

炉温曲线

摘要

本文研究建立回焊炉炉温曲线模型，求解相关问题，从机理模型角度研究炉内温度变化规律，对于指导制定焊接工艺流程具有重要意义。

问题一要求建立焊接区域温度变化模型，求解焊接区域过回焊炉炉时不同时空下的温度。设立简化坐标系，将泛炉内加热区温度与过炉速度作为参数分析。建立二维稳态传热方程，通过分析传热方式建立边界约束条件，使用集总参数分析得出目标函数 t_r 。结合二维传热方程、边界条件、目标函数建立温度变化模型。采用五点菱形差分法迭代求解，利用三分法遍历得到最优时间常数 $\tau_c=47.9567$ ，代入数据得到炉温曲线图，求解得到温区 3, 6, 7 中点和温区 8 结束点温度分别为 141.06°C , 172.78°C , 191.01°C 和 223.22°C 。

问题二要求根据设定的小温区温度求解最大过炉速度。本文对温度变化斜率、温度处于 $150\sim 190^{\circ}\text{C}$ 时长、超过 217°C 时长和峰值温度为 $240\sim 250^{\circ}\text{C}$ 等制程界限进行离散化表达。采用定步长遍历法，求得最大过炉速度为 $72.25\text{cm}/\text{min}$ 并得到炉温曲线。

问题三要求在超过 217°C 到峰值温度所覆盖面积最小的情况下，求解最优炉温曲线、过炉速度及各温区设定温度。以最小化阴影面积为目标函数，使用制程界限及题干焊接区域中心温度界限为约束，建立单目标优化模型。采用遗传算法迭代，求得最优炉温曲线，对应最小阴影面积指标为 461.14。温区 1~5, 6, 7, 8~9 温度分别为 179.59°C , 203.12°C , 244.49°C 和 265.00°C ，传送带过炉速度为 $99.98\text{cm}/\text{min}$ 。

问题四以最小化阴影面积 S 和对称度 E 为目标函数，确立制程界限与决策变量向量，建立多目标优化模型。采用多目标遗传算法 NSGA-II，运用 Matlab 迭代得到 Pareto 前沿图，求得最优对称度指标为 2.5，最小化面积指标为 469.24，对应温区 1~5, 6, 7, 8~9 温度分别为 167.72°C , 191.99°C , 228.72°C 和 265.00°C ，传送带速度为 $88.16\text{cm}/\text{min}$ 。

关键词：集总参数分析 五点菱形差分 三分法 多目标遗传算法 NSGA-II

1 问题背景及重述

1.1 问题背景及相关信息

回焊炉是一种主要用于生产集成电路板等电子产品的设备，它可以通过加热将电子元件自动焊接到电路板上。回焊炉内部设有不同小温区，在功能上分类可将其分成预热区，恒温区，回流区与冷却区 4 个大温区。工作时通过把电路板两侧搭上传送带进炉加热焊接。

当不同温区与传送带的温度设置完毕，可使用温度传感器测量某些位置上焊接区域中心的温度并形成炉温曲线（焊接区域中心温度曲线）。在电路板的焊接生产中，炉温曲线应该满足一定的制程界限。

在上述实验设定温度的基础上，各小温区设定温度可以进行 $\pm 10^{\circ}\text{C}$ 范围内的调整。调整时要求小温区 1~5 中的温度保持一致，小温区 8~9 中的温度保持一致，小温区 10~11 中的温度保持 25°C 。传送带的过炉速度调节范围为 $65\sim 100\text{ cm/min}$ 。

1.2 问题重述

1. 给定传送带过炉速度为 78 cm/min ，各温区温度的设定值分别为 173°C （小温区 1~5）、 198°C （小温区 6）、 230°C （小温区 7）和 257°C （小温区 8~9）。根据以上数据建立有关焊接区域温度变化规律的数学模型，同时给出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，并将每隔 0.5 s 焊接区域中心的温度存放在提供的 result.csv 中。

2. 设定各温区温度分别为 182°C （小温区 1~5）、 203°C （小温区 6）、 237°C （小温区 7）、 254°C （小温区 8~9），就这些数据得出允许的最大传送带过炉速度。

3. 已知在焊接过程中，焊接区域中心的温度高于 217°C 的时间不宜过长，峰值温度也不宜过高。在炉温曲线上表现为超过 217°C 到峰值温度所覆盖的面积最小。需要确定在该要求下的最优炉温曲线，各温区的设定温度和传送带的过炉速度，并给出相应的覆盖面积。

4. 为了使焊接完成的产品质量保持较高水准，除了考虑要求的制程界限，还应使峰值温度为中心线的两侧超过 217°C 的炉温曲线尽量对称。结合问题 3，进一步给出最优炉温曲线，各温区设定的温度及传送带过炉速度，并给出相应的指标值。

2 问题假设

- 假设回焊炉内部没有自发热源存在
- 假设上下区各对应加热温区参数均相同
- 假设各温区工作时不会发生自发热量传递
- 假设炉内原件不会因为持续工作发生损耗进而性能下降。
- 假设电路板材料不会因为经过不同温度区域而发生性质上的改变。

3 符号说明

符号	说明	量纲
τ	时间	s
τ_c	时间常数	°C
\mathbf{x}	决策变量	/
$t_I, t_{II}, t_{III}, t_{IV}$	温区温度	°C
v	传送带运行速度	cm/min
x_i, y_j	节点横纵坐标数值	1
u	泛炉内加热区温度	°C
L, H	泛炉内加热区长与宽	cm
t_0, t_f	初始温度，环境温度	°C
t	区域中心随传送带在不同位置温度	°C

4 问题分析

4.1 问题一的分析

本题要求建立焊接区域温度变化的模型，在给定过炉速度和设定各温区温度条件下，求解焊接区过炉时不同时空下的温度。首先界定泛炉内加热区范围（包含炉前区和炉内加热区），基于热力学原理，对不同区域传热规律进行探究。其二，在此基础上建立二维传热方程、边界条件，采用集总参数法建立目标函数，建立炉温数学模型，求解炉温曲线及相应温度。

4.2 问题二的分析

本题要求根据设定的小温区温度求解最大传送带过炉速度，其本质是一道优化问题。首先由于问题一求解为离散型解，延续前题对制程界限进行离散化表达；然后，基于题干所求为最大值，使用定步长遍历法算法，求解最大过炉速度。

4.3 问题三的分析

本题要求在超过 217°C 到峰值温度所覆盖面积最小的情况下求解最优炉温曲线、过炉速度及各温区设定温度。与问题二类似，本题亦是一道单目标优化问题。首先以制程界限及焊接区域中心温度界限为约束，建立以最小化阴影面积为目标的单目标优化模型；其次由于变量多、函数复杂等因素，迭代得到最优炉温曲线等目标。

4.4 问题四的分析

本题要求在峰值中心线的两侧超过 217°C 的炉温曲线尽量对称的情况，求解最优炉温曲线等问题，其本质是多目标优化问题。首先以最小化阴影面积 S 和对

称度 H 为目建立多目标优化模型。然后迭代求得最小化阴影面积、最优对称度，相应求得各小温区设定温度等目标。

5 模型的建立与求解

5.1 问题一模型建立与求解

已知在实际生产中回焊炉为一个三维物体 [1]，为了简化管理，本文首先做出截面示意图，并以炉前区域左端到传送带上投影点为原点，以传送带为 x 轴，以炉前左端纵向方向为 y 轴向建立直角坐标系如图 1：

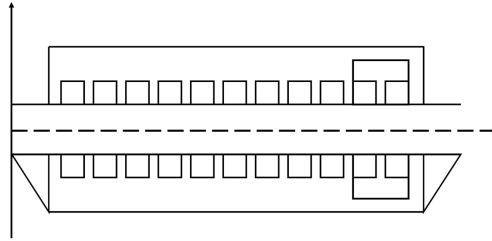


图 1: 回焊炉截面示意图

针对问题一，需要建立模型求解电路板焊接中心的温度变化，直观反映即为炉温曲线。本文确定泛炉内加热区域环境温度和传送带运行速度为炉温曲线的主要影响因素，对此分别进行分析。

5.1.1 泛炉内加热区分析

上下两排小温区中间的区域即为泛炉内加热区，该区域包括炉前区和炉内加热区，其形状为矩形，四边界定义如下：上、下两长边（设定为 L_u, L_d ），分别以上、下两排小温区朝向炉膛内的边界为长边；左宽边（设定为 H_l ）为炉前区左边与上下两长边的截断，右宽边（设定为 H_r ）为小温区 9、10 间隙中间的上下连线，具体位置如下图 2 蓝线所界定的区域：

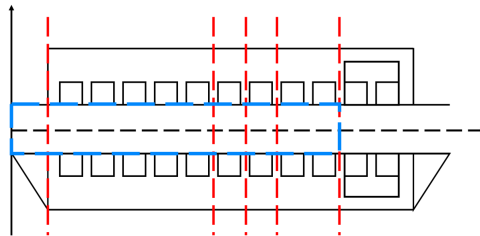


图 2: 中间区域示意图

对于该区域，由于回焊炉开始工作时其内部空气温度保持一个稳定值，区域

内无自发热源且区域，可以将该区域视为稳态区域。而所在平面用二维方式表示，即可根据直角坐标系建立二维稳态传热方程如下 [2]：

$$\frac{\partial u}{\partial \tau} = \alpha \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \quad (1)$$

区域为稳态，即 $\frac{\partial u}{\partial \tau} = 0$ ，因此有：

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \quad (2)$$

5.1.2 炉内加热区边界条件建立

除传热微分方程外，泛炉膛加热区与周边环境之间存在热量传导，这会影区域温度分布状态，须针对此情况进行分析。

区域边界包含两条宽边 H_l, H_r 与两条长边 L_u, L_d ，现在分别进行解释。其中 H_l 与坐标系 y 轴重合，原因在于炉前区域温度受到临近的小温区一影响无法到达相对稳态。其外部接触 25° 的室内环境，内部接触炉前区域。对于 H_r 由于冷却区（即小温区 10, 11）温度固定在 25° 与室温相同，选择小温区 9 与小温区 10 的边界作为该宽边。其外部接触 25° 冷却区环境，内部接触 257° 小温区 9。

