图解凸N边形划分问题—卡特兰数的一种应用

原创 软件研究所 夏宇 LOA算法学习笔记 2020-11-16 20:14

01 问题描述

给出一个N条边的凸多边形,我们可以将其划分为多少个分开的三角形?
N为4时，有两种不同的划分，N为5时，有五种不同的划分。
给出算法计算将凸N边形划分为三角形的方法数。

02 解决方案

求解过程:
从一个初学者的角度，从最“笨”的办法开始想，描述下我解这道题的思路和障碍：**如嫌啰嗦直接跳至“一种正确的解法”。**

1) 最简单的case会不会做?

N=3或4时。N=3显然结果为1；N = 4，既然是划分，那么就找能划分多边形的连线，原有的边不能划分，就找对角线，四边形有两条对角线，两种划分，会做。那问题就变成了找对角线的问题呗，猜想可以划分为多个多边形，然后将划分后各个区域的可能数相乘。

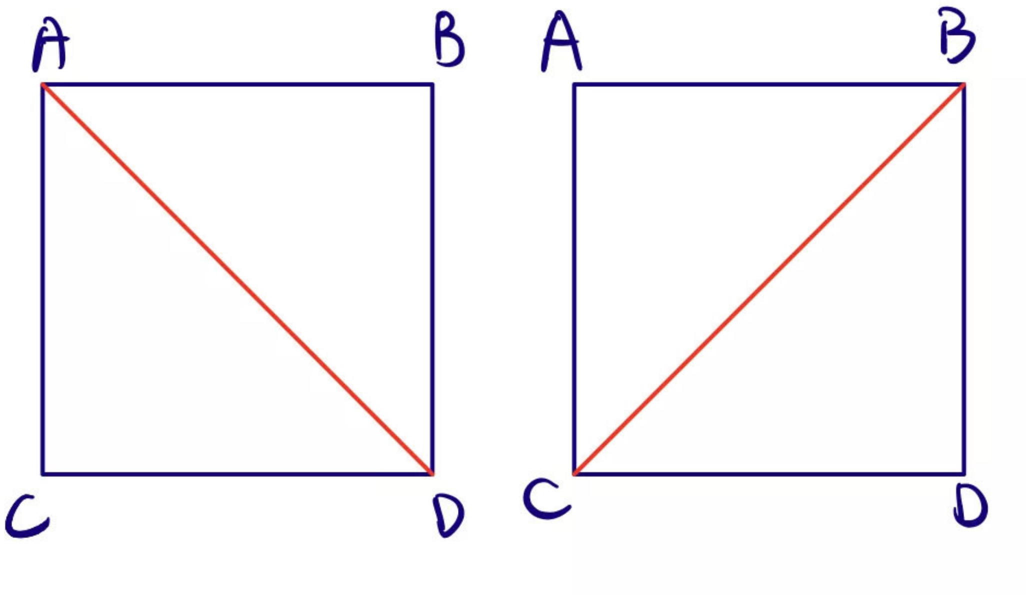

图1 分别以对角线AD、BC划分

2) 加大难度，N=5呢?

不难发现此时就出现了问题，还是用连线来划分的思路。连接AC后五边形被划分为了一个四边形和三角形，在四边形里，我们直接用之前的结论f(4) = 2；连接AD后情况类似，所以以A为顶点，连接不相邻的边有2+2种划分？不是的，从下图可以看出两种划分的冗余。

