基于非稳态热传导模型的炉内元件温度拟合与分析

Septempber 2020

摘要

本文解决了题目 A ——炉温曲线的问题一、问题二与问题三，并对问题四做了分析。基于傅里叶定律、牛顿冷却定律与能量守恒定律，建立非稳态热传导模型。本文考虑了温区之间的热导影响，解出了热传导偏微分方程的数值解，并较好地拟合了真实实验数据。在新的预设温度下，以固定步长遍历传送带的速度区间，在制程界限的限制下逐个检验每一个解，进行至多三次迭代，得到传送带运行速度的最大可行解。再将预设温度与传送带速度设为新的参数，利用微分法计算出目标函数的数值解，在约束条件下尝试参数的可能解，从而找到参数的最优解。

关键词

非稳态热传导模型，牛顿冷却定律，傅里叶定律，最小二乘法拟合，目标函数优化，迭代优化算法

1 问题重述

将安装电子元件的印刷电路板放置在回焊炉中，通过加热可与将电子元件自动焊接到电子板上。

焊炉内有 11 个小温区及炉前区域和炉后区域。小温区长 30.5 cm，相邻有 5 cm 间隙，炉前区域、炉后区域长 25 cm。回焊炉启动后，炉内空气温度视为瞬间达到稳定。炉前区域、炉后区域和小温区间隙温度与相邻温区的温度有关。生产车间温度保持在 25ºC。设定温度分别为：小温区 1-5，175ºC；小温区 6，195ºC；小温区 7，235ºC；小温区 8-9，255ºC；小温区 10-11，25ºC。传送带过炉速度为 70 cm/min；焊接区域厚度为 0.15 mm。

称焊接区域中心温度曲线为炉温曲线。焊接区域中心温度达到 30ºC 时开始，电路板进入回焊炉开始计时。

可调节各温区的温度和传送带的过炉速度。在原始设定温度基础上，各小温区温度可进行 ±10ºC 范围内的调整。调整时要求小温区 1-5、小温区 8-9、小温区 10-11 中的温度保持 25ºC。传送带的过炉速度调节范围为 65-100 cm/min。此外，炉温曲线应满足制程界限。

需解答下列问题：

（1）对焊接区域的温度变化规律建立数学模型。假设传送带过炉速度为 78 cm/min，各温区温度的设定值分别为：小温区 1-5，173ºC；小温区 6，198ºC；小温区 7，230ºC；小温区 8-9，257ºC。给出焊接区域中心的温度变化情况，列出小温区 3、6、7 中点及小温区 8 结束处焊接区域中心的温度，画出相应的炉温曲线，并记录每隔 0.5 s 焊接区域中心的温度。

（2）假设各温区温度的设定值分别为：小温区 1-5：182ºC、小温区 6：203ºC、小温区 7：237ºC、小温区 8-9：254ºC，确定允许最大传送带过炉速度。

（3）确定在“使超过 217ºC 到峰值温度所覆盖的面积最小”要求下的最优炉温曲线，以及各温区的设定温度和传送带的过炉速度，并给出相应面积。

（4）结合问题 3，进一步给出“以峰值温度为中心线的两侧超过 217ºC 的炉温曲线应尽量对称”最优炉温曲线，以及各温区设定的温度及传送带过炉速度，并给出相应指标值。

2 模型准备

2.1 背景知识

2.1.1 传热方式

热量在传递过程中主要有以下三种形式：

1. 热传导 热传导指能量在物体内部由高温区到低温区的传递或在高温物体到与之接触的低温物体的传递。其限制发生传热的物质间没有宏观位移。由傅立叶定律描述该现象。

2. 热对流 热对流指能量在流体的各部分间发生相对位移、冷热流体互相掺杂情况下的传递。其限制只发生在流体中。由牛顿冷却定律描述该现象。

3. 热辐射 热辐射指温度差产生的电磁波空间传递。由斯蒂芬-玻尔兹曼定理描述该物理现象。

2.1.2 傅里叶定律

傅里叶定律以微分方程的形式描述了热量在介质之间传导的规律。公式如下：

$$q = -\lambda \frac{\partial T}{\partial x}$$

其中，q 为热流密度，λ 为导热系数，温度 T 是关于位置 x 的函数。我们使用其微分形式，因为更加关注其在局部的能量传导率，即元件进入加热器的前接触面和后接触面的热量传递过程。

2.1.3 牛顿冷却定律

牛顿冷却定律描述了高于周围环境温度的物体在冷却过程中向周围媒质传递热量的规律。公式如下：

$$\triangle t = \left| t_w - t_f \right|$$
$$q = h \triangle t$$
$$\Phi = qA = Ah\triangle t = \tfrac{\triangle t}{1/hA}$$

其中 q 为热流密度，h 为对流传热系数，和物体本身的性质和物体厚度有关。ΔT 为物体和环境温度差。Φ 为传热功率，即单位时间内的传热量，A 为接触面积。通过观察可以发现物体和周围环境的热交换速率和温度密切相关，且在累计时间上呈指数递减的趋势。

2.1.4 微分法

在一个二维坐标平面上，求解一个固定区间内两曲线所夹的面积时，可以考虑使用微分法进行计算。以 x 轴区间为例，具体做法为：将原始区间划分为若干极小区间。对于每个小区间，用常规多边形计算方式近似求出该区间的覆盖面积。一种计算方法为用矩形近似表示区间面积：求出小区间左右两侧边界的长度（即两曲线 y 轴坐标绝对值），用两个长度的平均值表示矩形的长，用区间的长度（即边界 x 轴坐标绝对值）表示矩形的宽，用矩形面积计算公式获得该矩形面积，并用其近似表示该小区间覆盖的面积（如图 1）。随着划分的小区间个数增加，算法的精度会不断增长，计算的结果会不断逼近真实情况。当小区间个数足够多时（即极小区间的区间宽足够小），误差是极小的。

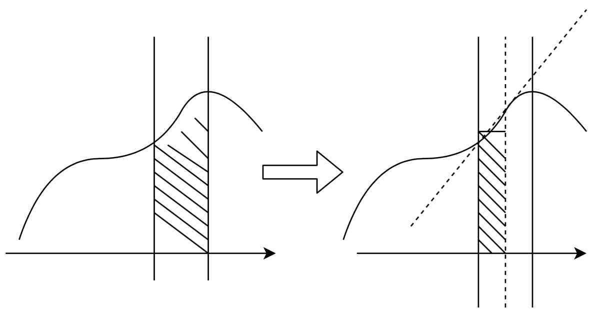

图 1: 微分法

2.2 模型抽象与假设

由于我们仅需要考虑物体在进入炉子之后的在运动方向上的运动变化，不需要考虑在垂直与运动方向上的热量扩散（物体的体积大小也允许我们忽略这部分的热量传导差异），同时由于元件高度较小，在竖直方向上也可以认为热量是均匀传导。因此，我们无需建立三维方向上的所有热量传导细节，只需考虑一维热传导的模型抽象。

除此之外，针对由于题目中提到各熔炉在启动后温度会快速达到稳定，所以也需要考虑熔炉之间的热量传递：

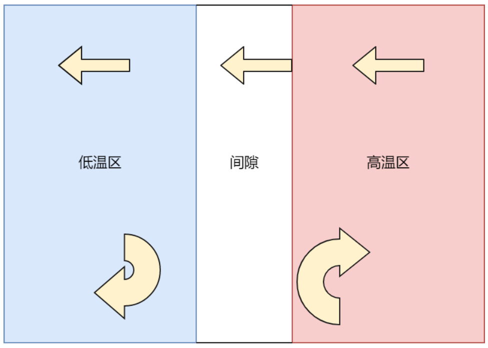

图 2: 温区之间传热的简易示意图

综上所述，我们排除建模过程中无需考虑的元素，将机理模型做适当简化，做出如下假设:

- 假设 1: 间隙温度为相邻小温区之间的平均温度;
- 假设 2: 不考虑接触面之间的接触热阻，并且认为接触面界面光滑连续;
- 假设 3: 简化问题为一维传热问题，不考虑其他不均匀热源和传热过程;

在物体运动方向上的一维简易模型示意图 3 所示。为了便于表示，这里展示的是二维的竖直截面的情况。此时，对水平方向上元件的空间移动，建立坐标系 $s - T_{fur}$，对元件内部，建立方向相反的坐标系（便于后续分析）$\alpha - T$。焊接中心厚度为 0.15mm，相对于估计的元件长度 $\alpha = 3cm$ 很小，可以认为元件的温度即是电路板中点 $\alpha/2$ 位置的温度，而忽略元件内部的热量传递情况。

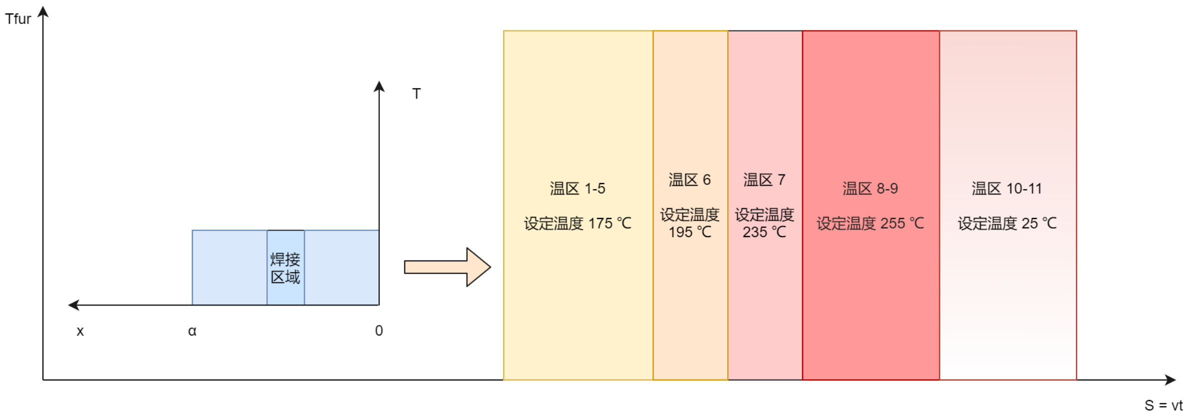

图 3: 焊接区域热传导的简易示意图

2.3 符号定义

本文构件的模型中出现的关键符号定义如表 1 所示。

符号	含义	值	单位
λ	电路板的导热系数	待拟合	$W/(m\cdot^\circ C)$
h_1, h_2	元件前、后侧与炉内空气间对流换热系数	待拟合	$W/(m^2\cdot^\circ C)$
θ	炉间空气之间的换热系数	待拟合	$W/(m^2\cdot^\circ C)$
$\mathcal{H}$	热量控制方程	-	-
t	从炉前区域开始的运动时间	-	s
v	传送速度	65 - 100	cm/min
s	相对起始位置的运动距离	vt	s
x	元件相对出发位置的横向坐标	-	cm
α	元件长度	3（估计）	cm
T_{env}	环境温度	25	$^\circ C$
$T(t,x)$	电子元件的温度控制方程	-	$^\circ C$
T_{fur_j}	第 j 个熔炉内部的空气温度	-	$^\circ C$
T_{gap_j}	第 j 个熔炉间隙的空气温度	-	$^\circ C$
L_{fur}	熔炉长度	30.5	cm
L_{gap}	熔炉间隙	5	cm
L_{edge}	炉前与炉后区域长度	25	cm
ρ	电子元件密度	7.469（估计）	g/cm^3
c	电子元件比热容	480（估计）	$J/g\cdot^\circ C$
$\mathcal{F}(T_{fur_j}, v)$	目标优化函数	-	-
$\mathcal{A}$	阴影区域面积	-	-

表 1: 符号定义

3 问题一

3.1 问题分析

3.1.1 模型构建-非稳态传热模型

如前文所述，本模型中主要存在的传热过程为热空气和电子元件间的热对流。传热的过程可以分为稳态传热和非稳态传热。其中，稳态传热不受时间的影响，控制方程中不引入时间；而稳态传热受时间的影响，控制方程中需要引入时间。在本模型中，由于焊接体的运动涉及到时间流动，在控制方程中需要引入时间参数，所以我们建立非稳态传热模型。建立控制方程。焊接面在进入不同的炉中时，会和炉内的高温气体相接触，形成对流传热。

