主定理的应用

求解递推方程：例1

例1 求解递推方程

$$T(n) = 9T(n/3) + n$$

解 上述递推方程中的

$$a = 9,\quad b = 3,\quad f(n) = n$$

$$n^{\log_3 9} = n^2,\quad f(n) = O(n^{\log_3 9-1})$$

相当于主定理的case1，其中ε=1.

根据定理得到 $T(n) = \Theta(n^2)$

求解递推方程：例2

例2 求解递推方程

$$T(n) = T(2n/3) + 1$$

解 上述递推方程中的

$$a = 1,\ b = 3/2,\ f(n) = 1,$$

$$n^{\log_{3/2} 1} = n^0 = 1$$

相当于主定理的Case2 .

根据定理得到 $T(n) = \Theta(\log n)$

求解递推方程：例3

例3 求解递推方程

$$T(n) = 3T(n/4) + n\log n$$

解 上述递推方程中的

$$a = 3,\quad b = 4,\quad f(n) = n\log n$$

$$n\log n = \Omega(n^{\log_4 3+\varepsilon}) \approx \Omega(n^{0.793+\varepsilon})$$

取 $\varepsilon = 0.2$ 即可.

条件验证

要使 $a\,f\,(n/b) \le c\,f\,(n)$ 成立，

代入 $f\,(n) = n\log n$，得到

$$\underline{3\,(n/4)\log(n/4) \le c\,n\log n}$$

只要 $\underline{c \ge 3/4}$，上述不等式可以对所有充分大的n 成立. 相当于主定理的 **Case3.**

因此有 $T(n)=\Theta(f\,(n))=\Theta(n\log n)$

递归算法分析

二分检索： $W(n)=W(n/2)+1,\ W(1)=1$

$$a=1,\ b=2,\ \underline{n^{\log_2 1}=1,\ \ f(n)=1},$$

属于Case2，

$$W(n)=\Theta(\log n)$$

二分归并排序：

$$W(n)=2W(n/2)+n-1,\ W(1)=0$$

$$a=2,\ b=2,\ \underline{n^{\log_2 2}=n,\ \ f(n)=n-1}$$

属于Case2，

$$W(n)=\Theta(n\log n)$$

不能使用主定理的例子

例4 求解 $T(n)=2T(n/2)+n\log n$

解 $a=b=2$, $n^{\log_b a}=n$, $f(n)=n\log n$

不存在$\varepsilon>0$ 使下式成立

$$n \log n = \Omega(n^{1+\varepsilon})$$

不存在 $c<1$ 使 $af(n/b)\le cf(n)$对所有充分大的 n 成立

$$2(n/2)\log(n/2)=n(\log n-1) \le cn\log n$$

递归树求解

$$n\log n$$

$$\frac{n(\log n-1)}{2} \qquad \frac{n(\log n-1)}{2}$$

$$\frac{n(\log n-2)}{4} \quad \frac{n(\log n-2)}{4} \quad \frac{n(\log n-2)}{4} \quad \frac{n(\log n-2)}{4}$$

$$\cdots\cdots\cdots$$

$$n\log n$$

$$n(\log n-1)$$

$$n(\log n-2)$$

$$\vdots$$

$$n(\log n-k+1)$$

求和

$$
\begin{aligned}
&T(n)\\
&= n\log n + n(\log n - 1) + n(\log n - 2)\\
&\quad + \ldots + n(\log n - k + 1)\\
&= (n\log n)\log n - n(\underline{1 + 2 + \ldots + k - 1})\\
&= n\log^2 n - n\underline{k(k-1)/2} = O(n\log^2 n)
\end{aligned}
$$

小结

- 使用主定理求解递推方程需要满足什么条件？
- 主定理怎样用于算法复杂度分析？