如何采样——蒙特卡洛树搜索（MCTS）

不论是在 策略梯度算法 还是 近端策略优化 中，采样轨迹都是非常重要的一个任务。常规的采样方法往往存在效率低、利用率低、不符合实际目标的特点。因此，蒙特卡洛树搜索方法被提出用来解决上述问题。

Rollout

预演策略（Rollout）是一种非常朴素的想法，我们随机选择动作直到终止状态，并得到反馈的奖励信号。一次 预演 可以看作一次采样，这有点类似于人类的假设推理法（下棋时，人类可能会假想未来的局面）。常规的采样方法就是 大量的使用 Rollout，得到大量的轨迹，从而帮助我们发现合适的动作。这样做并不能很好地利用之前的采样信息，而且有大量无意义的采样。实际上，人类进行预演时往往是有倾向性的选择的，下棋高手往往可以根据过往的训练判断出哪些步骤可能会通往胜利，并对这些动作进行推演，以发现最合适的步骤，而不是完全随机。

MCTS

鉴于直接给出理论分析难于理解，我将直接介绍搜索过程，随后再从理论角度分析。

首先，我们有初始状态 s_0 以及 初始的一系列动作 a_1、a_2、 $\ldots$、a_m ， s_0 可以通过这 m 个动作转移到其他状态（这个转移是确定的，也就是转移概率为 1），因此我们也就可以绘制出初始树。

随后，我们从根节点开始不停地选择孩子结点，逐渐深入树，直到找到叶子结点为止。这里，我们并非随机选择，而是贪心选择（选择价值最高的孩子结点）或是根据 UCB 值选择。
结点的 UCB 值按如下公式计算：

