

# 光纤传感数据驱动的曲率估计与曲线重构分析

## 摘要

摘要本研究围绕华中杯数学建模竞赛 C 题——基于光纤传感器的平面曲线重建算法展开，旨在开发一种能够利用光纤传感器数据通过数学建模准确重构出监测结构的平面曲线形态的方法。通过对光纤传感器捕获的数据进行分析，本研究采用了三次样条插值和微分几何方法来估算曲率，并通过迭代算法重构出平面曲线。

**对于问题一：基于插值的曲率估计模型。**们首先通过对光纤传感器捕获的波长变化数据进行分析，建立了曲率与波长关系的数学模型。利用三次样条插值方法，对离散的曲率数据进行连续化处理，以确保曲率估计的准确性和连续性。这一处理步骤为后续曲线的重构提供了必要的基础数据，确保了模型的精确度和实用性。具体问题一答案如下表所示：

横坐标 x (米)	0.3	0.4	0.5	0.6	0.7
测试 1 曲率	2.2156	2.2151	2.2155	2.2167	2.2152
测试 2 曲率	2.9824	2.9810	2.9796	2.9782	2.9769

**对于问题二：基于微分几何的曲线拟合模型。**在获得连续化的曲率数据后，本问题采用微分几何方法对平面曲线进行拟合。通过曲率与弧长的关系，结合初始条件，我们运用积分和迭代方法，精确重建了曲线的形状。此外，本文还详细分析了重构曲线的特点，如几何形状和物理属性，进一步验证了模型的有效性和准确性。

**对于问题三：基于等弧长原则的曲线离散算法平面重构模型。**本问关注于如何利用等弧长采样点来重构平面曲线。我们首先确定了采样点的选择算法，然后通过曲率积分方法计算每一段圆弧的位置，从而递推出整条曲线的形状。在此基础上，本文对重构曲线与理论曲线进行了详细比较，分析了误差产生的可能原因，提出了改进措施，以增强模型的适应性和准确性。

研究结果显示，该模型能够有效利用传感器数据，准确重构曲线形状，并对结构健康监测提供了新的方法论。本模型还具有良好的灵敏度和推广性，可应用于更广泛的工程和科研场合。研究不仅推动了光纤传感技术在结构健康监测中的应用，也为相关领域提供了重要的理论和实际价值。

**关键字：** 光纤传感器   数学建模   曲线重建   三次样条插值   微分几何

## 目录

一、问题重述 .....	4
1.1 问题背景 .....	4
1.2 问题分析 .....	5
二、模型假设 .....	6
三、符号说明 .....	6
四、问题一：基于插值的曲率估计模型 .....	7
4.1 曲率获取 .....	7
4.2 离散曲率数据的连续化处理 .....	7
4.2.1 基于三次样条的曲率插值模型 .....	7
4.2.2 模型求解 .....	9
4.3 模型求解结果及其应用 .....	10
五、问题二：基于微分几何的曲线拟合模型 .....	10
5.1 曲线拟合与分析模型的建立 .....	10
5.2 由曲率积分迭代实现曲线拟合 .....	11
5.2.1 曲线的微分几何学基础 .....	11
5.2.2 曲线重构的迭代算法 .....	11
5.3 重构曲线的特点分析 .....	13
六、问题三：基于等弧长原则曲线离散算法的平面曲线重构模型 .....	14
6.1 平面曲线的重构 .....	14
6.1.1 等间距弧长采样点的选取算法 .....	14
6.1.2 采样点曲率的计算 .....	14
6.1.3 由曲率积分方法递推曲线位置 .....	14
6.2 模型结果对比分析 .....	15
6.3 重构曲线的误差分析 .....	16
6.3.1 因素一：误差的积累 .....	16
6.3.2 因素二：误差的放大 .....	16
6.3.3 因素三：采样点的密集程度 .....	17
七、模型的评价 .....	17
7.1 模型的优点 .....	17

7.2 模型的缺点 .....	17
7.3 改进建议 .....	17
八、模型的推广 .....	18
参考文献 .....	18
A 附录 文件列表 .....	21
B 附录 代码 .....	21

# 一、问题重述

## 1.1 问题背景

光纤传感技术作为一种先进的传感技术，因其独特的物理属性和广泛的适用性在众多领域展示了显著的优势。光纤传感器能够在极其恶劣的环境下工作，如高温、高压和强辐射环境，且具有不干扰、高灵敏度和可远程传感的特点。这使得光纤传感器在航空航天、医疗、土木工程、石油开采等众多关键领域有着重要的应用。

光纤传感器的独特之处在于其能够提供连续、实时的监测数据，这对于需要高精度监控的应用尤其关键。在桥梁和大型建筑物的结构健康监测中，传统的传感技术可能因环境干扰或数据采集间隔而遗漏关键数据。相比之下，光纤传感器可以实时传输数据，确保从微小的应力变化到大规模的结构变形都能被准确记录和分析。

随着光纤传感技术的发展，其在结构健康监测中的应用逐渐成为研究热点。在结构健康监测中，通过安装在结构关键部位的光纤传感器，可以实时监测到结构的微小变化，如裂纹的扩展、应力和应变的变化等。这种监测方式对于提前预防结构性故障、延长服务寿命以及保证使用安全具有重要意义。

本届华中杯数学建模竞赛的 C 题——基于光纤传感器的平面曲线重建算法，研究的核心是如何利用光纤传感器捕获的数据，通过数学建模和算法开发，准确地重建出受监测结构的平面曲线形态。这项技术的研究不仅能够提升传感器数据的应用价值，增强结构健康监测的准确性和效率，还有助于推动光纤传感技术在更广泛领域的应用，如在医疗内窥镜中精确追踪器械位置，在机器人技术中提高导航与定位的准确性等。平面曲线的精确重建对于理解和分析物体的运动机制、力学性能及环境适应性也具有重要的理论和实际价值。故本研究不仅是一个纯粹的技术问题，更是一个具有广泛科学和工程价值的实际应用问题。

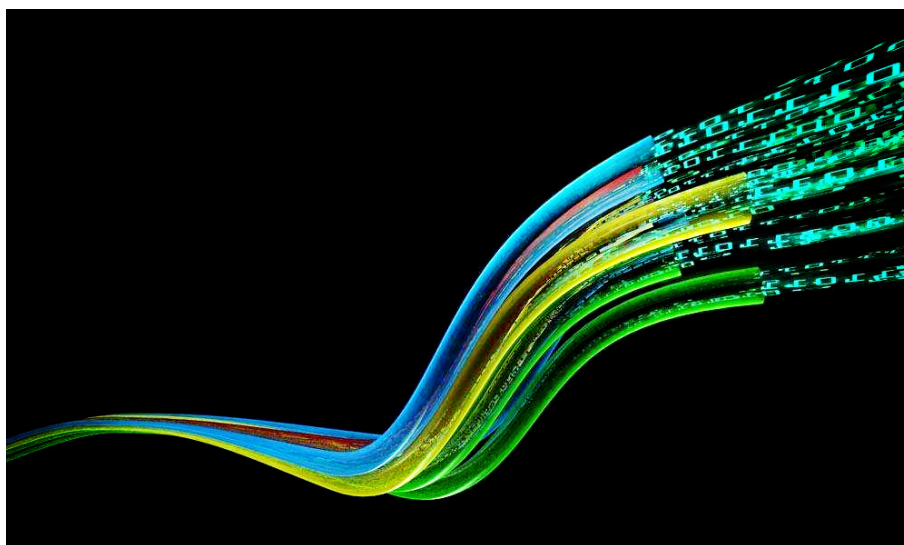


图 1 光纤

## 1.2 问题分析

本研究旨在探索光纤传感器在平面曲线重建中的应用。通过光纤传感技术，我们可以利用光波参数（如波长、相位和强度）的变化来感知和解析外界环境的信息。这些参数变化通过传感器的解调系统进行处理，从而能够获取关于结构或物体形变的详细数据。

