Lecture Notes of Probability Theory
Lesson 11

李沐阳
664789531@qq.com
上海交通大学 ACM 班

June 14, 2018

Contents

1 内容摘要

在 Lesson 11 中，吴老师所讲授的内容主要有：

- 接着上节课的内容，吴老师 assert 每个马尔科夫链都有起点，且起点长度有限，并留下习题。[1]
- 介绍邻居问题，并给出其证明。
- 介绍手机报废问题，并给出了其证明。
- 介绍了一个掷硬币的模型，并给出了一些相关性质及其证明。

笔者整理的 lecture notes 中如有疏漏之处，还请各位读者批评指正。

2 马尔科夫链起点习题

已知每条马尔科夫链的状态都有起点，那么马尔科夫链起点的长度是否有界。也即对于给定状态，则对于所有 $M > 0$，均存在一条马尔科夫链，使得其起点长度大于 M。

3 邻居问题

3.1 问题描述

已知有一个连通社交网络有 N 个人，其中有的人是坚定的（一定存在），有些人是不坚定的（有可能不存在）。他们给一部电影打分，每个人都开始都会有个起始分数。之后每轮都会选定一个人进行分数的更新：

- 若此人是坚定的，则其打分不变；
- 若此人不坚定，则其分数变为其社交网络中邻居的打分按照某种方式加权平均后的值。

若每个（不坚定的）人都有无穷次更新的机会，则所有人的打分都会收敛。

如图1所示社交网络，蓝点表示该人坚定，黄点表示该点不够坚定，影响方式即为算术平均。初始时每个人的打分即为他们旁边的数字。

- 第一轮更新 5 号，由于 5 号坚定，故每个人的打分不变。
- 第二轮更新 2 号，由于 2 号不坚定，故其打分变为 $(3+1+5+2)/4 = 2.75$。最后结果如图2所示。
- 第三轮更新 3 号，由于 3 号不坚定，故其打分变为 $(2.75+1+6)/3 = 3.25$。最后结果如图3所示。
- ...
- 最后收敛态，如图4所示。

[1]Lesson 10 的 notes 中提到此 notes 中会对此问题做出解答，然而由于老师实际上也没有详解，而是留作了习题，所以这题只有请大家自行思考了。