传热控制方程 考虑到电子元件内部的温度随着时间的变化而变化，所以在本问题中，时间对温度的影响无法消去，应当建立一个非稳态的热传导模型。

再考虑电子元件前进时，热空气以垂直于电子原件表面的方向射入，与物体表面发生一个热对流，产生了热量交换。在电子元件的运动过程中，其他方向的热空气影响虽然存在但并不显著，所以暂时假设其影响几乎为零。因此我们可以把此问题抽象成一个一维的非稳态热传导模型。

由能量守恒定律，系统总能量的改变等于传入或者传出该系统的能量。在本模型中，考虑对电子元件这个小型系统进行分析，其热力学能的变化等于流入或者流出该系统的热流量的差值，因此得到一个偏微分控制方程：

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}(\lambda \frac{\partial T}{\partial x})$$

其中，等式左边表示电子元件热力学能的变化，等式右边表示电子元件流入或者流出的热流量的差值。对于具体的参数来讲，ρ 表示电子原件内部体积微元的密度，c 表示电子元件内部各个位置的比热容，λ 表示电子元件材料的导热系数，进一步整理方程得到温度控制方程 $\mathcal{H}$：

$$\frac{\partial T}{\partial t} = a^2(\frac{\partial^2 T}{\partial x^2}),\ a^2 = \frac{\lambda}{\rho c}$$

边界条件以及初始条件 根据牛顿冷却定律，热量交换过程中热流量恒定，可以得到控制方程的边界条件：

$$\begin{cases} -\lambda \frac{\partial T}{\partial x}\Big|_{x=0} = h \cdot (Tf(t) - T(0,\ t)) \\ -\lambda \frac{\partial T}{\partial x}\Big|_{x=\alpha} = h \cdot (T(L,\ t) - Tf(t - \frac{\alpha}{v})) \end{cases}$$

其中，h 表示热对流系数，λ 表示热导系数，α 表示电子元件的长度，$Tf(t)$ 表示炉内温度，$T(0,\ t)$ 表示电子元件的头部位置，温度随时间变化的函数关系，$T(\alpha,\ t)$ 表示电子元件的尾部位置，温度随时间变化的函数关系。

根据已知的车间温度，用 T_{env} 表示，于是控制方程的初始条件可以写成

$$T(x,\ 0) = T_{env} = 25°C$$

于是总的初值条件与边界条件为：

$$\begin{cases} -\lambda \frac{\partial T}{\partial x}\Big|_{x=0} = h \cdot (Tf(t) - T(0,\ t)) \\ -\lambda \frac{\partial T}{\partial x}\Big|_{x=\alpha} = h \cdot (T(L,\ t) - Tf(t - \frac{\alpha}{v})) \\ T(x,\ 0) = T_{env} = 25°C \end{cases}$$

参数估计 在本问题的非稳态传热模型中，电子元件两端的换热系数 $h_1,\ h_2$ 是未知量，同时电子元件内部的导热系数 λ 同样未知。题目中给定了特定温度下。利用已知的焊接区域温度随时间的变化情况，可以使用最小二乘法拟合出估计值 $\hat{h_1},\ \hat{h_2}$。可以使用 MATLAB 中的 `lsqcurvefit` 来进行实现。

$$(\hat{h_1},\ \hat{h_2}) = arg\ min \sum_{i=1}^{N} [T(L,\ t_i;\ h_1,\ h_2) - T^*(t_i)]^2$$

对于其他参数，如元件的密度 ρ、元件的比热容 c。由于题目中没有给出元件的材料信息，我们使用铁金属的数据作为代入。这里不可避免会和实际情况出现误差，但由于在偏微分方程的控制方程中，变量 $a = \frac{\lambda}{\rho c}$ 含有将会拟合的未知量 λ，因此误差将会被拟合最小化，尽可能满足附件 1 中提供的实验数据，从拟合到的数据中填补模型的未知参数。

温区温度的变化 电子元件从低温区域进入高温区域，势必会造成温区的温度变化。温区的温度虽然随着时间的推移在基础温度上轻微地波动，但是当电子元件进入时，此温度会发生变化。在本模型中，环境温度认为恒定为 25 摄氏度，不受元件波动和炉的热辐射的影响。但在炉前具有的长为 25 cm 的区域部分，温度仍然会受到第一个炉的高温的影响，导致温度升高。同时，降温区和炉后区域的空缺也由于和相邻温区的温差过大，容易产生较强的热对流。根据牛顿冷却定律 $q = h\triangle T$，我们可以得知每个炉的温度都和其相邻的炉内的空气温度有关。接下来考虑该影响。

为了简化计算，我们首先假设在炉与炉之间的空隙的 5cm 部分的温度为相邻两个炉的设定温度的算术平均值。

$$T_{gapj}(t) = \frac{1}{2}(T_{furj-1} + T_{furj})$$

其次，令 $T_{furj}(t)$ 表示在 t 时间第 j 个炉的温度。从能量守恒定律我们知道每个炉从相邻炉获得的热流量将会转换成自己的热能，以温度的形式表现。同时，由于炉一直处在有外部能源加热状态，我我们可以认为所有炉的总能量在增加，但仍然存在流动。我们将牛顿冷却定律中的系数直接和温度正相关（这是因为在热流量和温度的转换中，温度和热量同样是线性的关系），将相关系数整合为炉间空气的换热系数 θ，且有以下常微分方程:

$$\frac{\mathrm{d}T_{furj}(t)}{\mathrm{d}t} = T_{furj}(t) + \theta(T_{furj}(t) - T_{furj-1}(t)) + \theta(T_{furj}(t) - T_{furj+1}(t))$$

使用四阶龙格-库塔法（Runge-Kutta Method）求解该常微分方程。在 t 和 $t+1$ 之间选取四个点求平均斜率。在微分方程 $\frac{\mathrm{d}T_{furj}(t)}{dt}$ 中，令:

$$\begin{aligned}
k_1 &= T_{furj}(t) \\
k_2 &= T_{furj}(t + 0.5\mathrm{d}tk_1) \\
k_3 &= T_{furj}(t + 0.5\mathrm{d}tk_2) \\
k_4 &= T_{furj}(t + \mathrm{d}k_3) \\
\text{则有 } T_{furj}(t+1) &= T_{furj}(t) + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).
\end{aligned}$$

在计算中取微分步长 $\mathrm{d}t = 0.5$，对温度流动过程离散化，得到常微分方程的数值解。该过程使用 Matlab 可视化如图 4 所示:

3.1.2 参数拟合与模型检验

将炉内温度设定为题目所要求的初始问题，通过若干次迭代，得到未知参数的拟合结果如表 2 所示。

参数	猜测值	最小二乘法拟合值
λ	$\lambda_{guess} = 50$	99.7946
$h_1 = h_2$	$h_{guess} = 1000$	1813.6
θ	$\theta_{guess} = 10$	21.2916

表 2: 参数拟合结果

将该拟合数据可视化，和原实验数据进行比对，得到如图 5 所示的结果。为了便于在二维图像中展

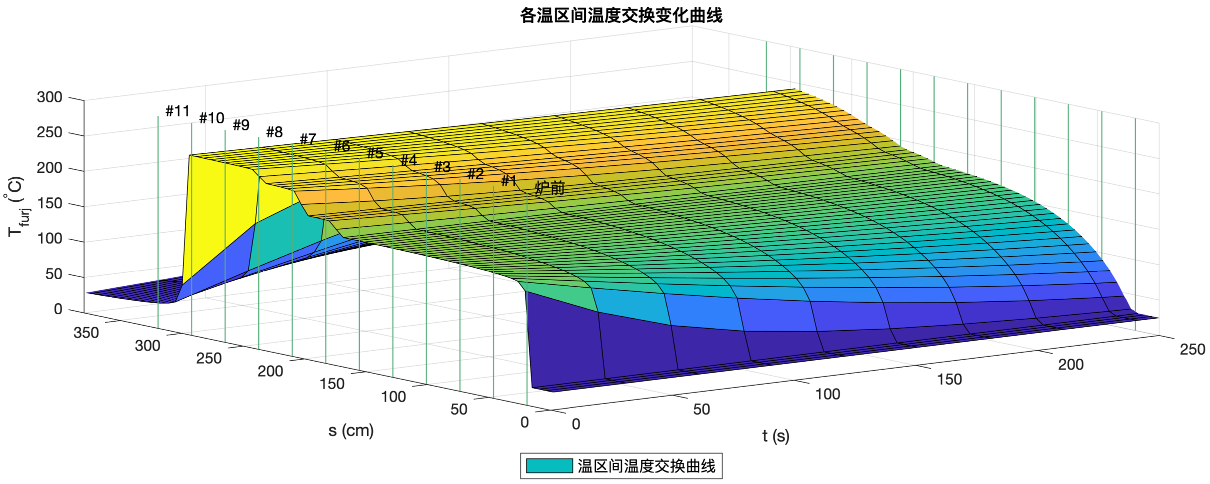

图 4: 温区间温度交换结果

示炉温的大致走向，引入炉温理想曲线的概念。即图中的红线为假设炉内温度不会受到旁边的温度的影响时的温度曲线，在实际计算中，使用的是上文提到的常微分方程计算。

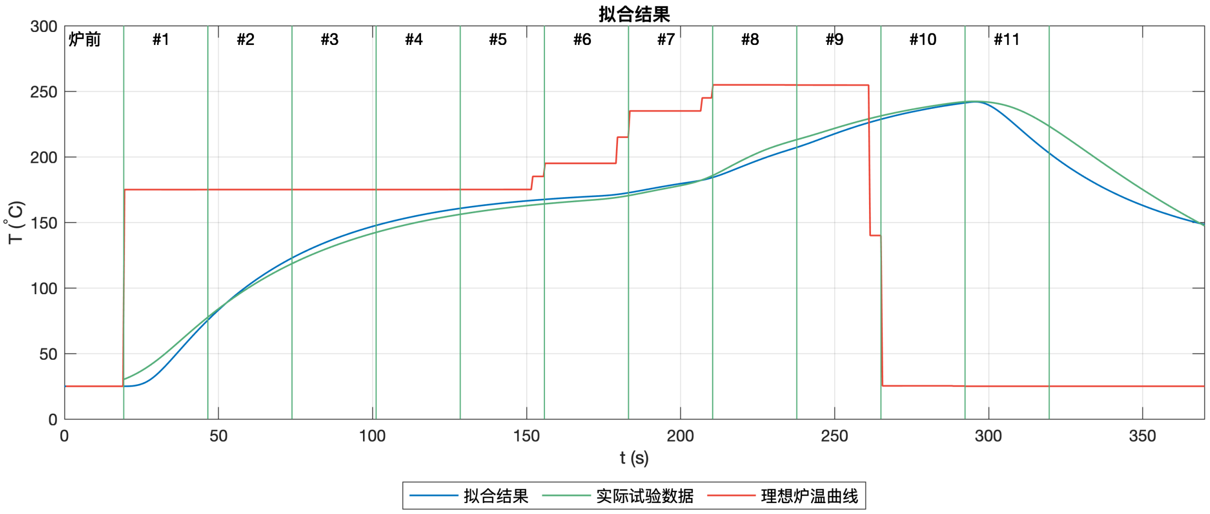

图 5: 拟合结果与实验结果

可以看到，该模型的拟合效果除了在最初升温阶段和后期的降温阶段以外，中间的拟合效果比较理想。考虑到我们在模型中极大地简化了炉间空气的交换流程，这样的结果已经可以接受。数值化地，我们根据参数拟合的结果计算了拟合数据与试验数据的均方差 $R^2 = 0.9212$，可以认为该模型具有较好的拟合效果。

3.1.3 方程求解

为了求解上述微分方程的数值解，可以使用 MATLAB 的 pdepe 函数求解。

考虑一般的情况，为了让 MATLAB 能够求解上述微分方程，首先需要将微分方程转换成以下标准形式：

$$c\left(x,t,u,\frac{\partial u}{\partial x}\right)\frac{\partial u}{\partial t} = x^{-m}\frac{\partial}{\partial x}\left(x^m f\left(x,t,u,\frac{\partial u}{\partial x}\right)\right) + s\left(x,t,u,\frac{\partial u}{\partial x}\right)$$