对上下两长边 L_u, L_d ，其由不同类型边界组成，包括炉前区域边、1~9 小温区边以及小温区间隙边。这些边对应温度变化也不尽相同：1~9 小温区边，其温度与对应小温区相同；1~5 小温区间隙温度恒定，8~9 小温区间隙温度恒定；炉前区域，5,6 小温区间隙，6,7 小温区间隙和 7,8 小温区间隙，其共同特征为两端温度不等，需要通过二维稳态传热方程求解。由这些间隙边水平可知 $\frac{\partial^2 u}{\partial y^2} = 0$ ，所以 u 是一个关于横坐标 x 的线性函数。具体长边温度随位置横坐标变化与中心线对比图如图 3：

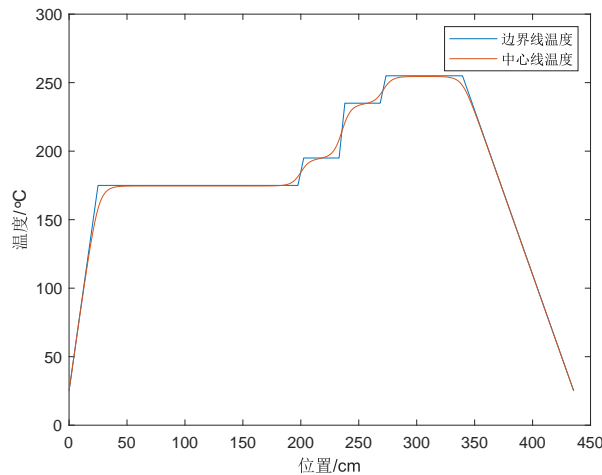


图 3: 边界条件与中心线温度对比图

现对边界条件进行具体的数学表述，记温度表达式为 $u(x, y)$ ，其中 x, y 分别代表位置横纵坐标。由上文所述 H, L 条件有：

$$\begin{cases} u\left(x, -\frac{d}{2}\right) = u\left(x, \frac{d}{2}\right) = u_b(x) \\ u(0, y) = u(l, y) = 25^\circ\text{C} \end{cases} \quad (3)$$

5.1.3 传送带速度分析

前两节分析了泛炉内加热区环境温度，现在分析传送带速度对于炉温曲线的影响。

已知传送带过炉速度恒定，它带动挂在上面需要焊接的电路板匀速运动，即焊接中心沿传送带依次经过不同温度区域，所以可视其为非稳态过程。对此使用集总参数方法 [2]，其本质在于将物体近似处理成为一个质点，忽略其实际体积，质量，且在同一时刻只有一个温度值。对应其解的形式为：

$$t = f(\tau) \quad (4)$$

本文以能量平衡作为着手点，将能量平衡运用到物体冷却过程中：

$$-\rho c V \frac{dt}{d\tau} = h A (t - t_f) \quad (5)$$

其中 ρ 代表物体密度， c 代表比热容， V, A 分别代表表面积和体积， t_f 为环境温度， h 为对流换热系数。本式说明空气吸收热流量等于物体降温释放的热流量。令过余温度 $\theta = t - t_f$ ，上式变为：

$$-\rho c V \frac{d\theta}{d\tau} = h A \theta \quad (6)$$

对应的初始条件可以表达成：

$$\theta(0) = t_0 - t_f = \theta_0 \quad (7)$$

使用分离变量法，对式6与式7求其定解：

$$\int_{\theta_0}^{\theta} \frac{d\theta}{\theta} = - \int_0^{\tau} \frac{h A}{\rho c V} d\tau \quad (8)$$

对以上微分式积分并且表达为指数函数：

$$\frac{\theta}{\theta_0} = \frac{t - t_f}{t_0 - t_f} = e^{-\frac{h A}{\rho c V} \tau} \quad (9)$$

最后得出 t 的表达式为：

$$t = (t_0 - t_f) e^{-\frac{hA}{\rho c V} \tau} + t_f \quad (10)$$

现将 $\frac{\rho c V}{hA}$ 记为时间常数 τ_c ， t 的表达式变为：

$$t = (t_0 - t_f) e^{-\frac{\tau}{\tau_c}} + t_f \quad (11)$$

由于本题所求为数值解，在此给出一种温度变化函数 t_τ 的数值解微分求法。取 $\tau = 0$ 附近微小变量 Δt ，可得：

$$\Delta t = t'_\tau \Big|_{\tau=0} \quad (12)$$

由于 Δt 微小，可近似边界温度 t_f 在保持不变。代入 t 表达式11，得到下式：

$$\Delta t = (t - t_f(v\tau)) \left(-\frac{1}{\tau_c} \right) \Delta \tau \quad (13)$$

化为微分式为：

$$dt = (t - t_f(v\tau)) \left(-\frac{1}{\tau_c} \right) d\tau \quad (14)$$

代具体点入该式积分即可得到对应 t_τ 。

5.1.4 问题一模型建立

所需目标函数即焊接区域中心温度变化函数 $t(\tau)$ 。约束条件即二维传热稳态方程和边界条件。有：

$$s.t. \begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0 \\ u\left(x, -\frac{d}{2}\right) = u\left(x, \frac{d}{2}\right) = u_b(x) \\ u(0, y) = u(l, y) = 25^\circ\text{C} \\ t = (t - t_f(v\tau)) \left(-\frac{1}{\tau_c} \right) d\tau \end{cases} \quad (15)$$

由以上条件可求出区域中心温度变化。

5.1.5 问题一模型求解

二维稳态导热式1为二维椭圆形方程中的拉普拉斯方程，由于约束条件方程较为复杂，因此不解析求方程精确解，而是求数值解 [3]。求解时的方法包括五点菱形差分，九点紧差分等。

本问题采用五点菱形差分的方法处理，思路在于结合二维稳态导热方程与边

界条件，将原来连续区域划分成为离散性区域，使方程仅在离散点成立，同时将方程中微商用差商代替得到数值解，具有高效性精确性等优点，具体步骤如下：

1. 分割矩形区域，在 x 方向进行步长为 Δx 等距剖分，分为 m 份。即 $\Delta x = l/m$ ， l 为长度 L 具体值，同时在 y 方向进行步长为 Δy 等距剖分，分为 n 份。即 $\Delta y = d/n$ ， d 为中间区域宽度 H 具体值。

可知对应节点横坐标纵坐标分别为：

$$\begin{cases} x_i = i \cdot \Delta x & i = 0, 1, 2 \dots m \\ y_j = -\frac{d}{2} + j \cdot \Delta y & j = 0, 1, 2 \dots n \end{cases} \quad (16)$$

在此基础上可以把该中间区域分为 mn 个小矩形，并得到节点 (x_i, y_j) 。

2. 将二位稳态导热方程弱化，使之只在离散的节点成立，对应下式：

$$\begin{cases} \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) \Big|_{(x_i, y_j)} = 0 & (x_i, y_j) \in \Omega^o \\ u(x, -\frac{d}{2}) = u(x, \frac{d}{2}) = u_b(x) \\ u(0, y) = u(l, y) = 25^\circ\text{C} \end{cases} \quad (17)$$

3. 再用差商近似取代微商，创建数值格式的方程如下，这里将 $\Delta x^2, \Delta y^2$ 分别记为 h^2, k^2 ：

$$\begin{cases} \left. \frac{\partial^2 u}{\partial x^2} \right|_{(x_i, y_j)} = \frac{u(x_{i-1}, y_j) - 2u(x_i, y_j) + u(x_{i+1}, y_j))}{h^2} + O(h^2) \\ \left. \frac{\partial^2 u}{\partial y^2} \right|_{(x_i, y_j)} = \frac{u(x_i, y_{j-1}) - 2u(x_i, y_j) + u(x_i, y_{j+1}))}{k^2} + O(k^2) \end{cases} \quad (18)$$

代入离散节点处方程，用数值解 $u_{i,j}$ 代替精确解 $u(x_i, y_j)$ ，取四个边中点计算，忽略高阶小项可得到数值格式：

$$\begin{cases} \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h^2} + \frac{u_{i,j-1} - 2u_{i,j} + u_{i,j+1}}{k^2} = 0 \\ 1 \leq i \leq m-1, 1 \leq j \leq n-1 \\ u_{i,j} = t(x_a, y_b) \quad a = 0, m \quad b = \pm \frac{d}{2} \end{cases} \quad (19)$$

该式的局部截断误差为 $O(h^2 + k^2)$ 。上式整理得：

$$\begin{cases} h^2(u_{i-1,j} + u_{i+1,j}) + k^2(u_{i,j-1} + u_{i,j+1}) - 2(h^2 + k^2)u_{i,j} = 0 \\ 1 \leq i \leq m-1, 1 \leq j \leq n-1 \\ u_{i,j} = u(x_a, y_b) \quad a = 0, m \quad b = \pm \frac{d}{2} \end{cases} \quad (20)$$

4. 这是一个高阶稀疏矩阵方程。理论上可以直接解该方程得到仿真结果，由于阶数过高本方程选用迭代方法求解。在此使用收敛速度较快的 SOR 迭代：

$$u_{i,j}^{(k+1)} = (1 - \omega)u_{i,j}^{(k)} + \omega \frac{h^2(u_{i-1,j}^{(k+1)} + u_{i+1,j}^{(k)}) + k^2(u_{i,j-1}^{(k+1)} + u_{i,j+1}^{(k)})}{2(h^2 + k^2)} \quad (21)$$

