高级人工智能——行为主义复习

——陈若愚

——2022.1.3

1. 群体智能

群体智能的关键是交互。

1.1 蚁群算法 ACO (Ant Colony Optimization)

一种解空间的搜索方法（在所有解中找到一个最优的）

适用于图上寻找最优路径

形式化：

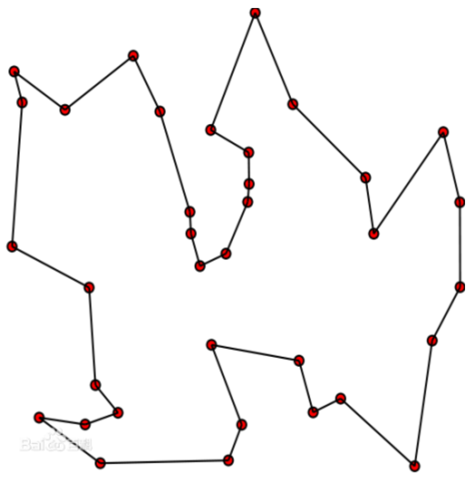
每个蚂蚁对应一个**计算智能体**

蚂蚁**按概率**选择候选位置

在经过的路径上留下**信息素**（Pheromone）

信息素**随时间挥发**

例题：使用蚁群算法解决旅行商问题。给定一系列城市和每对城市之间的距离，求解访问每一座城市一次并回到起始城市的最短回路。最终理想结果是如下图所示：



（随时间推移，蚂蚁一般都会选择最短路径，可能会出现一个连着的闭环，但不能保证，因为这是蚁群算法的局限性）

问题定义：

设总共用n个城市，城市用有向图$G=(V,E)$表示：

$$V=1,2,\dots,n \qquad\qquad E=(i,j)|i,j\in V$$

城市之间的距离体现知道，用$d_{i,j}$表示第i个城市和第j个城市之间的距离。

我们的目标函数为：

$$f(w)=\sum_{l=1}^{n} d_{i_l,i_{l+1}}$$

其中$w=(i_1,i_2,\dots,i_n)$，其中$i_{n+1}=i_1$。

参数描述：

将m个蚂蚁随机放置于n个城市（通常$m<n$），则城市i的第k个蚂蚁选择下一个城市j的概率为(*个人认为可能还有个条件，蚂蚁不可以选择原路返回*)：

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left(\tau_{ij}(t)\right)^{\alpha}\left(\eta_{ij}(t)\right)^{\beta}}{\sum_{k\in allowed}(\tau_{ik}(t))^{\alpha}(\eta_{ik}(t))^{\beta}} & j\in allowed\\ 0 \quad otherwise\end{cases}$$

其中τ_{ij}表示边(i,j)上信息素浓度，$\eta_{ij}(t)=1/d_{i,j}$，为先验。α和β为超参数，反应信息素和启发信息的相对重要性。

每一次蚂蚁周游后，更新信息素：

$$\Delta\tau_{ij}^{k}=f(x)=\begin{cases}\dfrac{Q}{L_k},(i,j)\in w_k\\ 0, \qquad otherwise\end{cases}$$

$$\Delta\tau_{ij} = \sum_{k=1}^{m} \Delta\tau_{ij}^{k}$$

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}$$

其中Q为常数，w_k表示第k只蚂蚁在本轮迭代中走过的路径，L_k为路径长度。ρ反应信息素挥发速度，通常小于 1。

算法流程：

```
⑴初始化 随机放置蚂蚁，
⑵迭代过程
k=1
while k=<ItCount do (执行迭代)
   for i = 1 to m do    (对 m 只蚂蚁循环)
     for j = 1 to n - 1 do    (对 n 个城市循环)
          根据式(1)，采用轮盘赌方法在窗口外选择下一个城市 j;
     end for
     将 j 置入禁忌表,蚂蚁 i 转移到 j;
   end for
   计算每只蚂蚁的路径长度;
   根据式(2)更新所有蚂蚁路径上的信息量;
   k = k + 1;
end while
⑶输出结果,结束算法.
```

缺点：

- 收敛速度慢
- 易陷入局部最优
- 对于连续的优化问题不适用

1.2 粒子群算法 PSO (Particle Swarm Optimization)

基于种群寻优的启发式搜索算法。

一种随机的优化方法，通过粒子群在解空间中进行搜索，寻找最优解。

构成要素：

粒子群：

  每个粒子对应所求解问题的一个可行解

  粒子通过其位置和速度表示：

    粒子i在第n轮的位置

    粒子i在第n轮的速度

记录：

  $p_{best}^{(i)}$粒子i历史最好位置

  g_{best}全局历史最好的位置

计算适应度函数$f(x)$

算法过程描述：

- **初始化**
  - **初始化粒子群：每个粒子的位置和速度，即$x_0^{(i)}$和$v_0^{(i)}$**
  - $p_{best}^{(i)}$**和**g_{best}

