

高温作业服装设计

摘 要

本文主要针对高温工作服与假人的传热系统进行了讨论。

针对问题一：建立**热传导模型**，计算温度分布。首先，将题中所给的各层介质与皮肤的温度分布函数用热传导方程来表示，并确定其边界条件与初值条件，构成一个热传导的定解问题。然后，利用有限差分法对该定解问题进行了**离散化处理**：其中，在空间上的差分利用了中心差分公式，在时间上的差分利用了四阶龙格-库塔法对偏导数作出估计，并完成迭代；对于边界条件，采用**网格傅里叶数与网格毕渥数**选取时间与空间上的迭代精度，使得数值结果稳定；对于初值条件，通过对题目数据的描述进行反证，确定传热系统在进入 75℃ 环境前所处环境的温度为 37℃，并得出高温-假人系统的初始温度分布。最后，利用计算机数值方法进行迭代，得出了系统内各处温度随时间的变化值，并与附件 2 所提供的皮肤表面的测量温度进行了比较，结果显示误差很小，数值解与实验值十分接近。

针对问题二：确定 II 层的最优厚度。显然，II 层的厚度越厚，则越容易满足题中给的约束条件。但是，考虑到工作服的用途，在满足约束条件的情况下，厚度越小越有利。因此，必存在一个满足题中约束条件的 II 层最小厚度，即为所求的最优厚度。首先，**根据上述结论构造出一个与 II 层厚度有关的单调函数**，仍采用问题一建立的热传导模型进行数值求解，并利用黄金分割查找的方式求出不满足约束条件的临界点，得到 II 层的最优厚度为 **9.9mm**。

针对问题三：确定 II 层与 IV 层的最优厚度。目标函数为 II 层与 IV 层的厚度之和，约束条件为题中所给出的条件。由于该问题存在非线性的约束条件，且无法用函数解析式表示，因此，**将约束条件构造成为一个新的目标函数**，并利用改进的无约束多维极值的方法求出目标函数的一组较优值。得到 II 层与 IV 层的较优厚度分别为 **15.7mm** 与 **3.5mm**。

本文的特色在于：在问题一中，在有限差分法的基础上结合四阶龙格-库塔法求解初边值问题，减小了离散化迭代的误差。同时利用了网格傅里叶数与网格毕渥数之间的不等关系选取了时间与空间的迭代精度，使求解结果稳定。采用了附件 2 及数值结果在四个特殊时刻边界的温度值的相对误差衡量数值结果与附件 2 数据的吻合程度，避免了大量的繁杂计算。在问题二中，利用了比二分查找效率更高的黄金分割查找的方法求取 II 层厚度的最优值。在问题三中，构造一目标函数化去原规划问题存在的非线性边界条件，利用了改进的无约束多维极值的方法求解模型。最后将模型推广至传热系统存在**一维非均匀介质**和需要**计算求取传热系统初始分布**时的情况。

关键词：热传导模型、有限差分法、龙格-库塔法、黄金分割查找、改进的无约束多维极值法

一、 问题重述

1.1 问题背景

在高温环境下工作时，人们需要穿着专用服装以避免灼伤。专用服装通常由三层织物材料构成，记为 I、II、III 层，其中 I 层与外界环境接触，III 层与皮肤之间还存在空隙，将此空隙记为 IV 层。

为设计专用服装，将体内温度控制在 37°C 的假人放置在实验室的高温环境中，测量假人皮肤外侧的温度。为了降低研发成本、缩短研发周期，请你们利用数学模型来确定假人皮肤外侧的温度变化情况。

1.2 要解决的问题

问题一：专用服装材料的某些参数值由附件 1 给出，对环境温度为 75°C 、II 层厚度为 6 mm 、IV 层厚度为 5 mm 、工作时间为 90 分钟的情形开展实验，测量得到假人皮肤外侧的温度（见附件 2）。建立数学模型，计算温度分布，并生成温度分布的 Excel 文件（文件名为 problem1.xlsx）。