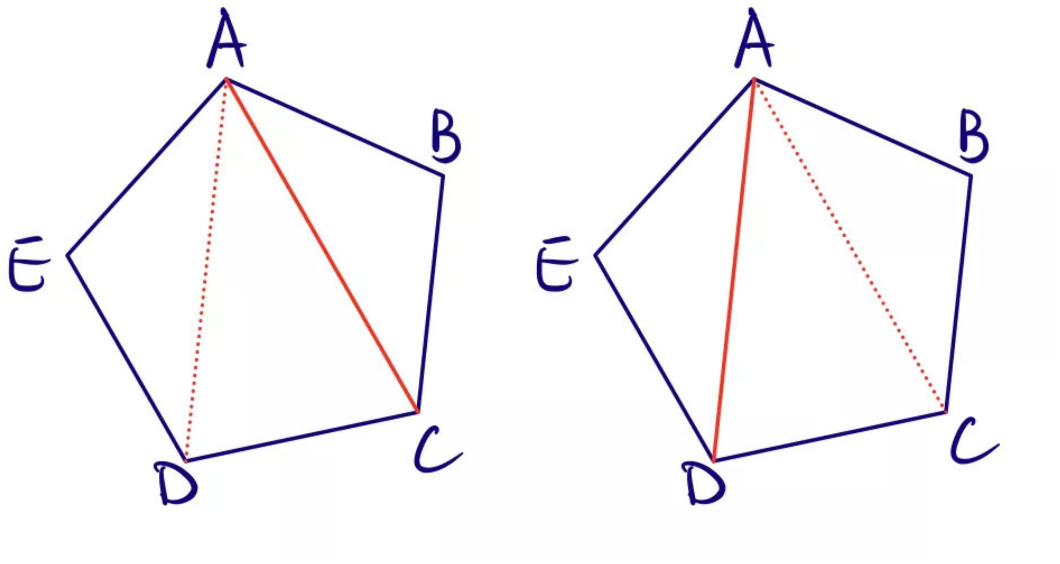

图2 分别以对角线AC、AD划分

而且以这种思路，要找到所有的对角线，当连接CE时，发现也存在冗余，笔者没有想到合适的办法来解决这个问题，因此改变思路。

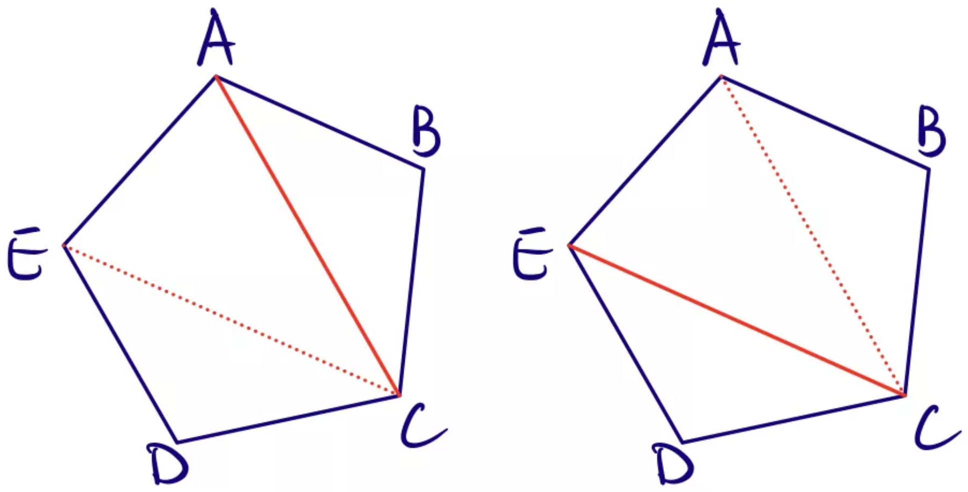

图3 分别以对角线AC、CE划分

将多边形划分为三角形，没法根据对角线划分，那三角形可以由什么来确定？三条边可以确定一个三角形，那在本题中有以下两种情况，两条边和一条对角线；或是两条对角线，一条边，如何记录这些对角线和三角形呢？又是一个问题，这条路又放弃了。