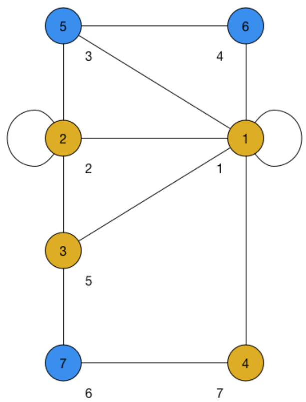

Figure 1: 社交网络图初始状态

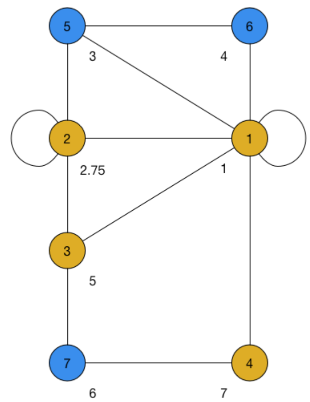

Figure 2: 第二轮社交网络

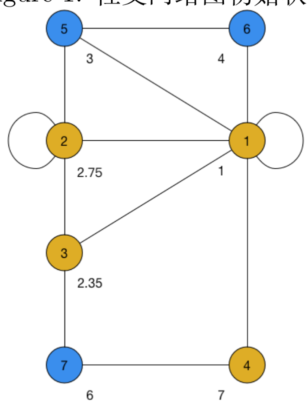

Figure 3: 第三轮社交网络图

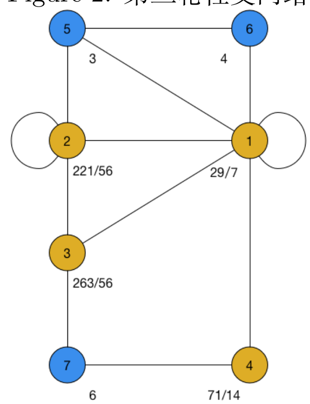

Figure 4: 社交网络图最终状态

3.2 解答与解答

Lemma 1. 若 $\mathbf{D}$ 为一个 n 阶非负方阵，且 $\mathbf{D}$ 每行行和严格小于 1，则

$$\lim_{s\to+\infty} \mathbf{D}^s = \mathbf{0}$$

Proof. 设 $\mathbf{D}^s(s>0)$ 中的最大元素为 g_s，$\mathbf{D}$ 的最大行和为 $t(0\le t<1)$，则由

$$\mathbf{D}^{s+1} = \mathbf{D}\mathbf{D}^s$$

故

$$(\mathbf{D}^{s+1})_{i,j} = \sum_{k=1}^{n} \mathbf{D}_{i,k}(\mathbf{D}^s)_{k,j} \le \sum_{k=1}^{n} \mathbf{D}_{i,k} g_s \le t g_s$$

因此

$$(\mathbf{D}^s)_{i,j} \le t^{s-1} g_1, \quad \forall 1\le i,j\le n$$

又 $\lim\limits_{s\to+\infty} t^{s-1}g_1 = 0$，所以

$$\lim_{s\to+\infty}\mathbf{D}^s = \mathbf{0}$$

□

设有 m 个不坚定的人，n 个坚定的人（$m+n=N$），显然证明过程与结果与编号无关系，不妨设 $1\sim m$ 号人为不坚定的人，$m+1\sim N$ 号人为坚定的人。根据当前状态每个人的评分 $x_1, x_2, \ldots, x_{N-1}, x_N$，我们可以构造一个 $N\times 1$ 的列向量

$$f = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{N-1} \\ x_N \end{pmatrix}$$

假设我们当轮更新第 i 个人，若 $i \le m$，则构造转移矩阵

$$\pi_i = \begin{pmatrix}
1 & 0 & \cdots & 0 & 0 & \cdots & 0 & 0 \\
0 & 1 & \cdots & 0 & 0 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 1 & 0 & 0 & 0 & 0 \\
P_{i,1} & P_{i,2} & \cdots & P_{i,i} & P_{i,i+1} & \cdots & P_{i,N-1} & P_{i,N} \\
0 & 0 & \cdots & 0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & 0 & \cdots & 1 & 0 \\
0 & 0 & \cdots & 0 & 0 & \cdots & 0 & 1
\end{pmatrix}$$

其中 $P_{i,j}$ 表示在更新 i 时，j 的打分对之影响的权重。以上图为例，则

$$\pi_2 = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

若 $i > m$，则 $\pi_i = \mathbf{I}_N$。

那么在更新第 i 个人后，评分列向量变为 f'，则

$$f' = \pi_i f$$

因此显然，假设我们按照序列 $v_1, v_2, v_3, \ldots$ 的次序进行更新，每个（非坚定）点都在序列中出现无限次，s 轮之后，那么最后的评分列向量 f_s 变为

$$f_s = \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1}f_0$$

其中 f_0 表示初始时每个人的打分。我们只需要证明

$$\lim_{s\to+\infty} f_s = \lim_{s\to+\infty} \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} f_0$$

存在就行了。

根据 $P_{i,j}$ 我们可以构建矩阵

$$\mathbf{P} = \begin{pmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,N-1} & P_{1,N} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,N-1} & P_{2,N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ P_{N-1,1} & P_{N-1,2} & \cdots & P_{N-1,N-1} & P_{N-1,N} \\ P_{N,1} & P_{N,2} & \cdots & P_{N,N-1} & P_{N,N} \end{pmatrix}$$

下面我们先证明

$$\lim_{s\to+\infty} \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} = \lim_{s\to+\infty} \mathbf{P}^s$$

也即对于所有 $\alpha \in \mathbb{R}^{1\times N}$，

$$\alpha \lim_{s\to+\infty} \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} = \alpha \lim_{s\to+\infty} \mathbf{P}^s \tag{1}$$

根据矩阵乘法的线性性，可以得 (1) 式等价于

$$\alpha_i \lim_{s\to+\infty} \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} = \alpha_i \lim_{s\to+\infty} \mathbf{P}^s, \quad \forall 1 \le i \le N$$

其中

$$\begin{array}{cc} & \begin{array}{ccccccc} & & & i\text{列} & & & \end{array} \\ \alpha_i = & \begin{pmatrix} 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{pmatrix} \end{array}$$