问题二：当环境温度为 65°C 、IV 层的厚度为 5.5 mm 时，确定 II 层的最优厚度，确保工作 60 分钟时，假人皮肤外侧温度不超过 47°C ，且超过 44°C 的时间不超过 5 分钟。

问题三：当环境温度为 80°C 时，确定 II 层和 IV 层的最优厚度，确保工作 30 分钟时，假人皮肤外侧温度不超过 47°C ，且超过 44°C 的时间不超过 5 分钟。

二、 问题分析

2.1 问题一的分析

在问题一中给定了某种隔热服的层厚分布，并给出了该隔热服在 75°C 的恒温环境中皮肤表层外侧的温度变化，求解目标为该隔热系统的温度分布。

可见该导热过程中物体的温度随时间而变化，所以该过程为一个非稳态导热过程，且物体的温度随时间的推移而逐渐趋于恒定的值。针对非稳态复合壁导热过程，基于傅里叶导热，对每层材料建立导热微分方程式，并求取相应的初值条件和边界条件，进行数值求解^[1,2]。以此为主要思路进行模型的建立和求解。

微分方程：该传热过程中各层仅是吸收从高温层传来的热量来获取自身温度的升高，其中各层并不会自发产热。且 IV 层虽为空气层，但其层厚较薄，且假人处于静止，一定程度上可以将该层视作普通的导热介质。

初值条件：根据题意实验假人在进入实验环境前的温度分布状态未知，即实验初始条件未知，所以为了确定初值条件，在模型建立过程中假定假人穿着隔热服在某室温环境下达到热稳态，然后进入实验环境。在此将室温作为一个自变量，初始状态会影响最终达到稳态的快慢。根据实验数据中体表温度达到稳态的时间来拟合初始的温度状态。

边界条件：隔热服最外层织物的外表面与实验环境间的边界条件为第三类边界条件；假定各层织物材料接触良好，这时在两种材料的分界面上应该满足界面连续条件；假人模型的核心保持 37°C ，皮肤内层与产热核心的边界为恒温条件，所以为第一类边界条件。

2.2 问题二的分析

对于问题二，由于空气层达到的最高温度与空气在 44 摄氏度以上的时间随着 II 层厚度的增加而减少，因此，存在一个 II 层的最小厚度能够同时满足题中所给的两个条件。显然，II 层的厚度越小，高温作业服的整体厚度越小，质量越轻，材料也越省。故 II 层的最优厚度为满足题目中条件的最小厚度。利用黄金分割比例搜索法可以获得 II 层介质的最小厚度。

2.3 问题三的分析

对于问题三，高温作业服的 II 层厚度与 IV 层厚度需要设计。根据对问题二的分析可知，高温作业服的厚度越小越好。故可以构造一个目标函数，即 II 层厚度与 IV 层厚度的总和。使得在满足题中的约束条件下，使该目标函数的值达到最小值。利用无约束多维极值法不断选取 II 层与 IV 层的厚度，使得总厚度最低。

三、 模型假设

3.1 对于高温服：

- 1、高温服的每层材料所用介质是均匀的，即其热参数不随厚度而变化。
- 2、高温服各材料间存在明显的分界面，且各分界面近似是平行的。
- 3、温度在介质中的分布为一个连续函数。

