递归式复杂度计算

例 如：

归并排序

采用分治算法计算归并计算复杂度的有如下的递归关系式

$$T(n) = 2T(n/2) + \Theta(n)$$

怎样比较各个分治算法的效率?

- 目前的一些方法:
  - 代换法
  - 递归树方法
  - 主方法

怎样解递归式?

递归式复杂度计算 | 代换法

忽略技术细节

$$T(n) = 2T(n/2) + \Theta(n).$$

- 假设n可能是非整数.
- 忽略掉上取整和下取整，即$n/2$可能是非整数.
- 忽略掉边界条件:

$$\begin{aligned} T(1) &= \Theta(1) \\ T(n) &= \Theta(1) \end{aligned}$$

对应足够小的问题

在进行分析时先忽略这些细节，而后再确定其重要与否

1.代换法

- 代换法（Substitution Method ）
- 主要思想：

1、猜测解的形式.

2、 通过数学归纳法找出使解真正有效的常数.

电子科技大学

代换法例子

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

- 猜测 $T(n)=O(n\lg n)$.
- 证明对某一常数$c>0$，有$T(n)\le cn\lg n$

电子科技大学

代换法例子

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

- 猜测 $T(n)=O(n\lg n)$.
- 证明对某一常数$c>0$，有$T(n)\le cn\lg n$

假设 $T(k)\le ck\lg k$，对$k<n$并
用归纳法证明 $T(n)\le cn\lg n$
(注意: c是相同值!)

代换法例子

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

- 猜测 $T(n)=O(n\lg n)$.
- 证明对某一常数$c>0$，有$T(n)\le cn\lg n$

假设 $T(k)\le ck\lg k$，对$k<n$并

用归纳法证明 $T(n)\le cn\lg n$

(注意: c是相同值!)

$$
\begin{aligned}
T(n) &= 2T(n/2)+n \\
&\le 2c(n/2)\lg(n/2)+n \\
&\le cn\lg(n/2)+n \\
&\le cn\lg n-cn\lg 2+n \\
&= cn\lg n-cn+n \\
&\le cn\lg n
\end{aligned}
$$

只要 $c\ge 1$

代换法名称来源于当归纳假设用较小的值时，用所猜测的值代替函数的解。这种方法很有效，但只能用于解的形式很容易猜测的情形。

代换法例子

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

- 猜测 $T(n)=O(n \lg n)$.
- 证明对某一常数$c>0$，有$T(n)\le cn\lg n$

假设 $T(k)\le ck\lg k$，对$k<n$，
用数学归纳法归纳法证明 $T(n)\le cn\lg n$ $(c>=1)$

用于解的形式很容易猜测的情形。

注意：做一个好的猜测

- 如果某个递归式与先前见过的类似，则可猜测该递归式有类似的解

$T(n)=2T(n/2+10)+n$

类似$T(n)=2T(n/2)+n$?

电子科技大学

注意：做一个好的猜测

- 如果某个递归式与先前见过的类似，则可猜测该递归式有类似的解

$T(n)=2T(n/2+10)+n$

1. 直觉$T(n/2)$和$T(n/2+10)$的差别不大：两者都将n均分为均匀的两半，因此可以猜测$T(n)=O(n\log n)$，再用代换法验证
2. 先证明递归式的较松的上下界，$T(n)=\Omega(n)$ and $T(n)=O(n^2)$然后再缩小不确定性区间，则猜测 $T(n)=\Theta(n\lg n)$

作 业

证明：$T(n)=2T(n/2+10)+n=\Theta(n\lg n)$

讨 论

证明：$T(n)=4T(n/2)+n$

猜测 $T(n)=O(n^2)$对不对，为什么？

作 业

证明：$T(n)=2T(n^{1/2})+\lg n$

提示：猜测T(n)的解是什么？如何用数学归纳法证明?