

大连理工大学

硕士学位论文

三次B样条数据拟合的节点自适应选取算法

姓名：李晓晔

申请学位级别：硕士

专业：计算数学

指导教师：罗钟铉

20080610

摘 要

样条作为数据拟合的有效工具，已经得到了迅速的发展和广泛的应用。

本文研究对于给定散乱数据的三次 B 样条拟合问题。即对于一组数据点，预先给定逼近精度，在三次 B 样条空间寻求函数，使这个函数与数据点距离的平方和小于给定精度。我们通过随机选取节点的方法，拟合出满足逼近精度要求的样条曲线，并在保证精度的条件下对样条节点进行优化，得到概率意义下优化的节点数目和节点分布。大量的数值实验表明通过我们的算法得到的拟合曲线具有逼近精度高，分段均匀，段数少的优点。

本文安排如下：在第一章，介绍文章的研究背景，以及数据拟合基本的逼近方法和逼近工具，提出本文研究的中心问题——三次自由节点 B 样条最小二乘拟合。并对这一问题已有的研究方法和研究成果进行综述。第二章，介绍参数样条的基础知识，为下文的研究做准备。第三章，分别介绍固定节点样条和自由节点样条的最小二乘逼近，着重介绍交替迭代法求解自由节点样条最小二乘问题。第四章，使用随机的方法，给出三次 B 样条最小二乘逼近的节点自适应选取算法。首先从理论上分析方法的可行性，然后给出节点的随机选取方法，将问题转化为固定节点样条的最小二乘逼近问题，并通过线性最小二乘法拟合出符合精度要求的样条曲线。接着对得到的节点向量从数量和分布上进行优化。本章最后给出了用该算法拟合三维空间数据点的数值实验，证明了方法的有效性。

关键词：数据拟合；最小二乘；固定节点；自由节点；样条逼近

An Adaptive Algorithm for Knots of Cubic B-Spline in Data Fitting

Abstract

As efficient in data fitting, spline has developed rapidly and been used extensively.

In this paper, we consider least squares adaptive approach to random data implemented by cubic B-spline, which is a problem concentrating on the construction of a cubic B-spline, whose err sum of squares is less than a given precision. Through choosing the knots randomly, we get the vector of knots that enables the B-spline up to the given precision. We also make the number and the distribution of knots optimized in the sense of probability.

The paper is structured as follows. The following section will introduce the research background, the basic method and tool used in data fitting, and the focus of this paper--least squares splines with free knots. Then we shall present the basic of B-spline as preliminary. Next, least squares splines with fixed knots and free knots will be interpreted respectively with a great emphasis on a algorithm called alternating iteration. Next we shall give an adaptive algorithm for knots of cubic B-spline in data fitting using stochastic method. We shall explain the feasibility of our method theoretically and present an algorithm for choosing the knots randomly, transforming the problem to linear least squares. Also the optimization of knots on the aspects of number and distribution will be given in this section. Finally, we shall examine some test examples and draw conclusions.

Key Words: Data Fitting; Least Squares; Fixed Knots; Free Knots; Spline Approach

独创性说明


作者郑重声明：本硕士学位论文是我个人在导师指导下进行的研究工作及取得研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写的研究成果，也不包含为获得大连理工大学或者其他单位的学位或证书所使用过的材料。与我一同工作的同志对本研究所做的贡献均已在论文中做了明确的说明并表示了谢意。

作者签名： 李晓晔 日期： 2008.6.26

大连理工大学学位论文版权使用授权书

本学位论文作者及指导教师完全了解“大连理工大学硕士、博士学位论文版权使用规定”，同意大连理工大学保留并向国家有关部门或机构送交学位论文的复印件和电子版，允许论文被查阅和借阅。本人授权大连理工大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，也可采用影印、缩印或扫描等复制手段保存和汇编学位论文。

作者签名: 李晓晔

导师签名: 

2008 年 6 月 26 日

1 绪论

1.1 数据拟合

在解决实际问题的生产（或工程）实践和科学实验过程中，通常需要通过研究某些变量之间的函数关系来帮助我们认识事物的内在规律和本质属性，而这些变量之间的未知函数关系又常常隐含在从试验、观测得到的一组数据之中。因此，能否根据一组试验观测数据找到变量之间相对准确的函数关系就成为解决实际问题的关键。

例如在航天、航空、汽车及其它机器制造工业中，有许多的零件轮廓曲线，如飞机机翼、整流罩、螺旋桨、凸轮样板、各种模具及叶片等，其轮廓形状是通过实验或测量的方法得到。这些通过实验或测量得到的数据，常以列表坐标点的形式给出，而不给出方程。常常需要从一组试验观测数据 $\{(x_i, y_i)\}_{i=0}^n$ 之中找到自变量 x 与因变量 y 之间的函数关系，一般可用一个近似函数 $y = f(x)$ 来表示。函数 $y = f(x)$ 的产生办法因观测数据和要求不同而异，通常可采用函数插值与逼近两种办法来实现，后者在工程上又称为数据拟合。

数据拟合主要是考虑到观测数据受随机观测误差的影响，进而寻求整体误差最小、能较好反映观测数据的近似函数 $y = f(x)$ ，此时并不要求所得到的近似函数 $y = f(x)$ 满足 $y_i = f(x_i)$ ， $i = 0, 1, \dots, n$ 。

在这里，数据常常受制于误差（即噪声）。根据问题的要求，逼近的度量可以有不同的标准，逼近的工具（即近似表达的函数类）也必须仔细选择。

1.2 最小二乘逼近

逼近离散数据的技术有很多，比如人工神经网络，支持向量基，遗传算法等等，其中一个基本方法就是曲线拟合，常采用最小二乘拟合。最小二乘法的原理是：求 $f(x)$ ，

使 $\delta = \sum_{i=0}^n (f(x_i) - y_i)^2$ 达到最小。

曲线拟合的实际含义是寻求一个函数 $y = f(x)$ ，使 $f(x)$ 在某种准则下与所有数据点最为接近，即曲线拟合得最好。最小二乘准则就是使所有散点到曲线的距离平方和最小。拟合时选用一定的拟合函数 $f(x)$ 形式，设拟合函数可由一些简单的“基函数”（例如幂函数，三角函数等等） $\varphi_0(x), \varphi_1(x), \dots, \varphi_m(x)$ 来线性表示： $f(x) = c_0\varphi_0(x) + c_1\varphi_1(x) + \dots + c_m\varphi_m(x)$ 。

现在要确定系数 c_0, c_1, \dots, c_m ，使 δ 达到极小。为此，将 $f(x)$ 的表达式代入 δ 中， δ 就成为 c_0, c_1, \dots, c_m 的函数，求 δ 的极小，就可令 δ 对 c_i 的偏导数等于零，于是得到 $m+1$

个方程组，从中求解出 c_i 。通常取基函数为 $1, x, x^2, x^3, \dots, x^m$ ，这时拟合函数 $f(x)$ 为多项式函数。当 $m=1$ 时， $f(x) = ax + b$ 称为一元线性拟合函数，它是曲线拟合最简单的形式。除此之外，常用的一元曲线拟合函数还有双曲线 $f(x) = a + b/x$ ，指数曲线 $f(x) = ae^{kx}$ 等，对于这些曲线，拟合前须作变量代换，转化为线性函数。

已知一组数据，用什么样的曲线拟合最好呢？可以根据散点图进行直观判断，在此基础上，选择几种曲线分别作拟合，然后比较，观察哪条曲线的最小二乘指标 δ 最小。通常以 B 样条作为基底。

1.3 样条曲线最小二乘逼近

尽管拟合方法有很多，但经过实践及综合分析各种方法后，目前倾向于采用三次参数样条函数对数据点进行拟合。样条，是指用压铁对一根弹性细梁加压，使梁通过给定的型值点而模拟出具有力学特性的曲线。在两个支点间的样条函数是一个以二阶导数为系数的三次样条函数，且其一阶导数和二阶导数都是连续的。将分段的三次样条曲线递推到整个型值点范围，便可拟合出整条三次样条曲线（处理过程从略）。三次样条具有一阶、二阶连续，是一种较好的拟合方法。但由于三次样条函数是建立在小挠度曲线的基础上的，处理大挠度时会产生较大的误差，甚至会出现多余的拐点；另外所拟合的曲线随坐标的变化而变化，不具备几何不变性的要求。解决这些缺陷的一个有效方法是采用三次参数样条拟合方法。

三次参数样条也是分段的三次多项式，且一阶、二阶导数连续，整体光滑，所不同的是用参数方程描述。每个三次参数样条段的边界条件为两个端点的位置矢量及切向矢量。当样条段参数 t 的取值范围为 $0 \leq t \leq 1$ 时，可求得曲线段两个端点的边界条件，从而可得任意三次样条段以位置矢量和切向矢量为系数的矢量方程。为使各曲线段光滑连接，各相邻两曲线段公节点处的曲率必须一致（二阶导数相等）的条件，以及整体曲线两端的端点条件，即可求解各样条段的三次参数样条曲线方程，且整体光滑。由于三次参数样条曲线与坐标系的选择无关，具有几何不变性，而且能对大挠度甚至封闭曲线进行拟合，所以被广泛应用。

人们研究样条逼近问题的传统方法是采用固定节点的样条函数。即在节点数目及位置给定的条件下，通过反求 B 样条控制点，得到逼近误差最小的 B 样条曲线。然而，固定节点样条的缺点是逼近误差严重依赖于既定节点，这引发了人们对自由节点样条的研究。即通过合理地选择节点数量及节点位置，构造样条函数，使它的逼近误差达到既定精度。

H. G. Burchard[1]在 1974 年就指出在一致逼近问题中, 自由节点样条比固定节点样条在逼近误差上有很大的提高。但是在此后的很长一段时期里, 自由节点样条并没有被广泛应用, 如何合理地确定样条节点位置一直是困扰人们的一个难题。C. De Boor[2]和 J. R. Rice[3]在 1968 年给出三次样条最小二乘拟合算法, 为后来的研究工作奠定了基础。M. J. D. Powell 研究自由节点样条最小二乘逼近问题的方法是首先用 k 次多项式作为最小二乘拟合的拟合函数, 并计算拟合函数与数值点相应的误差, 然后观察这些误差的分布, 在误差较大的区域设置一到两个节点, 再以上述点为内节点的 k 次样条函数作为拟合函数, 按同样的过程, 计算拟合函数与数值点相应的误差, 在误差较大的区域再设置一到两个节点, 以此类推, 直到得到的样条函数的逼近误差达到精度要求为止。M. J. D. Powell[4]的方法没有一次性地给出所有自由节点。后来许多学者从优化方法的角度考虑自由节点样条的拟合问题。P. D. Loach 和 A. J. Wathen[5]用局部优化的方法, 以自由节点的线性样条为拟合函数, 通过最速下降法求解局部最小值。但这种局部优化算法只能求出目标函数的近似最小值, 而且最终的节点分布严重依赖于初始节点的选取。W.C. Davidon[6]等人通过全局优化的方法考虑自由节点的分布, 将问题转化为一个非线性优化问题与线性最小二乘问题的求解。基本思想是预先给定内节点个数, 给出初始节点向量, 将初始节点向量带入目标函数, 将问题转化为线性问题, 通过最小二乘法求解矛盾方程组得到基函数的系数。再将求得的系数代入目标函数, 通过非线性优化方法(如 DFP 法, GNM 法)求出节点向量, 并且使迭代后的样条函数值与数据值的误差小于迭代前的误差。如此交替迭代下去, 最终得到全局最优解。后来许多学者的工作在这个方法的基础上加以改进。在求全局极小点的时候, 常会在约束区域的边界出现目标函数的驻点, 这些驻点可能是局部极小点或鞍点, 甚至是局部极大点。由于这些点的存在, 使求解发生困难。David L. B. Jupp 通过引入对数变换, 将有界的约束区域变换到无穷空间, 将问题转化为无约束问题, 使这一情况得到较大的改善。H. Schutze 和 T. Schwetlick[7, 8]通过阻尼高斯牛顿法和 Kaufman 逼近避免了对数变换, 但在每一步牛顿迭代过程中都要求解一个线性约束问题。P. Dierckx[9]用高斯牛顿法求得全局最小值, 并通过罚函数法来解决约束区域边界的“多驻点”问题。Gleb Beliakov[10]从确定性全局优化的角度, 利用切角法求得目标函数的全局最小值。这些后续工作使得自由样条最小二乘问题的求解方法日趋完善。