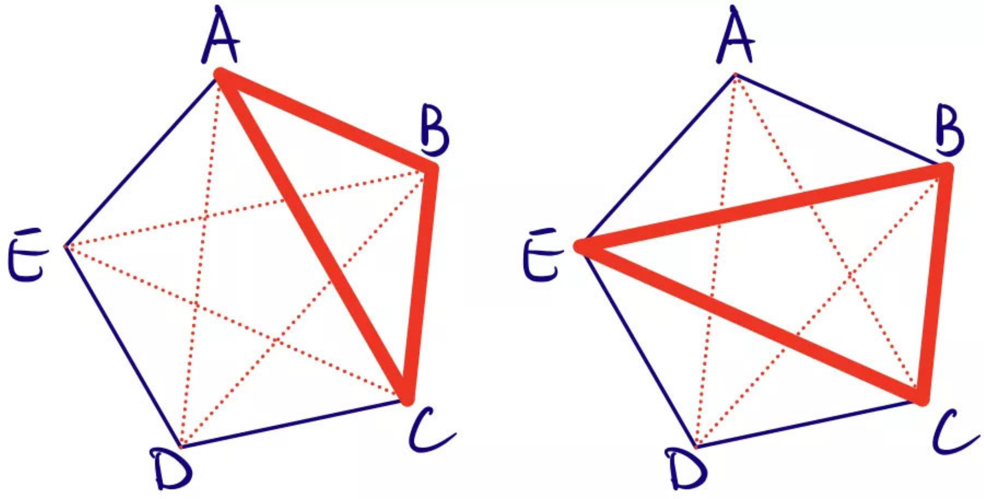

图4 尝试用边组成划分后的三角形

一种正确的解法：

三条边的路走不通，不妨试试由三个点来确定一个三角形。如下图要想将一个多边形完全划分为三角形，**不管我们怎么画，一定有两个点是挨着的，否则就不是正确的划分**，两个点挨着，也就是构造三角形时必用到一条已有的边。一条边为基准，找另一个点构成三角形，以这种思路展开。