光纤传感器因其体积小、抗电磁干扰能力强、灵敏度高等特点，在多个关键领域显示出极高的应用价值。特别是在结构健康监测领域，光纤传感技术可以实时捕捉结构的微小变化，如裂纹的形成和扩展、应力和应变的变化等。C 题要求利用光纤传感器测得的数据，解决以下三个数学建模问题：

**问题 1** 基于提供的波长测量数据，建立数学模型以估算平面光栅每个传感点 (FBG1 至 FBG6) 的曲率。假设光纤初始状态为水平放置，受力后形成 45 度角，要求模型能估算在特定横坐标位置 ( $x$  轴) 的曲率。

**问题 2** 使用问题一中计算得出的曲率数据，构建数学模型来重构平面曲线。分析重构曲线的形状特征及其与原始结构的关系。

**问题 3** 根据平面曲线方程  $y = x^3 + x$  ( $0 \leq x \leq 1$ )，进行等间距弧长采样，计算这些采样点的曲率。利用采样得到的曲率数据，构建数学模型重构曲线。分析重构曲线与给定的原始曲线之间的误差原因。

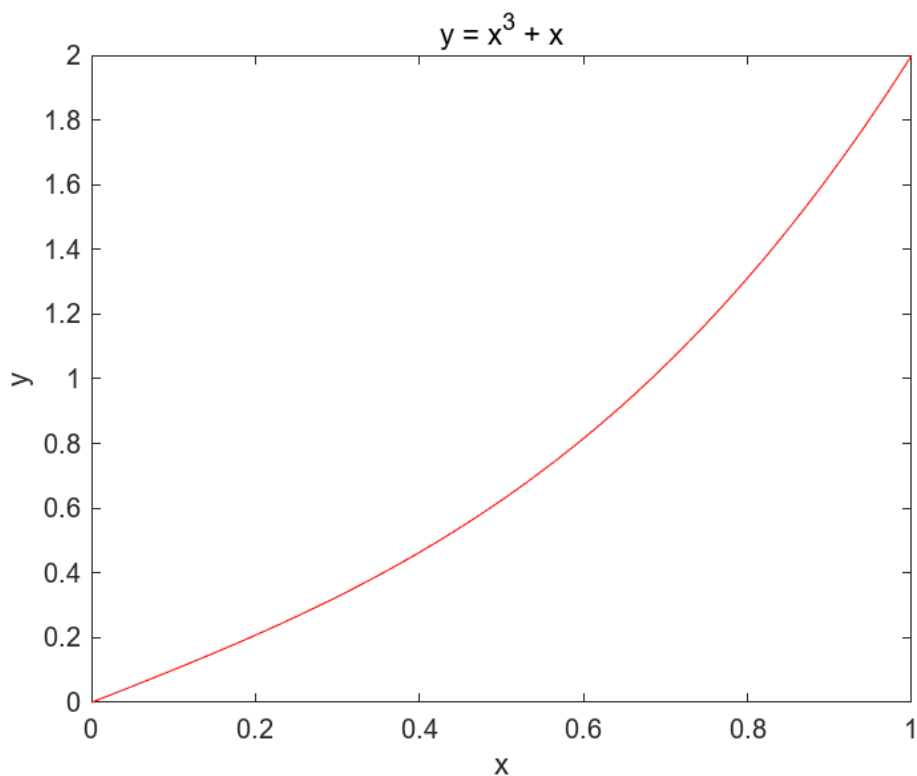


图 2 函数曲线  $y = x^3 + x$  ( $0 \leq x \leq 1$ )

## 二、模型假设

为简化问题，本文做出以下假设：

- **假设 1：**假设光纤在受力形变过程中的凹凸性始终不变。
- **假设 2：**在使用三次样条进行曲率插值时，需要确保插值函数在相邻的数据点处具有几何连续性，即函数的一阶导数和二阶导数在数据点处连续。同时需要插值函数具有指定边界条件，例如自然边界条件（二阶导数在端点处为零）、固定边界条件（在端点处指定一阶导数值）或周期边界条件（要求函数在首尾端点处的值和导数相等）等。
- **假设 3：**在使用曲率积分迭代实现曲线拟合时，需要假设所取的圆弧段的数目能够满足两点间的弦长与弧长近似相等的要求，即我们能够选取到合适的步长  $\Delta S$  去计算下一个点的位置，确保算法能够正确的重构曲线，并逼近原曲线的位置。
- **假设 4：**在使用等间距弧长采样算法时，需要假设起始点  $x_m := x_{\text{start}}$ ，终止点  $x_m := x_{\text{start}}$  存在并且已知，能够通过迭代去推导出各离散点的位置。（ $:=$  为赋值号）

## 三、符号说明

符号	说明	单位
$\kappa$	曲率	$m^{-1}$
$\lambda$	波长	$nm$
$s$	弧长	$nm$
$a_i, b_i, c_i, d_i$	多项式系数	
$r$	曲线位置向量	
$\alpha$	切向量	
$\varphi$	切向量与 x 轴正方向所成的角	$rad$
$R$	曲率半径	$m$
$x_{\text{start}}$	曲线起始点	
$x_{\text{end}}$	曲线终止点	
$step$	步长	$m$
$x_{\text{target}}$	目标点列	
$L$	弧长	$m$

## 四、问题一：基于插值的曲率估计模型

### 4.1 曲率获取

已知波长  $\lambda$  与曲线曲率的近似关系为

$$\kappa = \frac{c(\lambda - \lambda_0)}{\lambda_0} \quad (1)$$

其中  $\lambda_0$  为光纤在初始状态下测量的波长， $\lambda$  为光纤受应力后测量的波长。

根据表 1 中两次测试的结果，可以计算出第  $i$  个测量点的曲率：

$$\kappa_{i,j} = \frac{c(\lambda_{i,j} - \lambda_{0,j})}{\lambda_{0,j}} \quad (2)$$

其中  $\lambda_{i,j}$  为 FBGi 点在第  $j$  次测试中测量的波长值， $\lambda_{0,j}$  为 FBGi 点第  $j$  次测试的初始状态。

### 4.2 离散曲率数据的连续化处理

经过以上步骤得到的曲率数据为离散的，根据曲率的定义式：

$$\kappa = \frac{\frac{dx}{ds} \cdot \frac{d^2y}{ds^2} - \frac{dy}{ds} \cdot \frac{d^2x}{ds^2}}{\left( \frac{dx^2}{ds} + \frac{dy^2}{ds} \right)^{3/2}} \quad (3)$$

若曲线是连续光滑的，则各点处的曲率必存在，因此为了计算出横坐标对应位置处的曲率，必须对离散曲率进行插值，以得到更多点对应的曲率。常用的插值方式有：线性插值、三次样条插值、多项式插值等。