3.2 对于高温服与皮肤表层之间的气体 IV：

- 1、热辐射对于气体 IV 的影响较小，在模型中忽略^[3,12]。
- 2、气体厚度较薄，忽略气体对流对温度的影响，只简化为一导热介质。

3.3 对于皮肤：

- 1、假设皮肤为单层导热的形式，其热参数不随空间位置发生变化。
- 2、皮肤与发热核心之间存在明显的分界面，且其厚度近似为皮肤的平均厚度^[4,10]。

3.4 对于假人：

- 1、假人能够对于皮肤外的温度变化有较强的感知与调节能力，使得其核心区的温度能够保持恒定。
- 2、在初始实验前，穿着隔热服的假人热系统处于热稳态^[5]。

3.5 对于外界环境：

- 1、外界环境恒温，对流强度较小。
- 2、忽略热辐射的强度^[6]。

四、 符号说明

序号	符号	意义
1	c_i	第 i 层介质的比热容
2	ρ_i	第 i 层介质的密度
3	k_i	第 i 层介质的热传导率
4	T_i	第 i 层介质的温度分布函数
5	T_i^j	第 i 层介质在第 j 个格点的温度值
6	$T_i^j(t_k)$	第 i 层介质在第 j 个格点在 t_k 时刻的温度值
7	q	稳态时流经各介质的热流密度
8	x_i	各分界面的坐标
9	t	时间
10	ω_b	血液的灌注率
11	Q_m	代谢产热
12	h^a	对流引起的传热系数
13	t_w	I层外壁温度
14	t_f	外界流体环境温度
15	Δt	时间步长
16	Δx_i	各层的空间步长
17	Fo_Δ	傅里叶数
18	Bi_Δ	毕渥数
19	k^j	第 j 个节点的斜率值

表 1 符号说明

五、模型的建立与求解

5.1 问题一的求解

由高温作业服与皮肤的假设 2, 各层介质可由相互平行的长方体表示。首先, 以工作服的外表面为坐标原点, 由工作服的外表面指向皮肤的方向为 x 轴的正方向建立右手系, 可得到如图 1 所示的示意图:

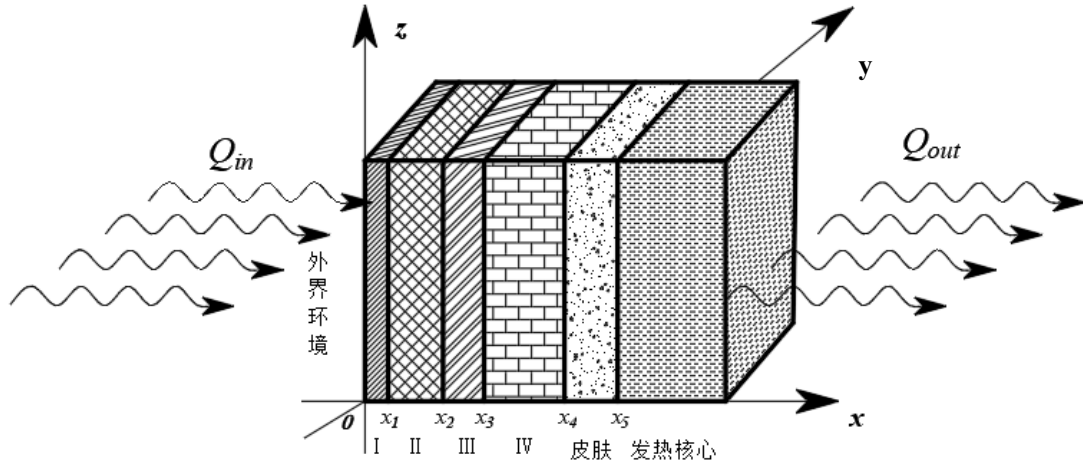


图 1 各层介质传热示意图

由于介质关于 y, z 轴不发生变化, 故等温面是平行于各介质的分界面的, 因此, 各介质内的温度分布函数不随 y, z 坐标而发生改变。各介质传热的过程伴随着温度的变化, 故其温度分布函数与时间有关, 故每层介质中的温度为坐标 x 与时间 t 的二元函数。设材料 I、II、III、空气层 IV 与皮肤的温度分布函数分别为 $T_1(x, t)$, $T_2(x, t)$, $T_3(x, t)$, $T_4(x, t)$, $T_5(x, t)$, 各分界面的坐标分别为 x_1 , x_2 , x_3 , x_4 , x_5 , 其示意图可简化如图 2 所示。