其中加括号的上标 k 代表迭代次数， $0 < \omega < 2$ 。

5. 在迭代一定次数或误差小于一定阈值时，迭代终止，将最终得到的 u^{k+1} 作为求解的 t 的结果。

取 $t_f(x) = u(x, 0) = u_{i,[n/2]}$ ，将 $y = 0$ 的线上作为焊接材料在回焊炉中移动时的环境。这样取到的 $t_f(x)$ 也是离散的点，将其记为 $(t_f)_i = t_f(x_i)$ 。

同样，对焊接原件进入回焊炉这一过程也进行离散。在时刻 τ 取短时间段 $\Delta\tau$ ，由式30，其对应一个 $\Delta t = t_i - t_{i-1}$ ，其中 t_i 为炉温曲线 $t(\tau)$ 的离散形式，可以得到如下的 t_i 的计算式：

$$t_i = t_{i-1} + (t_{i-1} - (t_f)_j) \left(-\frac{1}{\tau_c}\right) \Delta\tau \quad (22)$$

其中 j 满足 $x_{j-1} < v\tau \leq x_j$ ，且 $t_0 = t(0) = 25$ 。可得炉温曲线的离散形式。

在其他因素已知情况下，需要 τ_c 的值来确定炉温曲线。由式11，可知 τ_c 是反映焊点温度对环境温度响应速度的物理量。由附件数据拟合得到最优 τ_c 。

首先在大范围内进行遍历，观察 τ_c 对拟合误差的影响。其中拟合误差使用方均根误差进行计算。得到下面的误差与 τ_c 取值的关系图如图 4。

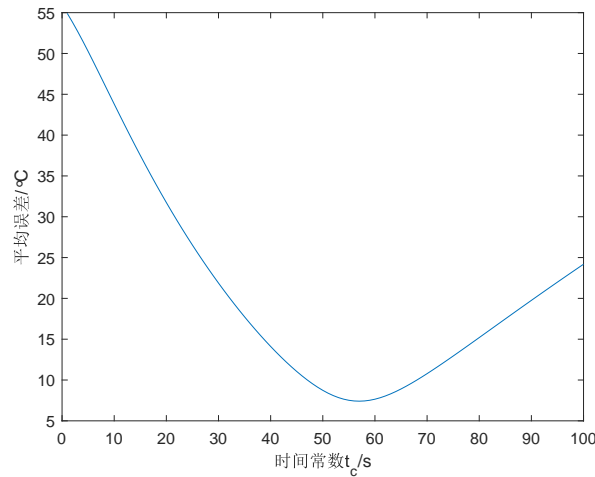


图 4: 时间常数 τ_c 与误差关系图

由图可见，随着 τ_c 的增大误差先升后降，且存在一个满足制程条件最小值位于 45-65s 之间。对于这种存在唯一最值的函数求最值的问题，使用三分法进行求解。其示意图如下图 5，其中 $err(\tau_0)$ 代表在 $\tau_c = \tau_0$ 时的方均根误差：

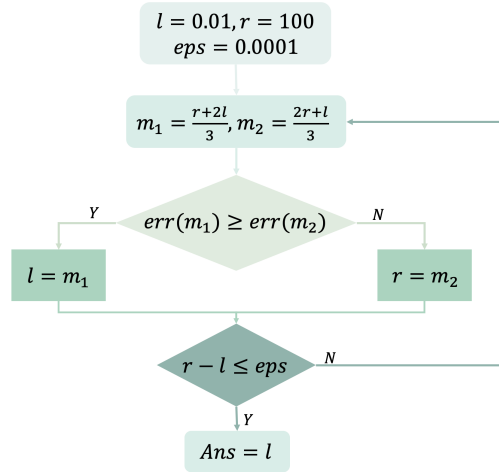


图 5: 三分法示意图

1. 确定搜索区间左右边界 $l = 0.01, r = 100$ ，确定终止阈值 $eps = 0.0001$ ；
2. 取区间的两个三等分点为 m_1, m_2 ，其中 m_1 在 m_2 的左边。
3. $err(m_1)$ 与 $err(m_2)$ 比大小。若 $err(m_1) \geq err(m_2)$ ，则取左边界 $l = m_1$ ；若 $err(m_1) < err(m_2)$ ，则取右边界 $r = m_2$ 。
4. 判断 $r - l$ 是否小于或等于 eps 值，若是，取 l 值为解，结束算法。若为否，则返回第二步。

通过该方法算出最优时间常数 $\tau_c = 47.9567s$ ，该常数值下炉温曲线与附件对比如图 6：

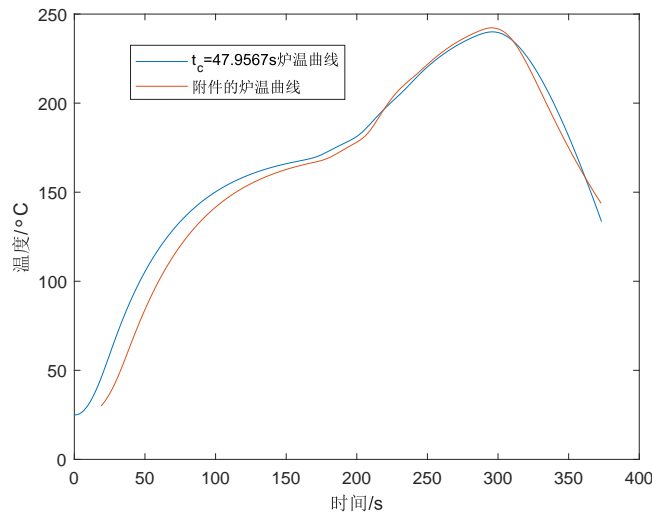


图 6: 炉温曲线与附件对比图

可见本文所得时间常数对应炉温曲线与附件数据误差较小。

下面代入本问题所给过炉速度与温区温度等数据，得到焊接区域中心温度变化如图 7：

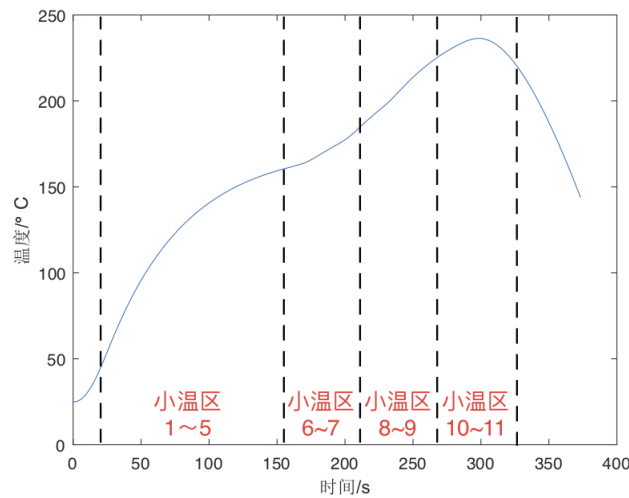


图 7: 焊接区域中心温度分布图

并给出焊接区域中心在传送带上处于与小温区 3、6、7 中点及小温区 8 结束点时的温度，对应表 1：

表 1: 小温区中点温度表	
位置	温度 (°C)
小温区 3 中点	141.06
小温区 6 中点	172.78
小温区 7 中点	191.01
小温区 8 结束处	223.22

可见焊接区域中心在经过不同小温区时温度变化的趋势。根据制程界限约束条件分析，温度从 1 至 10 温区以不同的趋势上升，最大斜率为 2.29，符合上升斜率在 0~3 的范围，温度在温区 10 与 11 交界处开始下降，其下降斜率最低值-2.38，符合下降斜率在-3~0 的范围。

温度上升时从温区 4，5 附近至温区 7 中点温度在 150~190°C，下降时温度在 150~190°C 的时段位于炉后区域内，其总和经过计算为 92.03s。温度处于 217°C 以上时的区域对应小温区 9~11 左右，其时长为 69.90 秒，两时间长度均在制程条件约束范围内。同时该温度极大值为 240.17°C，符合制程界限内。

5.2 问题二的求解

问题二要求根据一组数据，在制程界限的约束下，求解传送带允许的最大过炉速度。

5.2.1 约束条件的确定

本题约束条件即为制程界限，由于之前所求解为离散型解，约束条件亦亦进行离散化表示。

取一段关于时间的微元 $\Delta\tau$ ，每个微元对应一个离散型分布时间节点，现对制程界限进行解释。

首先关注温度变化的斜率，制程界限要求温度变化的斜率在 $[-3, 3]$ 范围内，对应斜率表达如下：

$$-3 \leq \frac{t_{i+1} - t_i}{\Delta\tau} \leq 3 \quad (23)$$

其中 i 作为下标对应取值范围为 $0 \leq i \leq [\frac{L}{v\Delta\tau}]$ 。（后面式中 i 的取值范围保持恒定）。

对于温度上升过程中，温度处于 $150 \sim 190^\circ\text{C}$ 的时长，有要求在 $60 \sim 120\text{s}$ 范围，表达式如下：

$$60 \leq \sum_{t_i \in [150, 190], t_{i+1} \geq t_i} \Delta\tau \leq 120 \quad (24)$$

对于温度超过 217°C 时间规定在 $40 \sim 90\text{s}$ 范围的条件，表达式如下：

$$40 \leq \sum_{t_i > 217} \Delta\tau \leq 90 \quad (25)$$

对于峰值温度要求在 $240 \sim 250^\circ\text{C}$ 的条件，表达式如下：

$$240 \leq \max_{i=1}^n \{t_i\} \leq 250 \quad (26)$$

将以上条件汇总，有：

$$s.t. \begin{cases} -3 \leq \frac{t_{i+1} - t_i}{\Delta\tau} \leq 3 \\ 60 \leq \sum_{\substack{t_i \in [150, 190], \\ t_{i+1} \geq t_i}} \Delta\tau \leq 120 \\ 40 \leq \sum_{t_i > 217} \Delta\tau \leq 90 \\ 240 \leq \max_{i=1}^n \{t_i\} \leq 250 \end{cases} \quad (27)$$