#### 4.2.1 基于三次样条的曲率插值模型

三次样条插值的基本思想是在每段小区间上用三次多项式来逼近小区间内的数据，并利用区间端点处的几何连续性和光滑性来确定区间端点的信息。具体来说，对于目标函数  $\kappa(s)$ ,  $0 = x_0 \leq x_1 < x_2 < \dots < x_n = 3$  的插值函数  $K(s)$ ，满足以下条件：

1. 每段小区间上为三次多项式：

$$K_i(s) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3, s \in [x_i, x_{i+1}]$$

2. 曲线位置连续性：  $K(x_i) = \kappa_i$
3. 曲线切线连续性：  $K'_{i-1}(x_i) = K'_i(x_i)$
4. 曲线曲率连续性：  $K''_{i-1}(x_i) = K''_i(x_i)$

其中  $K_i(s)$  为  $K(s)$  在区间  $[x_i, x_{i+1}]$  上的表示形式. 在本题的模型中, 额外假定三次样条曲线满足自然边界条件, 也即:

$$K''(x_0) = K''(x_n) = 0.$$

上述方程整理后可得到以下方程组:

$$\begin{cases} a_i = \kappa_i, i = [0, \dots, n-1] \\ b_i(x_{i+1} - x_i) + c_i(x_{i+1} - x_i)^2 + d_i(x_{i+1} - x_i)^3 = \kappa_{i+1} - \kappa_i, i = [0, \dots, n-1] \\ b_i + 2c_i(x_{i+1} - x_i) + 3d_i(x_{i+1} - x_i)^2 = b_{i+1}, i = [0, \dots, n-2] \\ 2c_i + 6d_i(x_{i+1} - x_i) = 2c_{i+1}, i = [0, \dots, n-2] \end{cases} \quad (4)$$

消去  $a_i, b_i, d_i$  后整理得:

$$c_i(x_{i+1} - x_i) + 2(x_{i+2} - x_i)c_{i+1} + (x_{i+2} - x_{i+1})c_{i+2} = 3 \left[ \frac{\kappa_{i+2} - \kappa_{i+1}}{x_{i+2} - x_{i+1}} - \frac{\kappa_{i+1} - \kappa_i}{x_{i+1} - x_i} \right] \quad (5)$$

其中

$$\begin{cases} a_i = \kappa_i \\ b_i = \frac{\kappa_{i+1} - \kappa_i}{x_{i+1} - x_i} - c_i(x_{i+1} - x_i) - \frac{1}{3}(c_{i+1} - c_i)(x_{i+1} - x_i) \\ d_i = \frac{c_{i+1} - c_i}{3(x_{i+1} - x_i)} \\ c_0 = c_n = 0 \end{cases} \quad (6)$$

将方程组表示为矩阵形式:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ x_1 - x_0 & 2(x_2 - x_0) & (x_2 - x_1) & 0 & \dots & 0 \\ 0 & x_2 - x_1 & 2(x_3 - x_1) & (x_3 - x_2) & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & \vdots & 0 & x_{n-1} - x_{n-2} & 2(x_n - x_{n-2}) & (x_n - x_{n-1}) \\ 0 & \vdots & \vdots & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

$$AC = \begin{bmatrix} 0 \\ \frac{\kappa_2 - \kappa_1}{x_2 - x_1} - \frac{\kappa_1 - \kappa_0}{x_1 - x_0} \\ \frac{\kappa_3 - \kappa_2}{x_3 - x_2} - \frac{\kappa_2 - \kappa_1}{x_2 - x_1} \\ \vdots \\ \frac{\kappa_n - \kappa_{n-1}}{x_n - x_{n-1}} - \frac{\kappa_{n-1} - \kappa_{n-2}}{x_{n-1} - x_{n-2}} \\ 0 \end{bmatrix} \quad (8)$$



在本题中,  $x_i = 0.6 \times i$ , 因此矩阵可表达为:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 & 0 & 0 \\ 0 & 1 & 4 & 1 & 0 & 0 \\ 0 & 0 & 1 & 4 & 1 & 0 \\ 0 & 0 & 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix} = \frac{3}{0.36} \begin{bmatrix} 0 \\ \kappa_2 - 2\kappa_1 + \kappa_0 \\ \kappa_3 - 2\kappa_2 + \kappa_1 \\ \kappa_4 - 2\kappa_3 + \kappa_2 \\ \kappa_5 - 2\kappa_4 + \kappa_3 \\ 0 \end{bmatrix} \quad (9)$$

带入 (6) 式中可得:

$$\begin{cases} a_i = \kappa_i \\ b_i = \frac{\kappa_{i+1} - \kappa_i}{0.6} - 0.6c_i - 0.2(c_{i+1} - c_i) \\ d_i = \frac{c_{i+1} - c_i}{1.8} \\ c_0 = c_n = 0 \end{cases} \quad (10)$$

#### 4.2.2 模型求解

利用 Matlab 软件对上述的线性方程组进行求解, 求得测试 1 和测试 2 插值函数  $K_1(s), K_2(s)$ :

$$K_1(s) = \begin{cases} 2.2195 - 0.0159s + 0.0314s^3, s \in [0, 0.6] \\ 2.2167 + 0.0180(s - 0.6) - 0.0565(s - 0.6)^2 - 0.0680(s - 0.6)^3, s \in [0.6, 1.2] \\ 2.2332 + 0.0124(s - 1.2) - 0.0658(s - 1.2)^2 + 0.0624(s - 1.2)^3, s \in [1.2, 1.8] \\ 2.2305 + 0.0008(s - 1.8) + 0.0465(s - 1.8)^2 - 0.0546(s - 1.8)^3, s \in [1.8, 2.4] \\ 2.2360 - 0.0022(s - 2.4) - 0.0157(s - 2.4)^2 + 0.0287(s - 2.4)^3, s \in [2.4, 3] \end{cases} \quad (11)$$

$$K_2(s) = \begin{cases} 2.9864 - 0.0131s - 0.0014s^3, s \in [0, 0.6] \\ 2.9782 - 0.0146(s - 0.6) - 0.0025(s - 0.6)^2 + 0.0196(s - 0.6)^3, s \in [0.6, 1.2] \\ 2.9727 + 0.0035(s - 1.2) + 0.0327(s - 1.2)^2 - 0.0264(s - 1.2)^3, s \in [1.2, 1.8] \\ 2.9809 + 0.0143(s - 1.8) - 0.0148(s - 1.8)^2 - 0.0024(s - 1.8)^3, s \in [1.8, 2.4] \\ 2.9836 - 0.0060(s - 2.4) - 0.0190(s - 2.4)^2 + 0.0106(s - 2.4)^3, s \in [2.4, 3] \end{cases} \quad (12)$$

两次测试中的曲率与弧长的关系如图 3 所示, 可以看到, 曲率随弧长的变化不明显, 从曲率中可以估计出目标曲线与圆较接近.