需要得到标准微分方程中的系数 m, c, f, s。改写热量控制方程 $\mathcal{H}$ 为标准形式如下:

$$\frac{\rho c}{\lambda}\frac{\partial T}{\partial t} - x^0\frac{\partial}{\partial x}(x^0\frac{\partial T}{\partial x}) = 0$$

对应标准形式求得标准 PDE 系数为:

$$\begin{cases} m = 1 \\ c(x,\ t,\ u,\ \frac{\partial u}{\partial x}) = 1 \\ f(x,\ t,\ u,\ \frac{\partial u}{\partial x}) = \frac{\partial u}{\partial x} \\ s(x,\ t,\ u,\ \frac{\partial u}{\partial x}) = 0 \end{cases}$$

此外对于边界条件和初始条件，标准形式为:

$$p(x,t,u) + q(x,t)f\left(x,t,u,\frac{\partial u}{\partial x}\right) = 0$$

将 $\mathcal{H}$ 的边界条件和初始条件化为标准形式得:

$$\begin{cases} \lambda\frac{\partial T}{\partial x}(0,t) + h(T_{fur}(t) - T(0,\ t)) = 0 \\ \lambda\frac{\partial T}{\partial x}(\alpha,t) + h(T(\alpha,\ t) - T_{fur}(t - \frac{\alpha}{t})) = 0 \\ T(x,\ 0) = T_{env} = 25°C \end{cases}$$

对应得到各个参数的情况如下:

$$\begin{cases} m = 0 \\ c(x,\ t,\ u,\ \frac{\partial u}{\partial x}) = \frac{\rho c}{\lambda} \\ s(x,\ t,\ u,\ \frac{\partial u}{\partial x}) = 0 \\ f(x,\ t,\ u,\ \frac{\partial u}{\partial x}) = \frac{\partial T}{\partial x} \end{cases}$$

得到标准形式 PDE 系数后即可使用 MATLAB 得到偏微分方程的数值解，用于后续的拟合和预测。

3.2 解答结果

根据拟合所得到的偏微分方程参数，使用有限元差分法，以步长 dt 为 0.5 s 进行计算，得到偏微分方程的数值解。题目中要求的几个特殊点的温度如表 3 所示。整次预测中每一时刻元件温度的详细结果文件见附件中的 `result.csv`。

位置	**x / cm**	**t / s**	**T °C**
小温区 3 中点	101.25	77.88	134.9095
小温区 6 中点	192.75	148.27	172.9624
小温区 7 中点	223.25	171.13	180.1546
小温区 8 结束处	269.00	206.92	203.7561

表 3: 部分位置的焊接点温度

预测的完整结果可视化地绘制成图像结果如图 6 所示。

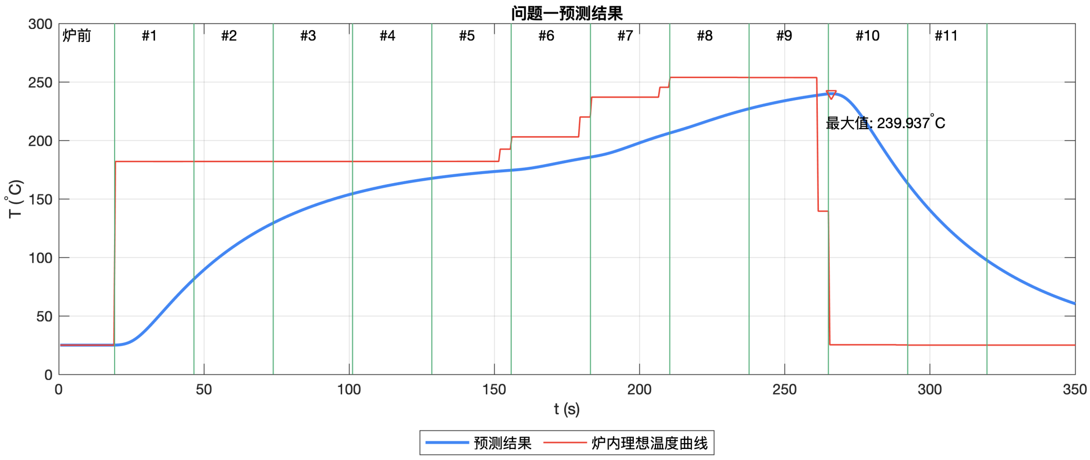

图 6: 问题一：焊接部分的温度曲线

4 问题二

4.1 问题分析

4.1.1 制程界限的数学表达

流程给出的四个制程的条件限制如表 4 所示。

界限名称	最低值	最高值	单位
温度上升斜率	0	3	ºC/s
温度下降斜率	−3	0	ºC/s
温度上升过程中在 150ºC-190ºC 的时间	60	120	s
温度大于 217ºC 的时间	40	90	s
峰值温度	240	250	ºC

表 4: 制程界限

条件规定了元件在全过程中的温度变化情况，以及特定区域的持续时间。我们通过每次迭代后依次计算 PDE 数值解的梯度和对应温区的持续时间，于是将约束条件形式化地改写如下：

$$
\begin{cases}
-3 \le \frac{\mathrm{d}T}{dt} \le 3 \\
60 \le t_{150^\circ C \le T \le 190^\circ C} \le 120 \\
40 \le t_{T \ge 217^\circ C} \le 90 \\
240 \le T_{max} \le 250
\end{cases}
$$

4.1.2 拟合与迭代

由于传送带速度 v 的取值范围为 [65, 100]，且需要找到同时满足以上条件的最大值，所以考虑从 100 cm/min 往 65 cm/min 按照 0.1 的步长进行迭代，依次判读新拟合出的函数 $T(t)$ 是否满足上述约束条件。找到一个可行解之后，继续以 0.02 或者 0.01 的步长迭代，如此反复，最终得到一个较为精确的可行最大速度 v_{max}。对于 cm/min 的速度单位和尺度而言，保留 2 位小数（即进行三次迭代）即能得到较高精度。

4.2 解答结果

先利用步长 0.5 cm/mim，从 100 cm/min 的速度逐渐递减。该过程可以得到以下速度与各指标（最大梯度、最小梯度、升温时间、高温时间、峰值温度）趋势如图 7 所示。完整的结果参考附件中的 `Q2_Trend.csv`。其内包含有的传送速度和指定温度条件下，所有满足条件的解析。

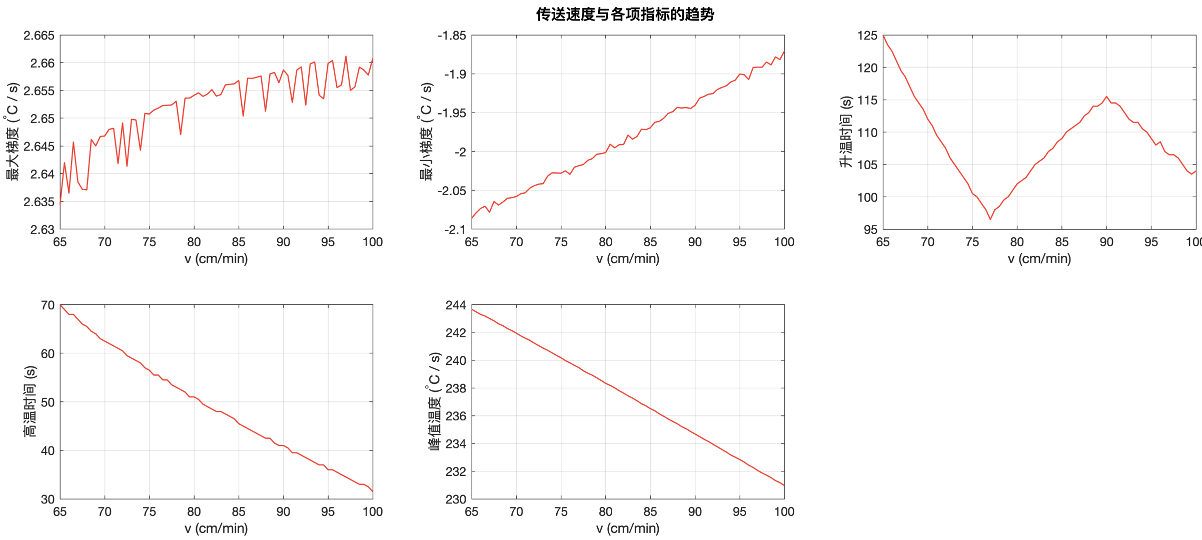

图 7: 传送速度与各项指标的趋势（粗粒度）

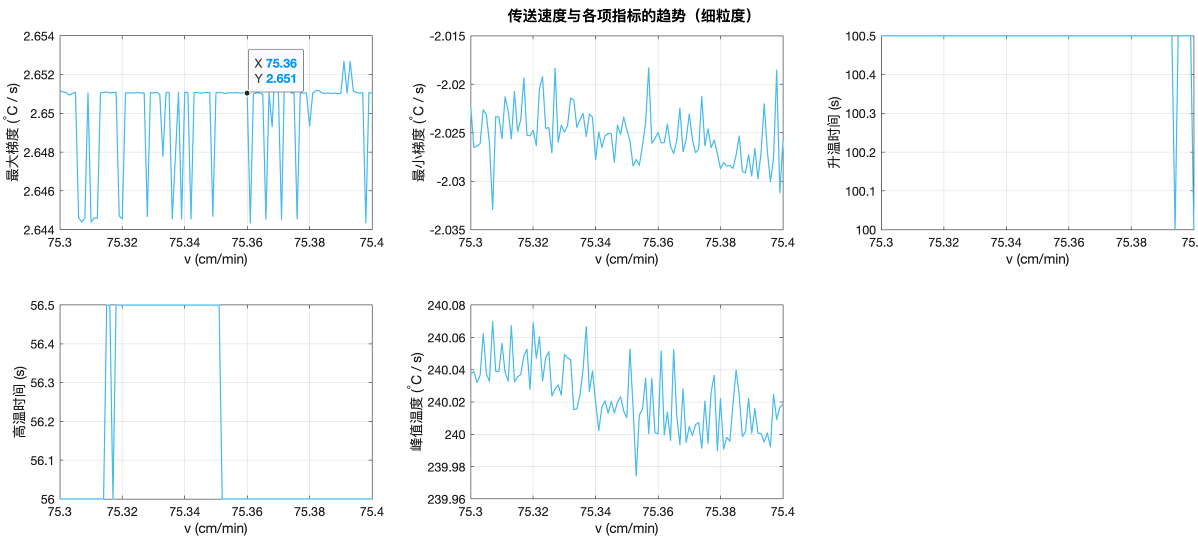

图 8: 传送速度与各项指标的趋势（细粒度）

于是可以发现在 65 cm/min 到 100 cm/min 之间的粗粒度计算，满足的最大速度在 76 cm/min 附近，于是我们在 76 cm/min 至 74 cm/min 以步长为 0.1 cm/min 逼近最大值，再以 75.4 cm/min 至 75.3 cm/min 以步长为 0.01 cm/min 逼近最大值。细粒度逼近的可视化结果如图 8 所示。

经过上述方法的计算, 我们得到能够**满足以上所有制程限制条件的最大传送速度为 75.3520 cm/min**。

5 问题三

5.1 问题分析

5.1.1 目标函数

目标优化函数 $\mathcal{A}=\mathcal{F}(T_{fur_j},v)$，物理意义为目标区域面积，其中：

$$T_{fur_j}=\begin{bmatrix}T_{fur_{1-5}} & T_{fur_6} & T_{fur_7} & T_{fur_{8-9}}\end{bmatrix}$$

