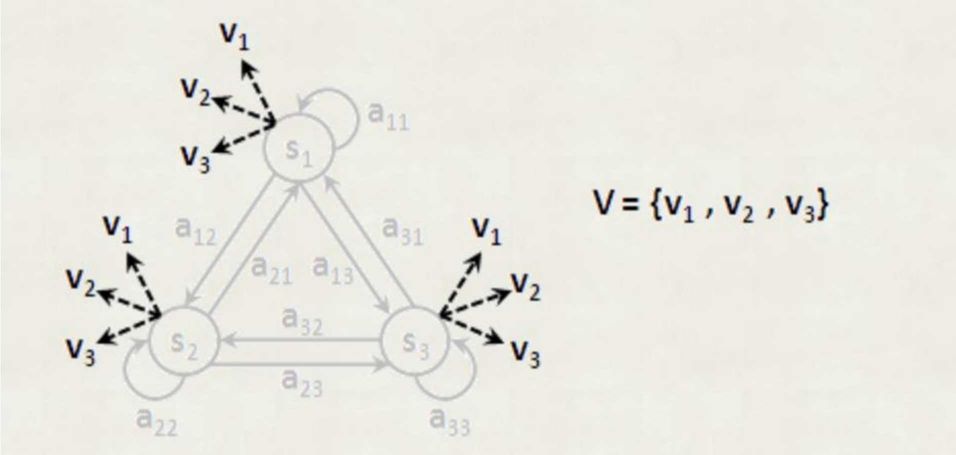
HMM模型介绍



模型的描述：

- 状态集合$S=\{s_1,s_2,...\}$，观察集合$O=\{o_1,o_2,...\}$
- 状态之间的转移矩阵a_{ij}，表示$P(s_t=a_j|s_{t-1}=a_i)$，s_i表示i时刻的状态，是一个随机变量
- 每一个状态对应观察的分布：$b_{ik}=P(o=o_k|s=s_i)$，表示在状态是s_i的情况下获得观察o_k的概率。

HMM模型介绍——问题1：endcodeing

问题：

(1)看到一个观察序列$o_1,...,o_T$，但是看不到状态序列$s_1,s_2...,s_T$，找出所有可能路径的概率总和？（初始状态的分布$\{\pi_1,...,\pi_n\}$）

其实这个问题没什么好说的，就是一个穷举，意思是T步中每一步都有可能是所有的状态，因此我们得先考虑一个有限的状态集合$\{s_1,s_2,...,s_n\}$，于是一共有n^T种状态的序列，然后对于每一个序列$\{s_{i_1},...,s_{i_T}\}$，$i_j\in\{1,...,n\}$，概率是：

$$
\begin{aligned}
P &= \pi_{i_1} * b_{i_1,o_1} * a_{i_1,i_2} * \cdots * a_{i_{T-1},i_T} * b_{i_T,o_T} \\
&= \pi_{i_1} b_{i_T,o_T} \prod_{k=1}^{T-1} a_{i_k,i_{k+1}} b_{i_k,o_k}
\end{aligned}
$$

于是总概率就是：

$$
P = \sum_{i_1,...,i_T\in\{1,...,n\}} \pi_{i_1} b_{i_T,o_T} \prod_{k=1}^{T-1} a_{i_k,i_{k+1}} b_{i_k,o_k}
$$

好像并没有什么启发性，看着也没有什么特别的。

HMM模型介绍——问题1：endcodeing

结果并不好看!

Proposition 1 (结果的丑陋性).