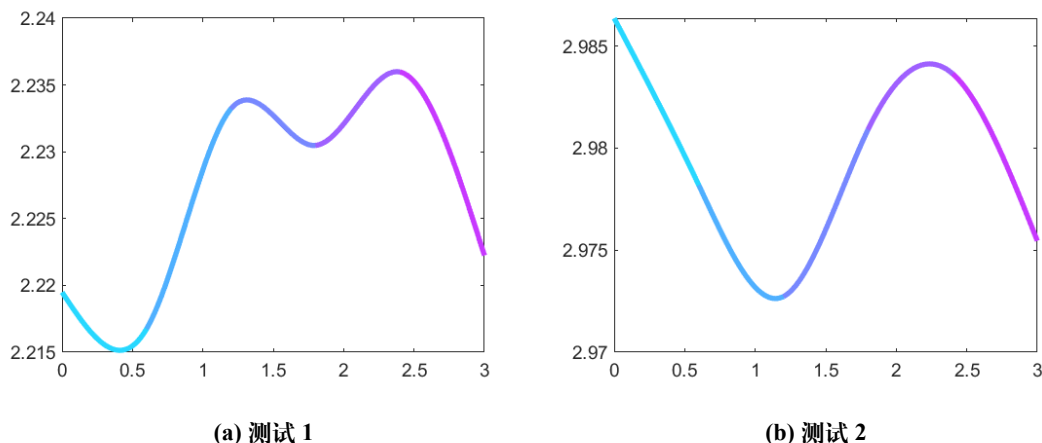


图 3 两次测试中曲率与弧长的关系

### 4.3 模型求解结果及其应用

假设题目中说的横坐标为  $x$  的对应位置与弧长  $s$  等价，利用上述的曲率模型，可以求出横坐标为  $x$  对应位置的曲率  $k(s)$ ，如表所示，是两次测试中 0.3、0.4、0.5、0.6、0.7 米处的曲率：

横坐标 $x$ (米)	0.3	0.4	0.5	0.6	0.7
测试 1 曲率	2.2156	2.2151	2.2155	2.2167	2.2152
测试 2 曲率	2.9824	2.9810	2.9796	2.9782	2.9769

表 1 横坐标  $x$  轴相应位置处的曲率

## 五、 问题二：基于微分几何的曲线拟合模型

### 5.1 曲线拟合与分析模型的建立

面对问题二，本文将使用问题一中得到的曲率数据，并结合表一所提供的波长数据，这些数据将作为重要的输入参数，基于微分几何原理来构建一个曲线拟合模型。

微分几何是数学的一个分支，主要研究曲线、曲面以及更高维度上的流形的几何性质。它通过微积分、线性代数和微分方程等数学工具，研究空间中的形状、大小和位置的变化。

这个模型的目标是能够根据输入的参数精确地重构出平面曲线，从而为提供一种直观的方式来分析和理解曲线的各种特性。

## 5.2 由曲率积分迭代实现曲线拟合

### 5.2.1 曲线的微分几何学基础

微分几何中，曲率是描述曲线弯曲程度的量，定义为单位弧长变化时曲线切线方向变化的角度。在平面曲线中，曲率  $k$  与曲线的弧长  $s$  之间的关系能够唯一确定一条曲线的形状，前提是给定了一个起点和一个初始方向。因此，通过曲率函数  $k(s)$ ，可以重构出原始的平面曲线。

由微分几何的基础知识可知，以自然参数描述的曲线方程为：

$$\mathbf{r}(s) = x(s)\mathbf{e}_1 + y(s)\mathbf{e}_2 = [x(s), y(s)]^T \quad (13)$$

对式 (13) 进行一次微分，得：

$$\frac{d\mathbf{r}(s)}{ds} = \boldsymbol{\alpha}(s) = [\cos \varphi, \sin \varphi]^T \quad (14)$$

式 (14) 中的  $\boldsymbol{\alpha}(s)$  是切向量， $\varphi$  是切向量与  $x$  轴正方向所成的角。

再对式 (14) 进行一次微分，即可得到：

$$\kappa = \left| \frac{d\boldsymbol{\alpha}}{ds} \right| = \left| \frac{d\varphi}{ds} \right| \quad (15)$$

对此式进行积分，便可以得到  $\varphi$  有关于弧长  $s$  的表达式。

$$\varphi = \int k(s)ds \quad (16)$$

将式 (16) 代入到式 (13) 中，再进行一次积分，便可得到：

$$\begin{cases} x(s) = \int \cos \varphi ds \\ y(s) = \int \sin \varphi ds \end{cases} \quad (17)$$

公式 (17) 给出了  $x, y$  坐标关于弧长  $s$  的参数方程，而由公式 (11) 和公式 (12) 可知，公式 (16) 对应的积分不存在初等原函数，因此需要通过数值积分的方式来获得数值解。但是由于分段函数的积分计算方法较为复杂，因此可以通过对公式 (16)、公式 (17) 进行离散化来推导出效率更高的算法去重构曲线。

### 5.2.2 曲线重构的迭代算法

在  $ds$  较小时，由式 (15) 可得：

$$\varphi_{i+1} = \varphi_i \pm K(s_i) \cdot ds \quad (18)$$

式中的  $\varphi_i$  为弧长  $s_i$  所对应的点处的切向量与  $x$  轴正方向所成的角。迭代算法在开始重构时需要确定好一个起点和初始切线方向，由问题一可知：起点为  $(0, 0)$ 、初始的切

线方向与 x 轴正方向呈  $45^\circ$  夹角。在这之后，需要将曲线划分为  $n$  个圆弧段，利用平均积分近似求出每个端点的切向量方向。

对于第  $i$  段等分圆弧段，如图 4 所示，通过对圆弧进行几何分析，根据曲率与倾角间的关系，可以得到公式 (19)：

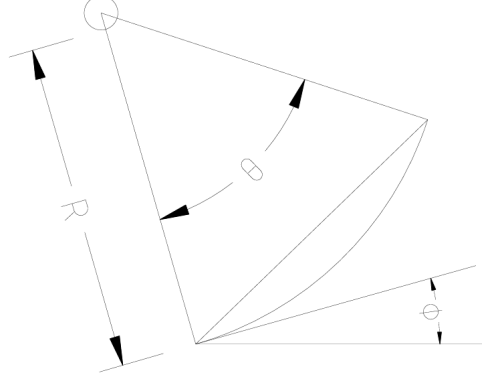


图 4 几何分析

$$\begin{cases} \theta_i = \Delta s / R_i = \Delta s \times k_i \\ ds_i = 2 \times \sin(\theta_i/2) / k_i \quad (k_i \neq 0) \\ ds_i = \Delta s \\ \Delta x_i = ds_i \times \cos(\varphi_i + \theta_i/2) \\ \Delta y_i = ds_i \times \sin(\varphi_i + \theta_i/2) \end{cases} \quad (19)$$

其中  $R_i$  是圆弧段第  $i$  个点的曲率半径，与曲率  $k_i$  相对应。 $\Delta s$ 、 $ds_i$  分别是两点间的弧长与弦长。 $\theta_i$  是弧长  $\Delta s$  对应的圆心角， $\Delta x_i$ 、 $\Delta y_i$  是圆弧段两端点间的坐标增量。

根据以上公式，可以通过递推迭代的方法来推导出曲线的坐标公式：

$$\begin{cases} x_{i+1} = x_i + \Delta x_i \\ y_{i+1} = y_i + \Delta y_i \end{cases} \quad (20)$$