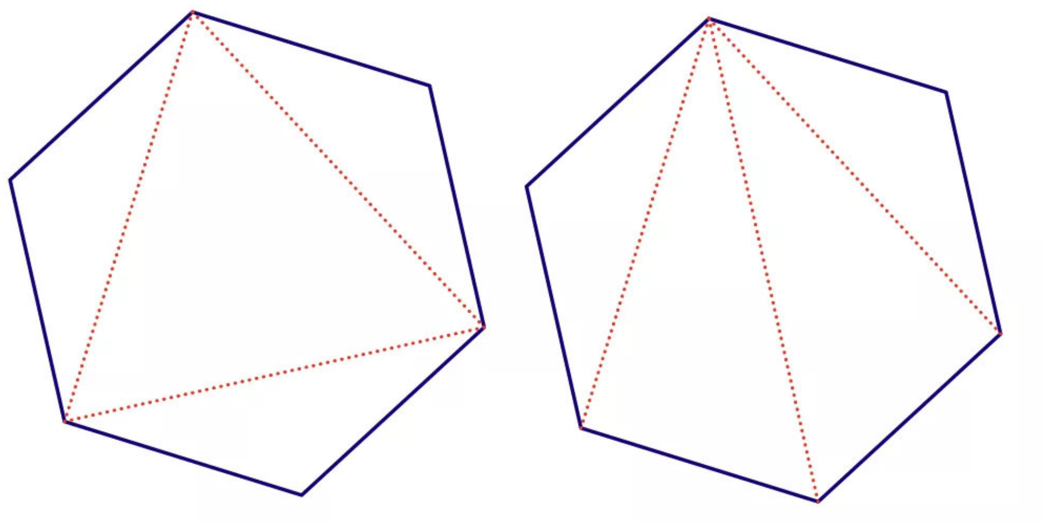

图5 两个点为什么必须相邻

1）最简单的case，会不会做？N=4

固定AB，分别选择C或D构成三角形，正确划分，会做。

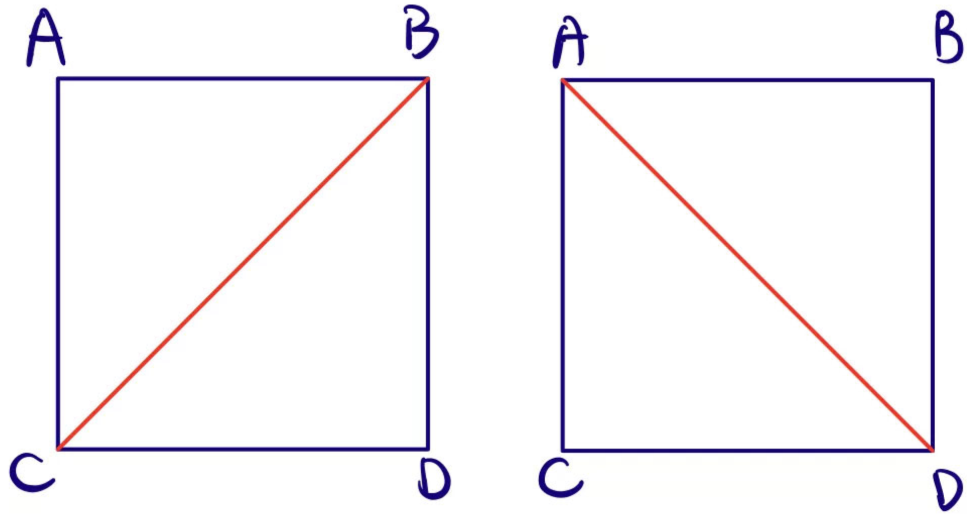

图6 N=4固定AB时的两种情况

2）加大难度，N = 5

固定AB，分别选择和A，B相邻的点C，E，问题都是划分为f(N-1)= f(4)。

固定AB，选择D，此时三角形ABD将五边形划分为三个三角形，仅此一种划分，所以：

$$F(5)=F(4)+F(3)*F(3)+F(4) = 5$$

共五种划分。

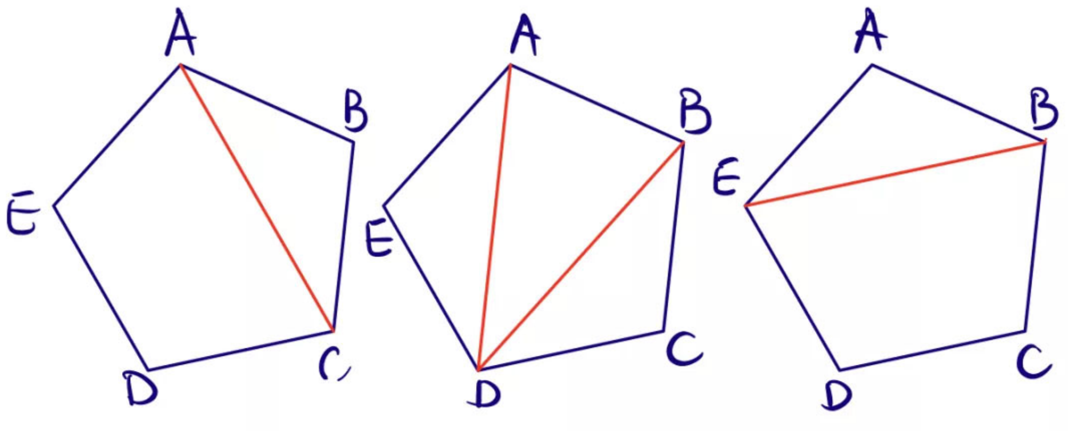

图7 N=5固定AB时的三种情况

没发现什么规律，再画一种，N=6时：

固定AB，选择C => f(N-1)= f(5)；

固定AB，选择D=> f(4)*f(3)；

固定AB，选择E=> f(3)*f(4)；

固定AB，选择F=>f(N-1) = f(5)；

F(6) = F(5) + F(3)*F(4)+ F(4)*F(3)+F(5)