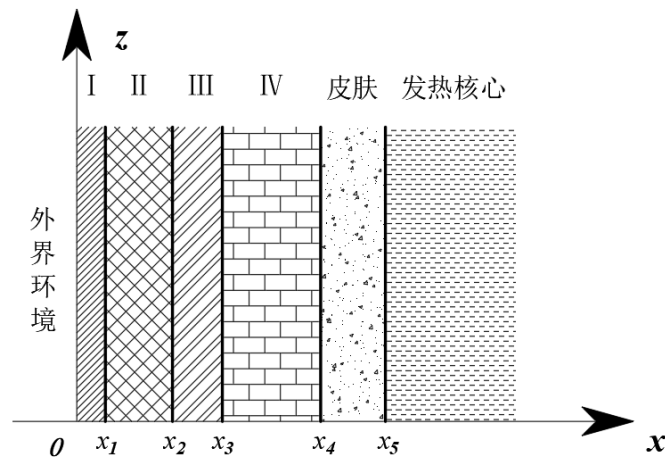


图 2 各层介质传热示意图简化

由传热学的理论可知,除皮肤外各层介质的温度分布满足热传导方程^[2],即:

$$\rho_i c_i \frac{\partial T_i}{\partial t} = k_i \cdot \frac{\partial^2 T_i}{\partial x^2} \quad (x_{i-1} \leq x \leq x_i) \quad (i=1,2,3,4,5) \quad (1)$$

对于皮肤的温度分布,采用 PENNES 单层模型^[7,8,9],其温度分布函数满足的关系可由以下的偏微分方程表述:

$$\rho_5 c_5 \frac{\partial T_5}{\partial t} = k_5 \cdot \frac{\partial^2 T_5}{\partial x^2} + \omega_b \rho_b c_b (T_a - T_5) + Q_m \quad (x_4 < x < x_5) \quad (2)$$

$\omega_b \rho_b c_b (T_a - T_5)$ 与 Q_m 分别为血管流动及皮肤代谢引起的温度变化^[11],为了简化对整个传热系统的研究,暂且不考虑这两部分温度变化的影响。故对皮肤的热力学模型可重新表述为:

$$\frac{\partial T_5}{\partial t} = \frac{k_5}{c_5 \rho_5} \cdot \frac{\partial^2 T_5}{\partial x^2} \quad (x_4 < x < x_5) \quad (3)$$

对于分层有界介质的传热问题,需要确定层间各边界所满足的边界条件。材料 I 与外界环境之间的分界面上存在着少许对流,由数学物理方程论中的边界条件类型可知,该边界属于第三类边界条件,满足:

$$-k \left(\frac{\partial T}{\partial n} \right)_w = h(t_w - t_f) \quad (4)$$

其中下角标 w 表示边界的位置函数, t_w 为边界上的温度值, t_f 为外界环境的温度值。本问题中各介质的温度分布函数在空间上仅为坐标 x 的函数,故可将上述边界条件改写为:

$$-\frac{\partial T_1}{\partial x} \Big|_{x=0} + \frac{h^a}{k_1} T_1 \Big|_{x=0} = \frac{h^a}{k_1} T_f \quad (5)$$

材料 II、III、空气层 IV 与皮肤之间的边界满足第二类边界条件,即:

$$k_i \left(\frac{\partial T_i}{\partial n} \right)_w = k_j \left(\frac{\partial T_j}{\partial n} \right)_w \quad (6)$$

其中 k_i 与 k_j 分别为边界两侧材料 i , j 的热传导率。同理,可将其改写为:

$$-k_i \frac{\partial T_i}{\partial x} \Big|_{x=x_{i-1}} = -k_{i-1} \frac{\partial T_{i-1}}{\partial x} \Big|_{x=x_{i-1}} \quad (i=2,3,4,5) \quad (7)$$