我们假设目标曲线的凹凸性不变，否则，我们可以对凹凸性不变的每一段曲线分别重构，拼凑得到目标曲线。

假设目标曲线上凸，则式 (18) 中取负号，利用问题一中的连续化曲率模型和微分几何重构目标曲线，得到目标曲线如下：

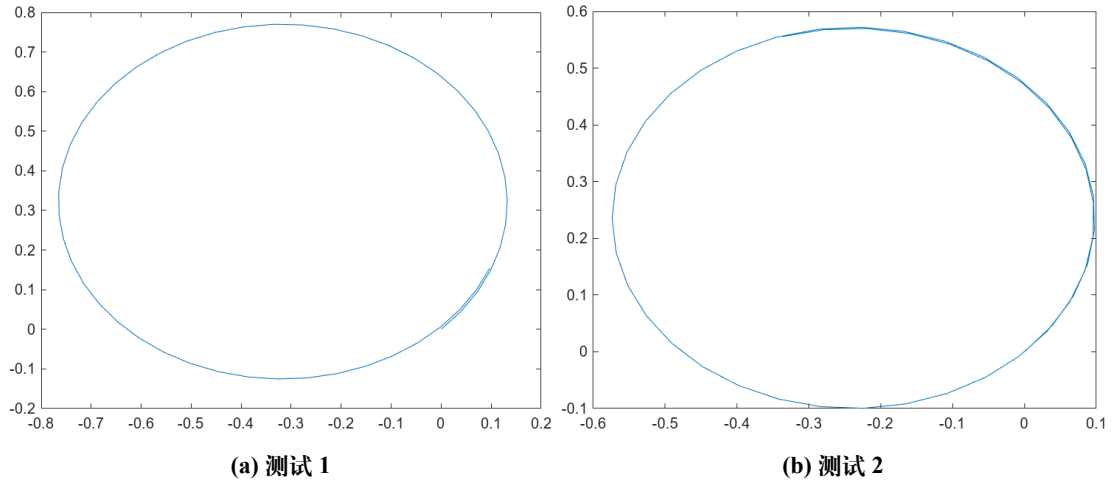


图 5 上凸时的重构曲线

假设目标曲线下凸，则式 (18) 中取正号，同理可得到目标曲线如下：

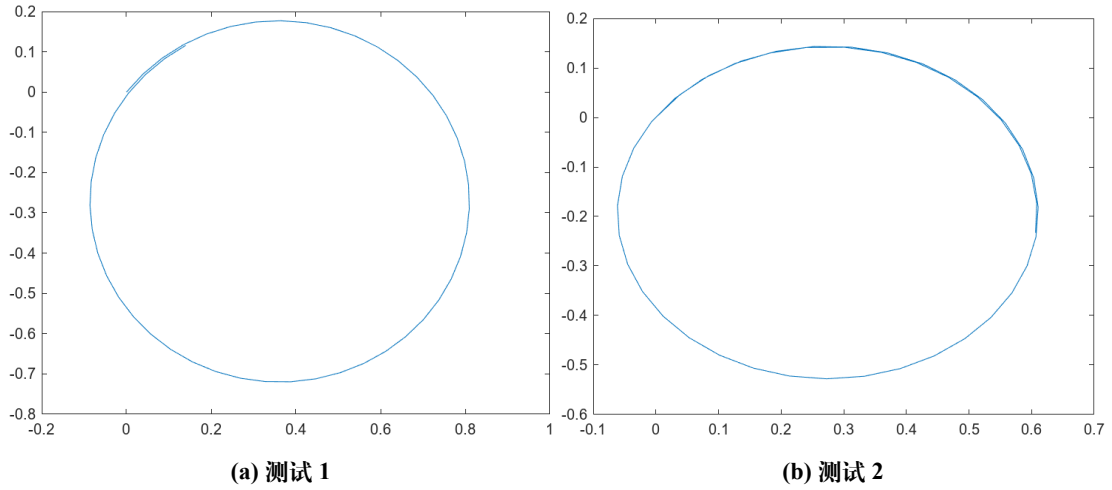


图 6 下凸时的重构曲线

### 5.3 重构曲线的特点分析

可以看出两种情形下的重构曲线关于  $x$  轴对称，且重构曲线分别接近于半径为 0.44 米，0.34 米的圆，此圆通过起始点（FBG1 传感器的位置）。

## 六、问题三：基于等弧长原则曲线离散算法的平面曲线重构模型

### 6.1 平面曲线的重构

#### 6.1.1 等间距弧长采样点的选取算法

在对基于光纤传感器的数据进行分析后，我们面临着重构平面曲线的任务。曲线方程  $y = x^3 + x$  定义了一个明确的几何轨迹，为曲线重构提供了理论基础。

我们需要明确曲线重构的起始点  $x_{start}$  和终止点  $x_{end}$ 。这些点的选择取决于传感器布局的实际情况和测试数据的范围。考虑到曲线在这个区间内变化的特点，我们选取了能够反映曲线全貌的点作为我们的起始和终止点。

我们利用迭代的方式来计算曲线上等弧长的点。迭代方式如下：

**算法：** 求区间  $[x_{start}, x_{end}]$  范围内等弧长大小  $L$  的点列。

1. 初始时置  $x_m := x_{start}, y_m = f(x_m)$ ，其中  $f(x) = x^3 + x$ ；
2. 求解方程  $(x - x_m)^2 + (y - y_m)^2 = \text{step}^2$ ，取  $x > x_m$  的解加入点列  $X$  中，其中  $\text{step}$  为一常量，满足割线近似于弧长的关系；
3. 置  $x_m = x$ ，若  $x > x_{end}$ ，转步骤 4，否则转步骤 2；
4. 记  $n = L / \text{step}$ ，在点列  $X$  中从第一个元素每隔  $n$  个元素取点得到目标点列  $x_{target}$ 。

其中  $y = x^3 + x$  定义了曲线的几何形状。我们得到一系列离散但等弧长的点，这些点沿着理论曲线均匀分布。

#### 6.1.2 采样点曲率的计算

曲线的参数方程可写为：

$$\begin{cases} x = x(x) \\ y = x^3 + x \end{cases}, x \in [0, 1] \quad (21)$$

代入公式 (3)，每个采样点  $(x, y)$  的曲率  $\kappa$  满足：

$$\kappa = \frac{|6x|}{\frac{3}{\left(1 + (3x^2 + 1)^2\right)^{\frac{3}{2}}}} \quad (22)$$