冷却区域设定温度 $T_{fur_{10-11}}$ 温度固定为 25 摄氏度，不纳入优化。

目标函数的优化可以形式化地写为

$$(\hat{T_{fur}},\ \hat{v})=arg\ min\mathcal{A}$$

5.1.2 迭代优化算法

使用 brute force 算法，枚举 4 个温区的温度和传送带速度，使用问题一中的模型，利用 T_{fur_j} 指定温度参数，计算出焊接区域的温度曲线，再依次审核制程界限，若不符合任意一条要求则不保留结果。对于满足制程界限的参数，计算积分面积 $\mathcal{A}$。具体地讲，考虑从 165ºC 到 185ºC 枚举小温区 1-5 的温度；从 185ºC 到 205ºC 枚举小温区 6 的温度；从 225ºC 到 245ºC 枚举小温区 7 的温度；从 245ºC 到 265ºC 枚举小温区 8-9 的温度；从 65 cm/min 到 100 cm/min 枚举传送带过炉速度。一个典型的问题在于，直接小步长枚举复杂度过高，不能接受。多次测试发现，当参数调整后，答案的变化应该是连续的。所以考虑可以先大步长枚举所有参数情况，得到最有解的大致范围，然后再在最优解左右两侧延拓后的区间以小步长枚举，得到精确的最优解。

5.2 解答结果

记步长为 dv，参数矩阵为 $[v,\ T_{fur_{1-5}},\ T_{fur_6},\ T_{fur_7},\ T_{fur_{8-9}},\ T_{fur_{10-11}},\ \mathcal{A}]$

在第一次粗粒度查找中，设置 $dv=3$，遍历得到如下参数矩阵：

$$[v,\ T_{fur_{1-5}},\ T_{fur_6},\ T_{fur_7},\ T_{fur_{8-9}},\ T_{fur_{10-11}},\ \mathcal{A}]=[95,\ 183,\ 200,\ 240,\ 263,\ 25,\ 401.7657]$$

第一次粗粒度查找的结果如图 9 所示。为了便于展示，横坐标只绘制了 v 和 T_{fur_1} 参数和阴影面积 $\mathcal{A}$ 表面的情况。

我们在最优结果的参数值上下浮动 3，即令在下一轮遍历中，各参数的取值范围为

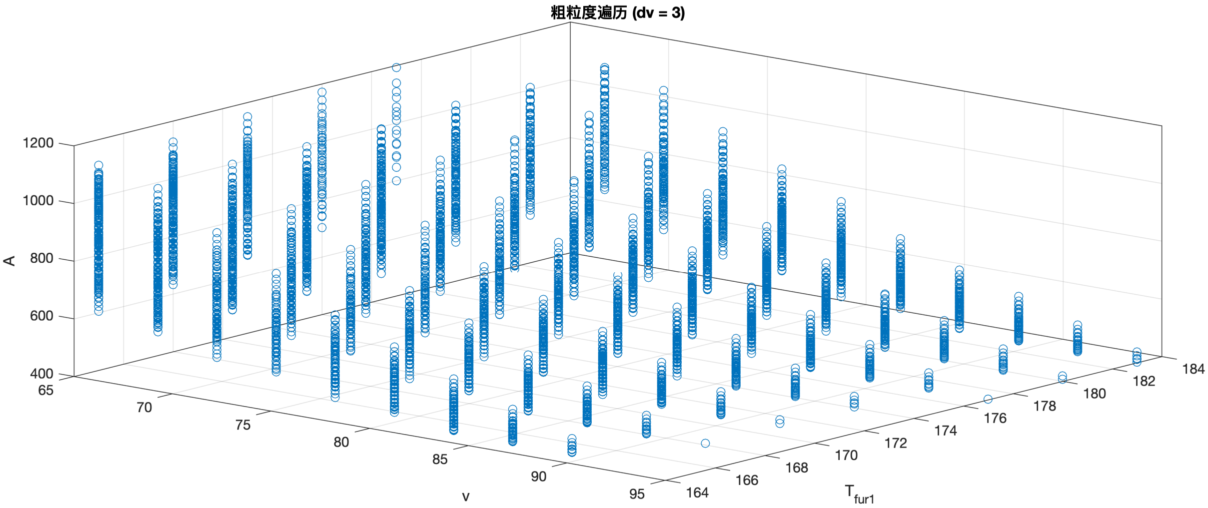

图 9: $T_{fur_1} - v - \mathcal{A}$ 优化结果（粗粒度）

$$\begin{cases} v \in [92,\ 98] \\ T_{fur_{1-5}} \in [180,\ 186] \\ T_{fur_6} \in [197,\ 203] \\ T_{fur_7} \in [237,\ 243] \\ T_{fur_{8-9}} \in [260,\ 265] \end{cases}$$

接下来在 $\mathcal{A}$ 具备最小值的附近进行中粒度遍历。设置 $dv = 1$，遍历得到具备最小 $\mathcal{A}$ 时的参数矩阵为:

$$[v,\ T_{fur_{1-5}},\ T_{fur_6},\ T_{fur_7},\ T_{fur_{8-9}},\ T_{fur_{10-11}},\ \mathcal{A}] = [98,\ 182,\ 203,\ 243,\ 264,\ 25,\ 393.5364]$$

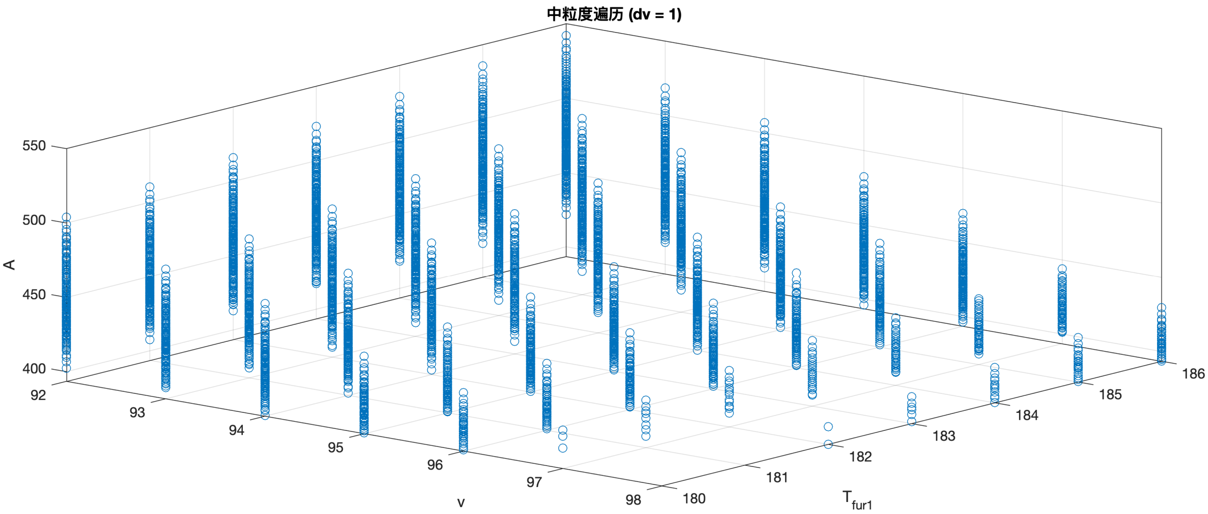

图 10: $T_{fur_1} - v - \mathcal{A}$ 优化结果（中粒度）

此时可视化的结果如图 10 所示。进一步地进行细粒度查找，以得到更精确的参数价值。我们在拟合的结果上下浮动 0.6，并将步长缩小至 0.3，在具备最低值的各个参数附近值进行遍历。此时遍历范围为 $2*dv=0.6$。各参数的取值范围为：

$$\begin{cases} v \in [97.4,\ 98.6] \\ T_{fur_{1-5}} \in [181.4,\ 182.6] \\ T_{fur_6} \in [202.4,\ 203.6] \\ T_{fur_7} \in [242.4,\ 243.6] \\ T_{fur_{8-9}} \in [263.4,\ 264.6] \end{cases}$$

在第三次粗粒度查找中，设置更小的步长 $dv=0.3$。经过遍历，此时的优化结果如图 11 所示。

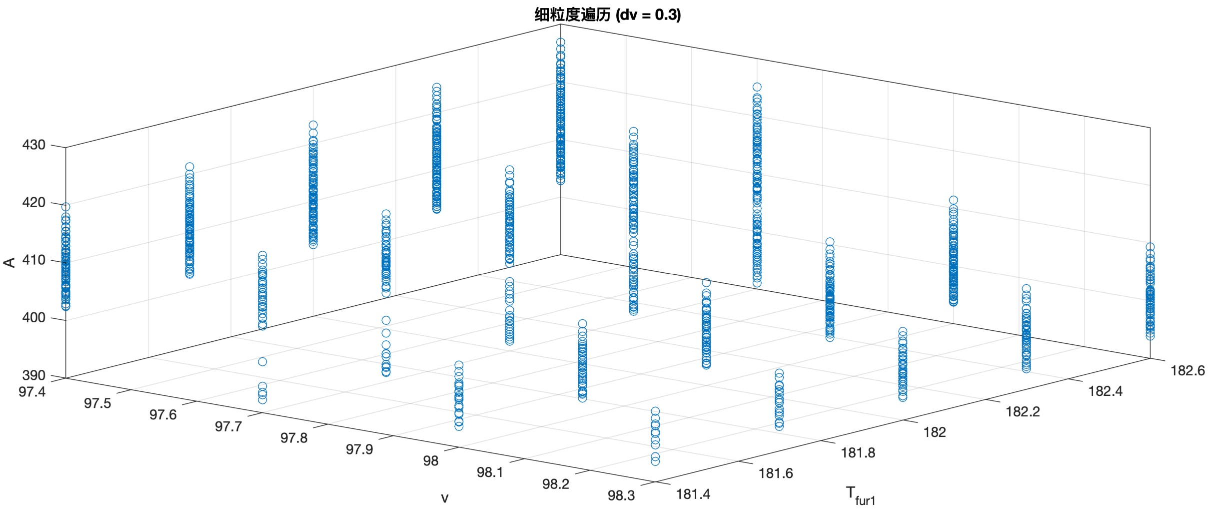

图 11: $T_{fur_1}-v-\mathcal{A}$ 优化结果（细粒度）

得到最优参数矩阵如下：

$$[v,\ T_{fur_{1-5}},\ T_{fur_6},\ T_{fur_7},\ T_{fur_{8-9}},\ T_{fur_{10-11}},\ \mathcal{A}] = [97.7,\ 182.6,\ 203.0,\ 242.4,\ 263.7,\ 25,\ 391.0421]$$

最终，我们可以得到阴影区域的最小面积 $\mathcal{A}$ 为 391.0421，这时传送带的速度为 97.7 cm/min，各个温度区的温度分别为：$T_{fur_{1-5}}=182.6°C, T_{fur_6}=203.0°C, T_{fur_7}=242.4°C, T_{fur_{8-9}}=263.7°C, T_{fur_{10-11}}=25°C$。

6 * 问题四

6.1 问题分析

6.1.1 目标函数

沿袭问题三的思路，有峰值左侧的区域面积为 $\mathcal{A}=\mathcal{F}(T_{fur_j},v)$，再设峰值右侧的区域面积为 $\mathcal{B}=\mathcal{F}(T_{fur_j},v)$，其中各参数仍然有：

$$T_{fur_j} = \begin{bmatrix} T_{fur_{1-5}} & T_{fur_6} & T_{fur_7} & T_{fur_{8-9}} \end{bmatrix}$$

冷却区域设定温度 $T_{fur_{10-11}}$ 温度固定为 25 摄氏度，不纳入优化。

考虑评价指标。要在问题 3 限制下使得 217 $°C$ 到峰值温度左右两侧的温度尽可能对称，我们可以用右侧函数以 $t = t_{peak}$ 轴对称的函数 f_R 和峰值温度左侧的函数 f_L 做分析，计算其方差。为问题 3 再增加一个优化条件，目标为使得方差 R^2 尽可能的小。目标函数的优化可以形式化地写成：

$$(\hat{T_{fur}}, \hat{v}) = arg\ min(\mathcal{A} - \mathcal{B})^2$$

通过不断调整参数来使 R^2 尽可能小。

6.1.2 迭代优化算法

仍然考虑使用 brute-force 算法，枚举四个温区的温度和传送带速度，检验制程界限，舍去不满足条件的参数元组。对于符合制程界限的参数元组，维护差值平方的最小值，利用迭代优化算法，设定恰当的大小步长，逐步迭代逼近最终解。