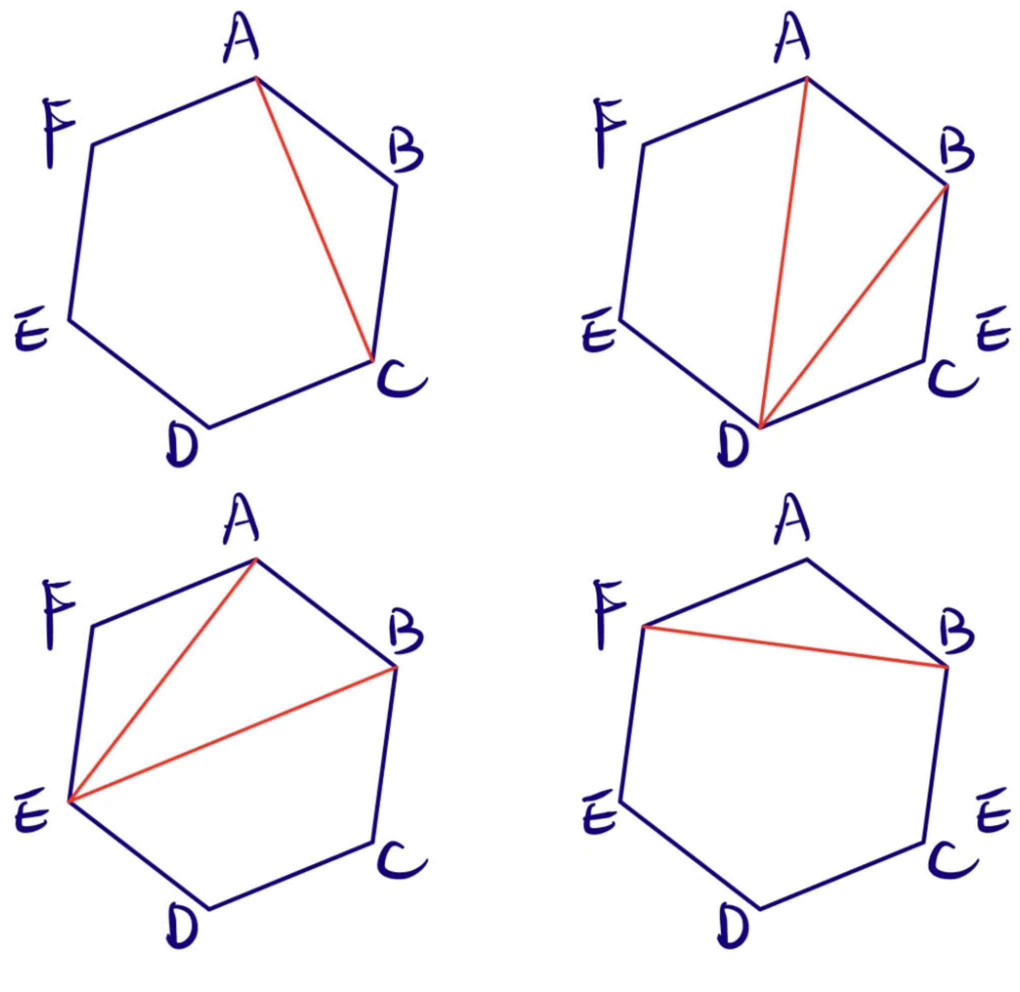

图8 N=6固定AB时的四种情况

3）N更大时

固定AB，选择和AB相邻的点，两个都是f(N-1)。

选择不相邻的点，对于N边形，不难找出规律，划分如图。

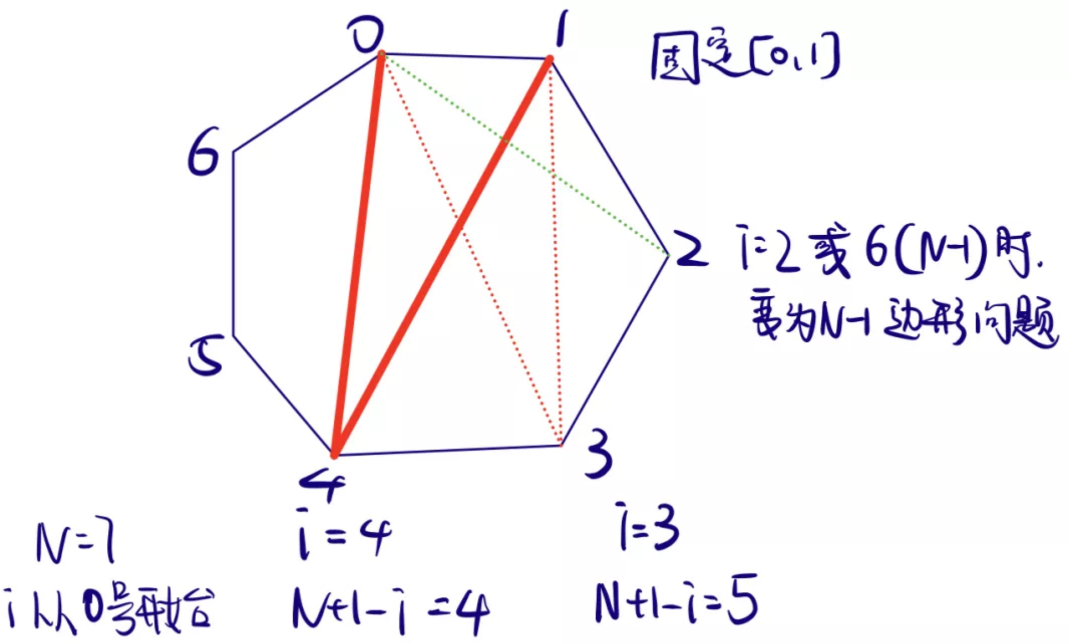

图9 固定AB令i=3或4的情况

我们可以在N逐渐增大时，保存每一步的结果，避免过多冗余的计算，最终结果(代码中因为数组是从0开始存的所以返回p[N-1])：

$$F(N) = 2 * F(N-1) + \sum_{3}^{N-2} F(\mathrm{i}) * F(N+1-\mathrm{i})$$

03 代码实现

```
int countTriangles(int n,int *p){
    if(n == 3)
        return 1;
    if(p[n - 1] == -1)//没算过，那就算一遍，存数组里,
    {
        p[n-1] = countTriangles(n-1,p)*2;//和指定的边相邻的两个点
        for(int i = 3; i < n-1; i++) //i从0开始算，i=2表示第三个点，也就是挨着初始边的点
            p[n-1]+=countTriangles(i,p)* countTriangles(n-i+1, p) ;
    }
    return p[n-1];
}
```

时间复杂度：相当于对数组的遍历O(N)

空间复杂度：用到了一个数组来记录某边形的结果O(N)

04 扩展（卡特兰数）

