{w_i}{n_i} + c * \sqrt{\frac{\ln t}{n_i}} = 1 - \frac{2}{4} + \sqrt{\frac{\ln 10}{4}} = 1.26$$

- 左二: 5/8 对应的局面奖赏值

$$1 - \frac{5}{6} + \sqrt{\frac{\ln 10}{6}} = 0.786$$

- A : 白棋失败的次数



蒙特卡洛树搜索: 选择实例

- 在节点 2/4，黑棋行棋，评估该节点下面的2个局面

- 左一：1/3 对应的局面奖赏值

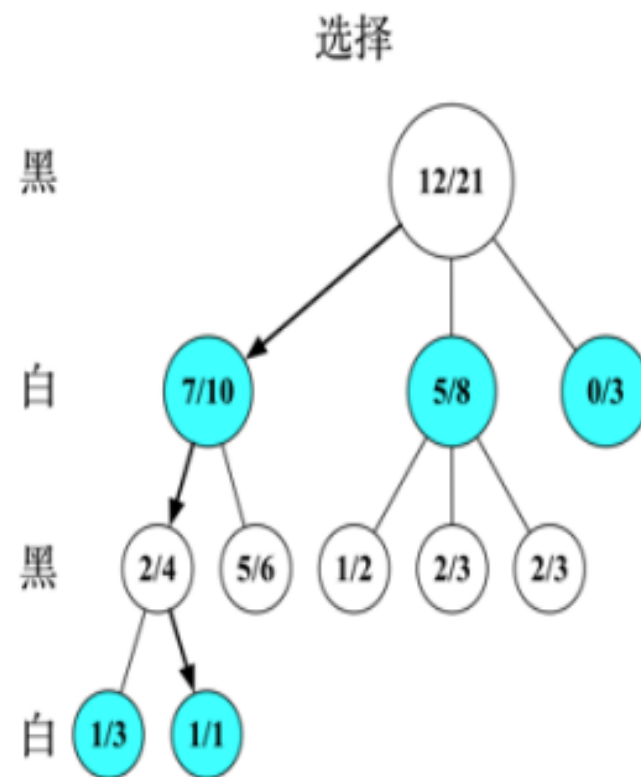
$$\frac{w_i}{n_i} + c * \sqrt{\frac{\ln t}{n_i}} = \frac{1}{3} + \sqrt{\frac{\ln 4}{3}} = 1.01$$

- 左二：5/8 对应的局面奖赏值

$$\frac{1}{1} + \sqrt{\frac{\ln 4}{1}} = 2.18$$

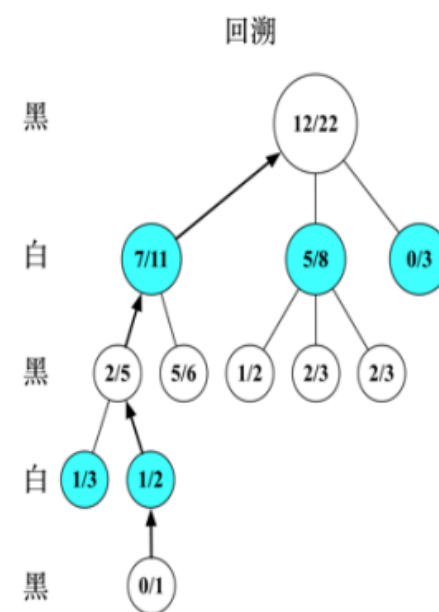
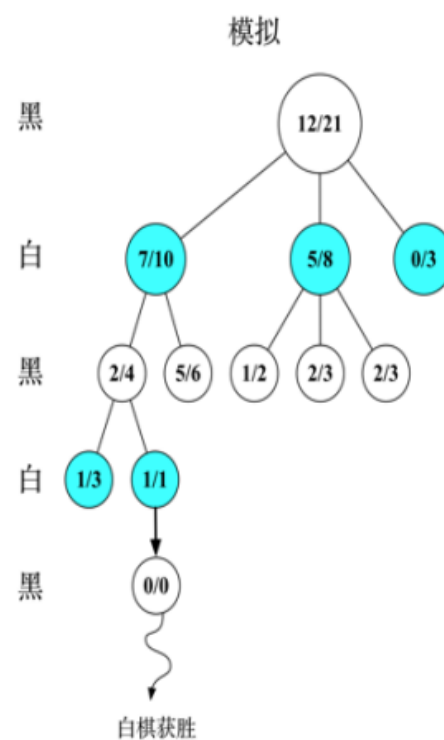
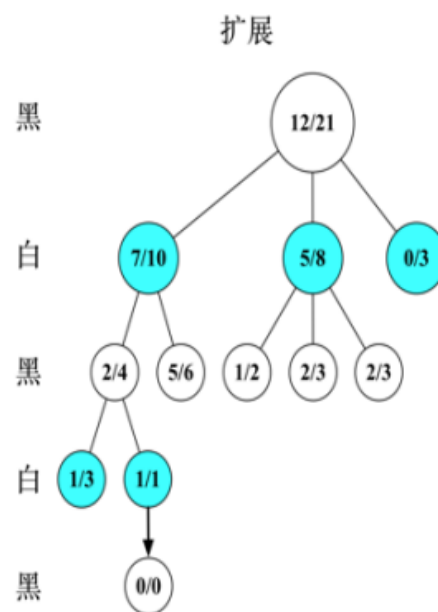
- 黑棋会选择局面 1/1 进行行棋