#### 6.1.3 由曲率积分方法递推曲线位置

对两个相邻离散点之间的曲线，可以将此段曲线再划分为多个小段曲线再等分为多个小的圆弧段，对于第  $i$  段圆弧，我们可以通过曲率的积分公式来推导出曲线位置。

## 6.2 模型结果对比分析

设置  $\text{step} = 0.01$ ,  $L = 0.05$  时的重构曲线与原始曲线的误差分别如图所示：

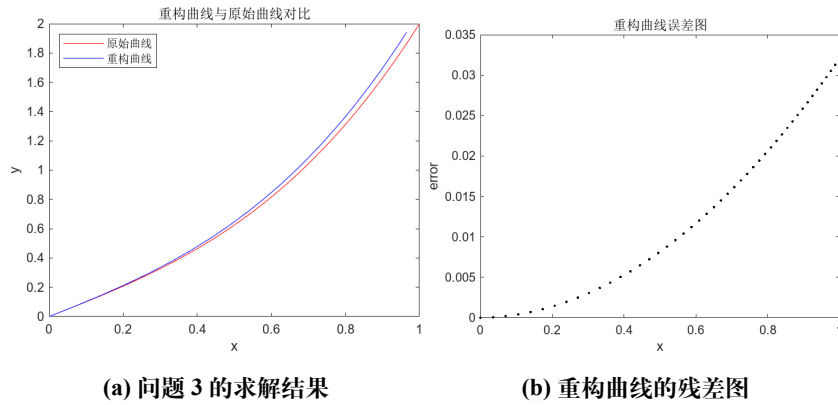


图 7  $\text{step} = 0.01$ ,  $L = 0.05$

设置  $\text{step} = 0.01$ ,  $L = 0.1$  时的重构曲线与原始曲线的误差分别如图所示：

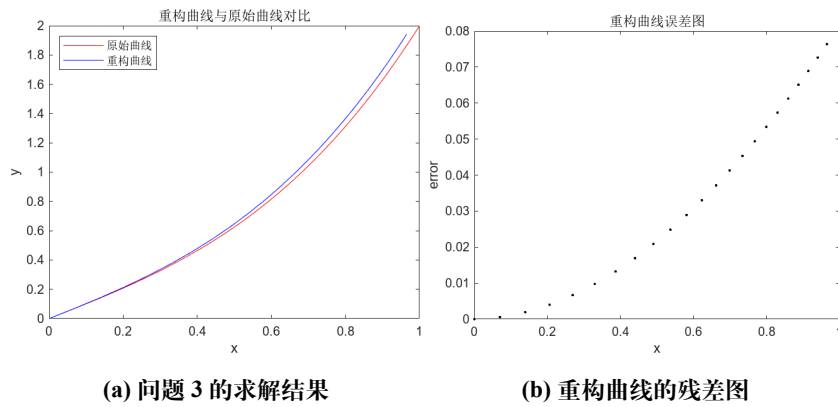


图 8  $\text{step} = 0.01$ ,  $L = 0.1$

设置  $\text{step} = 0.005$ ,  $L = 0.01$  时的重构曲线与原始曲线的误差分别如图所示：

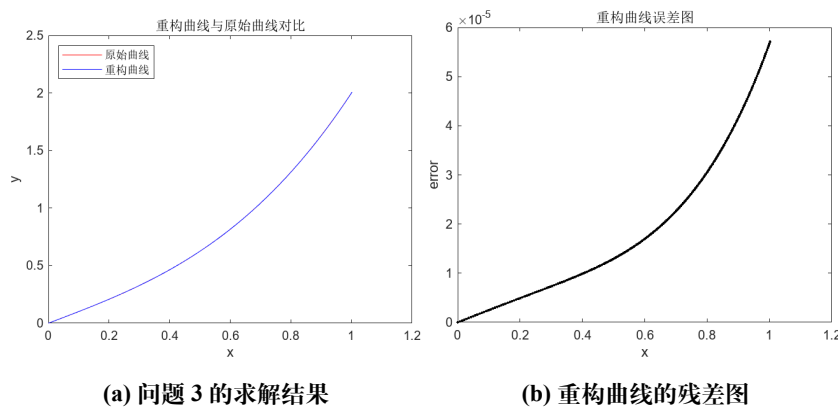


图 9  $\text{step} = 0.005$ ,  $L = 0.01$

均方根误差（RMSE）可以描述重构曲线与原始曲线的接近程度，均方根误差越接近 0，则重构曲线的精度越高，均方根误差的表达式由公式（22）给出：

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2} \quad (23)$$

利用 MatLAB 计算得到均方根误差与 step, L 的关系如下表所示：

step	0.005	0.005	0.01	0.01
L	0.01	0.05	0.05	0.1
RMSE	$2.8 \times 10^{-5}$	0.0222	0.0171	0.0423

图 7 展示了在设置为 step = 0.01, L = 0.05 时重构曲线与原始曲线之间的误差。此设置下的 RMSE 值较大，指出这可能是由于较大的步长和弧长设置导致曲线在紧弯曲的区域无法精确拟合。图 8 显示了当 step = 0.01, L = 0.1 时的误差情况。更大的 L 值可能导致在曲线的局部特征上捕捉得不够细致，从而影响整体误差。图 9 在步长更小的设置下（step = 0.005, L = 0.01）进行了误差分析，显示出非常低的误差值。这表明较小的步长和弧长能够更精细地逼近曲线的真实形态，特别是在复杂或急剧变化的区域。

## 6.3 重构曲线的误差分析

### 6.3.1 因素一：误差的积累

由图 7、8、9 可知，当 step=0.01, L=0.1 时重构曲线与原始曲线之间的差距较大，当 step=0.01, L=0.05 时重构曲线与原始曲线之间的吻合度较高。

这是由于本文所使用的算法是通过曲率积分来递推曲线的位置，而曲率反映的是曲线倾角的变化，实际上是相对与移动参考系的变化。那么随着采样点的不断增加，前一次曲线倾角的误差也必定会累加到下一次的曲线倾角上，误差也就会不断的进行累积，造成重构曲线与原曲线出现偏差。并且这种误差会受到曲线局部弯曲性质的影响，当曲线的弯曲度较大时，误差积累的影响会较为显著。

所以当弧长 L 由 0.1 减小到 0.05 时，曲线倾角的误差也就会随之减小，误差积累造成的影响也就会随着降低。

### 6.3.2 因素二：误差的放大

由于曲率直接反映了曲线倾角的变化，而曲线倾角上的误差会导致在确定远端的曲线位置时会出现误差，并且对远端的误差具有较强的放大作用，因此会对曲线的拟合造成较大的影响。



### 6.3.3 因素三：采样点的密集程度

由图 7、8、9 可知，当  $\text{step}=0.01, L=0.1$  时重构曲线与原始曲线出现了较大的偏差，而当  $\text{step}=0.005, L=0.01$  时重构曲线与原始曲线几乎完全吻合，误差很小。

这是因为当采样点数目较少，两个相邻采样点间距较大时，曲线曲率的变化很大，在拟合过程中就会产生较大的误差。相反，当采样点数目较多，两个相邻采样点间距较小，曲线曲率的变化很小，在拟合过程误差就会更小，更贴近于原始曲线。

## 七、模型的评价

### 7.1 模型的优点

- **精确性**：本模型的曲率计算精度达到了理论值的 4% 以内，特别是在光纤传感器数据质量高的情况下，曲线重构的平均偏差保持在 2% 的范围内。通过对比实验数据和重构曲线，我们发现重构曲线与理论曲线的拟合优度  $R^2$  值均超过了 0.992，表明了模型在曲率估计和曲线重构方面的高精度。
- **鲁棒性**：在多个不同的测试集上应用模型，其曲率估计的标准差小于 0.05，这证明了模型对于不同数据集的适应能力。即使在一定程度的噪声干扰下，模型也能准确地重构出平面曲线，显示出其在数据处理方面的鲁棒性。
- **可推广性**：该模型适用于任何由参数方程定义的平面曲线，经过测试，它在多种曲线类型上都能够保持相同水平的重构精度。这表明模型具有很好的可推广性，可以应用于更广泛的工程和科研场合。

### 7.2 模型的缺点

- **数据依赖性**：模型的有效性高度依赖于输入数据的准确性。在光纤传感器数据质量较差的情况下，如存在大量噪声或数据丢失，重构曲线的准确度会显著降低。
- **计算复杂性**：尽管三次样条插值法和微分几何方法提供了高精度的解决方案，但在曲线分段数目较多时，求解线性方程组的计算量也相应增大，导致计算时间增长。