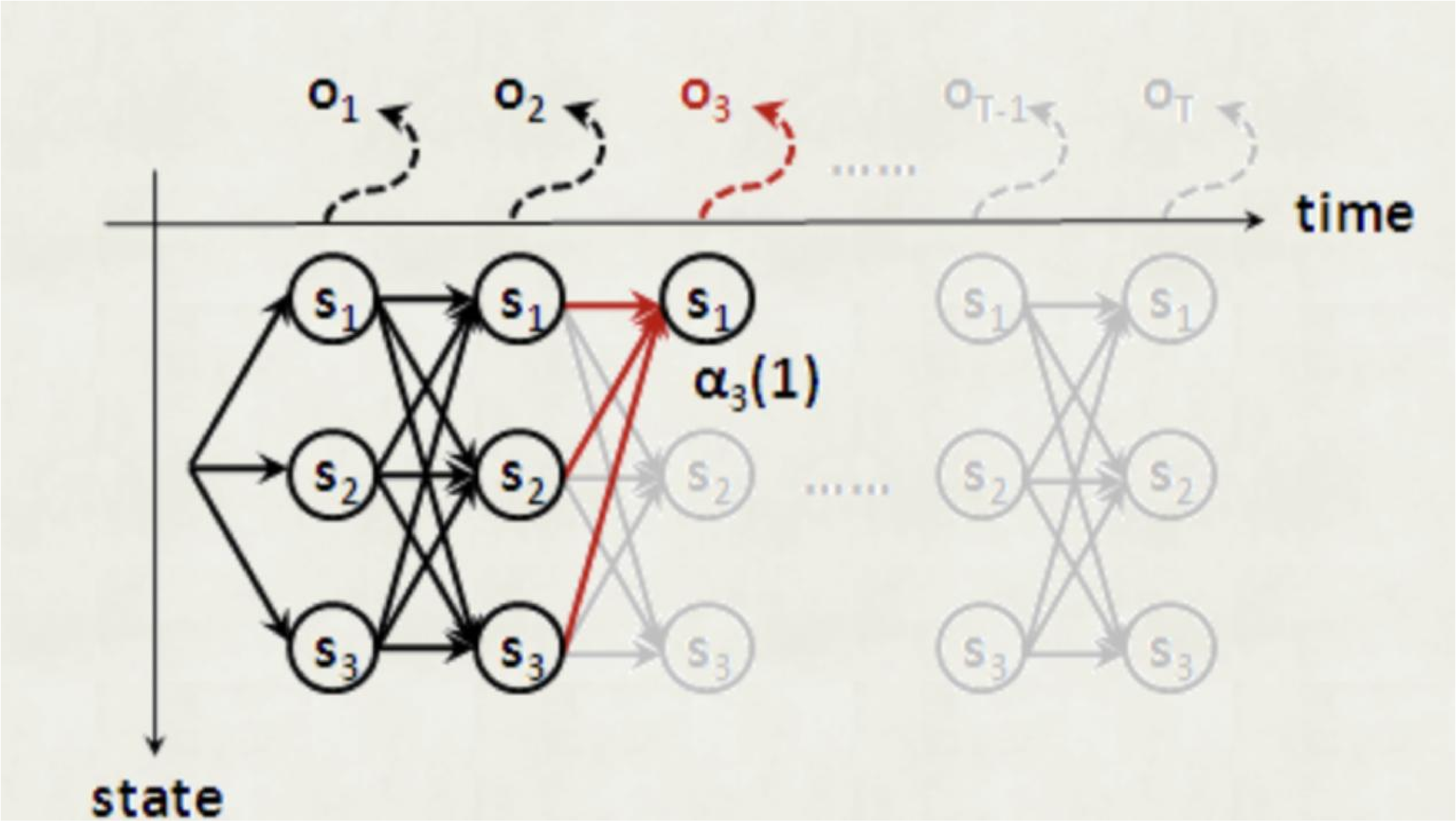
（1）求和约束条件的丑陋：下标的选择是permutation，阶乘级的运算量

（2）概率的连乘：极大量的概率乘积很容易导致结果趋于0，超出精度要求(甚至到了加log都不一定很有效的程度)

（3）最暴力的做法，本身就令人难以接受。。。

HMM模型介绍——问题1：endcodeing

结果并不好看！——引进新的性质：马尔可夫性



HMM模型介绍——问题1：endcodeing

结果并不好看！——引进新的性质：马尔可夫性

我们要求的是：能够产生观察序列$o_1, \ldots, o_T$的所有状态序列$s_1, \ldots, s_T$的概率之和。

并且我们注意到了这个过程的马尔可夫性，所以我们每一层只需要关心上一层的情况就好了，并且下一层也只关心我这一层的状态，和这个状态的历史完全无关，所以我们这一层也就没有必要区分是从上一层的哪个状态转移来的——这大大节约了我们的空间。

于是对于时间t的时候达到状态i的概率（并且保证t之前的观察序列和我们要求的一致）记为：$\alpha_t(i)$，我们要求的就是$\sum_{i=1}^{n} \alpha_T(i) * b_{i,o_T}$