1.4 本文的工作

不同于以往做法, 本文以一种随机的方法考虑这一问题。我们的基本思想是, 首先固定节点数目, 通过随机选取节点, 构造样条曲线。在这个数目下, 如果在概率 1 的意

义下不存在这样的节点向量使样条函数达到给定精度（即在这个数目下，我们尝试所有的节点分布，得到的拟合样条函数值与数据值的误差的最小值大于给定的精度），我们就增加节点的数目，继续上述过程；如果存在满足精度的节点向量，我们就在保证精度的条件下对节点的数目进行优化，优化的原则是尽可能地使节点数量少，分布均匀，最终得到概率意义下优化的节点数目和节点分布。

我们选择这个方法的原因有两点。第一，通过不断地增加样条节点，并在每一确定的节点数目下，尝试所有可能的节点分布，在理论上，一定会找到符合精度要求的样条函数。所以，我们的方法具有可行性。第二，在固定节点数下，随机选取节点相当于固定节点样条逼近问题，而这是一个线性最小二乘问题，用最小二乘法求解矛盾方程组，计算简单，易于程序实现。从大量的数值实验可以看出我们的算法是简单有效的。

2 基础知识

2.1 基本概念

样条是一类分段（片）光滑、并且在各段交接处也有一定光滑性的函数。样条一词来源于工程绘图人员为了将一些指定点连接成一条光滑曲线所使用的工具，即富有弹性的细木条或薄钢条。由这样的样条形成的曲线在连接点处具有连续的坡度与曲率。分段低次多项式、在分段处具有一定光滑性的函数插值就是模拟以上原理发展起来的，它克服了高次多项式插值可能出现的振荡现象，具有较好的数值稳定性和收敛性，由这种插值过程产生的函数就是多项式样条函数。样条函数的研究始于 20 世纪中叶，到了 60 年代它与计算机辅助设计相结合，在外形设计方面得到成功的应用。样条理论已成为函数逼近的有力工具。它的应用范围也在不断扩大，不仅在数据处理、数值微分、数值积分、微分方程和积分方程数值解等数学领域有广泛的应用，而且与最优控制、变分问题、统计学、计算几何与泛函分析等学科均有密切的联系。

设给定一组结点

$$-\infty = x_0 < x_1 < \cdots < x_N < x_{N+1} = \infty \quad (2.1)$$

又设分段函数 $S(x)$ 满足条件：于每个区间 $[x_j, x_{j+1}] (j=0, \cdots, N)$ 上， $S(x)$ 是一个次数不超过 n 的实系数代数多项式； $S(x)$ 于 $(-\infty, \infty)$ 上具有一直到 $n-1$ 阶的连续导数。则称 $y=S(x)$ 为 n 次样条函数。常把以 (2.1) 为结点的 n 次样条函数的总体记为 $S_n(x_1, x_2, \cdots, x_N)$ 。 x_1, \cdots, x_N 称为样条结点。

一个（奇次） $2n-1$ 次样条函数 $y=S(x)$ ，如果起在区间 $(-\infty, x_1]$ 与 $[x_N, \infty)$ 上的表达式都是 $n-1$ 次多项式（并不要求该两个 $n-1$ 次多项式相同），则特别称之为 $2n-1$ 次的自然样条函数。

定理 1 任一 $S(x) \in S_n(x_1, x_2, \cdots, x_N)$ 均可唯一地表现为

$$S(x) = p_n(x) + \sum_{j=1}^N c_j (x - x_j)_+^{n-1} \quad (-\infty < x \leq \infty) \quad (2.2)$$

其中， $p_n(x) \in P_n, c_j (j=1, \cdots, N)$ 为实数。

显然，由 (2.2) 式所给出的任一函数 $S(x)$ 必然满足 n 次样条函数的定义，亦即 $S(x) \in S_n(x_1, x_2, \cdots, x_N)$ ，因而定理 1 可进一步写成

定理 2 使 $S(x) \in S_n(x_1, x_2, \cdots, x_N)$ ，必须且只须存在 $p_n(x) \in P_n$ 和 N 个实数 c_1, c_2, \cdots, c_N ，使得 (2.2) 成立。

定理 1 和定理 2 说明函数系

$$1, x, x^2, \dots, x^n, (x-x_1)_+^n, \dots, (x-x_N)_+^n \quad (2.3)$$

构成 n 次样条函数类 $S_n(x_1, x_2, \dots, x_N)$ 的一组基底。

由定理 2 可知, 任一 $S(x) \in N_{2n-1}(x_1, x_2, \dots, x_N)$ 均可表示为

$$S(x) = p_{n-1}(x) + \sum_{j=1}^N c_j (x-x_j)_+^{2n-1} \quad (-\infty < x \leq \infty) \quad (2.4)$$

其中, $p_{n-1}(x) \in P_{n-1}$ 。

当然, 一函数 $S(x)$ 只是满足 (2.4) 还不足以保证它一定是一个自然样条函数。因为它在 $[x_N, \infty)$ 上是否为一个 $n-1$ 次的多项式尚不能保证, 为保证这点, 便须要求 $S(x)$ 于 $[x_N, \infty)$ 中的表达式 $p_{n-1}(x) + \sum_{j=1}^N c_j (x-x_j)_+^{2n-1}$ 为 $n-1$ 次多项式。即要求上述求和号这一项中 n 次以上的各方幂项之系数为 0。

定理 3 为使 $S(x) \in N_{2n-1}(x_1, x_2, \dots, x_N)$, 必须且只须存在 $p_{n-1}(x) \in P_{n-1}$ 和满足线性约束 (2.4) 的实数 c_1, c_2, \dots, c_N , 使得

$$S(x) = p_{n-1}(x) + \sum_{j=1}^N c_j (x-x_j)_+^{2n-1} \quad (-\infty < x \leq \infty) \quad (2.5)$$

2.2 参数曲线

2.2.1 参数表示

曲线、曲面可以用显式、隐式和参数表示, 由于参数表示的曲线、曲面具有几何不变性等优点, 计算机图形学中通常用参数形式描述曲线、曲面。

在几何造型系统中, 曲线曲面方程通常表示成参数的形式, 即曲线上任一点的坐标均表示成给定参数的函数。假定用 t 表示参数, 平面曲线上任一点 P 可表示为: $P(t)=[x(t), y(t)]$; 空间曲线上任一三维点 P 可表示为: $P(t)=[x(t), y(t), z(t)]$ 。

在曲线、曲面的表示上, 参数方程比显式、隐式方程有更多的优越性, 主要表现在:

- (1) 可以满足几何不变性的要求。
- (2) 有更大的自由度来控制曲线、曲面的形状。
- (3) 对非参数方程表示的曲线、曲面进行变换, 必须对曲线、曲面上的每个型值点进行几何变换; 而对参数表示的曲线、曲面对其参数方程直接进行几何变换。
- (4) 便于处理斜率为无穷大的情形, 不会因此而中断计算。

(5) 由于坐标点各分量的表示是分离的, 从而便于把低维空间中曲线、曲面扩展到高维空间中去。

(6) 规格化的参数变量 $t \in [0, 1]$, 使得界定曲线、曲面的范围十分简单。

(7) 易于用矢量和矩阵运算, 从而大大简化了计算。

2.2.2 参数化方法

过三点 P_0 、 P_1 和 P_2 构造参数多项式插值抛物线可以有无数条, 其原因是: 参数在 $[0, 1]$ 区间的分割可以有无数种, 因为 P_0 、 P_1 和 P_2 可对应: $[t_0 = 0, t_1 = \frac{1}{2}, t_2 = 1]$;

$$[t_0 = 0, t_1 = \frac{1}{3}, t_2 = 1] \dots$$

其中, 每个参数值称为节点(knot)。

对于一条插值曲线, 型值点 P_0, P_1, \dots, P_n 与其参数域 $t \in [t_0, t_1]$ 内的节点之间有一种对应关系。对于一组有序型值点, 确定一种参数分割, 称之对这组型值点参数化。

对一组型值点 (P_0, P_1, \dots, P_n) 参数化常用方法有以下几种:

(1) 均匀参数化(等距参数化)

使每个节点区间长度 $\Delta_i = t_{i+1} - t_i = \text{正常数}$ ($i = 0, 1, \dots, n-1$), 节点在参数轴上呈等距分布, $t_{i+1} = t_i + \text{正常数}$ 。

(2) 累加弦长参数化

$$\begin{cases} t_0 = 0 \\ t_j = t_{j-1} + |\Delta P_{j-1}|, \quad j = 1, 2, \dots, n \end{cases}$$

其中 $\Delta P_i = P_{i+1} - P_i$ 为向前差分矢量, 即弦边矢量。这种参数法此如实反映了型值点按弦长的分布情况, 能够克服型值点按弦长分布不均匀的情况下采用均匀参数化所出现的问题。

(3) 向心参数化法

$$\begin{cases} t_0 = 0 \\ t_j = t_{j-1} + |\Delta P_{j-1}|^{1/2}, \quad j = 1, 2, \dots, n \end{cases}$$

累加弦长法没有考虑相邻弦边的转折情况, 而向心参数化法假设在一段曲线弧上的向心力与曲线切矢从该弧段始端至末端的转角成正比, 加上一些简化假设, 得到向心参数化法。此法尤其适用于非均匀型值点分布。

(4) 修正弦长参数化法

$$\begin{cases} t_0 = 0 \\ t_j = t_{j-1} + K_j |\Delta P_{j-1}|, \quad j = 1, 2, \dots, n \end{cases}$$

$$\text{其中, } K_j = 1 + \frac{3}{2} \left(\frac{|\Delta P_{j-2}| \theta_{j-1}}{|\Delta P_{j-2}| + |\Delta P_{j-1}|} + \frac{|\Delta P_j| \theta_j}{|\Delta P_{j-1}| + |\Delta P_j|} \right),$$

$$\theta_j = \min(\pi - \angle P_{j-1} P_j P_{j+1}, \frac{\pi}{2}), \quad |\Delta P_{-1}| = |\Delta P_n| = 0$$

弦长修正系数 $K_j \geq 0$ 。从公式可知, 与前后弦长 $|\Delta P_{j-2}|$ 及 $|\Delta P_j|$ 相比, 若 $|\Delta P_{j-1}|$ 越小, 且与前后邻弦边夹角的外角 θ_{j-1} 和 θ_j (不超过 $\frac{\pi}{2}$ 时) 越大, 则修正系数 K_j 就越大。