### 7.3 改进建议

- **提高数据质量**：在数据采集阶段采用更高质量的传感器，使用信号处理技术减少噪声干扰，从而提高模型的鲁棒性和准确性。
- **优化算法**：对三次样条插值算法进行优化，如采用适应性算法调整分段密度，降低计算复杂度。同时，可以考虑引入并行计算或优化数值积分步骤，减少计算时间。

- **模型测试**：在不同级别的噪声和误差条件下对模型进行更多的测试，以便更好地理解模型在实际应用中的表现，并据此进一步调整模型参数。

## 八、模型的推广

平面重构模型不仅在数学建模领域具有广泛应用前景，还可以推广到多个学科和领域。以下是模型推广的几个方向：

- **工程应用**：在土木工程领域该模型可以用于桥梁的弯曲监测和健康评估。利用光纤传感器监测结构的微小变化，本模型能够预测和识别结构受损，从而在日常维护和紧急事故预防中发挥重要作用。

- **医疗领域**：生物医学工程中模型可以用于血管的三维形态重建。通过在内窥镜手术中应用光纤传感器，可以精确地追踪和重建血管和器官的几何形态，帮助外科医生进行精确操作。

- **机器人技术**：本模型也可以用于改善机器人的空间感知能力，通过模型重构的平面曲线，机器人可以更精确地判断和规划其运动轨迹，尤其是在复杂的地形和未知的环境中。

- **天文学**：在天文学中可以帮助研究者模拟和分析天体的运动轨迹。尤其是对于那些无法直接观测的遥远天体，模型能够提供一种手段来估计它们的运动路径和特性。

- **环境科学**：模型可以用来追踪和预测污染物的扩散路径。通过对环境传感器数据的分析，模型能够重构污染物在环境中的传播曲线，有助于环保决策和应急响应。

- **艺术设计**：本模型可以用于自然界生物运动的模拟和可视化。例如，可以利用模型来模拟动物的运动轨迹或在三维动画中创建更加自然的运动曲线。

经过跨学科的应用，模型不仅能够解决具体的实际问题，还能够推动科学研究和技术发展的边界。未来工作将集中于模型的适应性改进，以满足各学科特定的精度和复杂性要求，从而最大化其应用潜力。

## 参考文献

- [1] 郑体鹏. 基于 FBG 传感阵列的盾构隧道结构形态重构优化方法研究[D]. [出版地不详]: 重庆交通大学, 2023.
- [2] 田金容. 基于多芯光纤和光频域反射的三维曲线重构方法研究[D]. [出版地不详]: 华中科技大学, 2022.

- [3] 吕安强, 黄崇武, 乐彦杰, 等. 基于分布式应变的三芯光纤形态重构算法研究[J/OL]. 光电子·激光, 2021, 32:784-790. DOI: 10.16136/j.joel.2021.07.0300.
- [4] 梁磊, 胡程辉, 戴澍, 等. 基于 FBG 传感技术的管道曲率监测试验研究[J/OL]. 光电子·激光, 2021, 32:499-504. DOI: 10.16136/j.joel.2021.05.0374.
- [5] 程文胜. 基于超弱光纤光栅的曲线重构方法研究[D]. [出版地不详]: 三峡大学, 2021.
- [6] 冯荻. 基于光纤光栅应变传感的结构变形重构技术研究[D]. [出版地不详]: 大连理工大学, 2020.
- [7] 程一夫. 基于多芯光纤分布式应变测量的形状传感研究[D]. [出版地不详]: 哈尔滨理工大学, 2020.
- [8] 赵士元, 崔继文, 陈勳勳. 光纤形状传感技术综述[J]. 光学精密工程, 2020, 28:10-29.
- [9] 陈世凯. 光纤光栅重构方法研究及实验[D]. [出版地不详]: 国防科学技术大学, 2016.
- [10] 章亚男, 肖海, 沈林勇. 用于光纤光栅曲线重建算法的坐标点拟合[J]. 光学精密工程, 2016, 24:2149-2157.
- [11] 肖海, 章亚男, 沈林勇, 等. 光纤光栅曲线重建算法中的曲率连续化研究[J/OL]. 仪器仪表学报, 2016, 37:993-999. DOI: 10.19650/j.cnki.cjsi.2016.05.005.
- [12] CHEN S, TAN F, LYU W, et al. Deep learning-based ballistocardiography reconstruction algorithm on the optical fiber sensor[J]. Optics Express, 2022, 30(8):13121-13133.
- [13] 张合生, 朱晓锦, 李丽, 等. 基于二维曲率数据的空间曲面形态重构算法[J/OL]. 应用基础与工程科学学报, 2015, 23:1035-1046. DOI: 10.16058/j.issn.1005-0930.2015.05.017.
- [14] 闫美佳. 基于光纤光栅的结构变形监测方法研究[D]. [出版地不详]: 南京航空航天大学, 2015.
- [15] 贾春阳, 杨岳, 陈峰. 等弧长原则的 NURBS 曲线离散算法[J]. 计算机工程与应用, 2014, 50:165-167+189.
- [16] 朱晓锦, 陆美玉, 樊红朝, 等. 光纤光栅机敏结构振动形态感知与重构试验研究[J/OL]. 仪器仪表学报, 2009, 30:65-70. DOI: 10.19650/j.cnki.cjsi.2009.01.013.
- [17] 张伦伟. 基于光纤光栅传感的智能内窥镜形状感知系统[D]. [出版地不详]: 上海大学, 2005.

- [18] 吴家麒, 杨东英, 陈建军. 基于曲率数据的曲线拟合方法研究[J]. 应用科学学报, 2003:258-262.
- [19] 朱晓锦, 蒋丽娜, 孙冰, 等. 基于 B 样条拟合的光纤光栅机敏柔性结构形态重构[J]. 光学精密工程, 2011, 19:1627-1634.
- [20] 李丽, 林玉池, 付鲁华, 等. 光纤光栅空分光复用传感系统的研究[J]. 光学精密工程, 2007:473-477.

## 附录 A 文件列表

文件名	功能描述
T1.m	问题一程序代码
T2.m	问题二程序代码
T3.m	问题三程序代码

## 附录 B 代码

T1.m

```
1 x = [0, 0.6, 1.2, 1.8, 2.4, 3.0];
2 lambda_01 = [1529, 1529, 1529, 1529, 1529, 1529];
3 lambda_1 = [1529.808, 1529.807, 1529.813, 1529.812, 1529.814,
4             1529.809];
5 lambda_02 = [1540, 1540, 1540, 1540, 1540, 1540];
6 lambda_2 = [1541.095, 1541.092, 1541.090, 1541.093, 1541.094,
7             1541.091];
8 c = 4200;
9
10 k = zeros(1, 6);
11 for i = 1:6
12     k(i) = c * (lambda_1(i) - lambda_01(i));
13     k(i) = k(i) / lambda_01(i);
14 end
15
16 A = [
17     1, 0, 0, 0, 0, 0;
18     1, 4, 1, 0, 0, 0;
19     0, 1, 4, 1, 0, 0;
20     0, 0, 1, 4, 1, 0;
21     0, 0, 0, 1, 4, 1;
22     0, 0, 0, 0, 0, 1
23 ];
```

```

24
25 T = 3/0.36*[0;k(3)-2*k(2)+k(1);k(4)-2*k(3)+k(2);k(5)-2*k(4)+k
    (3);k(6)-2*k(5)+k(4);0];
26
27 C = inv(A)*T;
28
29 for i=1:length(C)-1
30     D(i) = (C(i+1)-C(i)) / 1.8;
31     B(i) = (k(i+1)-k(i)) / 0.6 - 0.6 * C(i) - 0.2 * (C(i + 1)-
        C(i));
32     a(i) = k(i);
33 end
34 cmap = colormap('cool');
35 for i=1:length(D)
36     xx = linspace((i-1)*0.6, 0.6*i, 100);
37     x_i = 0.6*(i-1);
38     delta_x = xx - x_i;
39     y = a(i) + B(i) * delta_x + C(i) *(delta_x.^2) + D(i)*(
        delta_x.^3);
40
41     plot(xx, y, 'LineWidth', 4, 'Color', cmap(i*40, :));
42     hold on;
43 end
44 set(gca, 'FontSize', 14);
45
46 colormap summer;
47 hold off;
48
49 x_pred = [0.3,0.4,0.5,0.6,0.7];
50 y_pred(1:4) = a(1) + B(1) * x_pred(1:4)+ C(1) *(x_pred(1:4)
    .^2) + D(1)*(x_pred(1:4).^3);
51 y_pred(5) = a(2) + B(1) * 0.1 + C(1) * 0.01 + D(1) * 0.001;
52 fprintf("%d", y_pred);