皮肤与发热核心之间的边界满足第一类边界条件,即:

$$T_w = f(\tau) \quad (8)$$

其中 $f(\tau)$ 为一不含时间变量的函数。同理,在本问题下,该边界的温度保持恒定。故可将其改写为:

$$T_5(x_5, t) = T_5^0 = \text{const} \quad (9)$$

显然，各层介质的初始温度分布对传热过程也有影响，故必须确定各层介质的初始温度分布。由对假人的假设 2 可知，初始该传热系统达到稳态。在此，我们推论室温为 37 摄氏度。采用反证法，假设室温不为 37 摄氏度，则传热系统达到稳态时皮肤表面与发热核心之间必存在一个温度差，而由题中所给出的数据可知，皮肤表面的初始温度为 37 摄氏度，故环境的温度为 37 摄氏度。即各层介质内的初始温度分布为：

$$T_i(x_5, 0) = T_5^0 = \text{const} \quad (10)$$

至此，整个传热系统的偏微分方程组，初值条件与边值条件均已确定，整个传热系统的温度分布函数的初边值问题的确定：

$$\begin{cases} \rho_i c_i \frac{\partial T_i}{\partial t} = k_i \cdot \frac{\partial^2 T_i}{\partial x^2} \quad (x_{i-1} \leq x \leq x_i) \quad (i = 1, 2, 3, 4, 5) \\ -\frac{\partial T_1}{\partial x} \Big|_{x=0} + \frac{h^a}{k_1} T_1 \Big|_{x=0} = \frac{h^a}{k_1} T_f \\ -k_i \frac{\partial T_i}{\partial x} \Big|_{x=x_i-1} = -k_{i-1} \frac{\partial T_{i-1}}{\partial x} \Big|_{x=x_i-1} \quad (i = 2, 3, 4, 5) \\ T_5(x_5, t) = T_5^0 = \text{const} \\ T_i(x_5, 0) = T_5^0 = \text{const} \end{cases} \quad (11)$$

对于该偏微分方程组，采用有限差分法将原方程及边界条件进行离散化处理。对介质的厚度方向进行等分， T_i^j 、 T_i^{j-1} 与 T_i^{j+1} 分别为介质 i 中点 j 与其左右相邻点 $j-1$ 、 $j+1$ 在 t^k 时刻的值，应用数值微分的三点公式离散，有：

$$\frac{k_i}{c_i \rho_i} \cdot \frac{\partial^2 T_i^j}{\partial x^2} = \frac{k_i}{c_i \rho_i} \cdot \frac{T_i^{j+1} - 2T_i^j + T_i^{j-1}}{(\Delta x_i)^2} \quad (i = 1, 2, 3, 4, 5) \quad (12)$$

偏微分方程左侧为对时间的偏导数，可采用四阶龙格-库塔法对 T^j 的第 t^{k+1} 时刻的值进行求取：

$$\begin{aligned} T_i^j(t_{k+1}) &= T_i^j(t_k) + \frac{1}{6}(k^1 + 2k^2 + 2k^3 + k^4)h \\ k^1 &= f(t_i + \frac{1}{2}\Delta\tau, y_i) \\ k^2 &= f(t_i + \frac{1}{2}\Delta\tau, y_i + \frac{1}{2}k^1\Delta\tau) \\ k^3 &= f(t_i + \frac{1}{2}\Delta\tau, y_i + \frac{1}{2}k^2\Delta\tau) \\ k^4 &= f(t_i + \frac{1}{2}\Delta\tau, y_i + k^3\Delta\tau) \end{aligned} \quad (13)$$