2.2.3 参数曲线的代数和几何形式

(1) 代数形式

一条三次曲线的代数形式是:

$$\begin{cases} x(t) = a_{3x}t^3 + a_{2x}t^2 + a_{1x}t + a_{0x} \\ y(t) = a_{3y}t^3 + a_{2y}t^2 + a_{1y}t + a_{0y} \\ z(t) = a_{3z}t^3 + a_{2z}t^2 + a_{1z}t + a_{0z} \end{cases}, \quad t \in [0, 1]$$

方程组中 12 个系数唯一地确定了一条 3 次参数曲线的位置与形状。上述代数式写成矢量式是:

$$P(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0, \quad t \in [0, 1] \quad (2.6)$$

其中 a_0, a_1, a_2, a_3 是代数系数矢量, $P(t)$ 是三次参数曲线上任一点的位置矢量。

(2) 几何形式

描述参数曲线的条件有: 端点位矢、端点切矢、曲率等。对三次参数曲线若用其端点位矢 $P(0), P(1)$ 和切点位矢 $P'(0), P'(1)$ 描述, 并将 $P(0), P(1)$ 和 $P'(0), P'(1)$ 简记为 P_0, P_1 和 P'_0, P'_1 , 代入(2.6)式得

$$\begin{cases} a_0 = P_0 \\ a_1 = P'_0 \\ a_2 = -3P_0 + 3P_1 - 2P'_0 - P'_1 \\ a_3 = 2P_0 - 2P_1 + P'_0 + P'_1 \end{cases}$$

代入(2.6)整理得到

$$P(t) = (2t^3 - 3t^2 + 1)P_0 + (-2t^3 + 3t^2)P_1 + (t^3 - 2t^2 + t)P_0' + (t^3 - t^2)P_1', \quad t \in [0,1]$$

$$\text{令 } F_0(t) = 2t^3 - 3t^2 + 1, \quad F_1(t) = -2t^3 + 3t^2, \quad G_0(t) = t^3 - 2t^2 + t, \quad G_1(t) = t^3 - t^2$$

将 F_0, F_1, G_0, G_1 代入 (3.1.3) 式, 可将其简化为:

$$P(t) = F_0P_0 + F_1P_1 + G_0P_0' + G_1P_1', \quad t \in [0,1]$$

上式是三次 Hermite(Ferguson)曲线的几何形式, 几何系数是 P_0, P_1 和 P_0', P_1' 。

2.3 B样条曲线

2.3.1 B样条的递推定义和分类

与 Bezier 曲线得定义方法类似, B 样条曲线方程定义为:

$$P(t) = \sum_{i=0}^n P_i N_{i,k}(t)$$

其中, $P_i(i=0,1,\dots,n)$ 是控制多边形的顶点, $N_{i,k}(t)$, $i=0,\dots,n$, 称为 k 阶($k-1$ 次)B 样条基函数, 其中每一个称为 B 样条, 它是一个称为节点矢量, 即非递减的参数 t 序列 $T: t_0 \leq t_1 \leq \dots \leq t_{n+k}$ 所决定的 k 阶分段多项式, 也即为 k 阶($k-1$ 次)多项式样条。

B 样条有多种等价定义, 在理论上较多地采用截尾幂函数的差商定义。我们只介绍作为标准算法的 de Boor-Cox 递推定义, 又称为 de Boor-Cox 公式。约定 $0/0=0$ 。

$$\begin{cases} N_{i,0}(t) = \begin{cases} 1 & \text{若 } t_i \leq x < t_{i+1} \\ 0 & \text{其它} \end{cases} \\ N_{i,k}(t) = \frac{(t - t_i)N_{i,k-1}(t)}{t_{i+k} - t_i} + \frac{(t_{i+k+1} - t)N_{i+1,k-1}(t)}{t_{i+k+1} - t_{i+1}} & k > 0 \\ 0/0 = 0 \end{cases}$$

该递推公式表明: 欲确定第 i 个 k 阶 B 样条 $N_{i,k}(t)$, 需要用到 t_i, \dots, t_{i+k} , 共 $k+1$ 个节点, 称区间 $[t_i, t_{i+k}]$ 为 $N_{i,k}(t)$ 的支承区间。曲线方程中, $n+1$ 个控制顶点 $P_i(i=0,1,\dots,n)$, 要用到 $n+1$ 个 k 阶 B 样条 $N_{i,k}(t)$ 。它们支撑区间的并集定义了这一组 B 样条基的节点矢量 $T=[t_0, t_1, \dots, t_{n+k}]$ 。

B 样条基函数有如下性质:

(1) 局部支承性

$$N_{i,k}(t) = \begin{cases} \geq 0 & t \in [t_i, t_{i+k}] \\ = 0 & \text{其它} \end{cases}$$

(2) 权性

$$\sum_{i=0}^n N_{i,k}(t) = 1, \quad t \in [t_{k-1}, t_{n+1}]$$

(3) 微分公式

$$N'_{i,k}(t) = \frac{k-1}{t_{i+k-1} - t_i} N_{i,k-1}(t) + \frac{k-1}{t_{i+k} - t_{i+1}} N_{i+1,k-1}(t)$$

曲线按其首末端点是否重合，区分为闭曲线和开曲线。闭曲线又区分为周期和非周期两种情形，周期闭曲线与非周期闭曲线的区别是：前者在首末端点是 C^2 连续的，而后者一般是 C^0 连续的。非周期闭曲线可以认为是开曲线的特例，按开曲线处理。

B 样条曲线按其节点矢量中节点的分布情况，可划分为四种类型。假定控制多边形的顶点为 $P_i, i=0, \dots, n$ ，阶数为 k （次数为 $k-1$ ），则节点矢量是 $T=[t_0, t_1, \dots, t_{n+k}]$ 。

(1) 均匀 B 样条曲线

节点矢量中节点为沿参数轴均匀或等距分布，所有节点区间长度 $\Delta_i = t_{i+1} - t_i = \text{常数}$ ， $i=(0, 1, \dots, n+k-1)$ ，这样的节点矢量定义了均匀 B 样条基。图 2.1 是均匀 B 样条曲线实例。



图 2.1 均匀 B 样条

Fig.2.1 Homogeneous B-Spline

(2) 准均匀 B 样条曲线

与均匀 B 样条曲线的差别在于两端节点具有重复度 k ，这样的节点矢量定义了准均匀的 B 样条基。

均匀 B 样条曲线在曲线定义域内各节点区间上具有用局部参数表示的统一的表达式，使得计算与处理简单方便。但它定义均匀 B 样条曲线没有保留 Bezier 曲线端点的几何性质，即样条曲线的首末端点不再是控制多边形的首末端点。采用准均匀的 B 样条曲线就是为了解决这个问题，使曲线在端点的行为有较好的控制，如图 2.2 所示。



图 2.2 准均匀 B 样条

Fig.2.2 Quasi-homogeneous B-Spline

(3) 分段 Bezier 曲线

节点矢量中两端节点具有重复度 k ，所有内节点重复度为 $k-1$ ，这样的节点矢量定义了分段的 Bernstein 基。

B 样条曲线用分段 Bezier 曲线表示后，各曲线段就具有了相对的独立性，移动曲线段内的一个控制顶点只影响该曲线段的形状，对其它曲线段的形状没有影响。并且 Bezier 曲线一整套简单有效的算法都可以原封不动地采用。其它三种类型的 B 样条曲线可通过插入节点的方法转换成分段 Bezier 曲线类型，缺点是增加了定义曲线的数据，控制顶点数及节点数都将增加。分段 Bezier 曲线实例如图 2.3 所示。



图 2.3 分段 Bezier 曲线

Fig.2.3 Sectional Bezier curve

(4) 非均匀 B 样条曲线

在这种类型里，任意分布的节点矢量 $T=[t_1, t_2, \dots, t_{n+k}]$ ，只要在数学上成立（节点序列非递减，两端节点重复度 $\leq k$ ，内节点重复度 $\leq k-1$ ）都可选取。这样的节点矢量定义了非均匀 B 样条基。

2.3.2 B样条曲线的性质

(1) 局部性

由于 B 样条的局部性， k 阶 B 样条曲线上参数为 $t \in [t_i, t_{i+1}]$ 的一点 $P(t)$ 至多与 k 个控制顶点 $P_j (j=i-k+1, \dots, i)$ 有关，与其它控制顶点无关；移动该曲线的第 i 个控制顶点 P_i 至多影响到定义在区间 (t_i, t_{i+k}) 上那部分曲线的形状，对曲线的其余部分不发生影响。

(2) 连续性

$P(t)$ 在 r 重节点 $t_i (k \leq i \leq n)$ 处的连续阶不低于 $k-1-r$ 。整条曲线 $P(t)$ 的连续阶不低于 $k-1-r_{\max}$ ，其中 r_{\max} 表示位于区间 (t_{k-1}, t_{n+1}) 内的节点的最大重数。

(3) 凸包性

$P(t)$ 在区间 $[t_i, t_{i+1})$ ， $k-1 \leq i \leq n$ 上的部分位于 k 个点 P_{i-k+1}, \dots, P_i 的凸包 C_i 内，整个曲线则位于各凸包 C_i 的并集 $\bigcup_{i=k-1}^n C_i$ 之内。

(4) 分段参数多项式

$P(t)$ 在每一区间 $[t_i, t_{i+1})$, $k-1 \leq i \leq n$ 上都是次数不高于 $k-1$ 的参数 t 的多项式, $P(t)$ 是参数 t 的 $k-1$ 次分段多项式。

(5) 导数公式

由 B 样条基的微分差分公式, 有

$$P'(t) = \left(\sum_{i=0}^n P_i N_{i,k}(t) \right)' = \sum_{i=0}^n P_i N'_{i,k}(t) = (k-1) \sum_{i=1}^n \left(\frac{P_i - P_{i-1}}{t_{i+k-1} - t_i} \right) N_{i,k-1}(t), \quad t \in [t_{k-1}, t_{n+1}]$$

(6) 变差缩减性

设平面内 $n+1$ 个控制顶点 P_0, P_1, \dots, P_n 构成 B 样条曲线 $P(t)$ 的特征多边形。在该平面内的任意一条直线与 $P(t)$ 的交点个数不多于该直线和特征多边形的交点个数。

(7) 几何不变性

B 样条曲线的形状和位置与坐标系的选择无关。

(8) 仿射不变性

对任一仿射变换 A :

$$A[P(t)] = \sum_{i=0}^n A[P_i] N_{i,k}(t), \quad t \in [t_{k-1}, t_{n+1}]$$

即在仿射变换下, $P(t)$ 的表达式具有形式不变性。

(9) 直线保持性

控制多边形退化为一 条直线时, 曲线也退化为一 条直线。

(10) 造型的灵活性

2.3.3 De Boor 算法

给定控制顶点 $P_i (i=0, 1, \dots, n)$ 及节点矢量 $T=[t_0, t_1, \dots, t_{n+k}]$ 后, 就定义了 k 阶 ($k-1$ 次) B 样条曲线。欲计算 B 样条曲线上对应一点 $P(t)$, 可以利用 B 样条曲线方程, 但是采用 de Boor 算法, 计算更加快捷。