已经到达叶子节点，需要进行扩展



蒙特卡洛树搜索: 扩展/模拟/回溯实例

- 随机扩展一个新节点
新节点初始化为 0/0
- 在新节点进行模拟, 假设白棋获胜
- 更新仿真节点路径上
每个节点的 A/B 值
新节点的A/B 值被更新为0/1
所有父辈节点 A 不变, B值加1



蒙特卡洛树搜索具体步骤：

- 1.由当前局面建立根节点，生成根节点的全部子节点，分别进行模拟对局；
- 2.从根节点开始，进行最佳优先搜索；
- 3.利用 UCB 公式计算每个子节点的 UCB 值，选择最大值的子节点；
- 4.若此节点不是叶节点，则以此节点作为根节点，重复 2；
- 5.直到遇到叶节点，如果叶节点未曾经被模拟对局过，对这个叶节点模拟对局；否则为这个叶节点随机生成子节点，并进行模拟对局；
- 6.将模拟对局的收益（胜 1 负 0）按对应颜色更新该节点及各级祖先节点，同时增加该节点以上所有节点的访问次数；
- 7.回到 2，除非此轮搜索时间结束或者达到预设循环次数；
- 8.从当前局面的子节点中挑选平均收益最高的给出最佳着法。



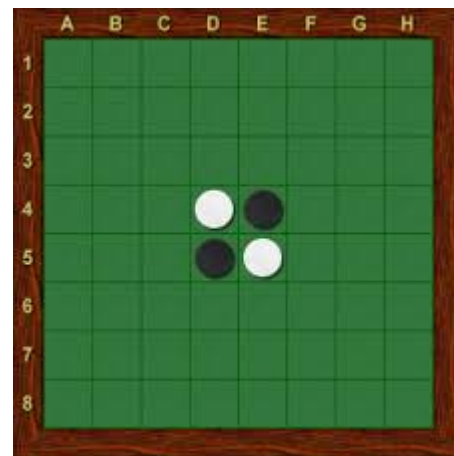
蒙特卡洛树搜索应用在游戏场景

- 信息对称：是指游戏的所有信息和状态都是所有玩家都可以观察到的
- 奖励和为0：最终状态胜利和失败的总奖励状态和为0（单独有正有负，总体上为0）
- 无可变影响：没有按照一定概率决定游戏发展方向的因素
- 动作按顺序：像下棋这样你一步我一步的，不存在同时启动等问题
- 动作离散



黑白棋规则

- 1. 黑方先行，双方交替下棋
- 2. 一步合法的棋步包括：
 - 在一个空格落下一个棋子，并且翻转对手一个或多个棋子
 - 对方被夹住的所有棋子都要翻转过来
 - 可以是横着夹，竖着夹，或是斜着夹
 - 夹住的位置上必须全部是对手的棋子，不能有空格
- 3. 如果一方没有合法棋步，那他只能弃权，而由他的对手继续落子直到他有合法棋步可下
- 4. 如果一方至少有一步合法棋步可下，他就必须落子，不得弃权。
- 5. 棋局持续下去，直到棋盘填满或者双方都无合法棋步可下。



传统黑白棋策略

- 贪心策略：每一步走子都选择使得棋盘上子最多的一步，而不考虑最终的胜负；
- 确定子策略：某些子一旦落子后就再也不会被翻回对方的子，最典型的是四个角上的子，这类子被称为确定子(Stable Discs)。每一步走子都选择使得棋盘上己方的确定子最多的一部。
- 位置优先策略。考虑到角点的重要性，把棋盘上的每一个子都赋予一个优先级，每一步从可走子里选择优先级最高的一个子。
- 机动性策略(mobility)。黑白棋每一步的可走子都是有限的，机动性策略是指走子使得对手的可走子较少，从而逼迫对手不得不走出差的一步(bad move)，使得自己占据先机。
- 消失策略(evaporation, less is more)。在棋盘比试的前期，己方的子越少往往意味着局势更优。因此在前期可采用使己方的子更少的走子。

谢谢