- **循环执行如下三步直至满足结束条件**
  - **计算每个粒子的适应度：$f\left(x_n^{(i)}\right)$**
  - **更新每个粒子历史最好适应度及其相应的位置，更新当前全局最好适应度及其相应的位置**
  - **更新每个粒子的速度和位置**

更新速度与位置：

$$v_{n+1}^{(i)} = v_n^{(i)} + c_1 * r_1 * \left(p_{best}^{(i)} - x_n^{(i)}\right) + c_2 * r_2 * \left(g_{best} - x_n^{(i)}\right)$$

$$x_{n+1}^{(i)} = x_n^{(i)} + v_{n+1}^{(i)}$$

Δt离散，为单位时间。r为随机参数，c为权重。

算法终止条件：

- 迭代的轮数
- 最佳位置连续未更新的轮数
- 适应度函数的值到达预期要求

优点

易于实现；

可调参数较少；

所需种群或微粒群规模较小；

计算效率高，收敛速度快。

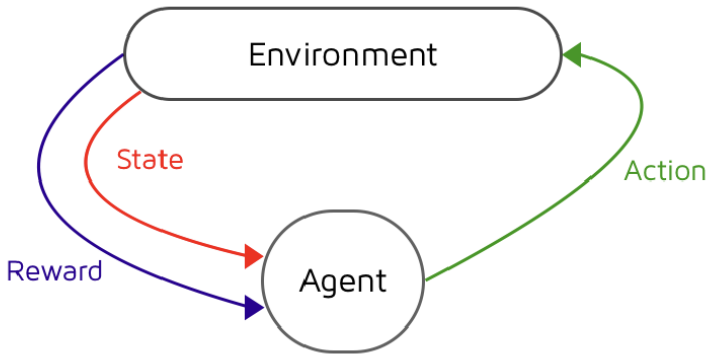
缺点

和其它演化计算算法类似，不保证收敛到全局最优解

2. 强化学习-贝尔曼方程

强化学习基本要素：

策略，奖励，价值，环境要素。



贝尔曼方程是一个状态估值函数。强化学习常见三种估值函数，状态估值，行为估值，和策略估值。贝尔曼方程属于状态估值。

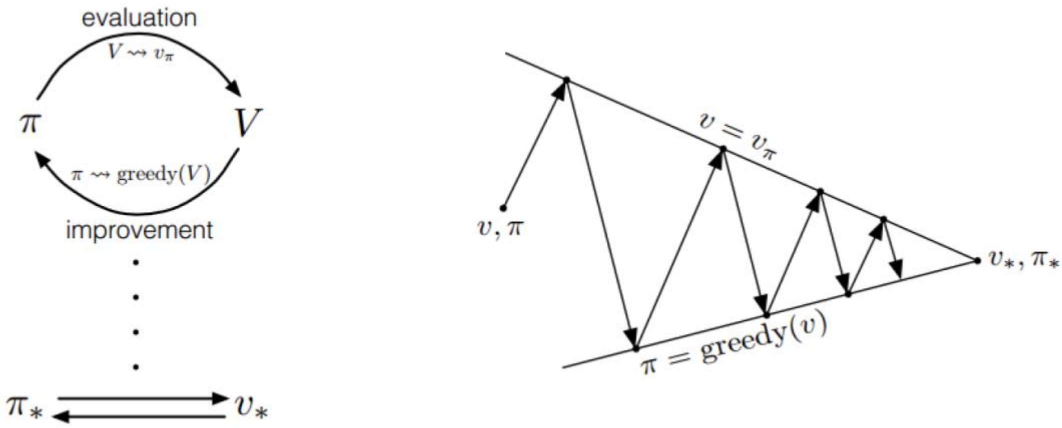
贝尔曼方程：

$$v_\pi(s) = \sum_a \pi(a|s) \sum_{s',r} p(s',r|s,a)[r + \gamma v_\pi(s')]$$

其中π为采取策略，例如格子游戏里面的上下左右，s为状态，格子游戏里面对应在哪个格子。$p(s',r|s,a)$为转移成功概率，给机器人指令向左，机器人可能向右边，但是格子游戏里面是确定的，所以概率为 1。r为这一步执行后的奖励，例如走一步-1 分。$v_\pi(s')$为转移状态的估值函数，γ为一个超参数，一般我们解题时候设置为 1 就好了。