先将 t 固定在区间 $[t_j, t_{j+1}]$, $k-1 \leq j \leq n$, 由 de Boor-cox 公式有:

$$\begin{aligned} P(t) &= \sum_{i=0}^n P_i N_{i,k}(t) = \sum_{i=j-k+1}^j P_i N_{i,k}(t) \\ &= \sum_{i=j-k+1}^j P_i \left[\frac{t-t_i}{t_{i+k-1}-t_i} N_{i,k-1}(t) + \frac{t_{i+k}-t}{t_{i+k-1}-t_{i+1}} N_{i+1,k-1}(t) \right] \\ &= \sum_{i=j-k+1}^j \left[\frac{t-t_i}{t_{i+k-1}} P_i + \frac{t_{i+k-1}-t}{t_{i+k-1}-t_{i+1}} P_{i+1} \right] N_{i,k-1}(t), \quad t \in [t_j, t_{j+1}] \end{aligned}$$

令

$$P_i^{[r]}(t) = \begin{cases} P_i & r=0, i=j-k+1, j-k+2, \dots, j \\ \frac{t-t_i}{t_{i+k-r}-t_i} P_i^{[r-1]}(t) + \frac{t_{i+k-r}-t}{t_{i+k-r}-t_i} P_{i-1}^{[r-1]}(t) & r=1, 2, \dots, k-1; i=j-k+r+1, \dots, j \end{cases}$$

则

$$P(t) = \sum_{i=j-k+1}^j P_i N_{i,k}(t) = \sum_{i=j-k+2}^j P_i^{[1]}(t) N_{i,k-1}(t)$$

上式是同一条曲线 $P(t)$ 从 k 阶 B 样条表示到 $k-1$ 阶 B 样条表示的递推公式，反复应用此公式，得到 $P(t) = P_j^{[k-1]}(t)$ 。

于是， $P(t)$ 的值可以通过递推关系式求得。这就是著名的 De Boor 算法。

3 B样条最小二乘数据拟合

本章首先介绍在节点给定的情况下,用B样条作为拟合函数的线性最小二乘问题,即固定节点样条最小二乘法。然后,介绍在样条函数的内节点可变的情况下,线性-非线性最小二乘问题,即自由节点样条最小二乘法。

3.1 固定节点B样条最小二乘问题

设给定一组数据点 $\{(t_i, y_i)\}_{i=1}^w$, $S_k(\Delta)$ 为 k 次 B 样条基函数 $\{N_{i,k}(t)\}_{i=0}^n$ 张成的空间, B 样条最小二乘数据拟合问题是在 $S_k(\Delta)$ 中,找到函数 $P(t)$, 使

$$\sum_{i=1}^w (P(t_i) - y_i)^2 = \min_{S(t) \in S_k(\Delta)} \sum_{i=1}^w (S(t_i) - y_i)^2$$

记 $Y = (y_1, y_2, \dots, y_w)^T$, $t = (t_1, t_2, \dots, t_w)^T$ 则 $P(t)$ 是 $S_k(\Delta)$ 中对 $\{y_i\}_{i=1}^w$ 的离散最佳逼近的充要条件是 $P(t) - Y$ 与所有的 $\{N_{i,k}(t)\}_{i=0}^n$ 正交, 即

$$\langle P(t) - Y, N_{i,k}(t) \rangle = 0 \quad (3.1)$$

设 $P(t) = \sum_{i=0}^n c_i N_{i,k}(t)$, 则 (3.1) 可以改写为

$$\langle N_{j,k}, Y - \sum_{i=0}^n c_i N_{i,k}(t) \rangle = 0, j = 0, 1, \dots, n \quad (3.2)$$

求 $P(t)$ 的问题就是求 B 样条系数 $\{c_i\}_{i=0}^n$ 。这样问题便可归结为求解线性方程组

$$\sum_{i=0}^n \langle N_{i,k}, N_{j,k} \rangle c_i = \langle N_{j,k}, Y \rangle \quad j = 0, 1, \dots, n \quad (3.3)$$

令 $A = (N_{i,k}(t_j))_{i=0, j=1}^n$, 则 (3.3) 可以写为 $A^T A c = A^T Y$, 所以 $c = A^+ Y$ 。

算法:

步骤 1: 给定样条的节点 $\{t_i\}_{i=0}^N$, 满足 $a = t_0 < t_1 < \dots < t_N = b$ 。

步骤 2: 由 B 样条的递推关系式

$$\begin{cases} N_{i,0}(t) = \begin{cases} 1 & \text{若 } t_i \leq x < t_{i+1} \\ 0 & \text{其它} \end{cases} \\ N_{i,k}(t) = \frac{(t - t_i) N_{i,k-1}(t)}{t_{i+k} - t_i} + \frac{(t_{i+k+1} - t) N_{i+1,k-1}(t)}{t_{i+k+1} - t_{i+1}} & k > 0 \\ 0 / 0 = 0 \end{cases}$$

计算 $N_{i,k}(t)$ 。

步骤 3: 计算 $N_{i,k}(t_j)$, 组成矩阵 A 。

步骤 4: 求解方程组 $A^T A c = A^T Y$ 。

步骤 5: 计算 $P(x) = \sum_{i=0}^n c_i N_{i,k}(t)$ 。

步骤 6: 计算 $\delta = A c - Y$, 及 $\langle \delta, \delta \rangle$ 。

3.2 自由节点样条最小二乘问题

3.2.1 非线性最小二乘法

非线性最小二乘法常把问题归结为函数 $F(\alpha) = \sum_{i=1}^m f_i^2(\alpha)$, $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$ 的极小化问题 (这里 α 为非线性参数)。即求 $\alpha = \alpha^*$, 使 $F(\alpha^*) = \min_{\alpha} F(\alpha)$ 。

平方和函数的极小问题有以下几种求解方法:

(1) 梯度法

① 最速下降法

② Newton-Raphson 法

③ 变尺度法 (DFP 法)

(2) Gauss-Newton 法

(3) Marquardt 修改的 Gauss-Newton 法 (GNM 法)

下面给出 GNM 法的计算步骤:

给定初值 $\alpha^{(0)}, \beta_0, \gamma$,

(1) $0 \Rightarrow k, \beta_0 \Rightarrow \beta$;

(2) $-1 \Rightarrow \omega, \beta / \gamma \Rightarrow \beta$;

(3) 求 $f = (f_1(\alpha^{(k)}), \dots, f_m(\alpha^{(k)}))^T$; $F(\alpha^{(k)})$; 以及

$$J = \begin{pmatrix} \frac{\partial f_1(\alpha^{(k)})}{\partial \alpha_1} & \frac{\partial f_1(\alpha^{(k)})}{\partial \alpha_2} & \dots & \frac{\partial f_1(\alpha^{(k)})}{\partial \alpha_n} \\ \frac{\partial f_2(\alpha^{(k)})}{\partial \alpha_1} & \frac{\partial f_2(\alpha^{(k)})}{\partial \alpha_2} & \dots & \frac{\partial f_2(\alpha^{(k)})}{\partial \alpha_n} \\ \vdots & \vdots & \dots & \vdots \\ \frac{\partial f_m(\alpha^{(k)})}{\partial \alpha_1} & \frac{\partial f_m(\alpha^{(k)})}{\partial \alpha_2} & \dots & \frac{\partial f_m(\alpha^{(k)})}{\partial \alpha_n} \end{pmatrix};$$

(4) 求 $p_k = -(J^T J + \beta I)^{-1} (J^T f)$, p_k 为下降方向

$$\alpha^{(k+1)} = \alpha^{(k)} + p_k$$

$$F(\alpha^{(k+1)})$$

(5) 若 $F(\alpha^{(k+1)}) < F(\alpha^{(k)})$, 则转(8); 否则转(6);

(6) $\omega+1 \Rightarrow \omega$, $\beta \cdot \gamma \Rightarrow \beta$, 转(7);

(7) 若 $\|J^T f\| \leq \varepsilon$, 则求出了最优解 $\alpha^* = \alpha^{(k)}$; 否则转(4);

(8) 若 $\|J^T f\| \leq \varepsilon$, 则求出了最优解 $\alpha^* = \alpha^{(k+1)}$; 否则, 令 $k+1 \Rightarrow k$, 转(2)。

3.2.2 自由节点样条最小二乘问题提出

给定区间 $[a, b]$, 样条内节点 $\tau = (\tau_{k+1}, \tau_{k+2}, \dots, \tau_{k+N})^T$ 的位置可以在区间 $[a, b]$ 之间变化, 且满足 $a = \tau_0 = \dots = \tau_k < \tau_{k+1} < \tau_{k+2} < \dots < \tau_{k+N} < \tau_{k+N+1} = \tau_{k+N+2} = \dots = \tau_{k+N+K} = b$, 即内节点序列满足 $N+1$ 个线性不等式约束 $\tau_{k+p} - \tau_{k+p-1} > 0$, $p = 1, 2, \dots, N+1$, 也就是内节点在 N 维单纯形 $S_N[a, b]$ 内部。记关于节点向量 τ 的第 j 个 k 次 B 样条函数为 $B_j(\tau; t)$, 给定数

据点 $\{(t_i, y_i)\}_{i=1}^m$, 令 $J(\alpha, \tau) = \sum_{i=1}^m (\sum_{j=0}^n \alpha_j B_j(\tau, t_i) - y_i)^2$, 求 α^* , τ^* , 使

$$J(\alpha^*, \tau^*) = \min_{\substack{\alpha \in R^{n+1} \\ \tau \in S_N}} J(\alpha, \tau) \quad (3.4)$$

在本章中, 对于自由节点样条最小二乘问题, 我们介绍一种交替求 τ, α 的方法 (即交替迭代法) 求解。这一方法在求非线性优化问题时容易比较在节点迭代前后的样条函数值与数据值的误差是否减少了。

设 α^* , τ^* 为问题 (3.4) 的解, 则 α^* 是固定 τ^* 时的线性最小二乘问题

$$\min_{\alpha} J(\alpha, \tau^*) \quad (3.5)$$

的解。

当 α^* 为与 τ 无关的常向量时,

$$\tilde{J}(\tau) = J(\alpha^*, \tau) \quad (3.6)$$

变成了只是向量 τ 的函数，这样就归结为非线性最小二乘问题的求解。求解的途径是：给出初始节点的值 $\tau^{(0)} = (\tau_{k+1}^{(0)}, \tau_{k+2}^{(0)}, \dots, \tau_{k+N}^{(0)})^T$ 后，求解线性问题

$$\min_{\alpha} J(\alpha, \tau^{(0)})$$

求出 $\alpha^{(0)}$ 。设 $\alpha^{(l)}$ ， $\tau^{(l)}$ 已求出，由非线性优化的方法求出 $\tau^{(l+1)}$ ，使

$$\tilde{J}(\tau^{(l+1)}) < \tilde{J}(\tau^{(l)})$$

再解线性最小二乘问题

$$\min_{\alpha} J(\alpha, \tau^{(l+1)})$$

求出 $\alpha^{(l+1)}$ 。这样就可以用迭代的方法求出 τ^* ，再由 (3.5) 求出 α^* 。

3.2.3 节点变换

自由节点样条最小二乘数据拟合问题归结为 $S_N[a, b]$ 中的优化问题，这实际上是一个约束优化问题。在求全局极小点的时候，常会在单纯形 $S_N[a, b]$ 的表面出现 $\tilde{J}(\tau)$ 的许多驻点，这些驻点可能是 $\tilde{J}(\tau)$ 的局部极小点或鞍点，甚至是局部极大点。而这些点的存在是与数据本身无关的。由于这些驻点的存在，使求解发生困难。