6.2 解答结果

由于时间限制，仅对此问题做了分析，未得出答案与实现。

7 分析与结论

本文完成了「炉温曲线」问题中的问题一、问题二、问题三。

问题一 问题一基于非稳态传热模型，分析了两个维度的温度变换：炉内空气和炉内的热对流；以及基于牛顿冷却定律的炉子与炉子之间温度的互相影响。通过构件热力学偏微分方程并利用其数值解拟合实际试验数据，得到了元件导热系数 λ 和空气的对流换热系数 h，取得了较好的拟合效果。

问题二 在问题一构建的模型基础上，我们采取了两次迭代，从允许的最大速度开始依次向下预测元件的炉温曲线，并审核其制程界限。在已知的界限范围内部继续递归缩小步长，找到能够满足制程界限的最大温度，得到结果。

问题三 使用迭代优化法，使用微分法近似计算题目要求部分的阴影面积，将计算结果的最小化作为优化函数 $\mathcal{A} = \mathcal{F}(T_{fur_j}, v)$ 的目标。对优化函数的五个参数先进行粗粒度的遍历，找到全局的最优解的取值范围，再在全局最优范围附近缩小步长，递归遍历。经过三次迭代，找到了精确到一位小数的全局最优解。

本模型已经考虑了对流传热及炉间温度传导的情况，利用偏微分方程的数值解能够较好地拟合实际试验的炉温曲线，在利用其它参数预测炉温曲线时也取得了较合理的结果。不过模型仍有可以改进之处，例如对于炉与炉缝隙处的温度传导过程，我们简化为了简单的左右瞬时温度的算术平方根，有不科学之处。如果能够将这部分的热量交换考虑进热量控制方程中，模型在炉后区域的温度下降阶段应当能够取得更好的结果。

此外，在模型构建中我们直接估计元件的长度为 3cm，进行所有的参数拟合。但实际这一参数题目并未给出，可以考虑将元件的长度也作为拟合参数加入最小二乘法的计算中，也有进一步提高拟合效果的可能性。

8 附录

8.1 源代码

本模型基于 MATLAB 实现，以下代码在 macOS 环境和 Windows 环境下的 MATLAB R2020a 版本成功运行。

TPDEFit.m 用于拟合偏微分方程。

```

%% 使用最小二乘方拟合 lambda 和 h

global TfurCalced;
TfurCalced = false;
lambdaguess = 50;
hguess = 1500;
thetaguess = 10;
dt = 0.5;
% 优化参数
options=optimset('TolX',1e-5,'TolFun',1e-5,'MaxFunEvals',1200,'Display','iter');
expdata = xlsread('temp.xlsx');
t = 19:0.5:373; % 生成横向坐标
xdata = expdata(:,1);
xdata = (xdata - 19) * 2 + 1;
ydata = expdata(:,2);

% LSQ
[coef,~, residual] = lsqcurvefit(@(para, data) PDF_for_fitting(para, data), ...
   [lambdaguess, hguess, thetaguess], xdata, ydata, [30 1000 0], [100 5000 100], options);

lambdafit = coef(1)
hfit = coef(2)
thetafit = coef(3)
function r = PDF_for_fitting(params, x)
%% 用于拟合的 PDF 函数
   res = TPDESolve(params(1), params(2), params(3), 70); % 70 = Velocity
   sol = res(:,15);
   sol = sol(19:728);
   r = sol(x);
end
```

TPDESolve.m 用于计算偏微分方程。

```
function [result] = TPDESolve(lambdaguess, hguess, thetaguess, velocity)
%% 使用 pdepe 求解元件温度的微分方程
% 参数配置
Tenv = 25; % 环境温度
V = velocity; % 传送速度
alpha = 3.2; % 物体长度 cm

% 物理参数
rho = 7.874; % 密度
C = 480; % 比热容
lambda = lambdaguess; % 导热系数
% h = lambda / (alpha / 100);
h = hguess; % 对流换热系数;
x = linspace(0, alpha, 30);
t = linspace(0, 400, 800);
m = 0;

sol = pdepe(m, @TPDE, @TPDE_IC, @TPDE_BC, x, t);
hold ;
grid on;
plot(t, sol(:,15), 'LineWidth', 1)
title('PDE 数值解');

result = sol;
    %% PDF
    % PDE 方程
    function [c, f, s] = TPDE(x, t, T, dudx)
        % 转换为标准 PDE
        c = rho * C / lambda;
        f = dudx;
        s = 0;
    end

    % PDE 初始条件
    function T0 = TPDE_IC(x)
        % Initial Condition
        T0 = Tenv;
    end

    % PDE 边界值
    function [pl, ql, pr, qr] = TPDE_BC(xl, ul, xr, ur, t)
        pl = h * Tfur2D(t, ul, thetaguess, V) - h * ul;
        ql = lambda;
        pr = h * ur - h * Tfur2D(t - alpha / V, ur, thetaguess, V);
        qr = lambda;
```

```
    end
end
```

Tfur2D.m 用于计算炉内温度。

```
function [tfur] = Tfur(t, m, theta, velocity)
global Tfurdata;
    furnaceLength = 30.5; % 熔炉长度
    furnaceGap = 5; % 熔炉间隙
    edgeGap = 25; % 炉前与炉后
    if (nargin < 4)
        % 指定默认速度
        V = 70;
    else
        V = velocity; % 速度， cm/min
    end
    tempEnv = 25; % 环境温度

     % 附件 - 拟合实验温区温度配置
%        t1 = 175;
%        t2 = 195;
%        t3 = 235;
%        t4 = 255;
%        t5 = 25;

     % Q1 - 预测温区温度配置
%      t1 = 173;
%      t2 = 198;
%      t3 = 230;
%      t4 = 257;
%      t5 = 25;

    % Q2 - 制程界限温区温度配置
      t1 = 182;
      t2 = 203;
      t3 = 237;
      t4 = 254;
      t5 = 25;

    hf = 1e-4; % 炉间温度传播系数
    if t < 0.5
        t = 0.5;
    end

    tempMat = TfurNew([tempEnv, t1, t2, t3, t4, t5], hf);
    distance = V / 60 * t + (furnaceLength + furnaceGap - edgeGap);
```

```
    furnaceTemp = tempMat(ceil(t * 2), 1:13);
    fCount = ceil(distance / (furnaceLength + furnaceGap));
    fCount(fCount < 1) = 1;
    dd = distance - (furnaceLength + furnaceGap) * (fCount - 1);
    if(fCount < 11)
        if dd <= furnaceLength
            tfur = furnaceTemp(fCount);
        else
            tfur = (furnaceTemp(fCount) + furnaceTemp(fCount + 1)) / 2;
        end
    else
        tfur = furnaceTemp(11);
    end
    if(fCount > 10)
          tfur = theta * nthroot((m - tfur), 3) + tfur;
    end
    Tfurdata(ceil(t * 2))= tfur;
end
```

Tfur.m 理想炉温曲线。

```
function [tfur] = Tfur(t)
    furnaceLength = 30.5; % 熔炉长度
    furnaceGap = 5; % 熔炉间隙
    edgeGap = 25; % 炉前与炉后
    V = 78; % 速度， cm/min

    tempEnv = 25; % 环境温度
     t1 = 175; % 附件 - 拟合实验温区温度配置
     t2 = 195;
     t3 = 235;
     t4 = 255;
     t5 = 25;
    hf = 0.00002; % 炉间温度传播系数

    if t < 0.5
        t = 0.5;
    end

    tempMat = TfurNew([tempEnv, t1, t2, t3, t4, t5], hf);
    distance = V / 60 * t - edgeGap;
    furnaceTemp = tempMat(ceil(t * 2), 2:12);
    if distance < 0
        tfur = tempEnv;
        return
    end
```

```
    fCount = ceil(distance / (furnaceLength + furnaceGap));
    fCount(fCount < 1) = 1;
    dd = distance - (furnaceLength + furnaceGap) * (fCount - 1);
    if(fCount < 11)
        if dd <= furnaceLength
            tfur = furnaceTemp(fCount);
        else
            tfur = (furnaceTemp(fCount) + furnaceTemp(fCount + 1)) / 2;
        end
    else
        tfur = furnaceTemp(11);
%          tfur = 25;
    end
end
```

TfurNew.m 用于计算炉内空气温度的动态变化。

```
function [tfur] = TfurNew(mat, h)
%% 考虑炉间热量传导的情况
% if exist('dp.mat', 'file')
%     load 'dp.mat' 'dp';
%     tfur = dp;
%     return;
% end
global dp;
global TfurCalced;

if TfurCalced
    tfur = dp;
    return;
end
% 13 * 800 矩阵
tfur = zeros(800, 13);
dp = zeros(800, 13);
for i = 1 : 13
    if i == 1
        dp(1, i) = mat(1);
    elseif i <= 6
        dp(1, i) = mat(2);
    elseif i == 7
        dp(1, i) = mat(3);
    elseif i == 8
        dp(1, i) = mat(4);
    elseif i <= 10
        dp(1, i) = mat(5);
    else
```

```
        dp(1, i) = mat(6);
    end
end
for i = 2 : 800
    for j = 1 : 13
        if j == 1
            dp(i, j) = dp(i - 1, j) - h * nthroot((dp(i - 1, j) - dp(i - 1, j + 1)), 3);
        elseif j <= 12
            dp(i, j) = dp(i - 1, j) - h * nthroot((dp(i - 1, j) - dp(i - 1, j + 1)), 3) - h *
                ↪ nthroot((dp(i - 1, j) - dp(i - 1, j - 1)), 3);
        else
            dp(i, j) = dp(i - 1, j) - h * nthroot((dp(i - 1, j) - dp(i - 1, j - 1)), 3);
        end
    end
end
tfur = dp;
TfurCalced = true;

end
```

Variables.m 用于存储模型常量。

```
%% 参数配置
global Tenv
global V
global alpha
global h
global rho
global c
global lambda
global furnaceLength
global furnaceGap
global edgeGap

furnaceLength = 30.5; % 熔炉长度
furnaceGap = 5; % 熔炉间隙
edgeGap = 25; % 炉前与炉后

Tenv = 25; % 环境温度
V = 70; % 传送速度
alpha = 3; % 物体长度

% 物理参数
h = 1500;
rho = 7.874;
c = 480;
```

```
lambda = 50;
```

Preview.m 用于预览拟合效果。

```
%% 设定温度曲线绘制

x = 0.5:0.5:400;
y = 1:800;
for i = 1:800
    y(i) = Tfur(x(i));
end
% figure;
box on; grid on; hold on;
width = 900;
height = 350;
set(gcf,'position',[0,0,width,height])
red = '#EB4537';
green = '#55AF7B';
blue = '#4286F3';

title('拟合结果')
xlabel('t (s)')
ylabel('T (^{\circ}C)')

t = xlsread('temp.xlsx')
plot(t(:,1), t(:,2), 'Color', green, 'LineWidth', 1)
plot(x, y, 'Color',red, 'LineWidth', 1)

for i=1:11
    lines = [lines, (25 + i * (furnaceLength + furnaceGap)) / (70 / 60)];
end

for i=1:12
    line([lines(i) lines(i)], [0 300], 'linewidth', 0.7, 'color', green);
    if i ~= 1
        text(lines(i) - 18, 290,['#', num2str(i - 1)]);
    else
        text(lines(i) - 18, 290,['炉前']);
    end
end

lgd = {'拟合结果', '实际试验数据', '理想炉温曲线'};
legend(lgd,'location','southoutside','NumColumns',3,'FontSize',10);
xlim([0 370])
exportgraphics(gcf,'pic/FitResult.png','Resolution',300)