由于

$$\alpha_i \pi_i = \alpha_i \mathbf{P}$$

$$\alpha_i \pi_j = \alpha_i (i \ne j)$$

故假设 $i \ne v_j (k < j \le s)$，而 $i = v_k$：

$$\begin{aligned} \alpha_i \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} &= \alpha_i \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_{k+1}}\pi_i\pi_{v_{k-1}}\cdots\pi_{v_1} \\ &= \alpha_i \mathbf{P} \pi_{v_{k-1}}\cdots\pi_{v_2}\pi_{v_1} \end{aligned}$$

当 s 趋于正无穷时，由于每个 π_i 都有无限个，每个都可以变成 P，因此

$$\lim_{s\to+\infty} \pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} = \lim_{s\to+\infty} \mathbf{P}^s$$

由于当 $i > m$ 时，$P_{i,i} = 1, P_{i,j} = 0(i \ne j)$，因此 $\mathbf{P}$ 可以用分块矩阵表示

$$\mathbf{P} = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix}$$

那么

$$\mathbf{P}^1 = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix} \quad \mathbf{P}^2 = \begin{pmatrix} \mathbf{Q}^2 & (\mathbf{Q}+\mathbf{I})\mathbf{R} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix}, \ldots$$

归纳可得

$$\mathbf{P}^s = \begin{pmatrix} \mathbf{Q}^s & (\sum_{i=0}^{s-1}\mathbf{Q}^i)\mathbf{R} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix}$$

由于 $\mathbf{P}$ 是个吸收马尔科夫链的转移矩阵[2]，故 $\mathbf{Q}^N$ 的每行行和严格小于 1。由引理1可得

$$\lim_{s\to+\infty}\mathbf{Q}^s = \lim_{s\to+\infty}(\mathbf{Q}^N)^s = \mathbf{0}$$

因此 $\lim\limits_{s\to+\infty}\sum_{i=0}^{s}\mathbf{Q}^i$ 存在，设其值为 $\mathbf{A}$。则

$$\mathbf{A} = \lim_{s\to+\infty}\sum\nolimits_{i=0}^{s}\mathbf{Q}^i \tag{2}$$

$$\mathbf{QA} = \lim_{s\to+\infty}\sum\nolimits_{i=1}^{s}\mathbf{Q}^i \tag{3}$$

$(2)-(3)$ 可得：

$$(\mathbf{I}-\mathbf{Q})\mathbf{A} = \mathbf{I} \Rightarrow \mathbf{A} = (\mathbf{I}-\mathbf{Q})^{-1}$$

由此

$$\lim_{s\to+\infty}\pi_{v_s}\pi_{v_{s-1}}\cdots\pi_{v_2}\pi_{v_1} = \lim_{s\to+\infty}\mathbf{P}^s = \begin{pmatrix} \mathbf{0} & (\mathbf{I}-\mathbf{Q})^{-1}\mathbf{R} \\ \mathbf{0} & \mathbf{I}_n \end{pmatrix}$$

即证。

4 手机报废问题

4.1 问题描述

假设我有一部手机，它每天可能处于三个状态——正常工作、维修和报废。当天处于正常工作状态，第二天有 99.5% 概率继续正常工作，有 0.5% 的概率第二天送去维修；处于维修状态的手机，第二天有 90% 概率可以正常使用，5% 概率第二天继续维修，5% 的概率第二天报废；处于报废阶段的手机，永远只能报废了。假设开始时手机是可以正常使用，那么在手机报废前期望有多少天手机可以正常使用?

4.2 解答与证明

我们构建该状态转移图的马尔科夫链转移矩阵

$$\mathbf{P} = \begin{array}{c} \\ \mathrm{W} \\ \mathrm{R} \\ \mathrm{S} \end{array}\begin{array}{c} \begin{array}{ccc} \mathrm{W} & \mathrm{R} & \mathrm{S} \end{array} \\ \begin{pmatrix} 0.995 & 0.005 & 0 \\ 0.9 & 0.05 & 0.05 \\ 0 & 0 & 1 \end{pmatrix} \end{array}$$

