§7.2 动态规划【自底向上】

一．思想：

有些问题不能分成少数几个子问题，使得这些子问题的解又可以组合成原问题的解。

我们可以把原问题按需要分成尽可能多的子问题，把子问题再分成子问题，如此下去。按这样做，我们可能得到指数时间的算法。

常常只有多项式数量的不同子问题，其中有些子问题我们需要解多次。我们可以把不同子问题的解保留在一张表中，需要时查一下表，这样我们就可以得到多项式时间的算法。这种方法叫作动态规划。

二．例1：世界系列差别赛

1. 问题：有两个队A和B进行一系列比赛，看哪个队先赢得n场比赛。

设是A队需要再赢场(则A队赢)，B队需要再赢场(则B队赢)，这时A队赢的概率。

我们假设A队和B队有相同的竞争力，各有50%赢某场的概率。

例如：，A队需要再赢场，B队需要再赢场，则 。

2. 的递归式

\*解释上式

解以上递归方程。令，是调用的最大次数，由(7.1)式，

和为常数，

解得。即为。即用递归过程计算需指数时间。实际上可证为数量级。

但递归计算有重复计算。

例如：计算需要计算和，而计算和，都需计算。

1. 列表计算

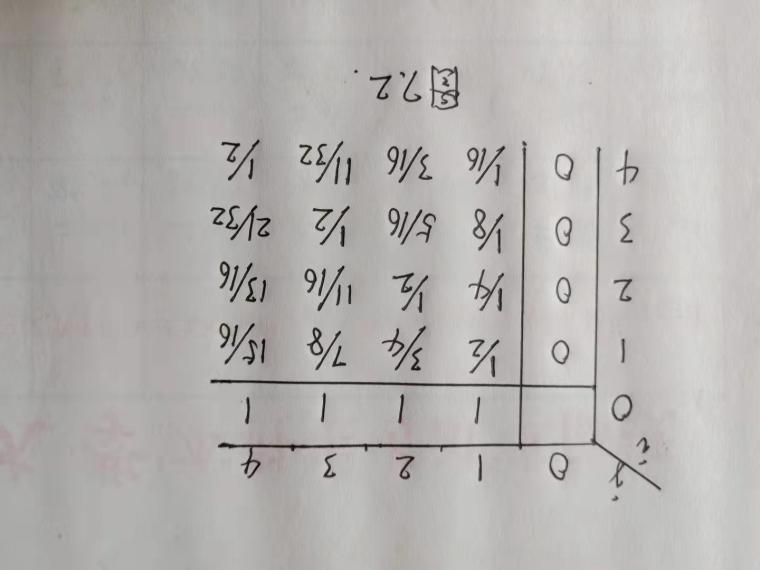


图7.2对角线顺序填表，每一个元素是其上边和左边相加÷2

\*解释计算规则

4. 算法：

FUNCTION odds (i , j : integer) : real;

VAR

s, k : integer;

BEGIN

1. FOR s := 1 TO i+j DO

BEGIN

1. P[0, s] := 1.0 ;
2. P[s, 0] := 0.0 ;
3. FOR k := 1 TO s-1 DO
4. P[k, s-k] := (P[k-1, s-k] + P[k, s-k-1])/2.0 ;

END;

1. RETURN( P[i, j] );

END;

5. 算法分析：

4-5行 : O(s)

2-3行 : O(1)

总时间 : 。

三．例2：多边形三角化问题

1. 问题：给出一个多边形，在不相邻的顶点对之间加边，使得每个区域成为三角形，并使所加边(弦)的长度之和最小。

2. 例子：

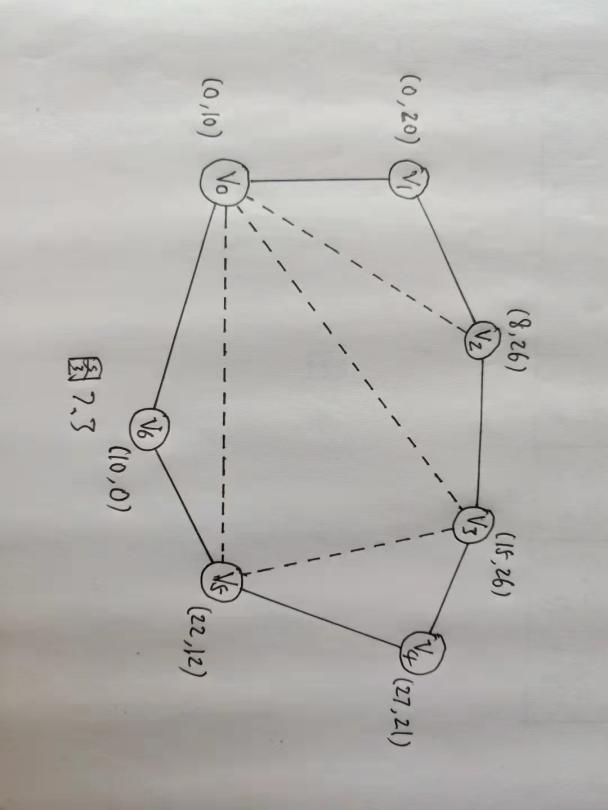


图7.3

距离计算用欧几里德距离。

例如：

\*图7.2的三角化不是最小的。

3. 求解方法：设在多边形上按顺时针方向排列。

几个事实：

1. 在一个多边形(多于三条边)的三角化中，任意两个相邻点中有一个顶点与一条弦关联。
2. 如果是三角化中的一条弦，那么必有某个，使得和或者是多边形上的一条边，或者是一条弦。