5.2.2 最大过炉速度求解

本问使用定步长遍历进行求解。遍历之前先做出对于各约束条件在速度为 65~100 时取值直线，如图 8：

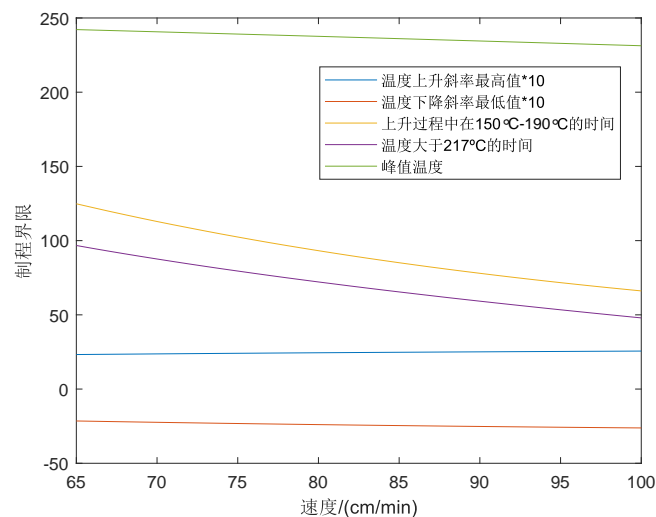


图 8: 不同下各约束条件制程界限图

进行遍历，由于所求为最大传送带过炉速度，先从 100 往 65 以定步长为 1 进行遍历，发现在 70 附近出现第一个符合条件值，即最大速度。

再进行步长为 0.01 的遍历，将取值范围精确在 72~73 内，最终确定 72.25 是符合条件最大速度，其对应炉温曲线如图 9：

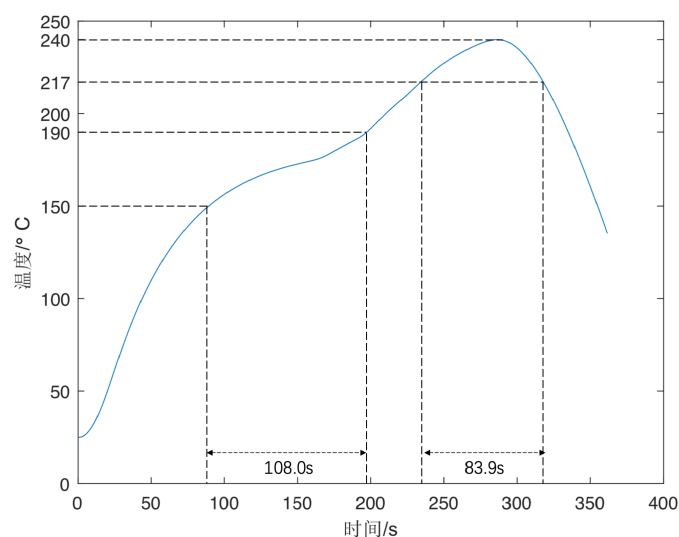


图 9: 最大速度对应炉温曲线

温度上升区斜率和下降区斜率最大值分别为 2.39 和 2.28。由图可知，温度上

升过程中在 150~190°C 的时间为 108.0s, 温度在超过 217°C 以上的时间为 83.9s, 且最大温度为 240°C。均在制程界限内。

5.3 问题三的求解

问题三重点在于求取最优炉温曲线, 其满足的前提为使超过 217°C 到峰值温度所覆盖的面积最小, 并需要给出最小值的具体值, 同时要求温区的设定温度和传送带的过炉速度。

5.3.1 问题三模型的建立

首先确定目标函数为阴影面积最小值。其影响因素可从水平与竖直两方向看。水平方向代表焊接区域中心超过 217°C 时长, 竖直方向代表最大峰值温度。现在就这两个影响因素给出目标函数计算方法。

同问题一问题二, 由于该区域解为离散型数值, 把阴影区域沿时间 τ 对应 x 轴分成宽度均匀的一个个小矩形, 每个矩形内部均为连续型且宽度为 $\Delta\tau$, 长度为 $t_i(\tau) - 217$ 。面积有如下表达:

$$\min S = \sum_i (t_i - 217) \Delta\tau \quad 217 \leq t_i \leq \max \{t_i\} \quad (28)$$

约束条件使用第二问所设即可, 有:

$$\text{s.t.} \begin{cases} -3 \leq \frac{t_{i+1} - t_i}{\Delta\tau} \leq 3 \\ 60 \leq \sum_{\substack{t_i \in [150, 190], \\ t_{i+1} \geq t_i}} \Delta\tau \leq 120 \\ 40 \leq \sum_{t_i > 217} \Delta\tau \leq 90 \\ 240 \leq \max_{i=1}^n \{t_i\} \leq 250 \end{cases} \quad (29)$$

5.3.2 算法使用和模型求解

本问的模型为单目标优化模型。但由于变量较多, 约束函数以及目标函数较复杂, 这里使用遗传算法进行求解。其决策变量为四个大温区温度以及传送带运行速度。设向量 $\mathbf{x} = (t_I, t_{II}, t_{III}, t_{IV}, v)$ 为决策变量向量, 亦即种群个体, 由题意, \mathbf{x} 需要满足下式的边界条件:

$$(165, 185, 225, 245, 65) \leq \mathbf{x} \leq (185, 205, 245, 265, 100) \quad (30)$$

其中的小于等于号表示向量元素在每一维上的关系。

当决策变量 \mathbf{x} 给定后, 就可以通过问题一建立的模型来求解得到相应的炉温曲线, 进而结合式29来判断其是否满足制程界限要求。如果满足, 其所对应的目标函数 $S = S(\mathbf{x})$ 也就可以结合式28进行求解, 即算法中所用目标函数。

单目标遗传算法在本题中思路如下：以一个随机的初始种群开始迭代，在每一代中，对于可行解进行交叉和变异，根据约束条件选择较优可行解作为下一代种群，在迭代一定次数后终止算法，得到最优可行解。具体算法过程不再赘述，使用 Matlab 内置的优化工具箱中的 ga 函数进行求解。在本问题中设置种群大小为 100，迭代 500 代。

另外，在本问题以及问题四中，需要大量求解不同边界温度下的 t_f 值，而这在之前的模型中是需要对微分方程求数值解才能得到。这一过程实际是非常耗时的。注意到由于回焊炉的宽度 d 不大，可以用边界温度代替泛炉内加热区温度 t_f ，这一点的合理性可以从图 3 中看出。

经过遗传算法迭代，产生最小阴影面积 S 为 $461.14^{\circ}\text{C}\cdot\text{s}$ ，对应的决策变量取值如下表 2：

表 2: 问题三变量取值表				
t_I	t_{II}	t_{III}	t_{IV}	v
179.59°C	203.12°C	244.49°C	265.00°C	$99.98\text{cm}/\text{min}$

对应的炉温曲线如图 10 所示：

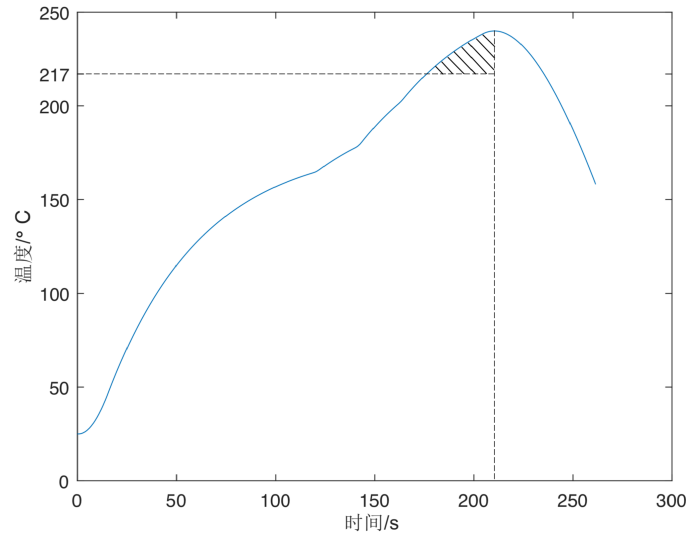


图 10: 最小阴影面积与炉温曲线示意图

5.3.3 模型制约界限验证

如图 10，温度上升区斜率和下降区斜率最大值分别为 2.80 和 2.74。温度上升过程中在 $150\sim 190^{\circ}\text{C}$ 的时间为 63.63s，温度在超过 217°C 以上的时间为 59.27s，且最大温度为 240°C 。均在制程界限内。由此可知，使最大温度尽可能低，过温区速度尽可能快（时间尽可能短）都是达到最低面积的必要条件。

5.4 问题四的求解

问题四希望在满足问题三最小阴影面积的基础上，使以峰值温度为中心线两侧超过 217°C 的炉温曲线形状尽量对称。求取该约束下最优炉温曲线，各温区设定的温度和传送带过炉速度。

5.4.1 目标函数和约束条件确定

与第三问类似，决策变量为四个大温区温度和传送带速度，仍用向量 \mathbf{x} 表示，约束条件同式29。

关于目标函数，我们仍有如式28的目标函数 $S = S(\mathbf{x})$ 。另外，对于问题四中要求的尽可能使曲线对称，我们采用如下的指标衡量其对称度：

$$E = \sqrt{\frac{\sum_i (t_{m+i} - t_{m-i})^2}{\sum_i 1}}, \quad t_{m-i}, t_{m+i} > 217 \quad (31)$$

