回顾

算法复杂性分析过程：

问题提出 — T（N,I)

↓

选用数学模型 — $T(N,I)=\sum_{i=1}^{k} t_i e_i(N,I)$

↓

对（N,I）分析 — 最坏，最好，平均

↓

渐进性态 — $\tilde{T}(N)$

渐进分析

渐近分析的符号

在下面的讨论中，对所有n，$f(n) \geq 0$，$g(n) \geq 0$。

（1）渐近上界记号O

$O(g(n)) = \{ f(n) \mid$ 存在正常数c和n_0使得对所有$n \geq n_0$有：$0 \leq f(n) \leq cg(n) \}$

（2）渐近下界记号Ω

$\Omega(g(n)) = \{ f(n) \mid$ 存在正常数c和n_0使得对所有$n \geq n_0$有：$0 \leq cg(n) \leq f(n) \}$

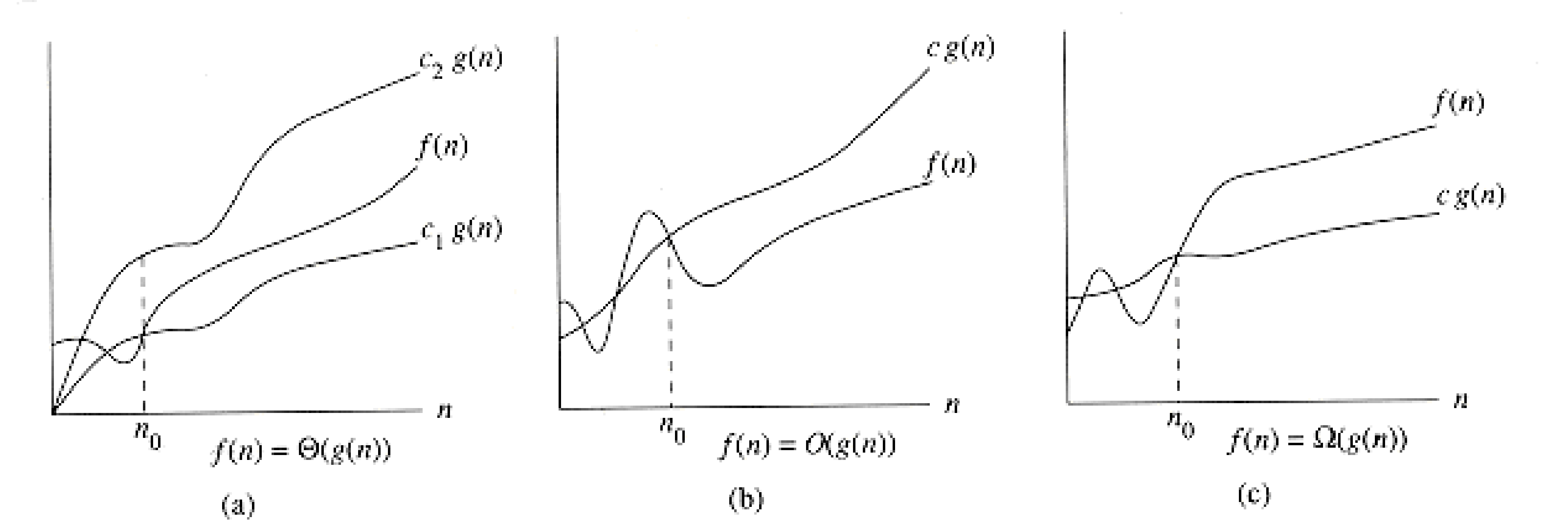
（3）紧渐近界记号Θ

$\Theta(g(n)) = \{ f(n) \mid$ 存在正常数c_1, c_2和n_0使得对所有$n \geq n_0$有：$c_1 g(n) \leq f(n) \leq c_2 g(n) \}$

如果 f(n)是集合 O(g(n))中的一个成员，我们说f(n) 属于 O(g(n))