\*开始取两个相邻顶点：和，其中至少有一个与弦关联，不妨设与弦关联。存在与相邻（两个点通过弦相连）。则和为两个多边形。再对它们进行三角化。有种选择。

定义具有s个顶点，从顶点开始的子问题为，将该多边形最小化。

解有三种选择：

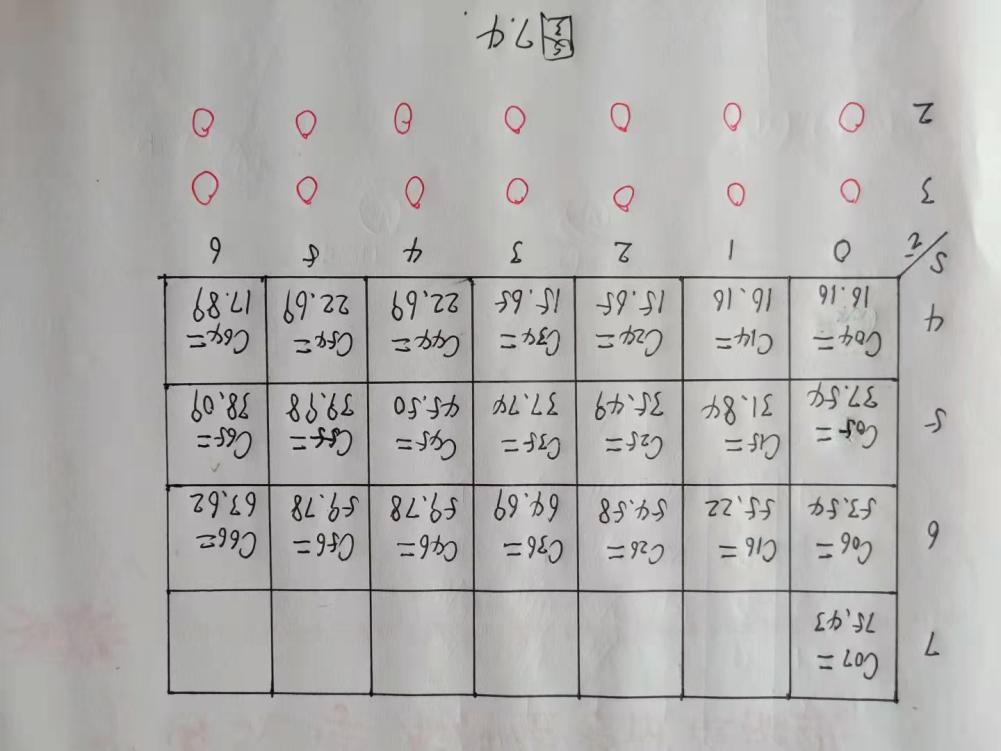
1. 取弦, 得三角形和子问题。
2. 取弦, 得三角形和子问题。
3. 对，取弦和，得三角形和两个子问题和。

\*图示

用递归方法需时间。

实际上只有个不同子问题。

令为将最小三角化时各弦的长之和。

其中：为弦的长度。当和在多边形上相邻时，。当时，。

4. 列表计算

图7.4

\*找出取表元运算的规律

5. 从表中找解：

需把最优解记在表中，每个表项记住最优解的k值，取弦和。

6.算法分析：

共有个不同子问题，求最小值需时间。计算时间复杂性为。

四．例3：矩阵连乘问题

1. 问题：给定n个矩阵，其中和是可乘的，。考查。由于矩阵乘法满足结合律，故有多种不同的计算次序。

例如：可以按以下次序计算

, , ,

,

共需次乘法。

n个矩阵连乘，不同的相乘次序计算量是不一样的。例如：

设分别是的矩阵。

需要次乘法；而

需要次乘法。

第二种乘法是第一种乘法计算量的10倍。

问题：如何安排的相乘次序，使得计算量最小？

2. 不同计算次序的个数

设是的矩阵，。设个矩阵连乘，不同计算次序有个。

由于可以先在第个和第个矩阵之间将原矩阵序列分为2个矩阵子序列，。然后分别对这两个矩阵子序列完全加括号，最后在两个子序列外加括号。因此的递推关系式如下：

解得是Catalan数，即，其中

是指数数量级的。（看所有乘积顺序再找最小时间花费）

3. 算法：

设表示。设计算的最优计算次序，在与之间将矩阵链断开，。

这个问题的一个关键特征是：计算的最优次序所包含的计算矩阵子链和的次序也是最优的。

设计算所需的最少乘法次数为。因此，

设计算的最小值，在k位置断开链，，故

。

计算只有个不同子问题，。

因此可用动态规划算法来计算。

4. 例子：设要计算矩阵连乘。

其中：

图7.5

5. 算法时间复杂度

共有个不同子问题，，每次计算至多需要时间，故总的计算时间为。

五．动态规划算法的基本要素

1. 最优子结构：

若的完全加括号方式在和之间将矩阵链断开，则由此确定的子链和的完全加括号方式也是最优的。

2. 重叠子问题：

在用递归算法自顶向下求解问题时，每次产生的子问题并不总是新问题，有些子问题被反复计算多次。不同子问题只有多项式个。

作业14：

1.设计一个时间的算法，找出由个数组成的序列中最长非递减子序列。

2.设计算个元素中取个元素的组合数的公式如下：

如果或

如果

1. 给出计算的递归函数；
2. 上述递归函数计算的最坏运行时间是什么？
3. 写出计算的动态规划算法；
4. (c)算法的运行时间是什么？