其中 $t_m = \max \{t_i\}$ 为炉温曲线的最高点。上式从直观上可以理解为，取最高点两侧对称的取样点之间的差距平均值来衡量对称度。这样定义的 E ，实际上也可以通过炉温曲线 t 间接的表示成 $E = E(\mathbf{x})$ 的形式。综合上述说明，问题四的优化模型如下：

$$\left\{ \begin{array}{l} \min \quad S = \sum_i (t_i - 217) \Delta\tau, \quad 217 \leq t_i \leq \max \{t_i\} \\ \min \quad E = \sqrt{\frac{\sum_i (t_{m+i} - t_{m-i})^2}{\sum_i 1}}, \quad t_{m-i}, t_{m+i} > 217 \\ \text{s.t.} \quad -3 \leq \frac{t_{i+1} - t_i}{\Delta\tau} \leq 3 \\ \\ 60 \leq \sum_{\substack{t_i \in [150, 190], \\ t_{i+1} \geq t_i}} \Delta\tau \leq 120 \\ 40 \leq \sum_{t_i > 217} \Delta\tau \leq 90 \\ 240 \leq \max_{i=1}^n \{t_i\} \leq 250 \end{array} \right. \quad (32)$$

其中的 t_i 由 \mathbf{x} 确定，而 \mathbf{x} 也要满足式30的约束条件。

5.4.2 多目标优化模型的求解

可以注意到，问题四是一个多目标的优化模型。因此本文采用多目标遗传算法 NSGA-II 进行求解。同样，Matlab 中优化工具箱自带 gamultiobj 函数，拥有可以实现这一算法的功能。设置初始种群大小为 300，迭代 1000 次，最终得到图 11 所示的 Pareto 前沿图，同时生成了 Q4.xls 文件储存各点的信息，以供进一步选择。

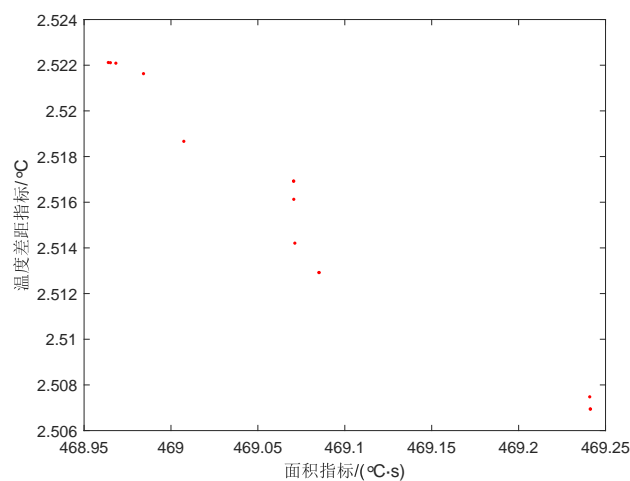


图 11: 问题四结果的 Pareto 前沿图

从图中可以看出，最优的对称度指标稳定在 2.5 左右，而面积指标稳定在 468 左右。本文选择图上最右下方的点为本问题的最终结果，而在实际生产中可以根据实际需要选择不同的 Pareto 前沿点分析。这一点对应的 x 取值如表 3:

表 3: 问题四变量取值表

t_I	t_{II}	t_{III}	t_{IV}	v
167.72°C	191.99°C	228.72°C	265.00°C	88.16cm/min

画出的炉温曲线如图 12 所示。

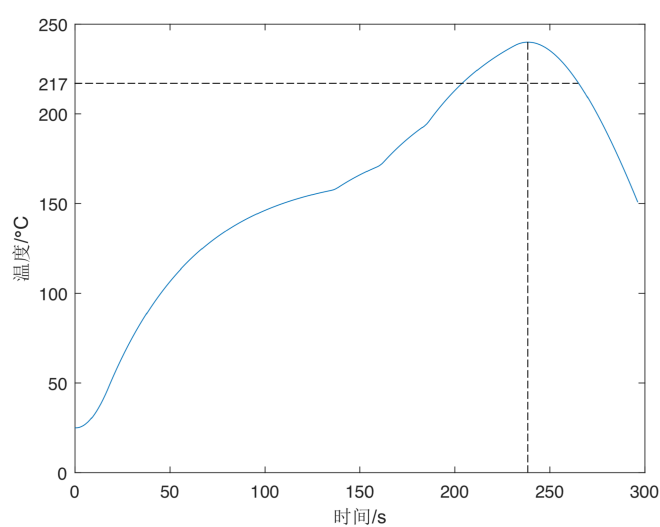


图 12: 问题四结果炉温曲线图

此时超过 217°C 区域面积为 $469.24^{\circ}\text{C}\cdot\text{s}$ ，两侧对称位置的温度平均差距为 2.51°C 。

同样本文对制程界限做检测，如上图，温度上升区斜率和下降区斜率最大值分别为 2.54 和 2.59。温度上升过程中在 $150\sim 190^{\circ}\text{C}$ 的时间为 70.44s，温度在超过 217°C 以上的时间为 61.42s，且最大温度为 240.0°C 。均在制程界限内。

6 模型的灵敏度分析及误差分析

分析对模型精确性的影响因素，主要有以下两点。

1. 模型使用的是结合附件拟合的时间常数 τ_c ，可能拟合的结果不够精确。同时，由于 τ_c 本身与焊接原件和周围空气的相对速度可能有关，因此在锅炉速度变化的前提下其值有可能发生改变。因此，有必要进行炉温曲线对于时间常数的灵敏度分析。

2. 三四问的模型均使用回焊炉的边界温度代替其中心的实际温度，这样也可能带来误差。因此有必要对其进行分析。

6.1 对时间常数 τ_c 进行的灵敏度分析

取 $t_c = 43, 48, 53$ 三个不同的数值，在附件所对应的温度、速度情况下，做出对应的炉温曲线图如图 13。

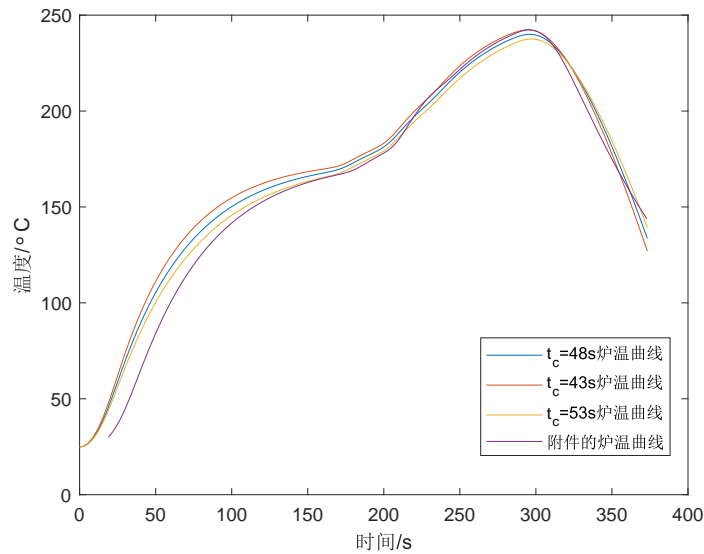


图 13: 不同 τ_c 下的炉温曲线

可以看出，在 200s 后四条曲线基本重合，误差非常小。而在初始的 200s 内，三个不同 τ_c 之间的差距也并不大。因此，可以认为炉温曲线对于时间常数的灵敏度并不高， τ_c 的值在 $\pm 10\text{s}$ 内变化时曲线误差并不大。

6.2 对近似炉心温度 t_f 的误差分析

仍在附件对应的条件下进行误差分析。原仿真的结果图与用边界温度代替炉心温度后的仿真结果图如图 14：

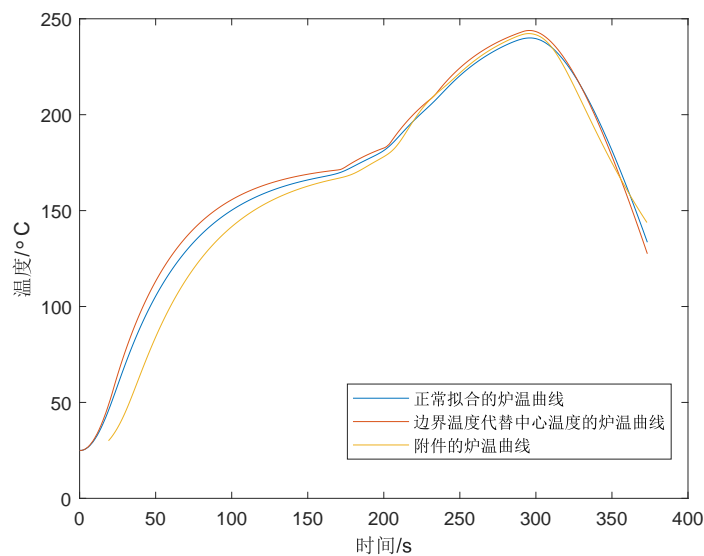


图 14: 不同近似方法温度对比图

其结果与上图类似，在 200s 后的部分几乎没有误差，而在 200s 前的误差也在可接受的范围内。因此，为了提高运行速度而做出的这一代替是可行的。

参考文献

- [1] 蔡海涛, 李威, 王浩. 回流焊接温度曲线控制研究 [J]. 微处理机, 2008, 29(005):24-26.
- [2] 赵镇南. 传热学 [M]. 第三版, 北京: 高等教育出版社, (2019)
- [3] 华东英, 李详贵. 微分方程的数值解法与程序实现 [M]. 第三版, 北京: 电子工业出版社, (2016)

附录

支撑材料目录

文件	说明
Q0.m	求解 τ_c 程序
Q1.m	问题一程序
Q2.m	问题二程序
Q3.m	问题三程序
Q4.m	问题四程序
ex.m	灵敏度分析程序
Q1-1.eps	图片
Q1-2.eps	图片
Q1-3.eps	图片
Q1-4.eps	图片
Q1-5.eps	图片
Q1-6.eps	图片
Q1-edge.eps	图片
Q2-1.eps	图片
Q2-2.eps	图片
Q3-shadow.eps	图片
Q4-2.eps	图片
Q4pareto.eps	图片
ex_2.eps	图片
ex_tc.eps	图片
result.csv	问题一结果
Q4.xlsx	问题四中间过程