```

```
saveas(gcf,['pic/','FurnacePreTemp','.png'])
```

Q1_Predict.m 问题一的主要求解代码。

```
clf; close all;
%% 特定条件下的预测

% 预测得到的参数
lambda = 99.7964;
h = 1813.6;
theta = 21.2916;
V = 78;
sol = TPDESolve(lambda, h, theta, V);
res = sol(:,15);
lines = [25 / (V / 60)];

furnaceLength = 30.5; % 熔炉长度
furnaceGap = 5; % 熔炉间隙
edgeGap = 25; % 炉前与炉后

% 绘制炉
% figure;
width = 900; height = 350;
set(gcf,'position',[0,0,width,height])
red = '#EB4537';
green = '#55AF7B';
blue = '#4286F3';
plot(0.5:0.5:400, res, 'LineWidth', 2, 'Color', blue);
grid on;
hold on;

% 温度曲线
x = 0.5:0.5:400;
y = 1:800;
for i = 1:800
    y(i) = Tfur(x(i));
end
plot(x, y, 'Color', red, 'LineWidth', 1)

for i=1:11
    lines = [lines, (25 + i * (furnaceLength + furnaceGap)) / (V / 60)];
end

for i=1:12
    line([lines(i) lines(i)], [0 300], 'linewidth', 0.7, 'color', green);
    if i ~= 1
```

```
        text(lines(i) - 18, 290,['#', num2str(i - 1)]);
    else
        text(lines(i) - 18, 290,['炉前']);

    end
end
title('问题一预测结果')
xlabel('t (s)');
ylabel('T (^{\circ}C)');

% Mark maximum.
amax = max(res);
tmax = find(res==amax) / 2;
plot(tmax,amax,'Color',red,'Marker','v','LineStyle','none','LineWidth',1);
text(tmax-2,amax*0.9,['最大值: ', num2str(amax), '^{\circ}C']);

lgd = {'预测结果', '炉内理想温度曲线'};
legend(lgd,'location','southoutside','NumColumns',2,'FontSize',10);
xlim([0, 350]);
exportgraphics(gcf,'pic/Q1.png','Resolution',300)



%% 3D Surface
figure;
t = 0.5:0.5:800;
% surf(1:10:300,t(1:10:800),sol(1:10:800,:));
surf(1:30:900,linspace(0,450,80),sol(1:10:800,:));

set(gcf,'position',[0,0,width,height])
title('各温区间温度交换变化曲线');
ylabel('s (cm)');
xlabel('t (s)');
zlabel('T_{furj} (^{\circ}C)');
xlim([0 250])
ylim([0 380])

for i=1:12
    line([0 0], [lines(i) lines(i)], [0 300], 'linewidth', 0.7, 'color', green);
    line([250 250], [lines(i) lines(i)], [0 300], 'linewidth', 0.7, 'color', green);

    if i ~= 1

        text(0, lines(i) - 6, 310,['#', num2str(i - 1)]);
    else
        text(0, lines(i) - 6, 310,['炉前']);

```

```
    end
end

lgd = {'温区间温度交换曲线'};
legend(lgd,'location','southoutside','NumColumns',2,'FontSize',10);
exportgraphics(gcf,'pic/TempExchange.png','Resolution',300)
```

Q2_Restrain.m 问题二的主要求解代码。

```
% 从最大速度开始筛选
detail = [];
red = '#EB4537';
green = '#55AF7B';
blue = '#4286F3';
for vv = 75.4:-0.001:75.3
% for vv = 100:-0.5:65
    out = vv
    % 指定拟合的模型参数
    lambda = 99.7964;
    h = 1813.6;
    theta = 21.2916;
    sol = TPDESolve(lambda, h, theta, vv);
    sol = sol(:,15);
    restrains = {@RestrainDiff @RestrainRising @RestrainPeakTime @RestrainPeakTemp};
    flag = 0;
    [r1, dd(2), dd(3)] = RestrainDiff(sol);
    [r2, dd(4)] = RestrainRising(sol);
    [r3, dd(5)] = RestrainPeakTime(sol);
    [r4, dd(6)] = RestrainPeakTemp(sol);

    dd(1) = vv;
    if ~r1
        flag = 1;
    end

    if ~r2
        flag = 2;
    end

    if ~r3
        flag = 3;
    end

    if ~r4
        flag = 4;
    end
```

```
    dd(7) = flag;
    detail = [detail; dd];

end

writematrix(detail, 'Q2.csv');

%% 可视化
figure;
hold on;
grid on;
box on;

set(gcf,'position',[0,0,1000,450])
subplot(2,3,1);
plot(detail(:,1), detail(:,2), 'Color', red, 'LineWidth', 1);
xlabel('v (cm/min)')
ylabel('最大梯度 (^{\circ}C / s)');
subplot(2,3,2);
plot(detail(:,1), detail(:,3), 'Color', red, 'LineWidth', 1);
xlabel('v (cm/min)')
ylabel('最小梯度 (^{\circ}C / s)');
subplot(2,3,3);
plot(detail(:,1), detail(:,4), 'Color', red, 'LineWidth', 1);
xlabel('v (cm/min)')
ylabel('升温时间 (s)');
subplot(2,3,4);
plot(detail(:,1), detail(:,5), 'Color', red, 'LineWidth', 1);
xlabel('v (cm/min)')
ylabel('高温时间 (s)');
subplot(2,3,5);
plot(detail(:,1), detail(:,6), 'Color', red, 'LineWidth', 1);
xlabel('v (cm/min)')
ylabel('峰值温度 (^{\circ}C / s)');
exportgraphics(gcf,'pic/Trend.png','Resolution',300);


function [res, valueMax, valueMin] = RestrainDiff(sol)
%% 斜率是否超过 3？
    tdiff = 2 * diff(sol);
    rising = tdiff(tdiff > 0);
    descending = tdiff(tdiff < 0);
    [rc,~] = size(rising(rising > 3));
    [dc,~] = size(descending(descending < -3));
    if rc ~= 0
        sprintf("升温速度过大。")
        res = false;
```

```
        return;
    end
    if dc ~= 0
        sprintf("降温速度过大。")
        res = false;
        return;
    end
    valueMax = max(tdiff);
    valueMin = min(tdiff);
    res = true;
end

function [res, value] = RestrainRising(sol)
%% 升温时间是否符合要求？
    % 只考虑升温过程
    sol = sol(1:600);
    count = sol(sol > 150 & sol < 190);
    [t, ~] = size(count);
    if t / 2 > 60 && t / 2 < 120
        res = true;
    else
        res = false;
        sprintf("升温时间不合要求。")
    end
    value = t / 2;
end


function [res, value] = RestrainPeakTime(sol)
%% 高温时间是否符合要求？

    count = sol(sol > 217);
    [t, ~] = size(count);
    if t / 2 > 40 && t / 2 < 90
        res = true;
    else
        res = false;
        sprintf("高温时间不合要求。")
    end
    value = t / 2;
end


function [res, value] = RestrainPeakTemp(sol)
%% 峰值温度是否符合要求？
    mt = max(sol);
    if mt > 240 && mt < 250
```

```
        res = true;
    else
        res = false;
        sprintf("峰值温度不合要求。")
    end
    value = mt;
end
```

Q3_Optimize.m 问题三的主要求解代码。

```
Output = [];
OutputFail = [];

% 模型参数
lambda = 99.7964;
h = 1813.6;
theta = 21.2916;

% 优化参数
% V = 70;
% t1 = 175;
% t2 = 195;
% t3 = 240;
% t4 = 257;
% t5 = 25; % t5 恒定为 25

%% ROUND 1
% dt = 3;
% %
% Vb = [89 100];
% t1b = [165 185];
% t2b = [185 205];
% t3b = [225 245];
% t4b = [245 265];

%% ROUND 2
dt = 1;
Vb = [94 96];
t1b = [180 186];
t2b = [197 203];
t3b = [237 243];
t4b = [260 265];

% ROUND 3
dt = 0.15;
Vb = [98.3 98.6];
```

```
t1b = [181.4 182.6];
t2b = [202.4 203.6];
t3b = [242.4 243.6];
t4b = [263.4 264.6];

for V=Vb(1):dt:Vb(2)
    for t1=t1b(1):dt:t1b(2)
        for t2=t2b(1):dt:t2b(2)
            for t3=t3b(1):dt:t3b(2)
                for t4=t4b(1):dt:t4b(2)
                    [Aera, sol] = FF(lambda, h, theta, V, [t1 t2 t3 t4 t5]);
                    [success, flag] = Restrain(sol(:,15));
                    if success
                        Output = [Output; [V, t1, t2, t3, t4, t5, Aera]];
                    else
                        OutputFail = [OutputFail; [V, t1, t2, t3, t4, t5, flag]];
                    end
                    disp([V t1 t2 t3 t4 success flag round(Aera)]);
                end
            end
        end
    end
end

% 参数上下界
%% 优化目标函数
function [aera, sol] = FF(lambda, h, theta, V, mat)
    sol = TPDESolveFO(lambda, h, theta, V, [mat(1) mat(2) mat(3) mat(4) mat(5)]);
    aera = getAera(sol(:,15));
    function res = getAera(sol)
        peakTemp = max(sol);
        peakTime = find(sol==peakTemp);
        sol = sol(1:peakTime);
        count = sol(sol > 217 & sol <= peakTemp);
        [l,~] = size(count);
        res = sum(count) / 2 - (l / 2) * 217;
    end
end

%% 约束条件
function [success, flag] = Restrain(sol)
    flag = 0;
    [r1] = RestrainDiff(sol);
    [r2] = RestrainRising(sol);
    [r3] = RestrainPeakTime(sol);
    [r4] = RestrainPeakTemp(sol);
    if ~r1
```

```
        flag = 1;
    end
    if ~r2
        flag = 2;
    end
    if ~r3
        flag = 3;
    end
    if ~r4
        flag = 4;
    end
    if flag == 0
        success = true;
    else
        success = false;
    end
    function [res, valueMax, valueMin] = RestrainDiff(sol)
    % 斜率是否超过 3？
        tdiff = 2 * diff(sol);
        rising = tdiff(tdiff > 0);
        descending = tdiff(tdiff < 0);
        [rc,~] = size(rising(rising > 3));
        [dc,~] = size(descending(descending < -3));
        if rc ~= 0
            res = false;

        end
        %if dc ~= 0
           % res = false;

        %end
        valueMax = max(tdiff);
        valueMin = min(tdiff);
        res = true;
    end

    function [res, value] = RestrainRising(sol)
        % 升温时间是否符合要求？
        % 只考虑升温过程
        sol = sol(1:600);
        count = sol(sol > 150 & sol < 190);
        [t, ~] = size(count);
        if t / 2 > 60 && t / 2 < 120
            res = true;
        else
            res = false;
        end
```

```
        value = t / 2;
    end

    function [res, value] = RestrainPeakTime(sol)
        % 高温时间是否符合要求？

        count = sol(sol > 217);
        [t, ~] = size(count);
        if t / 2 > 40 && t / 2 < 90
            res = true;
        else
            res = false;
        end
        value = t / 2;
    end

    function [res, value] = RestrainPeakTemp(sol)
        % 峰值温度是否符合要求？
        mt = max(sol);
        if mt > 240 && mt < 250
            res = true;
        else
            res = false;
        end
        value = mt;
    end
end
```

Plotter.m 用于绘制优化结果。

```
%% 寻找最小值

minRow = find(Output(:,7) == min(Output(:,7)));
Output(minRow,:)

%% 优化结果绘制
scatter3(Output(:,1), Output(:,2),Output(:,7));
view(40, 35);
set(gcf,'position',[0,0,1000,400]);
box on;
grid on;
% title("粗粒度遍历 (dv = 3)");
% title("中粒度遍历 (dv = 1)");
title("细粒度遍历 (dv = 0.3)");
xlabel('v');
ylabel('T_{fur1}');
```

```
zlabel('A');
exportgraphics(gcf,'OROUND3.png','Resolution',300);
```

