

绪论

计算模型：统一尺度

To measure is to know. If you can not measure it,
you can not improve it. - Lord Kelvin

洞察一个对象的形式性质，把它同它的质料部分完全分开，然后沉思它，
然后判断时间，即这一特殊形式的事物自然要持续的最长时间

邓俊辉

deng@tsinghua.edu.cn

算法分析

❖ 两个主要方面...

❖ **正确**: 算法功能与问题要求一致?

数学证明? 可不那么简单...

❖ **成本**: 运行时间 + 所需存储空间

如何度量? 如何比较?

❖ 将计算成本描述为函数, 比如...

$T_A(P)$ = 算法A求解问题实例P的计算成本

❖ 意义不大, 毕竟...

可能出现的问题实例太多

❖ 如何归纳简化、概括?

❖ 观察: 问题实例的**规模**, 往往是

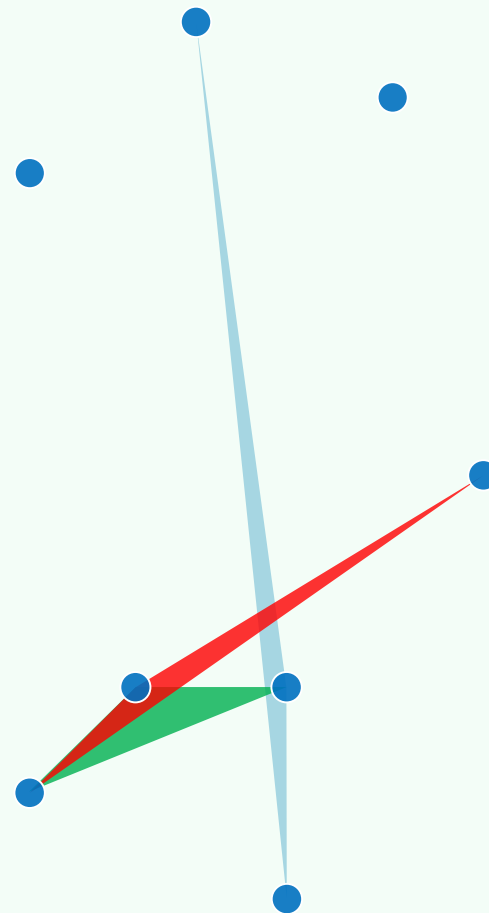
决定计算成本的**最主要**因素

❖ 通常: 规模接近, 计算成本也接近

规模扩大, 计算成本亦上升

特定算法 + 不同实例

- ❖ 令: $T_A(n)$ = 用算法A求解某一问题规模为n的实例, 所需的计算成本
讨论特定算法A (及其对应的问题) 时, 可简记作 $T(n)$
- ❖ 然而: 同等规模的不同问题实例, 计算成本也不尽相同, 甚至差异极大
- ❖ 例如: 任给平面上n个点, 在它们定义的 $\binom{n}{3}$ 个三角形中,
是否某一个的面积不超过5.0?
- ❖ 蛮力: 最坏情况下需枚举所有三角形; 但运气好的话...
- ❖ 既然如此, 又该如何定义 $T(n)$ 呢?
- ❖ 稳妥起见, 取 $T(n) = \max\{ T(P) \mid |P| = n \}$
亦即, 在规模同为n的所有实例中, 只关注最坏 (成本最高) 者



特定问题 + 不同算法

- ❖ 同一问题通常有多种算法，如何**评判**其优劣？
- ❖ 实验统计是最直接的方法，但并不足以准确反映算法的真正效率
 - 不同的算法，可能更适应于不同**规模**的输入
 - 不同的算法，可能更适应于不同**类型**的输入
 - 同一算法，可能由不同**程序员**、用不同**程序语言**、经不同**编译器**生成
 - 同一算法，可能实现并运行于不同的**体系结构**、**操作系统**...
- ❖ 为给出**客观**的评判，需要抽象出一个**理想**的平台或模型
 - 不再依赖于上述种种具体的因素
 - 从而直接而准确地描述、测量并评价算法