虽然在求解的过程中，求得局部极小点而非全局极小点通常不可避免，但通过变量变换，至少可以使这一情况得到较大的改善（因为在边界上与边界附近的局部极小点可以避免）。

$$\text{令 } \sigma_i = \ln(h_{i+1}/h_i), \quad i=1, 2, \dots, N \quad (3.7)$$

其中， $h_i = \tau_{k+i} - \tau_{k+i-1}$ 。记 $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_N)^T$ ，则上述变换 ($\tau \rightarrow \sigma$) 将 $S_N[a, b]$ 映为 R^N ；将 $\bar{S}_N[a, b]$ 的边界（即各个面）映为 R^N 的“无穷远点”。这时 R^N 中的原点对应于内节点 τ 的等距分布。这样，就把在 N 维空间中的单纯形 $S_N[a, b]$ 上的约束优化问题变成了 R^N 中的无约束优化问题。

下面导出上述变换的逆变换，即对任意的 σ 求出 τ 。令 $\lambda_i = e^{\sigma_i}$ ， $i=1, 2, \dots, N$ ，

则

$$h_{i+1} = \lambda_i h_i = \lambda_i \lambda_{i-1} h_{i-1} = \dots = \lambda_i \lambda_{i-1} \dots \lambda_1 h_1$$

因为

$$h_1 + h_2 + \dots + h_{N+1} = b - a$$

所以

$$h_1 + \lambda_1 h_1 + \lambda_1 \lambda_2 h_1 + \dots + \lambda_1 \lambda_2 \dots \lambda_N h_1 = b - a$$

因此得到

$$h_1 = (b - a) / Z, \quad \text{其中, } Z = 1 + \lambda_1 + \lambda_1 \lambda_2 + \dots + \lambda_1 \lambda_2 \dots \lambda_N$$

又因为

$$h_{i+1} = \lambda_i h_i, \quad i=1,2,\dots,N$$

所以

$$\tau_{k+1} = a + h_1, \quad \tau_{k+j} = \tau_{k+j-1} + h_j, \quad j=2,3,\dots,N.$$

由连锁规则, 可以得到

$$\frac{\partial \tilde{J}}{\partial \tau_{k+j}} = \sum_{i=1}^N \frac{\partial \tilde{J}}{\partial \sigma_i} \cdot \frac{\partial \sigma_i}{\partial \tau_{k+j}} = \nabla_{\sigma}^T \tilde{J}(\sigma) \cdot \frac{\partial \sigma}{\partial \tau_{k+j}}$$

所以

$$\nabla_{\tau}^T \tilde{J} = \nabla_{\sigma}^T \tilde{J}(\sigma) \left(\frac{\partial \sigma}{\partial \tau_{k+1}}, \frac{\partial \sigma}{\partial \tau_{k+2}}, \dots, \frac{\partial \sigma}{\partial \tau_{k+N}} \right)$$

记矩阵

$$G = \left(\frac{\partial \sigma}{\partial \tau_{k+1}}, \frac{\partial \sigma}{\partial \tau_{k+2}}, \dots, \frac{\partial \sigma}{\partial \tau_{k+N}} \right) = \left(\frac{\partial \sigma_i}{\partial \tau_{k+j}} \right)_{i=1, j=1}^{N,N}$$

因为

$$\sigma_i = \ln(h_{i+1}/h_i) = \ln h_{i+1} - \ln h_i = \ln(\tau_{k+i+1} - \tau_{k+i}) - \ln(\tau_{k+i} - \tau_{k+i-1})$$

所以

$$\frac{\partial \sigma_i}{\partial \tau_{k+j}} = \begin{cases} 0 & j \neq i-1, i, i+1 \\ 1/h_i & j = i-1 \\ -(1/h_i + 1/h_{i+1}) & j = i \\ 1/h_{i+1} & j = i+1 \end{cases}$$

由此

$$G = \begin{pmatrix} -(\frac{1}{h_1} + \frac{1}{h_2}) & \frac{1}{h_2} & & & \\ \frac{1}{h_2} & -(\frac{1}{h_2} + \frac{1}{h_3}) & \ddots & & \\ & \ddots & \ddots & \frac{1}{h_N} & \\ & & \frac{1}{h_N} & -(\frac{1}{h_N} + \frac{1}{h_{N+1}}) & \end{pmatrix}$$

上述矩阵简写为

$$-G = BH^{-1}B^T \quad (3.8)$$

其中

$$B = \begin{pmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \end{pmatrix}_{N \times (N+1)}, \quad H = \begin{pmatrix} h_1 & & & \\ & h_2 & & \\ & & \ddots & \\ & & & h_{N+1} \end{pmatrix}$$

这时

$$G \nabla_{\sigma}^T \tilde{J} = \nabla_{\tau}^T \tilde{J}$$

$$\text{令 } \varepsilon_j = \sum_{i=0}^n \alpha_j B_j(\tau, t_i) - y_i, \quad J(\tau) = \left(\frac{\partial \varepsilon_j}{\partial \tau_{k+l}} \right)_{j=1, l=1}^{m, n}, \quad J(\sigma) = \left(\frac{\partial \varepsilon_j}{\partial \sigma_i} \right)_{j=1, i=1}^{m, n}$$

类似地, 有

$$J(\tau) = J(\sigma) \cdot G$$

所以由方程组

$$G \nabla_{\sigma} \tilde{J} = \nabla_{\tau} \tilde{J} \quad (3.9)$$

$$G \cdot J^T(\sigma) = J^T(\tau) \quad (3.10)$$

便可解出 $\nabla_{\sigma} \tilde{J}$ 和 $J(\sigma)$ 。

3.3.3 自由节点样条拟合优化算法

自由节点样条最小二乘数据拟合，把问题化为轮流求线性最小二乘问题（这时 τ 固定）

$$\min_{\alpha} J(\alpha) \quad (3.11)$$

与平方和函数的非线性优化问题

$$\min_{\tau} \tilde{J}(\tau) \quad (3.12)$$

即给定初值 $\tau^{(0)}$ ，用 (3.10) 求出 $\alpha^{(0)}$ ，这时可以建立迭代格式

$$J(\alpha^{(k+1)}) = \min_{\alpha} J(\alpha, \tau^{(k)}) \quad (3.13)$$

$$\tilde{J}(\tau^{(k+1)}) = \min_{\tau} J(\alpha^{(k+1)}, \tau) \quad (3.14)$$

线性最小二乘问题用改进的 Gram-Schmidt 正交化方法求解，也可以用 Cholesky 分解法求解。非线性优化可以用 GNM 法求解，也可以用 DFP 法求解。

GNM 法求解自由节点样条最小二乘问题的计算步骤是：

(1) 根据数据点 $\{(t_i, y_i)\}_{i=1}^m$ ，计算 $a = \min_i \{t_i\}$ ， $b = \max_i \{t_i\}$ ，选取初始内节点

$$\tau^{(0)} = (\tau_{k+1}^{(0)}, \tau_{k+2}^{(0)}, \dots, \tau_{k+N}^{(0)})^T, \text{ 满足 } a < \tau_{k+1} < \tau_{k+2} < \dots < \tau_{k+N} < b.$$

(2) 计算 h_i ，并由 (3.8) 计算对应于初始内节点 $\tau^{(0)}$ 的 $\sigma^{(0)} = (\sigma_1^{(0)}, \sigma_2^{(0)}, \dots, \sigma_N^{(0)})^T$ 。

(3) 计算 B 样条的值，由 $A = (B_j(\tau; t_i))_{i=1, j=0}^{m, n}$ 形成矩阵 $A(\tau^{(0)})$ 。

(4) 用改进的 Gram-Schmidt 正交化方法求解线性最小二乘问题 (3.10)，求得 α 。

(5) 计算 $\varepsilon = Y - A\alpha$ 及 $J_0 = \|\varepsilon\|^2$ 。

(6) 计算 $\frac{\partial A(\tau)}{\partial \tau_{k+l}}$ ， $l = 1, 2, \dots, N$ ， $J(\tau) = \left(-\frac{\partial A(\tau)}{\partial \tau_{k+l}} \alpha \right)_{l=1}^N$ ， $\nabla_{\tau} J(\alpha, \tau) = -2J^T(\tau) \varepsilon$ 。

(7) 判断 $\|\nabla_{\tau} J(\alpha, \tau)\|$ 是否小于预定值，小于则停机；否则继续。

- (8) 由(3.8)计算矩阵 G ，并求解方程组 $G \cdot J^T(\sigma) = J^T(\tau)$ ，得 $J(\sigma)$ ，再计算向量 $g = -J^T(\sigma)\varepsilon$ （与负梯度方向 $-\nabla_{\tau} J(\alpha, \tau)$ 相同）。
- (9) 求解方程组 $(J^T(\sigma)J(\sigma) + \beta_0 I)p = g$ ，得方向 p 。
- (10) 令 $\sigma = \sigma^{(0)} + p$ ，用节点变换求出与 σ 对应的 τ 。
- (11) 形成矩阵 $A(\tau)$ 。
- (12) 计算 $\varepsilon = Y - A\alpha$ 及 $J = \|\varepsilon\|^2$ 。
- (13) 如果 $J < J_0$ ，做(14)；否则做(15)。
- (14) 求一维优化问题

$$\min_{\lambda} \tilde{J}(\sigma^{(0)} + \lambda p)$$

得 $\sigma = \sigma^{(0)} + \lambda p$ （ λ 是优化后的系数，一维优化可用抛物线法等），令 $\sigma^{(0)} = \sigma$ ，转(4)继续进行。

- (15) 令 $\beta = \beta_0 \gamma$ ，求解方程组 $(J^T(\sigma)J(\sigma) + \beta I)p = g$ ，得方向 p 。
- (16) 令 $\sigma = \sigma^{(0)} + p$ ，用节点变换求出与 σ 对应的 τ 。
- (17) 形成矩阵 $A(\tau)$ 。
- (18) 计算 $\varepsilon = Y - A\alpha$ 及 $J = \|\varepsilon\|^2$ 。
- (19) $\beta_0 = \beta$ 。
- (20) 若 $J \leq J_0$ ，则令 β_0 作为初值，令 $\sigma^{(0)} = \sigma$ 后转(4)；若 $J > J_0$ ，转(15)。

4 三次B样条数据拟合的节点自适应选取算法

我们从随机的方法重新考虑自由节点的三次 B 样条数据拟合。在实际生产中,人们通常预先给出一个逼近精度,希望拟合出的曲线与数据点的误差在精度范围以内。我们的方法分为两大步骤,第一步寻找样条的节点,使在这组节点下,样条可以达到精度要求;在得到了这组节点之后,我们发现曲线的有些地方(如平滑处)不需要很多节点也可以保证精度,所以第二步我们对得到的节点在保证精度地前提下进行优化,使节点的数量减少,分布均匀。因为有第二步的优化算法,为了提高运算速度,我们在第一步没有尝试所有的节点分布,而是对尝试的次数进行预设,如果在尝试了这些次之后没有找到合适的节点向量,我们就增加节点数量,继续寻找。虽然这么做会导致第一步得到的节点数不是最少的,但是通过第二步的优化过程,我们完全可以把多余的节点去掉。大量的数值试验也证明了我们的方法的有效性。

4.1 三次B样条自由节点

4.1.1 理论分析

在区间 $[a, b]$ 上给定一组数据点 $\{(x_i, y_i)\}_{i=1}^w$, 和非减序列 $\{u_i\}_{i=0}^N$, $N_{i,k}(x)$ 是以 $\{u_i\}_{i=1}^N$ 为样条内节点向量的 B 样条基函数, 则 k 次 B 样条 $S(x) = \sum_{i=0}^n \beta_i N_{i,k}(x)$ 。则最小二乘逼近问题可以描述为求解方程组

$$\begin{pmatrix} y_1 \\ \vdots \\ y_w \end{pmatrix} = \begin{pmatrix} N_{0,k}(x_1) & \cdots & N_{n,k}(x_1) \\ \vdots & & \vdots \\ N_{0,k}(x_w) & \cdots & N_{n,k}(x_w) \end{pmatrix} \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_n \end{pmatrix} + \delta$$