HMM模型介绍——问题1：endcodeing

结果并不好看！——引进新的性质：马尔可夫性

如果考虑向量(列向量)的运算，那么就是：

$$\vec{\alpha}_{t+1} = \vec{\alpha}_t^T \vec{c}$$

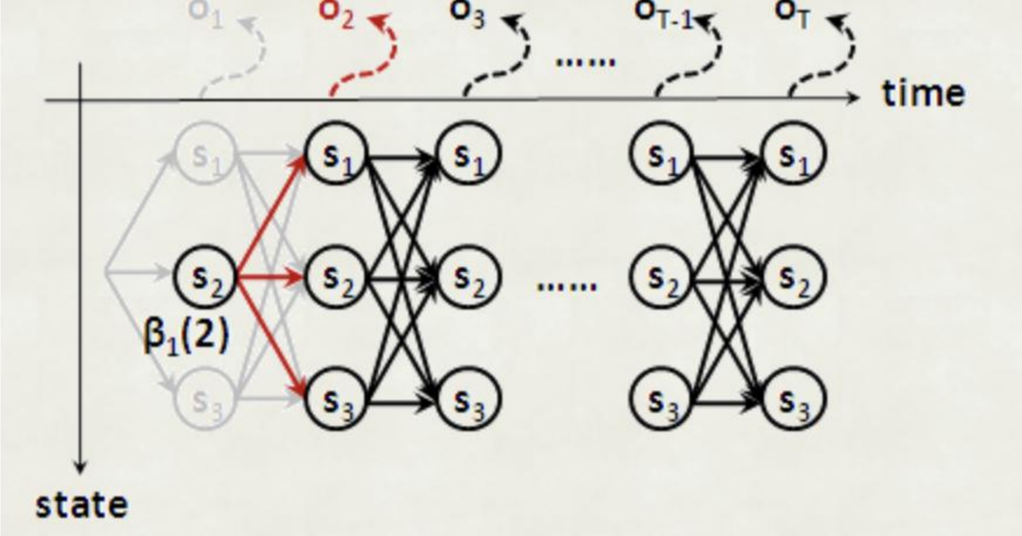
其中$c_i = b_{i,o_t} * a_{ix}$

这里我们可以看一下这个概率求解的本质——或者说古典型概率的求解特点：

因为无非是使用了加法和乘法原则，如果没有条件概率的事情，也大概不会引入除法——所以事件的概率无非就是样本点概率的线性组合

我们这里也是一样，我们其实已经用了所有样本点（当然因为马尔可夫性优化了一些），最后的结果无非就是最原本的样本的多次的线性组合。

HMM模型介绍——问题1：endcodeing



结果并不好看！——引进新的性质：马尔可夫性
反向求概率，依然可以求出

$\beta_t(i)$的定义就是形成$\{o_{t+1}, \ldots, o_T\}$，(如果t=T的话就是空)的所有状态序列(保证t之后的状态的序列和我们要求的一样），并且t时刻的状态是i的概率。

所以初始条件就是:$\beta_T(i) = 1$
递推条件是:

$$\beta_t(x) = \sum_{i=1}^{n} \beta_{t+1}(i) * a_{xi} * b_{i,o_{t+1}}$$

HMM模型介绍——问题2：dedcodeing

具体描述就是：在已知$\{o_1, o_2, ..., o_T\}$，未知状态序列的情况下，求概率最大的状态序列。

实际上这个问题中，所有状态序列都是有可能产生这个观察序列的，但是概率大小肯定是有区别的——这个时候我们要怎么做呢？似乎前面的α、β不太好做了——因为这两个都是具体路径的压缩，所以自然找不出具体的概率最大的路线。

实际上按照我们之前的做法中，很容易看到，每一条路线的概率就是取决于所有转移概率与观察产生的概率乘积。

这里有一个很好的性质：

Proposition 2 (转移的全链接性).

相邻两层的任意两个状态之间都是可以转移的

HMM模型介绍——问题2：dedcodeing

具体描述就是：在已知$\{o_1, o_2, ..., o_T\}$，未知状态序列的情况下，求概率最大的状态序列。

所以其实只要上一层有一条到达概率最大的路径，那么下一层完全没有理由不选择从那条路径继续。

所以我们本质上就是把每一层的$max_{i,j}(a_{ij} * b_{j,o_t})$找出来就好了

这个是$2n^2$次操作，然后每一层都做一遍，就是T次，总共时间复杂度$O(Tn^2)$，这个问题就解决了。

```
forward:
δ₁(j) = πⱼ × bⱼ(o₁)
δₜ₊₁(j) = max_{i=1…N}[ δₜ(i) × aᵢⱼ ] × bⱼ(oₜ₊₁)

path tracing:
ψₜ(j) = arg max_{i=1…N}[ δₜ₋₁(i) × aᵢⱼ ] × bⱼ(oₜ)
      = arg max_{i=1…N}[ δₜ₋₁(i) × aᵢⱼ ]
```