其中龙格-库塔法求解的示意图如图 3 所示。

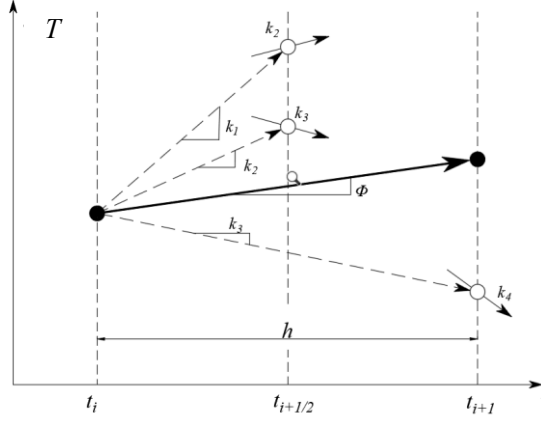


图 3 龙格-库塔法迭代求解示意图

古典四阶龙格-库塔法第一步先计算区间起点处的斜率 k^1 ，然后用这个值计算出过渡的中点值 y 和斜率 k^2 ，接下来回到区间起点用这个斜率计算出另一个过渡的中点 y 值和中点处的另一个斜率 k^3 。然后用这个斜率计算出区间终点的 y 值和斜率 k^4 。最后将四个斜率估计值组合，得到平均斜率。四阶龙格-库塔法的全局截断误差为 $O((\Delta t)^4)$ ，能够有效提高迭代精度。

对于第一类边界条件的离散化，直接将边界点处的温度设为定值即可，有：

$$T_5^j = T^0 = \text{const} \quad (j = x_5) \quad (14)$$

对于第二类边界条件的离散化，将边界条件两侧差分，有：

$$k_i \frac{T_i^{j+1} - T_i^j}{\Delta x} = k_{i-1} \frac{T_i^j - T_{i-1}^{j-1}}{\Delta x} \quad (i = 2, 3, 4, 5; j \in S, S = \{x_1, x_2, x_3, x_4\}) \quad (15)$$

经化简，可得：

$$T_i^j = \frac{k_i T_i^{j+1} + k_{i-1} T_i^{j-1}}{k_i + k_{i-1}} \quad (i = 2, 3, 4, 5; j \in S, S = \{x_1, x_2, x_3, x_4\}) \quad (16)$$

对于第三类边界条件的离散化^[13]，将等式左侧差分，有：

$$-\frac{T_1^{j+1} - T_1^j}{\Delta x} + \frac{h^a}{k_1} T_1^j = \frac{h^a}{k_1} T_{out} = \frac{h^a}{k_1} T^1 \quad (j = 0) \quad (17)$$

上式中差分项 $\frac{T_1^j - T_1^{j+1}}{\Delta x}$ 中的 T_1^{j+1} 与 T_1^j 分别为边界点与外界在 t_{k+1} 时刻的温度值，

$\frac{h^a}{k} T_1^j$ 项中的 T_1^j 为 t_{k+1} 时刻的温度值，化简可得在 $x=0$ 处的边界温度在 t_{k+1} 时刻的温度值：

$$T_1^0(t_{k+1}) = T^1 \left(1 - \frac{h^a}{k_1} \Delta x \right) + T_1^0(t_k) \frac{h^a}{k_1} \Delta x \quad (18)$$

至此，已完成对微分方程的离散化处理。

各介质层模型的参数，除了采用由题中附件给出的材料 I、II、III、空气层 IV 的数值外，皮肤层的参数如表 2 所示：

参数名称	数值	单位
c_5	4180	$J \cdot kg^{-1} \cdot ^\circ C^{-1}$
ρ_5	1000	$kg \cdot m^{-3}$
k_5	0.627	$W \cdot m^{-1} \cdot ^\circ C^{-1}$
h^a	1.0	1

表 2 皮肤层参数

除了上述未知量的值的选取情况，对时间与位置的迭代精度的选取也对最后结果的稳定性有较大的影响。在应用数值的方法解决传热问题时，常用网格傅里叶数 Fo_Δ 与网格毕渥数 Bi_Δ 来衡量传热模型数值解法的稳定性：

$$Fo_\Delta = \frac{k}{c\rho} \cdot \frac{\Delta \tau}{(\Delta x)^2} \quad (19)$$

$$Bi_\Delta = \frac{h^a}{k} \Delta x \quad (20)$$

网格傅里叶数 Fo_Δ 与网格毕渥数 Bi_Δ 需满足如下显式差分格式的稳定性条件：

$$Fo_\Delta \leq \frac{1}{2Bi_\Delta + 2} \quad (21)$$

在程序中，不同介质的热参数不一致。因此，若选取相同的时间与空间的迭代精度，最后的结果可能不收敛。程序中各层介质选取的时间与空间迭代精度如表 3 所示：

介质	×轴步长 Δx (m)	时轴步长 $\Delta \tau$ (s)	Fo_Δ	Bi_Δ
I	0.2×10^{-3}	0.02	0.099	0.00244
II	0.2×10^{-3}	0.02	0.102	0.000541
III	0.2×10^{-3}	0.02	0.176	0.00444
IV	1.0×10^{-3}	0.02	0.472	0.0357
皮肤	0.2×10^{-3}	0.02	0.104	0.000769