简写成 $Y = A\beta + \delta$, 使 δ 达到最小。

记 $Q(\beta) = \frac{1}{w} \sum_{i=1}^w (S(x_i) - y_i)^2$, 我们要通过 $Q(\beta)$ 来评价样条节点 $\{u_i\}_{i=1}^N$ 的合理性。给定逼近精度 ε , 随机选取样条节点 $\{u_i\}_{i=1}^N$, 计算概率 $P(Q(\beta) \geq \varepsilon)$, 这样我们可以得出 N 个随机分布节点的 B 样条函数无法达到逼近精度的概率。理论上, 这个概率可以算出, 记为 $\alpha(\varepsilon)$ 。我们做 m 次重复试验, 设第 i 次得到 $Q_i(\beta_i)$, 则 $P(Q_1 \geq \varepsilon, \dots, Q_m \geq \varepsilon) = P(Q_1 \geq \varepsilon) \cdots P(Q_m \geq \varepsilon) = (\alpha(\varepsilon))^m$, 可以得出 m 次试验都无法达到精度的概率。从而, $1 - (\alpha(\varepsilon))^m$ 就可以表示出在节点数目为 N 的情况下, 做 m 次实验至少有一次满足精度的概率。我们给出不满足精度的概率的容许值 α , 由 $(\alpha(\varepsilon))^m \leq \alpha$ 可以求

得 m 的最小值。即对固定的节点数 N ，大体上要做至少 m 次的随机选点，才能找到满足精度要求的样条函数。若 m 太大，则增加节点数目($N=N+1$)，重复上述过程。基于这个思想，我们将在后面的章节给出详细的算法。

4.1.2 样条节点构造方法

给定 N 的初值，在 $[A, B]$ 上构造点列 $\{t_i\}_{i=0}^N$ ，令 $t_0 = A$ ， $t_N = B$ ，取 $N-1$ 个随机数 $\{r_i\}_{i=1}^{N-1}$ ， $0 < r_i < 1$ ，令 $t_i = r_i t_{i-1} + (1-r_i)t_{i+1}$ ， $i=1, \dots, N-1$ ，则得到单调递增随即序列 $\{t_i\}_{i=0}^N$ 。我们把它作为样条节点。即求解如下三对角线性方程组

$$\begin{pmatrix} -1 & 1-r_1 & & & \\ r_2 & -1 & 1-r_2 & & \\ & \ddots & \ddots & \ddots & \\ & & r_{N-2} & -1 & 1-r_{N-2} \\ & & & r_{N-1} & -1 \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_{N-2} \\ t_{N-1} \end{pmatrix} = \begin{pmatrix} -r_1 A \\ 0 \\ \vdots \\ 0 \\ -(1-r_{N-1})B \end{pmatrix} \quad (4.1)$$

值得注意的是，由于我们希望节点分布尽量均匀，所以相邻节点之间的距离不能太近，从而，在选取随机数的时候，我们选择满足 $\min r_i \gg 0$ ，且 $\max r_i \ll 1$ 的随机数组。

我们以 $\{t_i\}_{i=0}^N$ 为节点，以累加弦长为参数，构造三次 B 样条，判断逼近误差是否满足精度要求。如果没有达到，则另取一组随机数 $\{r_i\}_{i=1}^{N-1}$ ，重新构造样条节点，如果这一组随机数与上一组随机数相差不大，为了提高运算效率，我们将舍弃这组随机数，再取随机数组，构造样条函数，判断逼近误差，这一过程我们循环多次，希望可以找到数目为 $N+1$ 且可以使样条达到既定精度的节点向量。然而如果在这个固定的节点数目下没有找到符合精度的节点向量，我们就增加节点数目，并重复以上过程，直到找到满足精度要求的节点向量为止。

在找到了一组符合精度要求的节点向量之后，我们的工作还没有结束。由于我们希望最终得到的节点尽量分布均匀，而我们对节点的选择具有随机性，所以我们多次重复以上过程，得到多组符合要求的节点，通过比较找到一组节点分布相对均匀的作为这一步我们找到的节点向量，为下一步节点的优化做准备。

算法 1

步骤 0 (初始化)

- 待拟合数据点 $\{x_i, y_i, z_i\}_{i=1}^w$ 。
- 节点数 $N+1$ 。
- 在相同节点数目下节点选取的最多次数 Num 。

d. 符合要求的节点组数 Matches。

e. 逼近精度 ϵ 。

步骤 1 取随机数 $\{r_i\}_{i=1}^{N-1}$; 若 $\min r \rightarrow 0$ 或 $\max r \rightarrow 1$, 重新取随机数。

步骤 2 用追赶法求解方程组(4.1), 得到内节点 $\{t_i\}_{i=1}^{N-1}$ 。

步骤 3 如果 $|t_{i+1} - t_i| < \delta$, 去掉一个节点。

步骤 4 将节点规一化。

步骤 5 构造三次 B 样条基函数。

步骤 6 对数据点进行累加弦长参数化, 将参数值带入基函数, 得到系数矩阵。

步骤 7 最小二乘法求解矛盾方程, 得到控制点。

步骤 8 判断逼近误差是否小于精度 ϵ 。若成立, 执行步骤 10。否则执行步骤 9。

步骤 9 $\text{num} = \text{num} + 1$, 若 $\text{num} < \text{Num}$, 执行步骤 1; 否则, $N = N + 1, \text{num} = 1$ 。

步骤 10 $\text{matches} = \text{matches} + 1$; 若 $\text{matches} < \text{Matches}$, 执行步骤 1; 否则执行步骤 11。

步骤 11 计算每个节点向量的相邻节点间距的最小值, 取间距最小值最大的节点向量。

步骤 12 程序结束。

4.1.3 样条节点优化方法

我们按 4.1.2 中介绍的方法得到了保证精度的节点向量, 但由于节点选取的随机性, 有些节点是多余的, 也就是去掉这些节点, 样条也可以达到精度要求。我们希望节点的数目尽量少, 所以要对得到的节点组进行优化。我们在保证精度的前提下, 首先处理节点分布密集的地方, 因为这些地方出现多余节点的概率较大。我们的做法是, 找到间距最小的点 u_i, u_{i+1} , 判断在保持其它节点不变的条件下, 分别去掉 u_i, u_{i+1} 后, 节点向量是否可以保证精度。如果去掉 $u_i (u_{i+1})$ 可以, 而去掉 $u_{i+1} (u_i)$ 不行, 则节点向量为去掉 $u_i (u_{i+1})$ 后剩下的节点构成的向量。如果两个点都可以去掉, 我们比较 $u_i - u_{i-1}$ 和 $u_{i+2} - u_{i+1}$ 。因为我们希望得到的节点分布尽量均匀, 所以若 $u_i - u_{i-1} < u_{i+2} - u_{i+1}$, 则优先去掉 u_i 。反之, 优先去掉 u_{i+1} 。如果两个点都不可以去掉, 节点向量不变。这样循环下去, 每次去掉一个点, 直到不能保证精度为止, 可以把稠密部分的节点进行约减。然而, 在节点分布稀疏的地方, 也有可能存在多余的节点, 所以我们接下来判断每个节点在保持其它节点条件下是否可以去掉。出于让节点尽量均匀分布的目的, 在所有可以去掉的节点中, 优先去掉与其相邻两节点的最小距离最小的节点。这样每次去掉一个节点循环做下去, 直到不能保证精度为止。

算法 2

- 步骤 1 在节点向量中找到相邻最近的两点 u_i, u_{i+1} 。若 $u_i = 0$ ，执行步骤 2。若 $u_{i+1} = 1$ ，执行步骤 3。否则执行步骤 4。
- 步骤 2 在保持其它节点不变的条件下，去掉 u_{i+1} ，判断此时的节点向量是否可以保证精度。如果可以，节点向量为去掉 u_{i+1} 后剩下的节点构成的向量。否则，节点向量不变。
- 步骤 3 在保持其它节点不变的条件下，去掉 u_i ，判断此时的节点向量是否可以保证精度。如果可以，节点向量为去掉 u_i 后剩下的节点构成的向量。否则，节点向量不变。
- 步骤 4 在保持其它节点不变的条件下，分别去掉 u_i, u_{i+1} ，判断节点向量是否可以保证精度。如果去掉 $u_i (u_{i+1})$ 可以，而去掉 $u_{i+1} (u_i)$ 不行，则节点向量为去掉 $u_i (u_{i+1})$ 后剩下的节点构成的向量。如果两个点都可以去掉，若 $u_i - u_{i-1} < u_{i+2} - u_{i+1}$ ，去掉 u_i 。否则，去掉 u_{i+1} 。如果两个点都不可以去掉，节点向量不变。
- 步骤 5 判断节点是否减少，如果减少，执行步骤 1；否则，执行步骤 6。
- 步骤 6 判断每个节点在保持其它节点条件下是否可以去掉。如果节点都不可去，则节点不变。否则，执行步骤 7。
- 步骤 7 在所有可去节点中，找到与相邻两节点的最小距离最小的节点。节点向量为去掉这个点后剩下的节点构成的向量。
- 步骤 8 判断节点是否减少，如果是，执行步骤 6；否则，执行步骤 9。
- 步骤 9 程序结束。

4.2 数值实验

我们首先用三次 B 样条逼近均匀分布的 150 个数据点，加上 0.001 范围内的随机扰动误差，规定逼近精度为 0.005。

设初始内节点数为 20，给出 20 个随机数，由(4.1)求出内节点向量。从而得到三次 B 样条基函数。对数据点进行累加弦长参数化，将参数值带入 B 样条基函数，得到系数矩阵。用最小二乘法求解矛盾方程，得到控制点。这样，我们可以写出三次 B 样条拟合曲线。计算拟合曲线的逼近误差，此时误差大于精度 ϵ 。于是，舍弃这组节点向量。重新给出一组随机数，构造新的节点向量。重复上述过程。在尝试了 300 次不同的随机分布节点之后，没有找到这样的节点向量，使拟合曲线的逼近误差满足精度要求。于是，将节点数增加一，即给出 21 个随机数，随机构造 21 个内节点，重复上述过程。最后，

在内节点数为 51 的时候，找到了满足精度要求的节点向量。对这个节点向量按照算法 2 进行优化，优化后的节点数目为 16。得到逼近曲线如图 4.1 所示。节点分布如图 4.2 所示。