8.2 结果文件

result.csv 问题一的解答:

```
时间(s),温度(摄氏度)
0,25
0.5,25
1,25.0000000002737
1.5000,25.0000000027697
2,25.0000000147783
2.5000,25.0000109376896
3,25.0000218606008
3.5000,25.0000327835121
4,25.0000437064234
4.5000,25.0000546293347
5,25.0000655522459
5.5000,25.0000764751572
6,25.0000873980685
6.5000,25.0000983209798
7,25.0001092438911
7.5000,25.0001465335839
8,25.0001838232767
8.5000,25.0002211129696
9,25.0002584026624
9.5000,25.0002956923552
10,25.0003329820481
10.500,25.0003702717409
11,25.0004075614337
11.500,25.0004448511266
12,25.0004821408194
12.500,25.0005263471963
13,25.0005745998678
13.500,25.0006265833630
14,25.0006823673214
14.500,25.0007420700437
15,25.0008058059695
15.500,25.0008736774747
16,25.0009444973978
16.500,25.0010195006578
17,25.0011012571675
17.500,25.0012693962684
18,25.0021959464443
18.500,25.0062010632229
19,25.0180887640338
19.500,25.0450112684105
20,25.0955138549657
20.500,25.1777823197750
21,25.2984508049384
21.500,25.4624697758199
22,25.6737213794105
22.500,25.9363642359910
23,26.2555862337525
23.500,26.6376069619147
24,27.0882408764163
24.500,27.6127068006648
25,28.2142704102695
25.500,28.8936713203521
26,29.6510998043214
26.500,30.4832567842266
27,31.3860230672907
27.500,32.3544569963148
28,33.3824995314981
28.500,34.4665005597781
29,35.6005892045057
29.500,36.7785739847711
30,37.9958590936096
30.500,39.2473850474030
31,40.5280238669835
31.500,41.8325580583534
32,43.1585085659515
32.500,44.5012539822777
33,45.8561053365349
33.500,47.2231034123935
34,48.5974816805724
34.500,49.9766820363965
35,51.3636844129079
35.500,52.7521159851482
36,54.1392987586635
36.500,55.5225547389999
37,56.8997481020511
37.500,58.2779064468243
38,59.6497898446136
38.500,61.0136467266850
39,62.3677255243048
39.500,63.7102746687393
40,65.0486904290100
40.500,66.3770429205360
41,67.6943715037984
41.500,68.9997397346458
42,70.2922111689269
42.500,71.5761084181703
43,72.8486327387099
43.500,74.1090430597860
44,75.3569163529188
44.500,76.5918295896288
45,77.8157534122866
45.500,79.0277025173891
46,80.2272063663173
46.500,81.4140994902194
47,82.5882164202441
47.500,83.7515166331817
48,84.9030331891921
48.500,86.0421838211457
49,87.1688838527103
49.500,88.2830486075533
50,89.3845934093423
50.500,90.4734335817447
51,91.5494844484282
51.500,92.6126613330602
52,93.6640603947093
52.500,94.7053555724001
53,95.7346893441542
53.500,96.7520852002864
54,97.7575666311116
54.500,98.7511597629447
55,99.7328801781004
55.500,100.702759274894
56,101.660817907639
56.500,102.607079566652
57,103.543905047131
57.500,104.469953750461
58,105.385128990116
58.500,106.289529425743
59,107.183253716992
59.500,108.066400523509
60,108.939068504943
60.500,109.801356320940
61,110.653362631150
61.500,111.495496065642
62,112.328041095056
62.500,113.150823763954
63,113.963963184928
63.500,114.767578470567
64,115.561788733461
64.500,116.346713086201
65,117.122470641377
65.500,117.889180511580
66,118.646951097377
66.500,119.395585640039
67,120.135428725067
67.500,120.866589811754
68,121.589178359392
68.500,122.303303827272
69,123.009075674687
69.500,123.706603360929
70,124.395996345290
70.500,125.077364087062
71,125.749482881087
71.500,126.412713645566
72,127.067998321836
72.500,127.715437773941
73,128.355132865927
73.500,128.987184461837
74,129.611693425716
74.500,130.228760621608
75,130.838486913558
75.500,131.440973165611
76,132.036320241810
76.500,132.624629006200
77,133.206000322826
77.500,133.780535055732
78,134.348334068963
78.500,134.909498226562
79,135.464128392575
79.500,136.012325431046
80,136.554190206019
80.500,137.089823581539
81,137.619326421650
81.500,138.142799590397
82,138.660343951824
82.500,139.172060369976
83,139.678049708896
83.500,140.178412832630
84,140.673250605222
84.500,141.161718514229
85,141.640885175875
85.500,142.114307150641
86,142.582067053741
86.500,143.044247499152
87,143.500931102709
87.500,143.952200479007
88,144.398138242953
88.500,144.838827009450
89,145.274349393405
89.500,145.704788009724
90,146.130225473311
90.500,146.550744399072
91,146.966427401912
91.500,147.377357096738
92,147.783616098453
92.500,148.185287021965
93,148.582452482178
93.500,148.975195093998
94,149.363597472329
94.500,149.747742232079
95,150.127711988151
95.500,150.503589355452
96,150.875456948887
96.500,151.243397383361
97,151.607493273780
97.500,151.967827235049
98,152.324481882074
98.500,152.672963188353
99,153.016299499842
99.500,153.355572535372
100,153.690843276178
100.50,154.022172703497
101,154.349621798566
101.50,154.673251542619
102,154.993122916892
102.50,155.309296902623
103,155.621834481047
103.50,155.930796633400
104,156.236244340917
104.50,156.538238584836
105,156.836840346391
105.50,157.132110606820
106,157.424110347357
106.50,157.712900549240
107,157.998542193704
107.50,158.281096261984
108,158.560623735318
108.50,158.837185594941
109,159.110842822089
109.50,159.381656397998
110,159.649687303904
110.50,159.914996521043
111,160.177645030652
111.50,160.437693813966
112,160.693261937585
112.50,160.943241219877
113,161.190275644736
113.50,161.434407863456
114,161.675680527331
114.50,161.914136287655
115,162.149817795723
115.50,162.382767702827
116,162.613028660262
116.50,162.840643319322
117,163.065654331301
117.50,163.288104347493
118,163.508036019192
118.50,163.725491997692
119,163.940514934286
119.50,164.153147480269
120,164.363432286935
120.50,164.571412005577
121,164.777129287490
121.50,164.980626783968
122,165.181947146304
122.50,165.381133025793
123,165.578227073729
123.50,165.773271941405
124,165.966310280116
124.50,166.157384741155
125,166.346537975816
125.50,166.533753779100
126,166.715865140531
126.50,166.895825858087
127,167.073666005783
127.50,167.249415657637
128,167.423104887664
128.50,167.594763769882
129,167.764422378306
129.50,167.932110786954
130,168.097859069841
130.50,168.261697300984
131,168.423655554400
131.50,168.583763904106
132,168.742052424117
132.50,168.898551188450
133,169.053290271122
133.50,169.206299746148
134,169.357609687547
134.50,169.507250169333
135,169.655251265525
135.50,169.801643050137
136,169.946455597187
136.50,170.089718980691
137,170.231463274665
137.50,170.371718553126
138,170.510514890091
138.50,170.647882359576
139,170.783851035598
139.50,170.917676813554
140,171.049520999602
140.50,171.179830293057
141,171.308622668490
141.50,171.435916100469
142,171.561728563565
142.50,171.686078032347
143,171.808982481385
143.50,171.930459885248
144,172.050528218507
144.50,172.169205455731
145,172.286509571490
145.50,172.402458540353
146,172.517070336891
146.50,172.630362935672
147,172.742354311267
147.50,172.853062438245
148,172.962416606991
148.50,173.070402019597
149,173.177092089900
149.50,173.282497901028
150,173.386630536105
150.50,173.489501078258
151,173.591120610614
151.50,173.691500216298
152,173.790642428523
152.50,173.888562333294
153,173.985329044208
153.50,174.080959909123
154,174.175521683415
154.50,174.269227418971
155,174.362496719620
155.50,174.456059541035
156,174.550580480067
156.50,174.647300912270
157,174.747378345672
157.50,174.851691670587
158,174.961002157481
158.50,175.075817870756
159,175.196747185934
159.50,175.324493528728
160,175.460308882688
160.50,175.604469246016
161,175.756695496163
161.50,175.918351892951
162,176.088816000117
162.50,176.268069117786
163,176.456096038443
163.50,176.652907865309
164,176.856758195378
164.50,177.068365293983
165,177.284869412884
165.50,177.503839434768
166,177.728433056689
166.50,177.958650278647
167,178.194491100642
167.50,178.435955522674
168,178.676524787498
168.50,178.917390618681
169,179.160965474056
169.50,179.407249353625
170,179.656242257386
170.50,179.907944185340
171,180.154594276544
171.50,180.401237194586
172,180.648167427554
172.50,180.895384975448
173,181.142889838268
173.50,181.389699405804
174,181.632380603231
174.50,181.873844694280
175,182.114091678951
175.50,182.353121557244
176,182.590934329159
```

```
176.50,182.826578382634
177,183.059402755334
177.50,183.290231173361
178,183.519063636715
178.50,183.745900145395
179,183.970293486327
179.50,184.192257478084
180,184.411798166202
180.50,184.629058613201
181,184.844046182092
181.50,185.056929587944
182,185.268150567089
182.50,185.478573103220
183,185.689485993355
183.50,185.902331774029
184,186.119132292278
184.50,186.341526684594
185,186.570780898419
185.50,186.808047901964
186,187.054108811533
186.50,187.309977357334
187,187.576924477687
187.50,187.856213455733
188,188.149255044717
188.50,188.456197004214
189,188.776671462004
189.50,189.111370451065
190,189.459971315888
190.50,189.821717115129
191,190.195564736937
191.50,190.582178468896
192,190.978247449414
192.50,191.384089680318
193,191.798559707274
193.50,192.212671182018
194,192.633589752347
194.50,193.061315418262
195,193.495848179762
195.50,193.937188036848
196,194.378823178170
196.50,194.817264837428
197,195.258200218482
197.50,195.701629321330
198,196.147552145974
198.50,196.595968692414
199,197.038056688476
199.50,197.477569728222
200,197.916230122954
200.50,198.354037872671
201,198.790992977375
201.50,199.226319103891
202,199.654713287553
202.50,200.080281371945
203,200.503023357067
203.50,200.922939242919
204,201.340029029501
204.50,201.753436905308
205,202.162184564500
205.50,202.566978343866
206,202.967628005885
206.50,203.363954198229
207,203.756062559616
207.50,204.143596964843
208,204.526829776326
208.50,204.905224289991
209,205.280249241609
209.50,205.652290210828
210,206.022151250448
210.50,206.390150732612
211,206.756966179589
211.50,207.123091080129
212,207.489219317882
212.50,207.855809301513
213,208.223552528238
213.50,208.592876193525
214,208.964533705588
214.50,209.338770579828
215,209.716652601163
215.50,210.098267516738
216,210.484206688341
216.50,210.874206245114
217,211.268279263037
217.50,211.666431020494
218,212.068661517486
218.50,212.473412212959
219,212.880611041420
219.50,213.290906851884
220,213.704299644351
220.50,214.118835285052
221,214.532959924775
221.50,214.948398475814
222,215.365150938170
222.50,215.782008976051
223,216.195943431617
223.50,216.609299725416
224,217.022077857448
224.50,217.433538425281
225,217.836993162444
225.50,218.238384182577
226,218.637711485681
226.50,219.034975071754
227,219.430174940798
227.50,219.823311092812
228,220.214383527796
228.50,220.603392245751
229,220.990337246675
229.50,221.373347774385
230,221.749272311358
230.50,222.121924581379
231,222.491304584447
231.50,222.857412320563
232,223.220247789726
232.50,223.579810991936
233,223.936101927193
233.50,224.289120595498
234,224.638866996850
234.50,224.984150416015
235,225.324687492264
235.50,225.661501364493
236,225.994592032704
236.50,226.323959496897
237,226.649603757070
237.50,226.971524813224
238,227.289722665359
238.50,227.604197313476
239,227.914948757573
239.50,228.222316841883
240,228.526172334874
240.50,228.826394366166
241,229.122983745759
241.50,229.415939663653
242,229.705262389848
242.50,229.990951924345
243,230.273008267143
243.50,230.551431418242
244,230.826221377642
244.50,231.102102572260
245,231.375160386375
245.50,231.645023966514
246,231.911693312677
246.50,232.175168424863
247,232.435449303072
247.50,232.692535947306
248,232.946428357563
248.50,233.197126533844
249,233.444630476149
249.50,233.688940184477
250,233.930055658829
250.50,234.167976899204
251,234.402703905604
251.50,234.634236678027
252,234.862575216473
252.50,235.087719520944
253,235.309669591438
253.50,235.528425427956
254,235.743987030497
254.50,235.956354399062
255,236.165527533651
255.50,236.372767392624
256,236.578299819705
256.50,236.781260765516
257,236.981650230056
257.50,237.179468213325
258,237.374714715323
258.50,237.567389736050
259,237.757951450744
259.50,237.946452134270
260,238.132819220201
260.50,238.317012407773
261,238.499036882608
261.50,238.678902868645
262,238.856642479729
262.50,239.032274485539
263,239.205670172027
263.50,239.375746639767
264,239.538956719386
264.50,239.688082344736
265,239.812079413165
265.50,239.899638911699
266,239.938010699850
266.50,239.917492598708
267,239.832082955587
267.50,239.677424818674
268,239.452437355801
268.50,239.156902457768
269,238.792273853934
269.50,238.360361553298
270,237.865732358561
270.50,237.313672624886
271,236.700867906306
271.50,236.040521434002
272,235.330127311447
272.50,234.572874217001
273,233.781203753630
273.50,232.948442377410
274,232.087970406422
274.50,231.199382298605
275,230.285506260074
275.50,229.364044269074
276,228.421718018414
276.50,227.458527508093
277,226.474472738112
277.50,225.494728491143
278,224.505719745238
278.50,223.506089606568
279,222.495838075134
279.50,221.487387025584
280,220.482038345985
280.50,219.474027723767
281,218.463355158929
281.50,217.454007775224
282,216.453708785057
282.50,215.456016427598
283,214.460930702847
283.50,213.468691925036
284,212.491644780127
284.50,211.520271666459
285,210.554572584033
285.50,209.594547532848
286,208.640196512903
286.50,207.691519524200
287,206.748516566738
287.50,205.817229666130
288,204.896012114046
288.50,203.982615523624
289,203.077039894865
289.50,202.179285227769
290,201.289351522335
290.50,200.407238778564
291,199.534293677060
291.50,198.671980346659
292,197.818403348001
292.50,196.973562681086
293,196.137458345913
293.50,195.310090342483
294,194.491458670796
294.50,193.681572271877
295,192.880577192671
295.50,192.088357270545
296,191.304912505498
296.50,190.530242897531
297,189.764348446643
297.50,189.007229152834
298,188.258885016105
298.50,187.515998399043
299,186.780231188559
299.50,186.052657710134
300,185.333277963769
300.50,184.622091949464
301,183.919099667219
301.50,183.224301117033
302,182.537696298908
302.50,181.859285212842
303,181.189067858836
303.50,180.527044236890
304,179.873214347004
304.50,179.227578189178
305,178.590135763412
305.50,177.960033855077
306,177.331368914081
306.50,176.709938265361
307,176.095741908915
307.50,175.488779844743
308,174.889052072846
308.50,174.296558593224
309,173.711299405877
309.50,173.133274510804
310,172.562483908005
310.50,171.998927597481
311,171.442605579232
311.50,170.893517853258
312,170.351664419558
312.50,169.817045278132
313,169.285735071758
313.50,168.756930481207
314,168.234318731605
314.50,167.717899822952
315,167.207673755248
315.50,166.703640528493
316,166.205800142687
316.50,165.714152597831
317,165.228697893923
317.50,164.749436030964
318,164.276367008955
318.50,163.809490827894
319,163.348807487782
319.50,162.894316988620
320,162.446019330406
320.50,161.997406092602
321,161.553169062267
321.50,161.114166592167
322,160.680398682300
322.50,160.251865332666
323,159.828566543266
323.50,159.410502314100
324,158.997672645167
324.50,158.590077536468
325,158.187716988003
325.50,157.790590999771
326,157.398699571773
326.50,157.012042704008
327,156.630620396477
327.50,156.253732571246
328,155.879545719961
328.50,155.509783062552
329,155.144387082806
329.50,154.783300264510
330,154.426465091451
330.50,154.073824047414
331,153.725319616187
331.50,153.380894281556
332,153.040490527307
332.50,152.704050837228
333,152.371517695105
333.50,152.042833584723
334,151.717940989871
334.50,151.396782394333
335,151.079300281898
335.50,150.765437136351
336,150.455135441479
336.50,150.148337681069
337,149.844986338906
337.50,149.545023898779
338,149.248392844472
338.50,148.955035659773
339,148.664894828469
339.50,148.377912834345
340,148.094032161189
340.50,147.813195292787
341,147.535344712925
341.50,147.262821246040
342,146.993996519281
342.50,146.728313233293
343,146.465725073647
343.50,146.206185725918
344,145.949648875679
344.50,145.696068208502
345,145.445397409962
345.50,145.197590165630
346,144.952600161082
346.50,144.710381081889
347,144.470886613625
347.50,144.234070441863
348,143.999886252177
348.50,143.768287730139
349,143.539228561324
349.50,143.312662431303
350,143.088543025650
```