电子科技大学

渐近分析的符号



$f(n)=\Theta(g(n))$	$f(n)=O(g(n))$	$f(n)= \Omega(g(n))$
$\Updownarrow \cong$	$\Updownarrow \cong$	$\Updownarrow \cong$
$f(n)=g(n)$	$f(n) \le g(n)$	$f(n) \ge g(n)$

更多渐近分析的符号

在下面的讨论中，对所有n，$f(n) \geq 0$，$g(n) \geq 0$。

（4）非紧上界记号o

$o(g(n))$ = { $f(n)$ | 对于任何正常数$c>0$，存在正数和$n_0 > 0$使得对所有$n \geq n_0$

有：$0 \leq f(n) < cg(n)$ }

等价于 $f(n) / g(n) \to 0$ ，as $n \to \infty$。

（5）非紧下界记号ω

$\omega(g(n))$ = { $f(n)$ | 对于任何正常数$c>0$，存在正数和$n_0 > 0$使得对所有$n \geq n_0$

有：$0 \leq cg(n) < f(n)$ }

等价于 $f(n) / g(n) \to \infty$ ，as $n \to \infty$。

渐近分析中函数比较

$f(n) = O(g(n)) \rightarrow a \le b;$

$f(n) = \Omega(g(n)) \rightarrow a \ge b;$

$f(n) = \Theta(g(n)) \rightarrow a = b;$

$f(n) = o(g(n)) \rightarrow a < b;$

$f(n) = \omega(g(n)) \rightarrow a > b.$

算法复杂性分析

例如：$N>=1, 3N<=4N$, 有$3N=O(N)$

$N>=1, N+1024<=1025N$,有$N+1024=0$（N）

$N>=10$，$2N^2+11N-10<=3N^2, 2N^2+11N-10=0(N^2)$

$N>=1, N^2<N^3, N^2=O(N^3)$

证明：　$N^3 \neq O(N^2)$　。

假设存在正的常数C和自然数N_0，使得当$N>=N_0$时有$N^3<=CN^2$。显然，当取$N=\max\{N_0, \lfloor C \rfloor+1\}$时不等式不成立，所以得证

再次分析

$100n^2=1.5n^2=n^2+4=10n^2-3n+6=\Theta(n^2)$.

课堂讨论

- 根据所学的渐进分析方法，同学们分析下下面的结论是否正确

例如： $f(n) = 32n^2 + 17n + 32.$

$f(n)$ 属于 $O(n^2)$, $O(n^3)$, $\Omega(n^2)$, $\Omega(n)$, $\Theta(n)$.

$f(n)$ 不属于 $O(n)$, $\Omega(n^3)$, $\Theta(n^2)$, or $\Theta(n^3)$.

渐进分析的算术运算：

- $O(f(n))+O(g(n)) = O(\max\{f(n),g(n)\})$；
- $O(f(n))+O(g(n)) = O(f(n)+g(n))$；
- $O(f(n))*O(g(n)) = O(f(n)*g(n))$；
- $O(cf(n)) = O(f(n))$；
- $g(n)= O(f(n)) \Rightarrow O(f(n))+O(g(n)) = O(f(n))$。

- 规则$O(f(n))+O(g(n)) = O(\max\{f(n),g(n)\})$ 的证明:
- 对于任意$f_1(n) \in O(f(n))$，存在正常数c_1和自然数n_1，使得对所有$n \geq n_1$，有 $f_1(n) \leq c_1 f(n)$ 。
- 类似地，对于任意$g_1(n) \in O(g(n))$，存在正常数c_2和自然数n_2，使得对所有 $n \geq n_2$，有$g_1(n) \leq c_2 g(n)$ 。
- 令$c_3=\max\{c_1, c_2\}$， $n_3 =\max\{n_1, n_2\}$，$h(n)= \max\{f(n),g(n)\}$ 。
- 则对所有的 $n \geq n_3$，有
- $f_1(n) + g_1(n) \leq c_1 f(n) + c_2 g(n)$
- $\leq c_3 f(n) + c_3 g(n) = c_3(f(n) + g(n))$
- $\leq c_3 2 \max\{f(n),g(n)\}$
- $= 2c_3 h(n) = O(\max\{f(n),g(n)\})$.