上文最后推导出的公式的是符合本题实际意义的，因为不存在两边形，所以N至少为3；如果令F(2)= 1，则也可以写成：

$$\begin{aligned} F(N) &= F(2) * F(N-1) + F(3) * F(N-2) + \ldots + F(N-1) * F(2) \\ &= \sum_{2}^{N-1} F(\mathrm{i}) * F(N+1-\mathrm{i}) \end{aligned}$$

这其实是一种著名数列，卡特兰数在多边形划分问题上的应用，下面给出卡特兰数的递推式：

$$f(n)=\sum_{i=0}^{n-1} f(i)\times f(n-i-1)$$

其通项有以下两种表示，其中后者在实际应用中更为常用：

$$f(n)=\frac{C_{2n}^{n}}{n+1}$$

$$f(n)=C_{2n}^{n}-C_{2n}^{n-1}$$

卡特兰数有多种应用，在维基百科里有更为详细的解释。

值得一提的是，在本学期其他的课程(周四晚的nlp)中也发现了卡特兰数的身影，查阅资料后发现是HW1的题目5竟是其应用。结合卜老师周五课上的建议，才有了这篇文章，希望能给您带来帮助。

宗成庆：《自然语言处理》讲义，第 9 章 9/15

9.2 短语结构分析

英语中的结构歧义随介词短语组合个数的增加而不断加深的，这个组合个数我们称之为开塔兰数(Catalan number，记作C_N)。

如果句子中存在这样 n (n为自然数)个介词短语，C_N 可由下式获得 [Samuelsson, 2000]：

$$C_N=\binom{2n}{n}\frac{1}{n+1}=\frac{(2n)!}{(n!)^2(n+1)}$$

作为算法上的小白，错谬在所难免，希望您在评论区不吝告知，将不胜感激。