Q2_trend.csv 问题二中速度在 65 - 100 期间的各项指标：

速度 v	最大梯度	最小梯度	升温时间	高温时间	峰值温度	满足状态（值为 0 时满足所有条件）
100	2.66075563748363	-1.87085565037546	104	31.5	230.964103878399	4
99.5	2.65776880341623	-1.88163670843562	103.5	32.5	231.169025428978	4
99	2.65869034628008	-1.87835952717262	104	33	231.319912826064	4
98.5	2.65919905538394	-1.8885499589926	105	33	231.531683144739	4
98	2.65562323539477	-1.88481239234665	106	33.5	231.710606381595	4
97.5	2.65503197393994	-1.89147742934426	106.5	34	231.865710600343	4

97	2.66117420266956	-1.89146653853464	106.5	34.5	232.053139819539	4
96.5	2.65600198795241	-1.89195556044842	107	35	232.268710443765	4
96	2.65552183442631	-1.90742277465745	108.5	35.5	232.430589489425	4
95.5	2.66038868757671	-1.90121646795154	108	36	232.653714410912	4
95	2.65987918908976	-1.90024459023903	109	36	232.838418973255	4
94.5	2.65349215785902	-1.90821313804122	110	37	233.006952052909	4
94	2.65414636149772	-1.91048795655468	110.5	37	233.160272418033	4
93.5	2.66012817983449	-1.9155068919369	111.5	37.5	233.37961247172	4
93	2.65979975994034	-1.91755337501138	111.5	38	233.562140982947	4
92.5	2.65238150789556	-1.92015288617404	112	38.5	233.753430607882	4
92	2.65923816841675	-1.92547120843028	113	39	233.942614575932	4
91.5	2.65867916172081	-1.92625039417436	114	39.5	234.136778357602	4
91	2.65282773427427	-1.92918970242636	114.5	39.5	234.303383149184	4
90.5	2.65772574158466	-1.93152324621025	114.5	40.5	234.494813015292	4
90	2.65868883292468	-1.94061147183663	115.5	41	234.676876413777	4
89.5	2.65639209739744	-1.94457004983184	114.5	41	234.859265610596	4
89	2.65824902679344	-1.9434744262652	114	41.5	235.057604239147	4
88.5	2.65797613425181	-1.94401031132418	114	42.5	235.213058793514	4
88	2.65124687999462	-1.94341059667914	113	42.5	235.422547966588	4
87.5	2.65760182514799	-1.94874453727073	112.5	43	235.584960787567	4
87	2.65737158877937	-1.95085433953608	111.5	43.5	235.75957138642	4
86.5	2.65715938621582	-1.95717842490097	111	44	235.953996906857	4
86	2.65722100533449	-1.96123497281388	110.5	44.5	236.132480539222	4
85.5	2.65038092884012	-1.96239359360288	110	45	236.349427167967	4
85	2.65678550386473	-1.96947985179179	109	45.5	236.503073587778	4
84.5	2.65619501877686	-1.97188910849223	108.5	46.5	236.708387512337	4
84	2.65610833327467	-1.97129931865516	107.5	47	236.861041412528	4
83.5	2.65598297112281	-1.98083014978278	107	47.5	237.071099832338	4
83	2.6542615055164	-1.98417235480986	106	48	237.268834621323	4
82.5	2.6539790062246	-1.97881528443253	105.5	48	237.430882561359	4
82	2.65514032338419	-1.99097948904347	105	48.5	237.623280257177	4
81.5	2.65433268873717	-1.99137541061043	104	49	237.802960618447	4
81	2.65393460564731	-1.99530355399565	103	49.5	238.005194174692	4
80.5	2.6545715724683	-1.99063778602999	102.5	50.5	238.180994843783	4
80	2.6541251050198	-2.00139224499975	102	51	238.327001104926	4
79.5	2.65361626647289	-2.00281583618545	101	51	238.532807904623	4
79	2.65362490531383	-2.00346661653816	100	52	238.727644879032	4
78.5	2.6470442680979	-2.00918894014143	99.5	52.5	238.917183762904	4
78	2.65305915937533	-2.01136075704216	98.5	53	239.058065300855	4
77.5	2.65238581400205	-2.01676135518989	98	53.5	239.237169950994	4
77	2.65232671537085	-2.0183100825052	96.5	54.5	239.463667981939	4
76.5	2.65226898401836	-2.02022599526725	98	54.5	239.624537372912	4
76	2.65182356281007	-2.02944001294009	99	55.5	239.801371590336	4
75.5	2.65146702530963	-2.02485079682725	100	55.5	239.962010875423	4
75	2.65075900527394	-2.02795097944846	100.5	56.5	240.176170410848	0
74.5	2.65086836522198	-2.02762225090237	102	57	240.340078729426	0
74	2.64423042192203	-2.02752693965209	103	58	240.528806721058	0
73.5	2.64968876695534	-2.03170899870577	104	58.5	240.714382280157	0
73	2.64977469762835	-2.04147550603085	105	59	240.870321775594	0
72.5	2.64135889961838	-2.04196946945081	106	59.5	241.048549745962	0
72	2.64915019027943	-2.04403830596794	107.5	60.5	241.228962165068	0
71.5	2.64182811487234	-2.04724289615507	108.5	61	241.433991903503	0
71	2.64817486711297	-2.05310840509554	109.5	61.5	241.581860851543	0
70.5	2.64799680823772	-2.05436442219491	111	62	241.752199372053	0
70	2.64679985678517	-2.05803711063368	112	62.5	241.945037449869	0
69.5	2.64669221489159	-2.05938021667782	113.5	63	242.124759587951	0
69	2.64500593977606	-2.06029908957112	114.5	64	242.279820223351	0
68.5	2.6461555468087	-2.06502276604988	115.5	64.5	242.480417000403	0
68	2.63708441742405	-2.06893640333874	117	65.5	242.62399125071	0
67.5	2.6371902017771	-2.06425669033877	118.5	66	242.835441808913	0
67	2.63851700825342	-2.07818757797895	119.5	67	243.007183015687	0
66.5	2.64570378597209	-2.07046943478827	121	68	243.181677638676	2
66	2.63650092414996	-2.07328769571973	122.5	68	243.298787113446	2
65.5	2.6419810414464	-2.07892992586602	123.5	69	243.480000282951	2

表 5: 速度在 65 - 100 期间的各项指标

8.3 数据来源

- 热导率数据
  - https://en.wikipedia.org/wiki/Thermal_conductivity
- 密度数据
  - https://en.wikipedia.org/wiki/Density

参考文献

[1] 杨世铭,. 传热学 [M]. 第 4 版. 北京: 高等教育出版社, 2006

[2] 李灿, 高彦栋, 黄素逸. 热传导问题的 MATLAB 数值计算 [J]. 华中科技大学学报 (自然科学版),2002(09):91-93.

[3] 代莹, 肖冰. 热传导方程中傅氏积分与傅氏变换的应用 [J]. 新疆师范大学学报 (自然科学板),2019(09):5-19.

[4] 孙浩然, 吴宏聪, 张嘉墀, 李明奇. 高温环境下热防护服的温度分布研究 [J]. 实验科学与技术,2020,18(04):1-6.

[5] 李灿, 高彦栋, 黄素逸. 热传导问题的 MATLAB 数值计算 [J]. 华中科技大学学报 (自然科学版),2002(09):91-93.

[6] Solve 1-D parabolic and elliptic PDEs, Mathworks (https://www.mathworks.com/help/matlab/ref/pdepe.html/), 2020