[2] $m+1 \sim N$ 号为该马尔科夫链的吸收点。

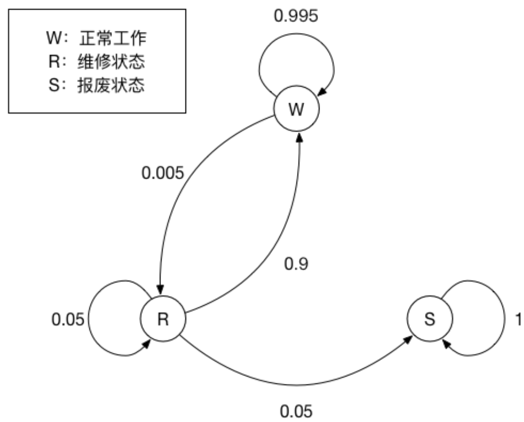

Figure 5: 手机状态转移图

则 $\mathbf{P}^n(\mathrm{W},\mathrm{W})$ 表示开始时正常工作，第 n 天依旧正常工作的概率。令

$$X_n = \begin{cases} 1 & \text{从正常工作开始第}N\text{天正常工作} \\ 0 & \text{其他情况} \end{cases}$$

则所求即为

$$EX = \sum_{n=0}^{\infty} E(X_n) = \sum_{n=0}^{\infty} \mathbf{P}^n(\mathrm{W},\mathrm{W})$$

设

$$\mathbf{Q} = \begin{pmatrix} 0.995 & 0.005 \\ 0.9 & 0.005 \end{pmatrix} \quad \mathbf{R} = \begin{pmatrix} 0 \\ 0.05 \end{pmatrix} \quad \mathbf{P} = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & 1 \end{pmatrix}$$

则 $\mathbf{P}^n(\mathrm{W},\mathrm{W}) = \mathbf{Q}^n(\mathrm{W},\mathrm{W})$。由 Section 3中结论，

$$EX = \sum_{n=0}^{\infty} \mathbf{P}^n(\mathrm{W},\mathrm{W}) = \sum_{n=0}^{\infty} \mathbf{Q}^n(\mathrm{W},\mathrm{W}) = (\mathbf{I}-\mathbf{Q})^{-1}(\mathrm{W},\mathrm{W}) = 3800$$

同理也可以求从正常工作状态开始，有多少天是在维修的；从维修状态开始，有多少天是正常工作的，有多少天处在维修状态的。

5 掷硬币问题

5.1 问题描述

一枚硬币有两面，分别是 H 和 T，且抛出两面的概率相等。现在给定一个长度一定的 HT 序列 w，然后开始掷硬币，将第 i 次掷硬币的结果记为 x_i，直到掷硬币的 HT 序列第一次出现 w 为止，将掷硬币的次数记为 $\mathcal{T}_w$。

举个例子，$w = \mathrm{HHT}$，且掷硬币序列如表1所示 则 $\mathcal{T}_w = 5$。

现在给定 w，问 $E(\mathcal{T}_w)$。

Table 1: 掷硬币序列

x_1	x_2	x_3	x_4	x_5
T	H	H	H	T

5.2 解答与证明

5.2.1 解法一

Lemma 2. 若 X 为非负整数随机变量，则

$$EX = \sum_{i=0}^{\infty} P(X > i)$$

Proof. 首先由于 X 为非负整数，故

$$X = \sum_{i=0}^{\infty} [X > i]$$

因此，

$$EX = E(\sum_{i=0}^{\infty}[X > i]) = \sum_{i=0}^{\infty} E([X > i]) = \sum_{i=0}^{\infty} P(X > i)$$

□

不妨设 w 的长度为 N，则

$$E(\mathcal{T}_w) = \sum_{1 \le i \le N} [pre_i = suf_i] 2^i$$

其中

$$pre_i = w_1 w_2 \dots w_i$$
$$suf_i = w_{N-i+1} w_{N-i+2} \dots w_N$$

Proof. 定义如下东西：

$$F_n = \{\text{不含}w\text{的长度为}n\text{的序列}\} \qquad f_n = |F_n|$$
$$\overline{F_n} = \{\text{长度为}n\text{的}w\text{只出现在末尾的序列}\} \qquad \overline{f_n} = |\overline{F_n}|$$