如何运用贝尔曼方程解问题：

一般为动态规划方法，即广义的策略迭代方法：



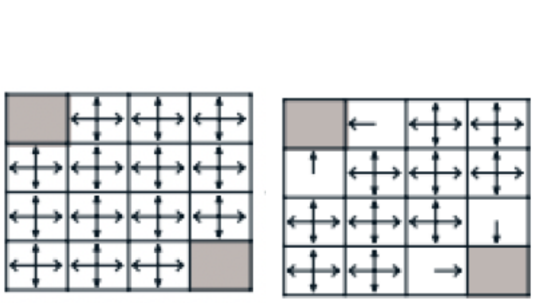
就是策略估值与策略提升两个交替，策略估值可以用**贝尔曼方程**去求解当前策略下的状态估值数，策略提升即用计算的估值去做策略调整，常用**贪心算法（朝着策略最高的状态移动）**。

策略估值：

$k = 1$

0.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	-1.0
-1.0	-1.0	-1.0	0.0

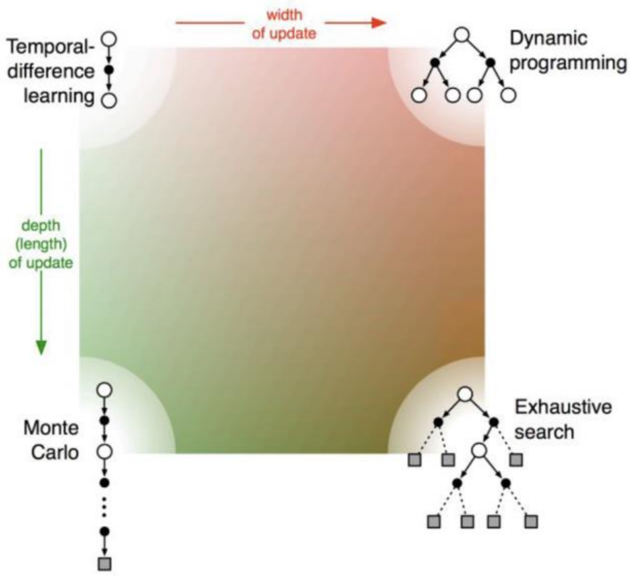
策略提升：



格子游戏问题的解题步骤就是如上（这里不写了，看看往年题），一般初始策略是上下左右都有，然后出口的奖励可以设置为 0。

注：强化学习通常有 4 种策略学习方法：动态规划，蒙特卡洛，时序差分和参数近似，虽然格子问题只能动态规划，但是生活中多用时序差分和参数近似。

时序差分，动态规划和蒙特卡洛的关系如下：



3. 博弈论

博弈论的要素：

局中人：在博弈中有权决定自己行动方案的博弈参加者

重要假设：局中人是自私的理性人

纳什均衡：如果一个局势下，每个局中人的策略都是相对其他局中人当前策略的最佳应对，则称该局势是一个纳什均衡，一个僵局。如果自己改变策略效用函数会下降。

混合策略：每个局中人以某个概率分布在其策略集合中选择策略

混合策略下的纳什均衡：给定其他局中人的策略选择概率分布的情况下，当前局中人**选择任意一个（纯）策略获得的期望效用相等**。

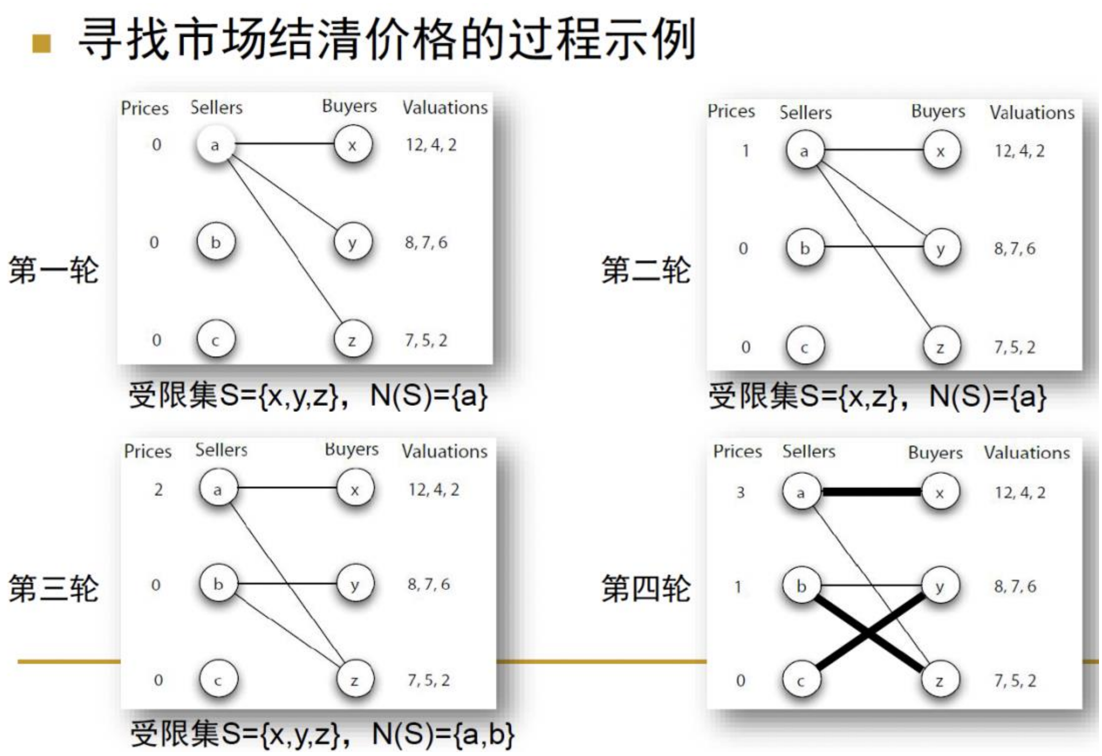
帕累托最优：不存在其他策略选择使**所有参与者**得到至少和目前一样高的回报，且至少一个参与者会得到严格较高的回报

