和人类下棋类似，AlphaGo在做决策前，需要在“大脑里”做预判，确保几步以后很可能会占优势。如果只根据当前格局做判断，不往前看，是很难获胜的。

AlphaGo 下棋非常“暴力”：每走一步棋之前，它先在“脑海里”模拟几千、几万局， 它可以预知它每一种动作带来的后果，对手最有可能做出的反应都在 AlphaGo 的算计之内。

**如何预判？**

MCTS的基本思想就是向前看，模拟未来可能发生的情况，从而找出当前最优的动作。AlphaGo每走一步棋，都要用MCTS做成千上万次模拟，从而判断出哪个动作的胜算最大。

**如何模型？**

做模拟的基本思想如下：

假设当前有三种看起来很好的动作。每次模拟的时候从三种动作中选出一种，然后将一局游戏进行到底，从而知晓胜负。重复成千上万次模拟，统计一下每种动作的胜负频率，发现三种动作胜率分别是48%、56%、52%。

那么AlphaGo应当执行第二种动作，因为它的胜算最大，此时第二步的移动即为算法真正下了下一步。

**模型流程：**

MCTS的每一次模拟选出一个动作a，执行这个动作，然后把一局游戏进行到底，用胜负来评价这个动作的好坏。

MCTS的每一次模拟分为四个步骤：选择(Selection)、扩展(Expansion)、求值(Evaluation)、回溯(Backup)。

1、选择：

第一步——选择——的目的就是找出胜算较高的动作，只搜索这些好的动作，忽略掉其他的动作。

指标：

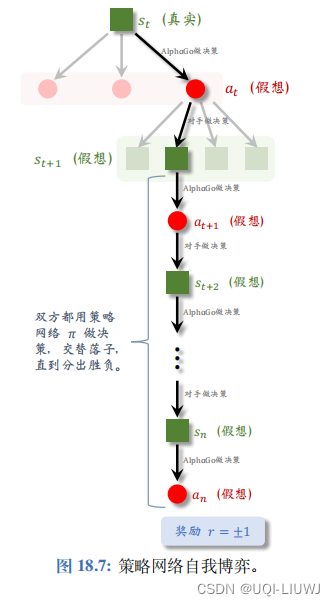
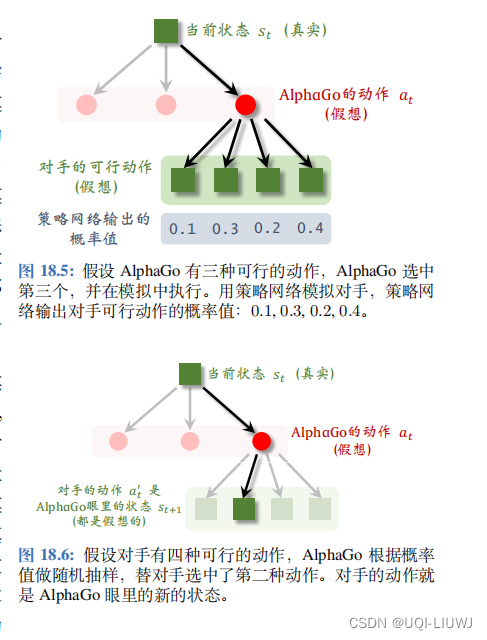
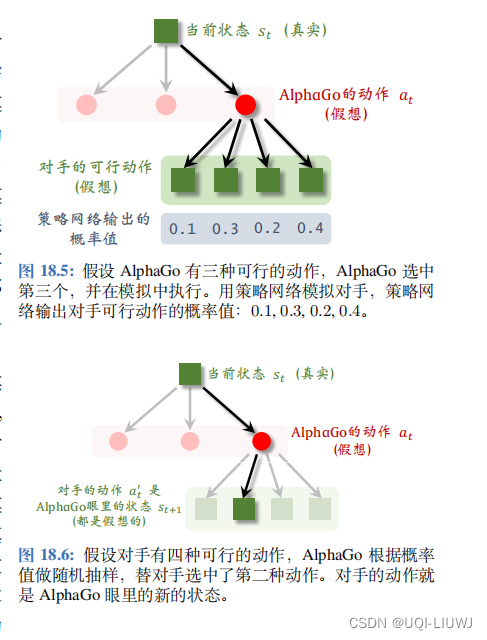
* η是个需要调的超参数
* N(a)是动作a已经被访问过的次数。
* 初始的时候，对于所有的a，令N(a)←0。
* 动作a每被选中一次，我们就把N(a)加一：N(a)←N(a)+1。
* Q(a)是之前N(a)次模拟算出来的动作价值，主要由胜率和价值函数决定。
* Q(a)的初始值是0；
* 动作a每被选中一次，就会更新一次Q(a)；

如何理解？

动作a选择次数低时，主要由策略网络起作用，鼓励探索次数少的选择

动作a选择次数多时，主要由动作价值起作用，鼓励选择探索后胜率高的

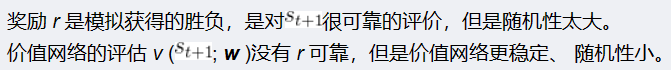
2、扩展：



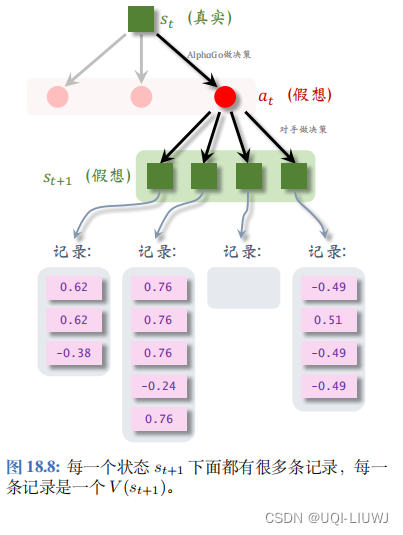
3、求值：（算出某一状态的价值）

AlphaGo的解决方案是把奖励r与价值网络的输出v(s_{t+1};**w**)取平均，作为对状态s_{t+1}的评价

https://img-blog.csdnimg.cn/fbf444f0997848fd92303db59a08055f.png



4、回溯：



第 t 步的动作 at 下面有多个可能的状态（子节点），每个状态下面有若干条记录。把 a t 下面所有的记录取平均，记作价值 Q( a t)，它可以反映出动作 a t 的好坏。

第 t 步的动作 at 下面有多个可能的状态（子节点），每个状态下面有若干条记录。把 a t 下面所有的记录取平均，记作价值 Q( a t)，它可以反映出动作 a t 的好坏。