**6.1 简介:结序列,de Boor控制点** 2020年7月23日10点01分

多项式曲线对于CAGD具有很多优点,但在许多方面也不能令人满意:

1. 给定一个包含个边(个顶点)的控制多边形,由这些控制点确定的多项式曲线的度为.我们知道计算曲线上的点需要个步骤.当很大时,这可能太昂贵,因此不切实际.
2. 移动任何控制点都会影响整个曲线.从移动控制点仅影响曲线的较小区域的意义上,希望具有更好的局部控制.
3. 如果我们对插值感兴趣,而不仅仅是对形状进行近似,则必须从曲线上的点计算控制点.这导致了线性方程组,而当曲线的度数较大时,求解这样的系统可能是不切实际的.

由于上述原因,人们考虑将由复杂控制多边形指定的曲线分段为更小和更易于管理的段.这就是样条线背后的想法.在本章中,我们介绍了一类称为B样条的样条曲线,并说明了如何使用结来控制形成样条曲线的曲线段之间的连续程度.我们展示了结序列和de Boor控制点序列如何可用于指定B样条，我们介绍了B样条的de Boor算法以及有用的结插入算法.我们还将讨论使用B样条的插值,并简要介绍基本函数方面更传统的B样条方法.

曲线的每个线段都将由全局多边形的某个小子多边形控制,并且这些线段将以某种平滑的方式连接在一起.为了实现这一点,有必要考虑多个连续的参数区间,而不是像多项式曲线那样考虑单个区间. 例如,我们可以有区间

如果要使用三次分段,则每个区间需要四个控制点.让我们将相应的三次曲线段表示为.由于我们要连接这些三次曲线段,因此我们假设曲线段的第四个控制点等于曲线段的第一个控制点,即曲线段的第四个控制点等于曲线段的第一个控制点,依此类推,最后,曲线段的第四个控制点等于曲线段的第一个控制点.

通常,我们会希望在结点（2、3、5、6）的连续性比单纯的接触更好.正如我们稍后将解释的那样,这可以实现.

为什么我们选择区间的间距不均匀？因为它并不能真正简化任何事情,而且这种灵活性实际上很有用,我们将在后面看到.显然,我们可能还需要更加注意端点(在本例中为1和9).就目前而言,我们可以假设我们正在处理连续区间的无限序列,其中该序列在两个方向上都延伸到无穷大.当看有限的序列时

我们只关注某些双无限序列的特定子序列.也可以处理循环序列.

请注意,仅列出这些区间的连接点来指定区间的顺序稍微更经济,如以下顺序:

这样的序列称为结序列.

我们将看到折叠间隔很有用,因为这是降低连接连续性的一种方法.通过使一个结出现不止一次来捕获.例如,如果我们希望将区间[3,5]折叠为[3,3],则上述结序列变为

我们还允许折叠几个连续的时间间隔.例如,我们也可以将[2,3]折叠为[3,3],以获得序列

结的连续出现次数称为其乘数[multiplicity].因此，在上面的序列中,3具有乘数3.稍微扩展我们的序列,

我们可以折叠几个区间,并得到多个乘数,如将[10,12]折叠为[10,10]所示:

上面的序列具有一个三结3和一个双结10.乘数为1的结称为简单结,乘数大于1的结称为多重结.如我们将看到的,结乘数超过毫无意义,其中是每个曲线段的度数(实际上,乘数对应于与该多重结相关的控制点处的不连续性).

现在的问题是找到一种方便的方法来指定我们想要的每个连接的连续程度,并找到比曲线段的贝塞尔控制点更方便的控制点(在我们的示例中,).幸运的是,两个问题都有很好的答案.**引理5.5.2**将为连接的连续性问题提供一个非常令人愉悦的答案，并且该答案还将表明存在称为de Boor点的自然控制点,事实证明这些控制点比贝塞尔曲线段的控制点更方便(且更经济).我们还将看到如何扩展de Casteljau算法,以便直接从de Boor控制点计算任何曲线段上的点,而不必首先计算给定曲线段的B´ezier控制点.但是,我们可以使用这种称为de Boor算法的算法计算贝塞尔控制点.

由于我们现在使用结序列来表示连续的区间,因此可以通过使用对应于区间开始的结的索引(位置)作为该区间上的曲线段的索引,来简化曲线段的表示法.例如,给定结序列

我们用来表示,简化为和.注意我们需要小心多重结,如序列

我们只考虑严格增加的结的子序列,然后用对应于其区间域左侧的最后一次出现的结的索引(位置)来索引每个曲线段,因此,我们有曲线段:.

现在我们必须解释de Boor控制点是如何产生的.有几种可能的演示.更加数学的表示是考虑与区间相关的每个曲线段的极形式,并找出处连续性施加在极形式和的条件.如果所有曲线段的度数,并且为简单起见,我们允许最多为的结节乘数(尽管很容易容纳个结节),结果证明我们考虑了长度为的连续节点的序列形式

并且对于所有,极形式的值

以结开头的间隔关联的是常数,表示为.这些点是de Boor点.然后,对于任何简单的结,即的结,个连续结的序列

产生个连续结的序列

其中每一个长度为.这些序列证明与中间区间相关的曲线段定义了个De Boor控制点.实际上,如果是的极性形式,则这些de Boor点是极值

其中.

例如,给定以下(一部分)结序列

如果,且,我们有序列

由个节点组成,因此,中间区间为[6,8],,且.

由于,则序列

是渐进式的，因此可以使用第5.3节中介绍的de Casteljau算法的渐进式版本(也称为de Boor算法)来计算曲线上的点,更一般地,计算在参数上的任何极值.下面的6个结点渐进顺序说明了这一点:

观察到,如果是上述的简单结,则个连续结的序列

其中每一个长度为,与序列

的中间区间精确的重叠.

由于Ken Shoemake,此属性是de Boor算法的另一个相当直观的解释的出发点.现在让我们忘记曲线段.取而代之的是,我们将要修正上升的曲线段的最大度m，并假定给定了一个双无限结序列,

为了简单起见,它仅由简单结(即对于所有的)和(不同的)控制点的数量双无限序列

组成,但有一个附加的约束:我们假设从控制点的序列到结序列有一个双射函数,其特性是,如果映射到结,则映射到.

由于连续的控制点映射到连续的结,显然存在仿射双射,将区间映射到线段,定义为

其中.因此,我们可以将控制多边形的每一条边划分为个子段.

例如,假设m = 3,则给出以下(部分)结序列

与前面的说明的联系更加清楚.由于任意两个连续的线段和对应于在结序列的个连续子区间上重叠的区间,因此我们可以通过个结的序列对每个控制点进行索引,其中被映射到.