社会最优：使参与者的回报之和最大的策略选择

Maxmin 策略：最大化自己最坏情况时的效用

minmax 策略：最小化对手的最大收益

匹配：

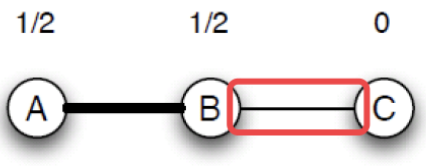


议价权，谁的议价最大：

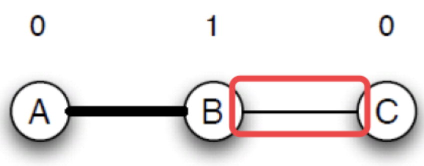
结局的稳定性，稳定结局

对于结局中**未参与配对的边**，如果边的两个端点获得的收益之和**小于1**，则称这条边为不稳定边。

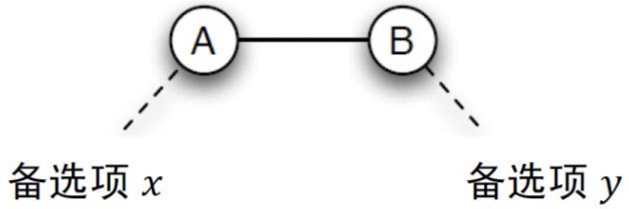
如果一个结局中**不存在不稳定边**，则称该结局为**稳定结局**。



不稳定的结局
B和C的收益之和小于1



稳定的结局



剩余价值：$s = 1 - x - y$

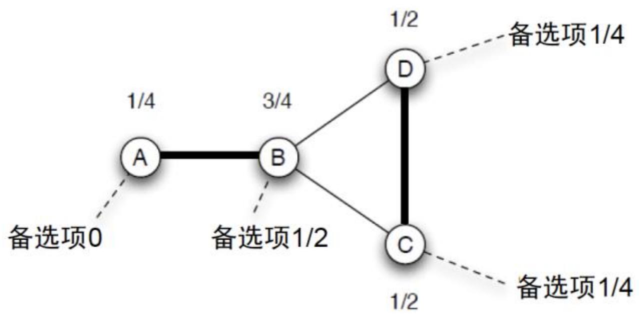
纳什议价解：

A 的收益：$x + \frac{s}{2} = \frac{1+x-y}{2}$

B 的收益：$y + \frac{s}{2} = \frac{1+y-x}{2}$

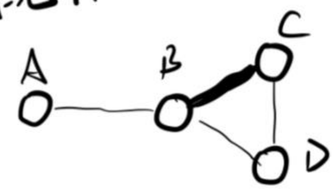
均衡结局：给定一个结局，如果结局中的任意一个参与配对的边都满足纳什议价解的条件，则称该结局是均衡结局。

计算均衡结局案例：



1. 先两两匹配

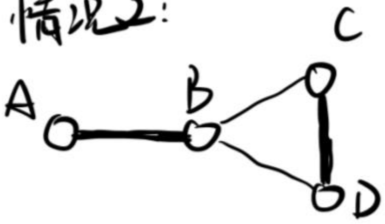
情况1:



此时 A、D收益为0，而 $B+C=1$

B或C必有1个小于1，而边AB与CD值必有1条小于1，不是稳定结局，不可行

情况2:



此时，匹配AB与CD

A备选项为0，B备选项为 $1-C$

C与D因对称，故收益必一致（证明略）为$\frac{1}{2}$

则B备选项为$\frac{1}{2}$，A收益：$\frac{1-\frac{1}{2}}{2}+0=\frac{1}{4}$

B收益：$\frac{1-\frac{1}{2}}{2}+\frac{1}{2}=\frac{3}{4}$

故 A $\frac{1}{4}$　B $\frac{3}{4}$　C $\frac{1}{2}$　D $\frac{1}{2}$