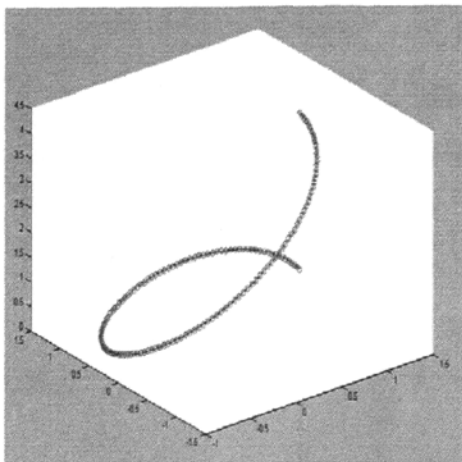


图4.1 三次样条逼近150个均匀有噪数据点

Fig.4.1 B-spline fitting curve and 150 uniform data points with random noisy

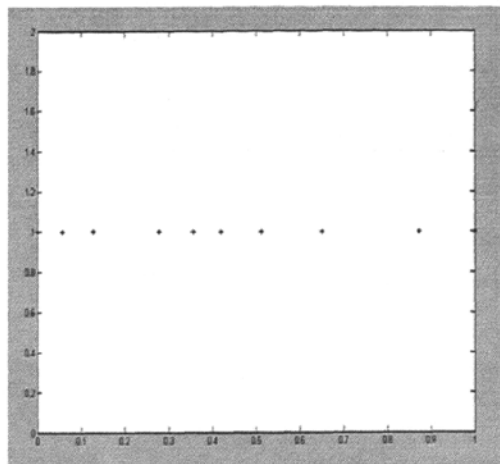


图4.2 节点分布

Fig.4.2 Distribution of knots

我们在数据点不变的条件下，提高逼近精度到 0.001。由算法 1，算法 2 得到逼近曲线如图 4.3 所示。节点数目为 23，分布如图 4.4 所示。

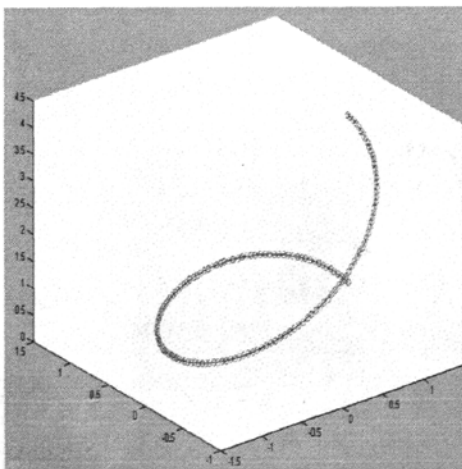


图4.3 三次样条逼近150个均匀数据点

Fig.4.3 B-spline fitting curve and 150 uniform data points

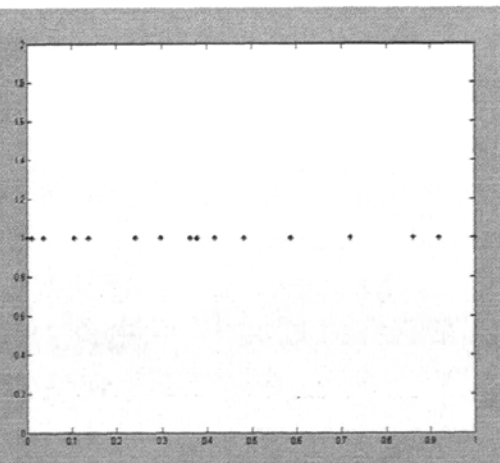


图4.4 节点分布

Fig.4.4 Distribution of knots

我们取随机分布的 150 个数据点，加上 0.001 范围内的扰动误差，规定逼近精度为 0.001，得到逼近曲线如图 4.5 所示。节点数目为 24，分布如图 4.6 所示。

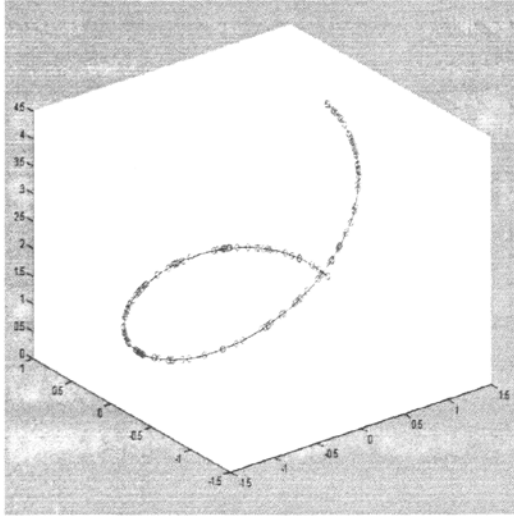


图4.5 三次样条逼近150个非均匀数据点

Fig.4.5 B-spline fitting curve and 150 non-uniform data points

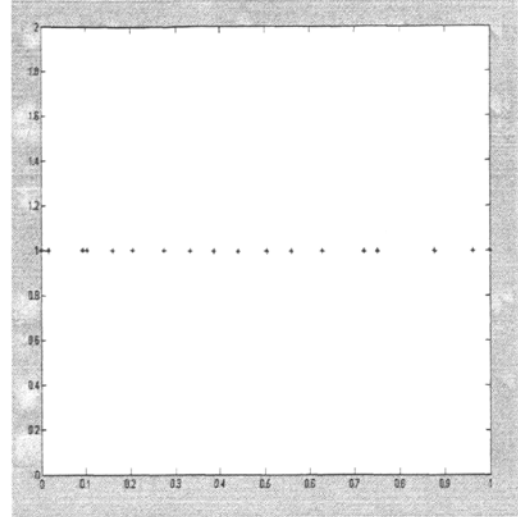


图4.6 节点分布

Fig.4.6 Distribution of knots

在运行过程中，由于节点的随机性，我们需要经过多次尝试，才能找到符合精度要求的节点组。如果节点的数目过少，样条达到精度的概率就很小，从而延长了选择节点的时间。而如果节点数目过多，会直接导致控制点数过多，从而在用最小二乘法求解矛盾方程组反求样条控制点时的计算量过大，计算时间增加。优化节点的时间也会增加。考虑到这些因素，我们希望合理地增加节点数目，提高算法的计算效率。已知三维空间数据点的数目 w ，由控制点与节点在数量上的关系，最小二乘逼近三次 B 样条曲线的节点数最多为 $w+4$ ，最少为 2（区间的两个端点）。我们取均匀分布的 150 个数据点，加上 0.001 范围内的扰动误差，规定逼近精度为 0.001，连续做 15 次实验。

我们考察两种不同的节点数目变化方式对运行时间的影响。

第一种做法是首先对 N 赋初值，若在这个数目下没有找到合适的节点向量，则 $N = N + 1$ 。继续寻找节点向量，直到找到满足精度要求的节点向量为止。

第二种做法是首先赋初值 $low = 2, high = w + 4, N = \text{ceil}((low + high) / 2)$ ，若在这个数目下没有找到合适的节点向量，则 $low = N; N = \text{ceil}((low + high) / 2)$ 。

下面我们对第二种方法的 N 进行分析。

考察序列 $\{x_n\}_{n=1}^{+\infty}$ ，满足 $x_n = \frac{x_{n-1} + a}{2}$ ，则

$$|x_n - a| = \left| \frac{x_{n-1} + a}{2} - a \right| = \left| \frac{x_{n-1} - a}{2} \right| = \left| \frac{x_{n-2} - a}{2^2} \right| = \dots = \frac{|x_1 - a|}{2^{n-1}}$$

所以 $x_n \rightarrow a, (n \rightarrow \infty)$ ，且 $x_n < a$ 。

所以，按第二种方式增加内节点数，有 $N \rightarrow high$ ，且 $N < high$ 。继续寻找节点向量，直到找到满足精度要求的节点向量为止。

两种方法的运行时间和节点数如表 4.1 所示。（PC 配置：2.93GHz,512MB）

表4.1 运行时间和节点数目

Tab. 4.1 Time of running and number of knots

<i>Time(s)</i>	<i>No. of Knots</i>	<i>Time(s)</i>	<i>No. of Knots</i>
298.0620	23	58.4540	23
387.1250	25	39.1250	25
315.6090	22	57.2190	22
206.3280	23	52.1560	23
293.2030	25	102.9220	23
390.8900	23	48.8750	23
342.6250	23	53.5310	21
487.0160	22	141.9070	21
430.8910	25	42.0780	25
388.8280	24	62.7180	23
213.8280	22	70.0470	22
586.3910	23	53.4220	23
477.8280	22	68.2340	25
503.9850	23	62.0150	22
389.1720	22	48.3750	24
Average	Average	Average	Average
Time	Number	Time	Number
378.3894	23	60.0035	23

由于节点的随机性，我们去掉每一组运行时间的最大值和最小值，然后计算平均时间。由表 1 的数据可以看出，第二种方法很大程度上节省了时间，而且这种运行效率的

提高并没有以节点数目的增加为代价。也保证了节点的优化。这种优势在大规模数据拟合的过程中体现得更加明显。

通过多次数值实验，我们发现内节点数在 100 至 300 的范围内的时候，可以保持较快的运算速度。所以当拟合的数据点数较少（如小于 300）的时候，我们按照第二种方式增加内节点数。当数据点数比较多（如大于 300）的时候，为了保证计算速度，我们希望将内节点数控制在 300 以内，可以找到满足精度要求的节点向量。于是我们令 $high = 300$ ，按照第二种方式增加 N ，则 $N < 300$ ，且 $N \rightarrow 300$ 。实验证明我们的方法是可行的。

图 4.7 表示均匀分布并在 0.001 范围内加随机扰动误差的 1500 个数据点以及三次 B 样条逼近曲线，逼近误差为 0.0008697。子图为局部放大图。节点数目为 23，分布如图 4.8 所示。

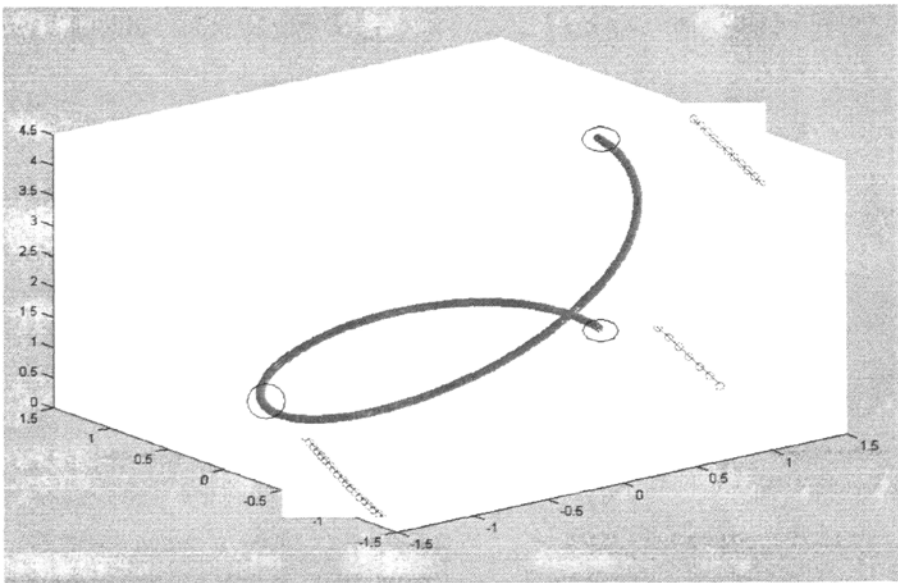


图4.7 三次样条逼近1500个均匀有噪数据点
Fig.4.7 B-spline fitting curve and 1500 uniform data points
with random noisy