$$ucb = \bar{v_i} + 2 * \sqrt{\frac{lnN}{n_i}}$$

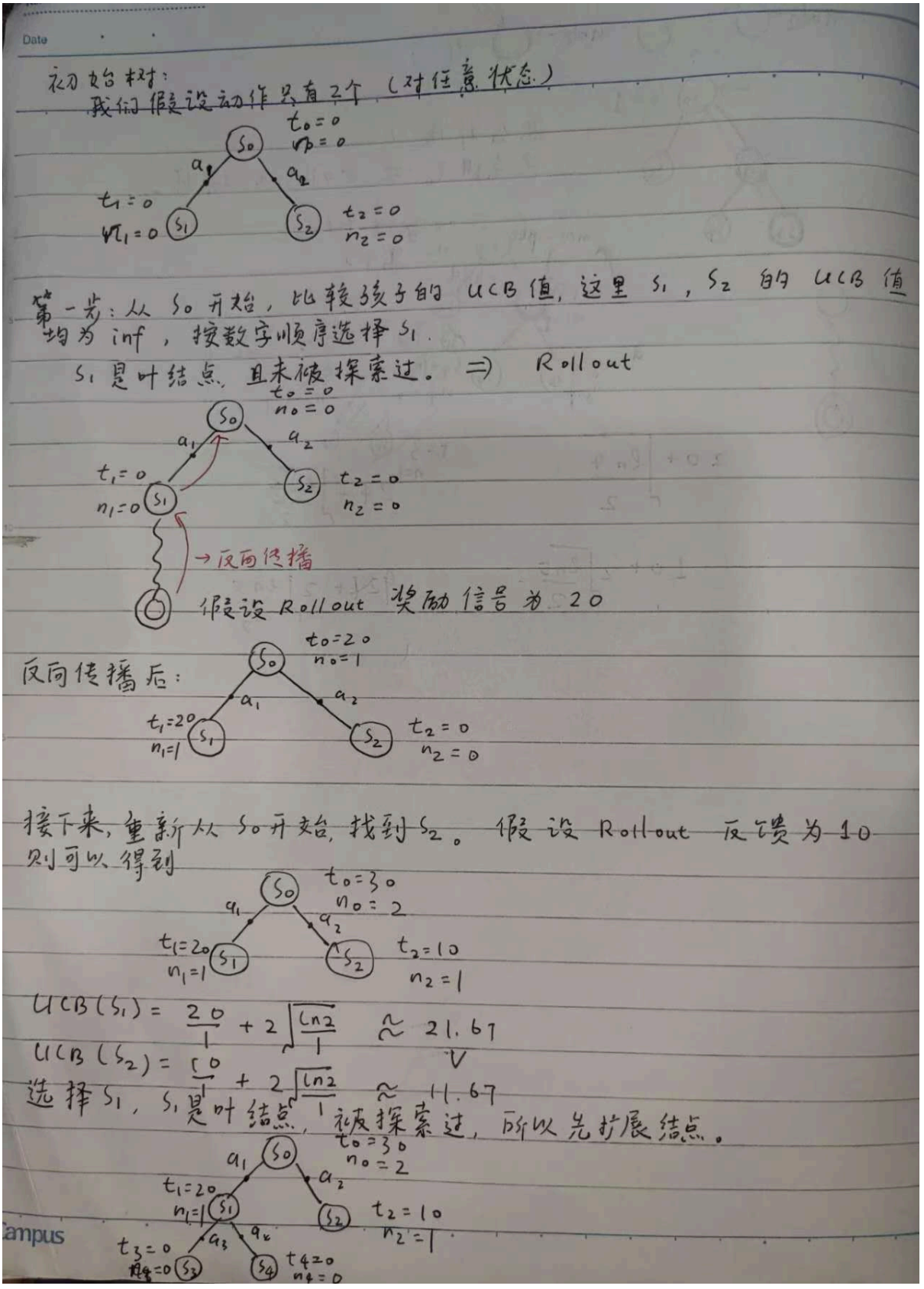
其中，$\bar{v_i}$ 表示结点 i 的平均价值（或得分），N 表示总共探索的次数（根节点被选择的次数），n_i 表示 结点 i 被选择的次数。
我们可以发现，$a)$ 结点被探索的次数越少，UCB 值也就越大；$b)$ 结点的平均价值越大，UCB 值越大。所以，我们往往倾向于选择 平均价值更高 或是 被探索次数很少的结点。当一个结点的平均价值很高时，我们不停地探索这个结点，增加了这个结点的探索次数，也就降低了这个结点被再次选择的可能性（探索次数越大，UCB 值越低）。而当其他价值更高的结点探索次数太大时，我们会去选择未被充分探索的结点。
这是一种启发式搜索方法，UCB 值平衡了探索（exploration） 和 利用（exploitation）。我们既考虑到过去的知识（结点的平均价值是根据过往的探索得到的），也充分考虑了对未知的探索（对于探索次数很少的结点，UCB 值可以很大（一次也没被探索过时，UCB 值无穷大））

找到叶子结点后，我们讨论该结点是否曾经被探索过，如果没有，我们对该叶子结点进行Rollout（预演操作），得到奖励信号；如果曾经被探索过，我们首先扩展结点，然后将扩展的结点中的某一个进行Rollout操作（通常默认第一个），得到奖励信号。

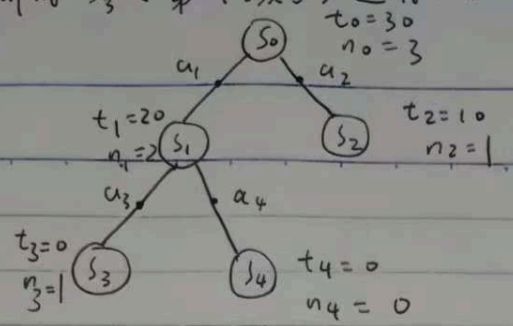
BackPropagation：每一次我们进行 预演 后，我们需要将奖励信号向前传播。

例子

我将给出一个实际的例子，便于理解。



立即对 s_3（第一个孩子）进行 Rollout 并反向传播（假设奖励信号为0）



t_i：所有被选过的奖励信号之和（针对结点 i 以及其子结点）

n_i：被选择次数