算法分析与设计基础作业8

软件71 骆炳君 2017013573

Exercises 32.4-8

首先进行证明：

由定义可得，$\delta(q,a)=\sigma(P_q a)=max\{k:P_k \sqsupset P_q a\}$

$\pi[q]=max\{k:k<q \; and \; P_k \sqsupset P_q\}$

如果$P[q+1]\neq a$，因为$\delta(q,a)=\sigma(P_q a)$，而$\delta(\pi[q],a)=\sigma(P_{\pi[q]}a)\leq\sigma(P_q a)$，又有$\pi[q]\geq\sigma(P_q a)-1$且$P[\sigma(P_q a)]=a$，所以$\sigma(P_{\pi[q]}a)\geq\sigma(P_q a)$，最终可证得$\sigma(P_{\pi[q]}a)=\sigma(P_q a)$，即$\delta(q,a)=\delta(\pi[q],a)$.

如果$q=m$，此时已匹配到了T的一个子串P，应继续寻找下一个P，可等价认为$P[m+1]\neq a$，即可化为第一种情况.

如果$P[q+1]=a$且$q\neq m$，则$\delta(q,a)=q+1$.

因此可设计出如下的算法：

```
COMPUTE-TRASITION-FUNCTION(P,Sigma)
m = P.length
pi = COMPUTE-PREFIX-FUNCTION(P)
for a in Sigma:
    if P[1] != a:
        delta(0, a) = 0
    else:
        delta(0, a) = 1
for q = 1 to m:
    for a in Sigma:
        if P[q+1] != a or q == m:
            delta(q, a) = delta(pi[q], a)
        else:
            delta(q, a) = q + 1
return delta
```

Problems 32-1

a.

首先计算出P的前缀函数π，可得当$\rho(P_i)>1$时，$\pi[i]$为$\rho(P_i)-1$个最小循环节的长度，即$\rho(P_i)>1$等价于$i-\pi[i]$能整除i.

如果$i-\pi[i]$不能整除i，那么$\rho(P_i)=1$.否则，$P_{i-\pi[i]}$就是P_i中的最小循环节，那么$\rho(P_i)=i/(i-\pi[i])$.

运行时间取决于π的计算，其时间$T(n)=\Theta(m)$.

b.

$\rho(P_i)=r$的概率为$\frac{2^{i/r}}{2^i}$（r能整除i时）.

因为$\rho(P_i)>r$的概率为 $\sum_{r'<r,r'|i}\frac{2^{i/r'}}{2^i}=\frac{1}{2^i}\sum_{r'<r,r'|i}2^{i/r'}=\frac{1}{2^i}\sum_{k=1}^{i/r}2^k=\frac{2^{i/r}-1}{2^{i-1}}\leq 2^{i/r-i+1}$

所以 $P(\rho^*(P)=r)\leq P(\rho^*(P)\geq r)\leq\sum_{i=0}^{m}2^{i/r-i+1}\leq 2^{2-1/r}$

因为$E(\rho^*(P))$是与m无关的常数，即$E(\rho^*(P))=O(1)$.

c.

因为s始终在增加且最多为n-m，而在每次s增加前q最多增加$\rho^*(P)$次，类似KMP算法我们可以得出 $T=O(\rho^*(P)n+m)$.

因为该算法在从左至右扫描T的过程中会扫到所有可能与P_k（k满足$P_k=max\rho(P_i)$）匹配的位置，也就是所有可能与P匹配的位置，所以能找出P在T中所有出现位置.