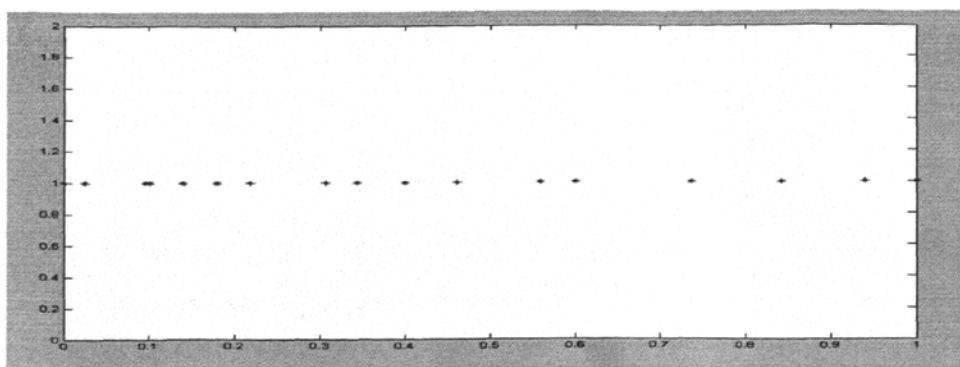


图8 节点分布

Fig.8 Distribution of knots

图 4.9 是对 1000 个数据点拟合的 B 样条曲线, 逼近误差为 0.0076064, 节点数为 26, 分布如 4.10 所示。

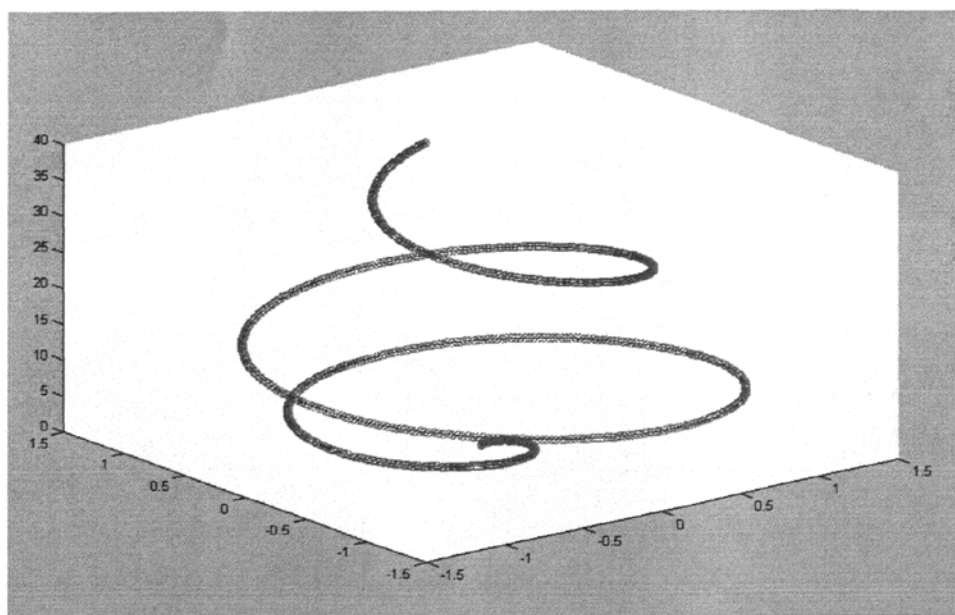


图4.9 三次样条逼近1000个均匀有噪数据点

Fig.4.9 B-spline fitting curve and 1000 uniform data points with random noisy

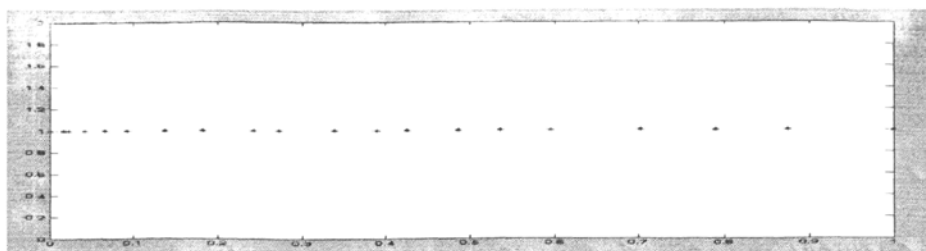


图4.10 节点分布

Fig.4.10 Distribution of knots

图 4.11 的数据点轨迹为非正则曲线，数据点数为 150，逼近误差为 0.0031119。子图为局部放大图。节点数为 13，分布如图 4.12 所示。

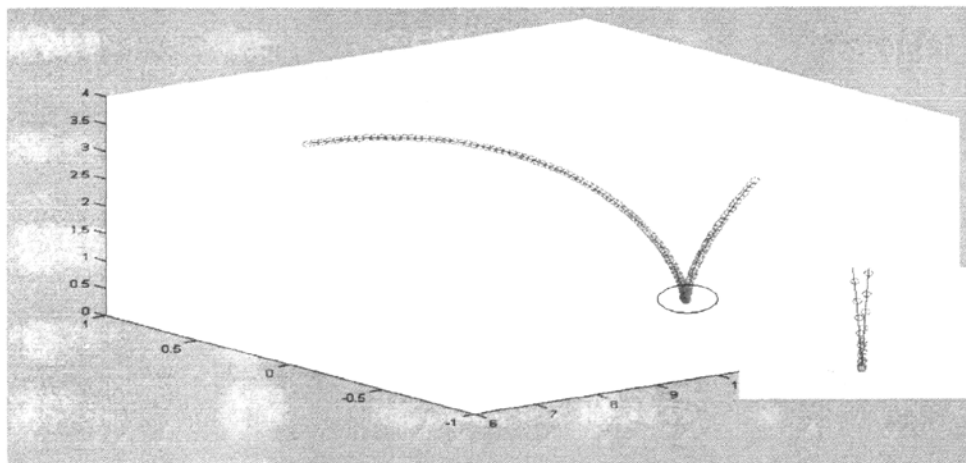


图4.12 三次样条逼近150个均匀数据点

Fig.4.11 B-spline fitting curve and 150 uniform data points

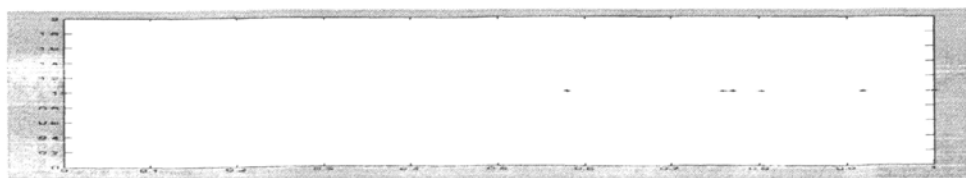


图4.12 节点分布

Fig.4.12 Distribution of knots

结 论

本文以一种随机的方法研究三次 B 样条最小二乘逼近问题。我们首先在一定数目的样条节点下, 随机选取节点, 寻找满足精度要求的节点向量。如果在概率 1 的意义下没有找到这个数目的节点向量, 就增加节点数目, 直到找到可以使三次 B 样条曲线达到预先给定的逼近精度为止。在得到节点向量后, 在保证逼近精度的前提下, 减少节点数量, 优化节点分布, 从而使逼近曲线具有逼近精度高, 分段均匀, 段数少的优点。本文计算简单, 易于程序实现。由于我们使用的是随机方法, 因此在理论上应该可以计算出在每一给定的节点数目下, 运行 m 次可以找到最优拟合样条的概率。这将是我们的下一步的重点工作。

参 考 文 献

- [1] H. G. Burchard, Spline (with optimal knots) are better, *Applicable Anal.*, 3 (1974), p. 309.
- [2] C. De Boor, Least squares cubic spline approximation, I: Fixed knots, CSD TR 20, Purdue Univ., Lafayette, IN, 1968.
- [3] J. R. Rice, Least squares cubic spline approximation, II: Variable approximation, CSD TR 21, Purdue Univ., Lafayette, IN, 1968.
- [4] M. J. D. Powell, A rapidly convergent knots, CSD TR 21, Purdue Univ., 1968.
- [5] P. D. Loach, A. J. Wathen, On the best least squares approximation of continuous functions using linear spline with free knots, *IMA J. Numer. Anal.* 11 (1991) 393-409.
- [6] W. C. Davidon, Variable metric method for minimization, A. E. C. Research and Development Report ANL-5990, Atomic Energy Commission, Argonne, TN, 1959.
- [7] T. Schatzke, H. Schwetlick, Constrained approximation by spline with free knots, *BIT* 37 (1997) 105-137.
- [8] H. Schwetlick, Least squares approximation by spline with free knots, *BIT* 35 (1995) 361-384.
- [9] P. Dierckx, Curve and surface fitting with spline, New York, Heidelberg, 1978.
- [10] Gleb eliakov, Least squares splines with free knots: global optimization approach, *Appl. Math. and Computation* 149 (2004) 783-398.
- [11] M. Andramonov., Cutting angle method in global optimization, *Appl. Math. Lett.* 12 (1999) 95-100.
- [12] A. Torm, A. Zilinskas, Global optimization, Springer, Heilelberg, 1989.
- [13] P. G. Guest, Numerical Methods of Curve Fitting, Cambridge Univ. Press, Cambridge, 1961.
- [14] C. R. Hobby, J. R. Rice, Approximation from a curve of functions, *Arch. Rational Mech. Anal.*, 27 (1967), 91-106.
- [15] D. L. B. Jupp, Curve fitting by splines as an example for centre to limb data, *Astronom. Astrophys.*, 30 (1974), 185-188.
- [16] J. Kowalik, M. R. Osborne, Method for unconstrained optimization problems, Elsevier. New York, 1968.
- [17] D. W. Marouardt, An algorithm for least-squares estimation of nonlinear parameters, *J. Soc. Indust, Appl. Math.*, 11 (1963), 431-441.
- [18] M. R. Osborne, D. M. Ryan, On penalty function methoes for nonlinear programming problem, *J. Math. Anal. Appl.*, 31 (1970) 559-578.
- [19] J. K. Reid, A note on the least squares solution of a band system of linear equations by Householder reductions, *Comput. J.*, 10 (1967), 188-189.

- [20] F. Bessaoud, J. P. Daures, N. Molinari. Free Knot Spline for Logistic Models and Threshold Selection. Computer Methods and Programs in Biomedicine (2005).
- [21] C. De Boor. A Practical Guide to Spline. Springer-Verlag, New York, 1978.
- [22] 程正兴, 数据拟和, 西安: 西安交通大学出版社, 1986.
- [23] 王仁宏, 数值逼近, 北京: 高等教育出版社, 2003.
- [24] 徐树方, 高立, 数值线性代数, 北京: 北京大学出版社, 2004.
- [25] 叶慈南, 应用数理统计, 北京: 机械工业出版社, 2004.

攻读硕士学位期间发表学术论文情况

李晓晔，罗钟铉，三次 B 样条数据拟合问题的节点自适应选取算法（第四章），大连理工大学研究生网络学刊（已接收）。

致 谢

首先由衷地感谢我的导师罗钟铉教授。几年来，罗老师在学习、研究和生活方面给予了我巨大的支持和帮助。罗老师一直本着科研与实践相结合的理念来培养我分析解决问题的能力，科研实践创新的能力。也正是他的悉心的指导和不倦的教诲，才有了这篇论文的完成。

两年的研究生生涯，同学间的友谊和关爱令我终生难忘。在朝夕相处的生活中，留下了许多美好的回忆。感谢孟兆良、薛均晓、刘宇、冯二宝、于冉、王倩、尹显文、戴仕全、曾辉、夏静、林树、刘杰、百华睿、陈露等师兄姐妹对我学习和生活上的关心和帮助，大家共同学习，互帮互助，度过了许多难忘的时光。

感谢父母多年的培养和无私的奉献，是他们的关怀和鼓励使我不断地进步。

谨以此文献给所有关心、支持和帮助过我的人们！

最后，要感谢评阅，评议硕士论文和出席硕士论文答辩会的各位老师，感谢他们在百忙的工作中能给予指导。

李晓晔

2008 年 6 月于大工校园