动态规划
算法设计

动态规划设计要素

1. 问题建模，优化的目标函数是什么？约束条件是什么？
2. 如何划分子问题（边界）？
3. 问题的优化函数值与子问题的优化函数值存在 着什么依赖关系？(递推方程)
4. 是否满足优化原则？
5. 最小子问题怎样界定？其优化函数值，即初值等于什么？

矩阵链相乘

问题：设 $A_1, A_2, \ldots, A_n$ 为矩阵序列，A_i 为 $P_{i-1} \times P_i$ 阶矩阵，$i = 1, 2, \ldots, n$. 试确定矩阵的乘法顺序，使得元素相乘的总次数最少.

输入：向量

$$P = < P_0, P_1, \ldots, P_n >,$$

其中 $P_0, P_1, \ldots, P_n$ 为 n 个矩阵的行数与列数

输出：矩阵链乘法加括号的位置.

矩阵相乘基本运算次数

矩阵A： i 行 j 列，B: j 行 k 列， 以元素相乘作基本运算,计算 AB的工作量

$$
\begin{bmatrix} & \cdots & \\ \boxed{a_{t1}\ a_{t2}\ \ldots\ a_{tj}} \\ & \cdots & \end{bmatrix}
\begin{bmatrix} \vdots & \boxed{\begin{matrix} b_{1s} \\ b_{2s} \\ \vdots \\ b_{js} \end{matrix}} & \vdots \end{bmatrix}
=
\begin{bmatrix} & \boxed{c_{ts}} & \\ \end{bmatrix}
$$

$$c_{ts} = a_{t1}\,b_{1s} + a_{t2}\,b_{2s} + \ldots + a_{tj}\,b_{js}$$

AB: i 行 k列， 计算每个元素需要做 j 次乘法，总计乘法次数 为 ijk

实例

实例： $P = <10, 100, 5, 50>$

A_1: 10×100, A_2: 100×5, A_3: 5×50,

乘法次序

$(A_1A_2)A_3$: $10\times100\times5+10\times5\times50 = \boxed{7500}$

$A_1(A_2A_3)$: $10\times100\times50+100\times5\times50=75000$

第一种次序计算次数最少.

蛮力算法

加n个括号的方法有 $\frac{1}{n+1}\binom{2n}{n}$ 种，

是一个Catalan数，是指数级别

$$W(n)=\Omega(\frac{1}{n+1}\frac{(2n)!}{n!n!})$$

代入
Stirling
公式

$$=\Omega(\frac{1}{n+1}\frac{\sqrt{2\pi 2n}(\frac{2n}{e})^{2n}}{\sqrt{2\pi n}(\frac{n}{e})^{n}\sqrt{2\pi n}(\frac{n}{e})^{n}})=\Omega(2^{2n}/n^{\frac{3}{2}})$$

动态规划算法

- 子问题划分

  $A_{i..j}$：矩阵链 $A_i A_{i+1} \dots A_j$，边界i, j

  输入向量：$<P_{i-1}, P_i, \dots, P_j>$

  其最好划分的运算次数：$m[i, j]$

- 子问题的依赖关系

  最优划分最后一次相乘发生在矩阵 k 的位置，即

  $$A_{i..j} = A_{i..k}\ A_{k+1..j}$$

  $A_{i..j}$最优运算次数依赖于$A_{i..k}$与$A_{k+1..j}$的最优运算次数

优化函数的递推方程

递推方程：

$m[i,j]$：得到 $A_{i..j}$ 的最少的相乘次数

$$\underline{m[i,j]} = \begin{cases} 0 & i = j \\ \min\limits_{i \le k < j} \{\underline{m[i,k]} + \underline{m[k+1,j]} + P_{i-1}P_kP_j\} & i < j \end{cases}$$

A_i … A_k	A_{k+1} … A_j
$P_{i-1} \times P_k$	$P_k \times P_j$

该问题满足优化原则

小结

动态规划算法设计要素

- 多阶段决策过程，每步处理一个子问题，界定子问题的边界
- 列出优化函数的递推方程及初值
- 问题要满足优化原则或最优子结构性质，即：一个最优决策序列的任何子序列本身一定是相对于子序列的初始和结束状态的最优决策序列