考虑在 F_n 每个元素末尾拼上 w，将这些新拼出来的串构成的集合定义为 $F_n + w$，则

$$F_n + w = \{x_1 x_2 \dots x_n w \mid x_1 x_2 \dots x_n \in F_n\}$$

考虑 $F_n + w$ 中每个元素的 $\mathcal{T}_w$ 值，则按照 F_n 的定义

$$\mathcal{T}_w = n + 1, n + 2, \dots, n + N$$

然而也并不是 $[n+1, n+N]$ 之间每个整数都能取得到。比如 w = THT 时 $\mathcal{T}_w \neq n+2$。因为若存在序列 $s \in F_n + w$，使得 $\mathcal{T}_w = n+2$。则由 $s \in F_n + w$，得

$$s_{n+2} = \mathrm{T}$$

又由 $\mathcal{T}_w = n+2$，

$$s_{n+2} = \mathrm{H}$$

显然二者是矛盾的。

下面我们考虑 $[n+1, n+N]$ 中哪些值是可以取得的。假设存在 $s \in F_n + w$，使得 $\mathcal{T}_w = n+i (1 \leq i \leq N)$。那么，由 $s \in F_n + w$，

$$s_{n+1} s_{n+2} \dots s_{n+N} = w$$

由 $\mathcal{T}_w = n+i$, 得

$$s_{n+i-N+1} s_{n+i-N+2} \dots s_{n+i} = w$$

考虑二者重叠部分（如图 5.2.1 所示阴影部分），则

$$w_1 w_2 \dots w_i = w_{N-i+1} w_{N-i+2} \dots w_N$$

也即 $pre_i = suf_i$。

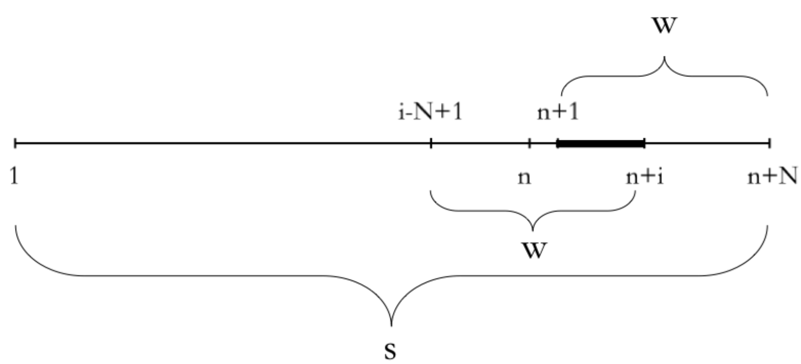

Figure 6: 数列示意图

因此，记

$$S = \{i \mid 1 \leq i \leq N, pre_i = suf_i\}$$

易得

$$f_n = \sum_{i \in S} \overline{f_{n+i}} \Rightarrow \frac{f_n}{2^n} = \sum_{i \in S} 2^i \frac{\overline{f_{n+i}}}{2^{n+i}} \tag{4}$$

由于 HT 等概率掷出，因此 (4) 式可以化为

$$P(\mathcal{T}_w > n) = \sum_{i \in S} 2^i P(\mathcal{T}_w = n+i)$$

因此

$$
\begin{aligned}
E(\mathcal{T}_w) &= \sum_{n=0}^{\infty} P(\mathcal{T}_w > n) \qquad (5)\\
&= \sum_{n=0}^{\infty} \sum_{i\in S} 2^i P(\mathcal{T}_w = n+i)\\
&= \sum_{i\in S} \sum_{n=0}^{\infty} 2^i P(\mathcal{T}_w = n+i)\\
&= \sum_{i\in S} 2^i P(\mathcal{T}_w \geq i)
\end{aligned}
$$

其中 (5) 运用了引理2的结论。由于 $i \leq N$，而 T_w 显然不小于 N，故

$$EX = \sum_{i\in S} 2^i P(\mathcal{T}_w \geq i) = \sum_{i\in S} 2^i = \sum_{1\leq i\leq N} [pre_i = suf_i] 2^i$$

□

其实有个赌徒模型也可以解决上面的问题，为了方便叙述，假设 $w = \mathrm{HTH}$：

现在赌场有一中奖序列即为 $w = \mathrm{HTH}$，这是对外公开的。现在 3 位赌徒，均押注 1 元，按照 w 的顺序竞猜。每轮赌场会公布一个字母，赌徒猜对奖励翻倍，但是若猜就没有任何奖励了。当序列结束时可以给每位赌徒结算奖励。第一位赌徒在中奖序列开始前就进入了赌场竞猜，第二位赌徒在公布完第一个 H 后进入，第三位赌徒在公布完 T 后进入，则赌场最后一共要发放 $8 + 2 = 10$ 元奖励，如图5.2.1，即 $E(\mathcal{T}_w)$。