HMM模型介绍——问题3：training problem

问题具体的描述：还是已知观察序列$\{o_1, o_2, \ldots, o_T\}$，转移概率矩阵A、观察产生概率矩阵B、初始状态分布Π这三个的初始值，但是不知道状态序列，我们需要更新A、B、Π使得第一个问题中算出来的产生这个观察序列的概率最大。

这个相比于前面两个问题就显得更加统计。

采用的想法就是最大似然法——老祖宗的智慧

并且这里难点就是我们第一个问题执着于求出这个概率的值，也就是用递推的方式，但是我们缺少一个显式的表达式，也就是其实损失了每个参数的信息，

HMM模型介绍——问题3：training problem

问题具体的描述：还是已知观察序列$\{o_1, o_2, \ldots, o_T\}$，转移概率矩阵A、观察产生概率矩阵B、初始状态分布Π这三个的初始值，但是不知道状态序列，我们需要更新A、B、Π使得第一个问题中算出来的产生这个观察序列的概率最大。

这里的想法就是，我们还有状态序列$y = (y_1, \ldots y_n)^T$，以及有一系列的参数θ（在HMM中是A、B、Π三个矩阵），在一般的MLE中，我们希望：

$$max_\theta \ \ell(\theta; y) = \log p(y|\theta)$$

显然，在HMM问题中，我们无法得到$p(y|\theta)$的显式表达式，因此不能够处理，实际上我们可以知道的是：

$$\ell(\theta; y) = \log \sum_x p(y, x|\theta)$$

其中x是观测序列$x = (x_1, \ldots, x_n)^T$，我们可以知道的是y与x的条件联合概率。

HMM模型介绍——问题3：training problem

MLE通常的处理是进行求导，但是这里的函数形式并不利于这样做，于是考虑将求和号拿出log：

$$
\begin{aligned}
\ell(\theta; y) &= \log \sum_x p(y, x|\theta) \\
&= \log \sum_x q(x) \left(\frac{p(y, x|\theta)}{q(x)} \right) \\
&\geq \sum_x q(x) \log \left(\frac{p(y, x|\theta)}{q(x)} \right) \\
&= E_{q(x)}[\log(y, x|\theta)] + Entropy(q(x)) \\
&= L(q, \theta; y)
\end{aligned}
$$

按照上面的形式，我们可以将最大似然函数写成这样的表达：

$$
\ell(\theta; y) = E_{q(x)}[\log(y, x|\theta)] + KL[q(x)||p(x; y, \theta)] + Entropy(q(x))
$$

HMM模型介绍——问题3：training problem

实际上，我们得到的下界中仅有$E_{q(x)}[\log(y,x|\theta)]$与我们希望优化的参数有关，因此在引入q之后，我们考虑这样的迭代优化步骤，希望至少能够得到局部最优的数值解

第一步：优化q——jensen不等式的取等条件

$$q^t = argmax_q L(q, \theta^{t-1}; y) = p(x|y, \theta^{t-1})$$

第二步：优化θ——类似MLE，但因为形式化简，可操作性强多了：

$$\theta^t = argmax_\theta L(q^t, \theta; y) = argmax_\theta E_{q^t(x)}[\log p(y, x|\theta)]$$

迭代这两步，可以在数学上证明可以收敛到局部最优解或者鞍点。

Wu, C. F. Jeff (Mar 1983). "On the Convergence Properties of the EM Algorithm". *Annals of Statistics*. **11** (1): 95–103. doi: 10.1214/aos/1176346060. JSTOR 2240463. MR 0684867

HMM模型介绍——问题3：training problem

我们直接将前面的显式表达式加log求和，转化为EM优化的形式：

$$
\begin{aligned}
\log P &= \sum_{i=1}^{N} q_0^i \log \pi_i + \sum_{t=1}^{T} \sum_{i,j=1}^{N} q_t^i q_{t+1}^j \log a_{ij} + \sum_{t=1}^{T} \sum_{i,j}^{N,M} q_t^i y_t^i \log b_{i,j} \\
&= \sum_{i=1}^{N} (q_0^i) \log \pi_i + \sum_{i,j}^{N} \left(\sum_{t=1}^{T} q_t^i q_{t+1}^j \right) \log a_{i,j} + \sum_{i,j}^{N,M} \left(\sum_{t=1}^{T} q_t^i y_t^j \right) \log \eta_{i,j}
\end{aligned}
$$

其中，q_t^i是一个0-1变量，代表了在t时刻达到状态i的概率，y_t^i也是一个0-1变量，代表了在t时刻获得i观察值的概率——显式的表达式我们可以用第一二问的结果求出来，先保留这样的记号，继续优化问题：