求解 τ_c

```

xls = xlsread('附件.xlsx', 'A2:B710');
tim = xls(:,1);
t = xls(:,2);
v = 7/6*0.01;
len = 435.5e-2;
% plot(tim,t);

```

```

KK = 273.15;

ll = 15e-2;    % 宽度
h = 0.5 / 100;
k = 0.1 / 100;
x = 0:h:(len);
y = 0:k:(ll);
%% 炉心温度
temp = zeros(1,length(x));
for i = 1:length(x)
temp(i) = get_t(x(i));
end
plot(x*100,temp)
hold on
iteration = 50000;
n = size(x,2);
m = size(y,2);
z_p = ones(n,m) * 0;
z_c = ones(n,m) * 0;
A = [h^2,h^2,k^2,k^2];
Eps = 1e-3;
w = 1.7;
for int = 1:iteration
for j = 1:m
z_c(1,j) = 25 + KK;
z_c(n,j) = 25 + KK;
end
for i = 1:n
z_c(i,m) = get_t(x(i)) + KK;
z_c(i,1) = get_t(x(i)) + KK;
end
for i = 2:n-1
for j = 2:m-1
z_c(i,j) = (1-w)*z_p(i,j) +
    w*(dot(A,[z_c(i,j-1),z_p(i,j+1),z_c(i-1,j),z_p(i+1,j)])))/2/(h^2+k^2);
end
end
err = sqrt(sum((z_p - z_c).^2)/length(z_c));
if (err<Eps)

```

```

break;
end
z_p = z_c;
end
z_p = z_p - KK;
% [x,y]=meshgrid(x,y);
% mesh(x',y',z_p);
tt = ll/2 / k;
xlswrite('z0.xls',z_p(:,tt));
plot(x*100,z_p(:,tt))
legend('边界线温度','中心线温度')
xlabel('位置/cm')
ylabel('温度/\circ C')
%% 大范围搜索t_c
step = 0.01;
t_e = xlsread('z0.xls') + KK;
hold on
ti = 0:step:(len/v);    % 时间
t_c = 1:1:100;
t_0 = zeros(1,length(ti));
err = zeros(1,length(t_c));
for kkkk = 1:length(t_c)
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-t_c(kkkk));
end
ccc = 1;
err(kkkk) = 0;
for i = 1:length(ti)
if(ccc>length(t))
break;
end
if(abs(ti(i) - tim(ccc))<step/2)
err(kkkk) = err(kkkk) + (t_0(i)-KK - t(ccc))^2;
ccc = ccc + 1;
end
end
err(kkkk) = err(kkkk) / length(t);

```

```

err(kkkk) = sqrt(err(kkkk));
end
plot(t_c,err)
xlabel('时间常数t_c/s');
ylabel('平均误差/\circ C');
%% 三分法
l = 0.01;r = 100;
step = 0.01;
t_e = xlsread('z0.xls') + KK;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
while(r-l>=1e-6)
m1 = (2*l+r)/3;
m2 = (l+2*r)/3;
% e1
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-m1);
end
ccc = 1;
e1 = 0;
for i = 1:length(ti)
if(ccc>length(t))
break;
end
if(abs(ti(i) - tim(ccc))<step/2)
e1 = e1 + (t_0(i)-KK - t(ccc))^2;
ccc = ccc + 1;
end
end
e1 = e1 / length(t);
e1 = sqrt(e1);
re = check(t_0,ti);
if(re(5)-KK<240)
e1 = 1e7;
end
% e2
t_0(1) = 25 + KK;

```

```

for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-m2);
end
ccc = 1;
e2 = 0;
for i = 1:length(ti)
if (ccc>length(t))
break;
end
if (abs(ti(i) - tim(ccc))<step/2)
e2 = e2 + (t_0(i)-KK - t(ccc))^2;
ccc = ccc + 1;
end
end
e2 = e2 / length(t);
e2 = sqrt(e2);
re = check(t_0,ti);
if (re(5)-KK<240)
e2 = 1e7;
end
if (e1>e2)
l = m1;
else
r = m2;
end
end
l
r
e1
e2
%% 最优t_c = 47.9567画图(err = 9.55)
tc_real = 47.9567;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);

```



```

end
plot(ti,t_0-KK);
hold on
plot(tim,t);
legend('t_c=47.9567s炉温曲线','附件的炉温曲线');
xlabel('时间/s')
ylabel('温度/\circ C')

%% 边界温度分布函数
function temp = get_t(x)
to = 25;
t1 = 175;
t2 = 195;
t3 = 235;
t4 = 255;
x = x * 100;
len = 435.5;
if(x<=25)
temp = to + (t1 - to)/25 * x;
elseif (x<=25 + 30.5*5 + 5*4)
temp = t1;
elseif (x<=25 + 30.5*5 + 5*5)
temp = t1 + (t2-t1)/5 * (x - (25 + 30.5*5 + 5*4));
elseif (x<=25 + 30.5*6 + 5*5)
temp = t2;
elseif (x<=25 + 30.5*6 + 5*6)
temp = t2 + (t3-t2)/5 * (x - (25 + 30.5*6 + 5*5));
elseif (x<=25 + 30.5*7 + 5*6)
temp = t3;
elseif (x<=25 + 30.5*7 + 5*7)
temp = t3 + (t4-t3)/5 * (x - (25 + 30.5*7 + 5*6));
elseif (x<=25 + 30.5*9 + 5*8)
temp = t4;
else
temp = (len - x)/(len - (25 + 30.5*9 + 5*8))*(t4 - to) + to;
end
end

%% 时间对应环境温度

```

```

function idx = get_t2(tim,v,x)
l = 1;r = length(x);
while(r-l>=5)
mid = floor((l+r)/2);
if (tim*v<x(mid))
r = mid;
else
l = mid;
end
end
for i = l:r
if (tim*v>=x(i))
idx = i;
break;
end
end
end
%% 炉温曲线检验
function res = check(t, ti)
delta = ti(2) - ti(1);
res = [0,1e9,0,0,0];
for i = 2:length(t)
if (i>1)
res(1) = max(res(1),(t(i)-t(i-1))/delta);
res(2) = min(res(2),(t(i)-t(i-1))/delta);
if (t(i)-t(i-1)>=0 && t(i)<=190 && t(i)>=150)
res(3) = res(3) + delta;
end
end
if (t(i)>217)
res(4) = res(4) + delta;
end
end
res(5) = max(t);
end

```

Q1

```

v = 78/60*0.01;
len = 435.5e-2;

```

```

KK = 273.15;

ll = 15e-2;    % 宽度
h = 0.5 / 100;
k = 0.1 / 100;
x = 0:h:(len);
y = 0:k:(ll);
%% 炉心温度
iteration = 50000;
n = size(x,2);
m = size(y,2);
z_p = ones(n,m) * 0;
z_c = ones(n,m) * 0;
A = [h^2,h^2,k^2,k^2];
Eps = 1e-3;
w = 1.7;
for int = 1:iteration
    for j = 1:m
        z_c(1,j) = 25 + KK;
        z_c(n,j) = 25 + KK;
    end
    for i = 1:n
        z_c(i,m) = get_t(x(i)) + KK;
        z_c(i,1) = get_t(x(i)) + KK;
    end
    for i = 2:n-1
        for j = 2:m-1
            z_c(i,j) = (1-w)*z_p(i,j) +
                w*(dot(A,[z_c(i,j-1),z_p(i,j+1),z_c(i-1,j),z_p(i+1,j)])))/2/(h^2+k^2);
        end
    end
    err = abs(z_p - z_c);
    if (all(err<Eps))
        break;
    end
    z_p = z_c;
end
z_p = z_p - KK;
tt = ll/2 / k;

```

```

xlswrite('z1.xls',z_p(:,tt));
%% 炉温曲线
tc_real = 47.9567;
step = 0.01;
t_e = xlsread('z1.xls') + KK;
ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1)+step*(t_0(i-1)-t_f-2)/(-tc_real);
end
plot(ti,t_0-KK);
xlabel('时间/s')
ylabel('温度/\circ C')
res_time = [];
res_temp = [];
for i = 1:length(t_0)
if(mod(i,50)==1)
res_time = [res_time, ti(i)];
res_temp = [res_temp, t_0(i)];
end
end
xlswrite('result.xlsx',[res_time;res_temp-KK]');
ch = check(t_0-KK,ti);
t_0 = t_0 - KK;
ch(1)
ch(2)
ch(3)
ch(4)
ch(5)
%% 炉温曲线检验
function res = check(t, ti)
delta = ti(2) - ti(1);
res = [0,1e9,0,0,0];
for i = 2:length(t)
if(i>1)
res(1) = max(res(1),(t(i)-t(i-1))/delta);
res(2) = min(res(2),(t(i)-t(i-1))/delta);

```

```

if (t(i)-t(i-1))>=0 && t(i)<=190 && t(i)>=150)
res(3) = res(3) + delta;
end
end
if (t(i)>217)
res(4) = res(4) + delta;
end
end
res(5) = max(t);
end
%% 边界温度分布函数
function temp = get_t(x)
to = 25;
t1 = 173;
t2 = 198;
t3 = 230;
t4 = 257;
x = x * 100;
len = 435.5;
if (x<=25)
temp = to + (t1 - to)/25 * x;
elseif (x<=25 + 30.5*5 + 5*4)
temp = t1;
elseif (x<=25 + 30.5*5 + 5*5)
temp = t1 + (t2-t1)/5 * (x - (25 + 30.5*5 + 5*4));
elseif (x<=25 + 30.5*6 + 5*5)
temp = t2;
elseif (x<=25 + 30.5*6 + 5*6)
temp = t2 + (t3-t2)/5 * (x - (25 + 30.5*6 + 5*5));
elseif (x<=25 + 30.5*7 + 5*6)
temp = t3;
elseif (x<=25 + 30.5*7 + 5*7)
temp = t3 + (t4-t3)/5 * (x - (25 + 30.5*7 + 5*6));
elseif (x<=25 + 30.5*9 + 5*8)
temp = t4;
else
temp = (len - x)/(len - (25 + 30.5*9 + 5*8))*(t4 - to) + to;
end
end