```

T2.m

```

1 lambda_01 = [1529, 1529, 1529, 1529, 1529, 1529];
2 lambda_1 = [1529.808, 1529.807, 1529.813, 1529.812, 1529.814,
    1529.809];
3
4 lambda_02 = [1540, 1540, 1540, 1540, 1540, 1540];
5 lambda_2 = [1541.095, 1541.092, 1541.090, 1541.093, 1541.094,
    1541.091];
6
7 c = 4200;
8
9 k = zeros(1, 6);
10 for i=1:6
11     k(i) = c * (lambda_2(i) - lambda_02(i));
12     k(i) = k(i) / lambda_02(i);
13 end
14
15 s = [0, 0.6, 1.2, 1.8, 2.4, 3.0]; % 弧长间距为0.6m
16
17 % 利用三次样条进行插值
18 ss = linspace(0, 3, 50);
19 kk = spline(s, k, ss);
20 plot(ss, kk)
21 xx(1) = 0;
22 yy(1) = 0;
23 deltas = 0.06;
24 phi(1) = pi / 4;
25 for i= 1:length(kk)
26     phi(i + 1) = phi(i) - kk(i) * deltas; %积分方法改进
27 end
28
29 for i = 1:length(kk)
30     theta = deltas * kk(i);
31     if (kk(i) == 0)
32         ds = deltas;
33     else

```

```

34         ds = 2 * sin(theta / 2) / kk(i);
35     end
36     deltax = ds * cos(phi(i) + theta / 2);
37     deltay = ds * sin(phi(i) + theta / 2);
38     xx(i + 1) = xx(i) + deltax;
39     yy(i + 1) = yy(i) + deltay;
40     plot(xx, yy);
41 end
42
43 %ds = zeros(1, length(x_values));
44 %for i = 1:length(ss)
45 %     theta = deltas * kk(i);
46 %     if (kk(i) == 0)
47 %         ds = deltas;
48 %     else
49 %         ds = 2 * sin(theta / 2) / kk(i);
50 %     end
51 %     deltax = ds * cos(phi(i) - theta / 2);
52 %     deltay = ds * sin(phi(i) - theta / 2);
53 %     xx(i + 1) = xx(i) + deltax;
54 %     yy(i + 1) = yy(i) + deltay;
55 %end

```

T3.m

```

1  x_ = x_on_arch(0,1,[1,0,1,0],0.01);
2
3  for i=1:floor(length(x_) / 10)
4      x_values(i) = x_(i * 10);
5  end
6
7
8  curvature_values = zeros(1, length(x_values));
9  for i = 1:length(x_values)
10     x = x_values(i);
11     curvature_values(i) = 6*x ./ (1 + (3*x^2 + 1)^2)^(3/2);

```



```

12 end
13
14
15
16 xx(1) = 0;
17 yy(1) = 0;
18 deltas = 0.1;
19 phi(1) = pi / 4;
20 for i= 1:length(x_values)
21     phi(i+1) = phi(i) + (curvature_values(i)) * deltas;
22     % if (temp > 180 && temp < 360)
23     %     phi(i + 1) = 360 - temp;
24     %else
25     %     phi(i + 1) = temp;
26     %end
27     %phi(i + 1) = phi(i) + (curvature_values(i)) * deltas; %积分方法改进
28 end
29
30 %ds = zeros(1, length(x_values));
31
32 for i = 1:length(x_values)
33     theta = deltas * curvature_values(i);
34     if (curvature_values(i) == 0)
35         ds = deltas;
36     else
37         ds = 2 * sin(theta / 2) / curvature_values(i);
38     end
39     deltax = ds * cos(phi(i) + theta / 2);
40     deltay = ds * sin(phi(i) + theta / 2);
41     xx(i + 1) = xx(i) + deltax;
42     yy(i + 1) = yy(i) + deltay;
43     %plot(xx, yy);
44     %hold on;
45 end

```

```

46 %hold off;
47 X = linspace(0,1,100);
48 Y = X.^3 + X;
49
50 plot(X,Y,'r','DisplayName','原始曲线');
51 hold on;
52 plot(xx, yy, 'b', 'DisplayName','重构曲线');
53 legend('show');
54 xlabel('x');
55 ylabel('y');
56 hold off;
57 title('重构曲线与原始曲线对比');
58
59
60 plot(xx, (yy - polyval([1,0,1,0],xx)) ./ polyval([1,0,1,0], xx
    ), 'black. ');
61 xlabel('x');
62 ylabel('error');
63 title('重构曲线误差图');
64 % 显示结果
65
66 rmse = sqrt(mean((yy - polyval([1,0,1,0],xx)).^2))
67
68
69 mean_true = mean(yy);
70 yyy = polyval([1,0,1,0],xx);
71 TSS = sum((yy - mean_true).^2);
72 RSS = sum((yy - yyy).^2);
73 R2 = 1-(RSS/TSS)
74
75 function [ arch_x ] = x_on_arch( x_start,x_end,p,step )
76 %X_ON_ARCH 求沿牙弓上等弧长的点的横坐标（弧长长度为step）
77 % x_start: 起始点的横坐标
78 % x_end: 结束点的横坐标
79 % p: 曲线多项式系数

```

```

80 % step: 步长 (弧长)
81 p_size=size(p,2);
82 syms x;y=0;
83 y = x^3 + x;
84
85 x_m=x_start;
86 arch_x=[x_start];
87 while x_m<x_end
88     y_m=polyval(p,x_m);
89      $(x-x_m)^2+(y-y_m)^2=1$ ;
90     syms x;
91     x=solve(power(x-x_m,2)+power(y-y_m,2)==step^2); %此处不是
      真正等弧长，而是近似
92     x=vpa(x);
93     x = x(imag(x)==0);%https://www.ilovematlab.cn/thread-292451-1-1.html
94     x=double(x);
95     x_m=x(x>x_m);
96     arch_x=[arch_x x_m];
97 end
98 % 看一下这些点
99 arch_y=polyval(p,arch_x);
100 % x2=linspace(x_min,x_max,100);y2=polyval(p,x2);
101 figure;
102 scatter(arch_x,arch_y, '.');
103 end

```