	H T H				说明
第一位	H	T	H			第一位全部猜对，赢了8元
第二位		H	T	H		第二位由于第一位没有猜对，没有奖励
第三位			H	T	H	第三位由于只猜对第一位，赢了2元

Figure 7: 赌徒模型图示

5.2.2 解法二（暂时解体了）

本题还可以用生成函数做，需要用到一些动态规划[3] 和 KMP[4]或 AC 自动机[5]的知识。

不妨设 w 的长度为 N。用 $f_{i,j}$ 表示长度为 i 的掷硬币序列，在未出现 w 的情况下，最后末尾最多匹配 w 的前 j 位的概率；$c_{i,j}$ 表示当前末尾最多匹配 w 前 i 位，下一位填 j(H 或 T) 末尾最多可以匹配 w 前多少位。

[3]关于动态规划详情可见Dynamic Programming

[4]KMP 即为 Knuth–Morris–Pratt algorithm

[5]AC 自动机即为 Aho–Corasick algorithm

$c_{i,j}$ 用 KMP 算法或 AC 自动机都可以求，于是我们可以很轻松的得到转移方程:

$$f_{i,j}=\frac{1}{2}\sum_{k=0}^{N-1}[c_{k,0}=j]f_{i-1,k}+\frac{1}{2}\sum_{k=0}^{N-1}[c_{k,1}=j]f_{i-1,k}$$

初始条件:

$$f_{0,0}=1$$

最后

$$EX=\sum_{i=1}^{\infty}if_{i,N}$$

令

$$F_j(x)=\sum_{i=0}^{\infty}f_{i,j}x^i$$

则显然

$$EX=\sum_{i=1}^{\infty}if_{i,N}=F'_N(1)$$

由上面的转移，我们可以得到一个新的转移

$$F_j(x)=\frac{1}{2}x\sum_{k=0}^{N-1}[c_{k,0}=j]F_k(x)+\frac{1}{2}x\sum_{k=0}^{N-1}[c_{k,1}=j]F_k(x)$$

根据上面的式子列出方程，我们就可以得出 $F_N(x)$ 了。

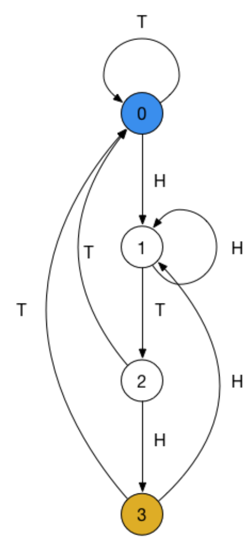

Figure 8: AC 自动机示意图

以 $w = \mathrm{HTH}$ 为例，造出如图5.2.2AC 自动机，易得

$$\begin{cases} c_{0,\mathrm{H}} = 1 \\ c_{0,\mathrm{T}} = 0 \end{cases} \quad \begin{cases} c_{1,\mathrm{H}} = 1 \\ c_{1,\mathrm{T}} = 2 \end{cases} \quad \begin{cases} c_{2,\mathrm{H}} = 3 \\ c_{2,\mathrm{T}} = 0 \end{cases} \quad \begin{cases} c_{3,\mathrm{H}} = 1 \\ c_{3,\mathrm{T}} = 0 \end{cases}$$

于是方程

$$\begin{cases} F_0(x) = \dfrac{1}{2}xF_0(x) + \dfrac{1}{2}xF_2(x) \\ F_1(x) = \dfrac{1}{2}xF_0(x) + \dfrac{1}{2}xF_1(x) \\ F_2(x) = \dfrac{1}{2}xF_1(x) \\ F_3(x) = \dfrac{1}{2}xF_2(x) \end{cases}$$

解得 $F_3(x) = 0$，故

$$EX = F_3'(1) = 10$$

（解体原因：没有定义初值，本人才疏学浅，欢迎有志之士前来 fix）

5.3 思考题

对于给定序列 w，构造序列 w' 使得

$$\begin{cases} P(\mathcal{T}_w < \mathcal{T}_{w'}) > \dfrac{1}{2} \\ E(\mathcal{T}_w) > E(T_{w'}) \end{cases}$$