```

```

%% 时间对应温度
function idx = get_t2(tim,v,x)
l = 1;r = length(x);
while(r-l>=5)
mid = floor((l+r)/2);
if (tim*v<x(mid))
r = mid;
else
l = mid;
end
end
for i = 1:r
if (tim*v>=x(i))
idx = i;
break;
end
end
end

```

Q2

```

v = (65:100)/60/100;
len = 435.5e-2;
KK = 273.15;

ll = 15e-2;    % 宽度
h = 0.5 / 100;
k = 0.1 / 100;
x = 0:h:(len);
y = 0:k:(ll);
%% 炉心温度
iteration = 50000;
n = size(x,2);
m = size(y,2);
z_p = ones(n,m) * 0;
z_c = ones(n,m) * 0;
A = [h^2,h^2,k^2,k^2];
Eps = 1e-3;
w = 1.7;

```

```

for int = 1:iteration
for j = 1:m
z_c(1,j) = 25 + KK;
z_c(n,j) = 25 + KK;
end
for i = 1:n
z_c(i,m) = get_t(x(i)) + KK;
z_c(i,1) = get_t(x(i)) + KK;
end
for i = 2:n-1
for j = 2:m-1
z_c(i,j) = (1-w)*z_p(i,j) +
    w*(dot(A,[z_c(i,j-1),z_p(i,j+1),z_c(i-1,j),z_p(i+1,j)])))/2/(h^2+k^2);
end
end
err = abs(z_p - z_c);
if (all(err<Eps))
break;
end
z_p = z_c;
end
z_p = z_p - KK;
tt = ll/2 / k;
xlswrite('z2.xls',z_p(:,tt));
%% 大步长画图
tc_real = 47.9567;
step = 0.01;
t_e = xlsread('z2.xls') + KK;
ch = zeros(5,length(v));
for ttt = 1:length(v)
ti = 0:step:(len/v(ttt));    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v(ttt),x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
ch(:, ttt) = check(t_0-KK,ti);
end

```

```

plot(v*100*60,ch(1,:)*10);
hold on
plot(v*100*60,ch(2,:)*10);
hold on
plot(v*100*60,ch(3,:));
hold on
plot(v*100*60,ch(4,:));
hold on
plot(v*100*60,ch(5,:));
legend('温度上升斜率最高值*10','温度下降斜率最低值*10','上升过程中在150\circC-190\circC的时间','温
xlabel('速度/(cm/min)')
ylabel('制程界限')
%% 搜索最大可行v
v = (70:0.01:75)/60/100;
tc_real = 47.9567;
step = 0.01;
t_e = xlsread('z2.xls') + KK;
ch = zeros(5,length(v));
for ttt = 1:length(v)
    ti = 0:step:(len/v(ttt));    % 时间
    t_0 = zeros(1,length(ti));
    t_0(1) = 25 + KK;
    for i = 2:length(ti)
        t_f = t_e(get_t2(ti(i),v(ttt),x));
        t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
    end
    ch(:, ttt) = check(t_0-KK,ti);
end
res = 0;
for i = length(v):-1:1
    if(ch(1,i)<=3 && ch(2,i)>=-3 && ch(3,i)>=60 && ch(3,i)<=120 &&...
    ch(4,i)>=40 && ch(4,i)<=90 && ch(5,i)>=240 && ch(5,i)<=250)
        res = v(i);
        break
    end
end
res*60*100
%% 最优v=72.25画图
v = 72.25/60/100;

```



```

ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
plot(ti,t_0-KK);
xlabel('时间/s')
ylabel('温度/\circ C')
ch = check(t_0-KK,ti)

%% 炉温曲线检验
function res = check(t, ti)
delta = ti(2) - ti(1);
res = [0,1e9,0,0,0];
for i = 2:length(t)
if(i>1)
res(1) = max(res(1),(t(i)-t(i-1))/delta);
res(2) = min(res(2),(t(i)-t(i-1))/delta);
if(t(i)-t(i-1)>=0 && t(i)<=190 && t(i)>=150)
res(3) = res(3) + delta;
end
end
if(t(i)>217)
res(4) = res(4) + delta;
end
end
res(5) = max(t);
end

%% 边界温度分布函数
function temp = get_t(x)
to = 25;
t1 = 182;
t2 = 203;
t3 = 237;
t4 = 254;
x = x * 100;

```

```

len = 435.5;
if (x<=25)
temp = to + (t1 - to)/25 * x;
elseif (x<=25 + 30.5*5 + 5*4)
temp = t1;
elseif (x<=25 + 30.5*5 + 5*5)
temp = t1 + (t2-t1)/5 * (x - (25 + 30.5*5 + 5*4));
elseif (x<=25 + 30.5*6 + 5*5)
temp = t2;
elseif (x<=25 + 30.5*6 + 5*6)
temp = t2 + (t3-t2)/5 * (x - (25 + 30.5*6 + 5*5));
elseif (x<=25 + 30.5*7 + 5*6)
temp = t3;
elseif (x<=25 + 30.5*7 + 5*7)
temp = t3 + (t4-t3)/5 * (x - (25 + 30.5*7 + 5*6));
elseif (x<=25 + 30.5*9 + 5*8)
temp = t4;
else
temp = (len - x)/(len - (25 + 30.5*9 + 5*8))*(t4 - to) + to;
end
end

%% 时间对应温度
function idx = get_t2(tim,v,x)
l = 1;r = length(x);
while(r-l>=5)
mid = floor((l+r)/2);
if (tim*v<x(mid))
r = mid;
else
l = mid;
end
end
for i = l:r
if (tim*v>=x(i))
idx = i;
break;
end
end
end

```

end

Q3

```
x0 = [175,195,235,255,80];
lb = [165,185,225,245,65];
ub = [185,205,245,265,100];

options=optimoptions('ga','PopulationSize',100,'MaxGenerations',500);
[x,fval,exitflag,output] =
    ga(@(x)fun(x),5,[],[],[],[], lb,ub,@(x)nonlcon(x),options)
t1 = x(1);
t2 = x(2);
t3 = x(3);
t4 = x(4);
v = x(5)/100/60;
len = 435.5e-2;
KK = 273.15;
h = 0.5 / 100;
x = 0:h:(len);
% 炉心温度
t_e = zeros(1,length(x));
for i = 1:length(t_e)
    t_e(i) = get_t(t1,t2,t3,t4,x(i)) + KK;
end
% 求解炉温曲线
tc_real = 47.9567;
step = 0.01;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
    t_f = t_e(get_t2(ti(i),v,x));
    t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
t_0 = t_0 - KK;
plot(ti,t_0);
check(t_0,ti)
```

```

% fun(ub)
%%
function area = fun(X)
t1 = X(1);
t2 = X(2);
t3 = X(3);
t4 = X(4);
v = X(5)/100/60;
len = 435.5e-2;
KK = 273.15;
h = 0.5 / 100;
x = 0:h:(len);
% 炉心温度
t_e = zeros(1,length(x));
for i = 1:length(t_e)
t_e(i) = get_t(t1,t2,t3,t4,x(i)) + KK;
end
% 求解炉温曲线
tc_real = 47.9567;
step = 0.01;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
t_0 = t_0 - KK;
[~,idx] = max(t_0);
area = 0;
for i = 1:idx
if (t_0(i)>217)
area = area + (t_0(i)-217) * step;
end
end
end
%% a
function [c,ceq] = nonlcon(X)
t1 = X(1);

```

```

t2 = X(2);
t3 = X(3);
t4 = X(4);
v = X(5)/100/60;
len = 435.5e-2;
KK = 273.15;
h = 0.5 / 100;
x = 0:h:len);
% 炉心温度
t_e = zeros(1,length(x));
for i = 1:length(t_e)
t_e(i) = get_t(t1,t2,t3,t4,x(i)) + KK;
end
% 求解炉温曲线
tc_real = 47.9567;
step = 0.01;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
ch = check(t_0-KK,ti);
c = [ch(1) - 3;
- 3 - ch(2);
ch(3) - 120;
60 - ch(3);
ch(4) - 90;
40 - ch(4);
ch(5) - 250;
240 - ch(5)];
ceq = [];
end
%% 炉温曲线检验
function res = check(t, ti)
delta = ti(2) - ti(1);
res = [0,1e9 ,0,0,0];
for i = 2:length(t)

```

```

if(i>1)
res(1) = max(res(1),(t(i)-t(i-1))/delta);
res(2) = min(res(2),(t(i)-t(i-1))/delta);
if(t(i)-t(i-1)>=0 && t(i)<=190 && t(i)>=150)
res(3) = res(3) + delta;
end
end
if(t(i)>217)
res(4) = res(4) + delta;
end
end
res(5) = max(t);
end
%% 边界温度分布函数
function temp = get_t(t1,t2,t3,t4,x)
to = 25;
x = x * 100;
len = 435.5;
if(x<=25)
temp = to + (t1 - to)/25 * x;
elseif(x<=25 + 30.5*5 + 5*4)
temp = t1;
elseif(x<=25 + 30.5*5 + 5*5)
temp = t1 + (t2-t1)/5 * (x - (25 + 30.5*5 + 5*4));
elseif(x<=25 + 30.5*6 + 5*5)
temp = t2;
elseif(x<=25 + 30.5*6 + 5*6)
temp = t2 + (t3-t2)/5 * (x - (25 + 30.5*6 + 5*5));
elseif(x<=25 + 30.5*7 + 5*6)
temp = t3;
elseif(x<=25 + 30.5*7 + 5*7)
temp = t3 + (t4-t3)/5 * (x - (25 + 30.5*7 + 5*6));
elseif(x<=25 + 30.5*9 + 5*8)
temp = t4;
else
temp = (len - x)/(len - (25 + 30.5*9 + 5*8))*(t4 - to) + to;
end
end