我们记：

$$
m_{ij} = \sum_{t=1}^{T} q_t^i q_{t+1}^j, \ n_{ij} = \sum_{t=1}^{T} q_t^i y_t^j
$$

HMM模型介绍——问题3：training problem

优化的限制条件，也就是概率的归一性：

$$\sum_{i=1}^{N} \pi_i = 1,\ \sum_{j=1}^{N} a_{ij} = 1,\ \sum_{j=1}^{M} b_{ij} = 1$$

接下来就是常规的拉格朗日乘子法，直接给出结果：

$$\begin{cases} \hat{\pi}_i = q_1^i \\ \hat{a}_{ij} = \frac{m_{ij}}{\sum_{k=1}^{N} m_{ik}} \\ \hat{\eta}_{ij} = \frac{n_{ij}}{\sum_{k=1}^{N} n_{ik}} \end{cases}$$

接下来表示参数q，也就是用我们之前求的两个问题的结果代入：

$$q_t^i = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)}$$

代入上式就可以得到我们的结果。

HMM模型介绍——一些细节

之后有几个小优化，一个是对概率取log，这样是为了防止概率越乘越小，最后超出我们的精度。对于加法而言，log有这样的公式：

$$
\begin{aligned}
&\text{if } p \geq q \\
&\quad \log_b(p+q) = \log p + \log_b\left(1 + b^{\log_b q - \log_b p}\right) \\
&\text{else} \\
&\quad \log_b(p+q) = \log q + \log_b\left(1 + b^{\log_b p - \log_b q}\right)
\end{aligned}
$$

smooth问题，和神经网络中曾经为了防止梯度消失使用了类似的想法：在需要更新某个参数为0的时候，不真正更新为0，而是一个很小的数字，同时保证概率和为1的限制。

之后还有连续版本的HMM，我就不多说了，基本想法是类似的（连续版本并不需要smoothing，可以思考一下为什么）

作业最后一道附加题讲解

题目简述：有两类节点，选中点和未选中点，每个未选中点归属并且仅归属于一个选中点，选中点归属于本身——每一天，随机选中两个选中点，再随机选其中一个变为另一个的未选中点，被选中的选中点的“儿子”变为选中点——问：到达只剩一个选中点的期望天数

分析：就是求停时的期望而已，考虑停时定理以及相应的势能函数方法

停时定理

设 t 为鞅过程 $\{X_0, X_1, \cdots\}$ 的停时，当下面三个条件之一成立时，有 $E(X_t) = X_0$：

- t 几乎必然有界；
- $\mid X_{i+1} - X_i \mid$ 一致有界，$E(t)$ 有限；
- X_i 一致有界，t 几乎必然有限。

作业最后一道附加题讲解

势能函数

对于随机时间序列 $\{A_0, A_1, \cdots\}$，t 为其停时，终止状态为 A_t，求 $E(t)$。

构造势能函数 $\Phi(A)$，满足：

- $E(\Phi(A_{n+1}) - \Phi(A_n) \mid A_0, A_1, \cdots, A_n) = -1$;
- $\Phi(A_t)$ 为常数，且 $\Phi(A_i) = \Phi(A_t)$ **当且仅当** $i = t$。

构造序列 $X_i = \Phi(A_i) + i$，则 $E(X_{n+1} - X_n \mid X_0, X_1, \cdots, X_n) = 0$，即 $\{X_0, X_1, \cdots\}$ 是鞅。

根据停时定理，我们可以得到 $E(X_t) = E(X_0)$，即 $E(t) = E(\Phi(A_0)) - \Phi(A_t)$。

作业最后一道附加题讲解

设 $f(x)$ 为跟随有 x 个未选中点的选中点的势能函数，整个局面的势能函数为 $\Phi(A)$，每个选中点的势能函数的和。

显然每次操作的势能变化量只和随出来的两个点有关，u, v 的儿子数设为 x, y。

为满足 $E(\Phi(A_{n+1}) - \Phi(A_n) \mid A_0, A_1, \cdots, A_n) = -1$，有

$$f(x) + f(y) - 1 = \frac{1}{2}(f(x+1) + yf(0)) + \frac{1}{2}(f(y+1) + xf(0))$$

因为我们要对于任意 x, y 成立，所以

$$f(x) - \frac{1}{2} = \frac{1}{2}f(x+1) + \frac{x}{2}f(0)$$

取 $f(0) = 0$，则 $f(x) = 1 - 2^x$。

设停时为 t，注意到 $\Phi(A_t) = f(n-1) = 1 - 2^{n-1}$ 为常数，所以 $E(t) = \Phi(A_0) - \Phi(A_t)$。