```

```

%% 时间对应温度
function idx = get_t2(tim,v,x)
l = 1;r = length(x);
while(r-l>=5)
mid = floor((l+r)/2);
if (tim*v<x(mid))
r = mid;
else
l = mid;
end
end
for i = 1:r
if (tim*v>=x(i))
idx = i;
break;
end
end
end

```

Q4

```

x0 = [175,195,235,255,80];
lb = [165,185,225,245,65];
ub = [185,205,245,265,100];

options=optimoptions('gamultiobj','ParetoFraction',0.5,
    'PopulationSize',300,'MaxGenerations',1000);
[x,fval, exitflag ,output] =
    gamultiobj(@(x)fun(x) ,5,[],[],[], lb,ub,@(x)nonlcon(x),options)
delete 'Q4_x'.xlsx
xlswrite('Q4_x.xlsx',x);
delete 'Q4_fval'.xlsx
xlswrite('Q4_fval.xlsx', fval);
%% pareto前沿
fval = xlsread('Q4_fval.xlsx');
for i = 1:size(fval,1)
plot(fval(i,1), fval(i,2), '.r');
hold on
end
xlabel('面积指标/(\circ C \cdot s)')

```

```

ylabel('温度差距指标/\circC')
%% 选点 画图
x = xlsread('Q4_x.xlsx');
fval = xlsread('Q4_fval.xlsx');
[~,idx] = min(fval(:,2));
x = x(idx,:);
t1 = x(1);
t2 = x(2);
t3 = x(3);
t4 = x(4);
v = x(5)/100/60;
len = 435.5e-2;
KK = 273.15;
h = 0.5 / 100;
x = 0:h:(len);
% 炉心温度
t_e = zeros(1,length(x));
for i = 1:length(t_e)
t_e(i) = get_t(t1,t2,t3,t4,x(i)) + KK;
end
% 求解炉温曲线
tc_real = 47.9567;
step = 0.01;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
t_0 = t_0 - KK;
plot(ti,t_0);
xlabel('时间/s')
ylabel('温度/\circC')
check(t_0,ti)
fval(idx,:)
%%
function res = fun(X)
t1 = X(1);

```



```

t2 = X(2);
t3 = X(3);
t4 = X(4);
v = X(5)/100/60;
len = 435.5e-2;
KK = 273.15;
h = 0.5 / 100;
x = 0:h:len);
% 炉心温度
t_e = zeros(1,length(x));
for i = 1:length(t_e)
t_e(i) = get_t(t1,t2,t3,t4,x(i)) + KK;
end
% 求解炉温曲线
tc_real = 47.9567;
step = 0.01;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
t_0 = t_0 - KK;
% 面积
[~,idx] = max(t_0);
area = 0;
for i = 1:idx
if(t_0(i)>217)
area = area + (t_0(i)-217) * step;
end
end
% 对称
err = 0;
cnt = 0;
for i = 1:length(ti)
if(t_0(idx - i)<=217 || t_0(idx + i)<=217)
break
end
end

```

```

err = err + (t_0(idx-i) - t_0(idx+i))^2;
cnt = cnt + 1;
end
err = sqrt(err / cnt);
res = [area,err];
end
%% a
function [c,ceq] = nonlcon(X)
t1 = X(1);
t2 = X(2);
t3 = X(3);
t4 = X(4);
v = X(5)/100/60;
len = 435.5e-2;
KK = 273.15;
h = 0.5 / 100;
x = 0:h:(len);
% 炉心温度
t_e = zeros(1,length(x));
for i = 1:length(t_e)
t_e(i) = get_t(t1,t2,t3,t4,x(i)) + KK;
end
% 求解炉温曲线
tc_real = 47.9567;
step = 0.01;
ti = 0:step:(len/v); % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
ch = check(t_0-KK,ti);
c = [ch(1) - 3;
- 3 - ch(2);
ch(3) - 120;
60 - ch(3);
ch(4) - 90;
40 - ch(4);

```

```

ch(5) - 250;
240 - ch(5)];
ceq = [];
end
%% 炉温曲线检验
function res = check(t, ti)
delta = ti(2) - ti(1);
res = [0,1e9,0,0,0];
for i = 2:length(t)
if(i>1)
res(1) = max(res(1),(t(i)-t(i-1))/delta);
res(2) = min(res(2),(t(i)-t(i-1))/delta);
if(t(i)-t(i-1)>=0 && t(i)<=190 && t(i)>=150)
res(3) = res(3) + delta;
end
end
if(t(i)>217)
res(4) = res(4) + delta;
end
end
res(5) = max(t);
end
%% 边界温度分布函数
function temp = get_t(t1,t2,t3,t4,x)
to = 25;
x = x * 100;
len = 435.5;
if(x<=25)
temp = to + (t1 - to)/25 * x;
elseif(x<=25 + 30.5*5 + 5*4)
temp = t1;
elseif(x<=25 + 30.5*5 + 5*5)
temp = t1 + (t2-t1)/5 * (x - (25 + 30.5*5 + 5*4));
elseif(x<=25 + 30.5*6 + 5*5)
temp = t2;
elseif(x<=25 + 30.5*6 + 5*6)
temp = t2 + (t3-t2)/5 * (x - (25 + 30.5*6 + 5*5));
elseif(x<=25 + 30.5*7 + 5*6)
temp = t3;

```

```

elseif (x<=25 + 30.5*7 + 5*7)
temp = t3 + (t4-t3)/5 * (x - (25 + 30.5*7 + 5*6));
elseif (x<=25 + 30.5*9 + 5*8)
temp = t4;
else
temp = (len - x)/(len - (25 + 30.5*9 + 5*8))*(t4 - to) + to;
end
end

%% 时间对应温度
function idx = get_t2(tim,v,x)
l = 1;r = length(x);
while(r-l>=5)
mid = floor((l+r)/2);
if (tim*v<x(mid))
r = mid;
else
l = mid;
end
end
for i = l:r
if (tim*v>=x(i))
idx = i;
break;
end
end
end

```

灵敏度分析

```

xls = xlsread('附件.xlsx','A2:B710');
tim = xls(:,1);
t = xls(:,2);
v = 7/6*0.01;
len = 435.5e-2;
% plot(tim,t);
KK = 273.15;

ll = 15e-2; % 宽度
h = 0.5 / 100;

```

```

k = 0.1 / 100;
x = 0:h:(len);
y = 0:k:(ll);
%% t_c灵敏度分析
t_e = xlsread('z0.xls') + KK;
tc_real = 48;
ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
    t_f = t_e(get_t2(ti(i),v,x));
    t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
plot(ti,t_0-KK);
hold on
t_e = xlsread('z0.xls') + KK;
tc_real = 43;
ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
    t_f = t_e(get_t2(ti(i),v,x));
    t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
plot(ti,t_0-KK);
hold on
t_e = xlsread('z0.xls') + KK;
tc_real = 53;
ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
    t_f = t_e(get_t2(ti(i),v,x));
    t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
plot(ti,t_0-KK);
hold on
plot(tim,t);
legend('t_c=48s炉温曲线','t_c=43s炉温曲线','t_c=53s炉温曲线','附件的炉温曲线');

```

```

xlabel('时间/s')
ylabel('温度/\circ C')
%% 边界代替t_f的灵敏度分析
t_e = xlsread('z0.xls') + KK;
tc_real = 47.9567;
ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
plot(ti,t_0-KK);
hold on
t_e = xlsread('z0.xls') + KK;
for i = 1:length(t_e)
t_e(i) = get_t(x(i)) + KK;
end
tc_real = 43;
ti = 0:step:(len/v);    % 时间
t_0 = zeros(1,length(ti));
t_0(1) = 25 + KK;
for i = 2:length(ti)
t_f = t_e(get_t2(ti(i),v,x));
t_0(i) = t_0(i-1) + step * (t_0(i-1) - t_f) / (-tc_real);
end
plot(ti,t_0-KK);
hold on
plot(tim,t);
legend('正常拟合的炉温曲线','边界温度代替中心温度的炉温曲线','附件的炉温曲线');
xlabel('时间/s')
ylabel('温度/\circ C')

%% 边界温度分布函数
function temp = get_t(x)
to = 25;
t1 = 175;
t2 = 195;
t3 = 235;

```

```

t4 = 255;
x = x * 100;
len = 435.5;
if (x<=25)
temp = to + (t1 - to)/25 * x;
elseif (x<=25 + 30.5*5 + 5*4)
temp = t1;
elseif (x<=25 + 30.5*5 + 5*5)
temp = t1 + (t2-t1)/5 * (x - (25 + 30.5*5 + 5*4));
elseif (x<=25 + 30.5*6 + 5*5)
temp = t2;
elseif (x<=25 + 30.5*6 + 5*6)
temp = t2 + (t3-t2)/5 * (x - (25 + 30.5*6 + 5*5));
elseif (x<=25 + 30.5*7 + 5*6)
temp = t3;
elseif (x<=25 + 30.5*7 + 5*7)
temp = t3 + (t4-t3)/5 * (x - (25 + 30.5*7 + 5*6));
elseif (x<=25 + 30.5*9 + 5*8)
temp = t4;
else
temp = (len - x)/(len - (25 + 30.5*9 + 5*8))*(t4 - to) + to;
end
end

%% 时间对应环境温度
function idx = get_t2(tim,v,x)
l = 1;r = length(x);
while(r-l>=5)
mid = floor((l+r)/2);
if (tim*v<x(mid))
r = mid;
else
l = mid;
end
end
for i = 1:r
if (tim*v>=x(i))
idx = i;
break;

```

end

end

end