表 3 各阶段时间与空间迭代精度与网格参数的大小

程序执行的框图可由图 4 表示：

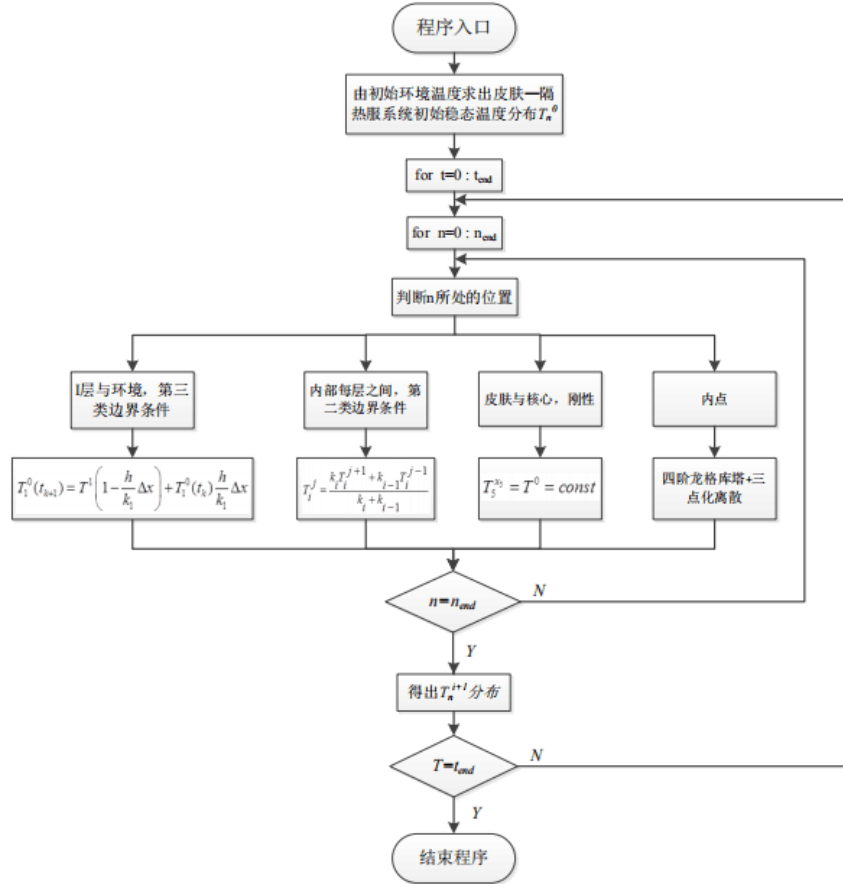


图 4 问题 1 的程序框图

由题中所给的数据（附录 2）可知，皮肤表面的温度在 15s 时开始发生变化，1376s 时达到稳定；而数值解法的皮肤表面温度在 16s 时开始发生变化，在 1478s 时达到稳态。选取附录 2 的数据与数值解的数据在上述四个时刻的温度值进行比较，并求取相对误差，数值由表 4 所示：

	15s 时边界的 温度	16s 时边界的 温度	1376s 时边界 的温度	1648s 时边界 的温度
附录 2 数据	37.00	37.01	48.05	48.08
数值结果	37.003	37.005	48.03	48.03
相对误差	0.0081%	0.014%	0.042%	0.10%

表 4 附录 1 数据与数值分析结果对比

经比较，两者相差不大，可以在一定程度上说明模型与实际情况吻合得较好。将两温度变化曲线做在一张图上，如图 5 所示：

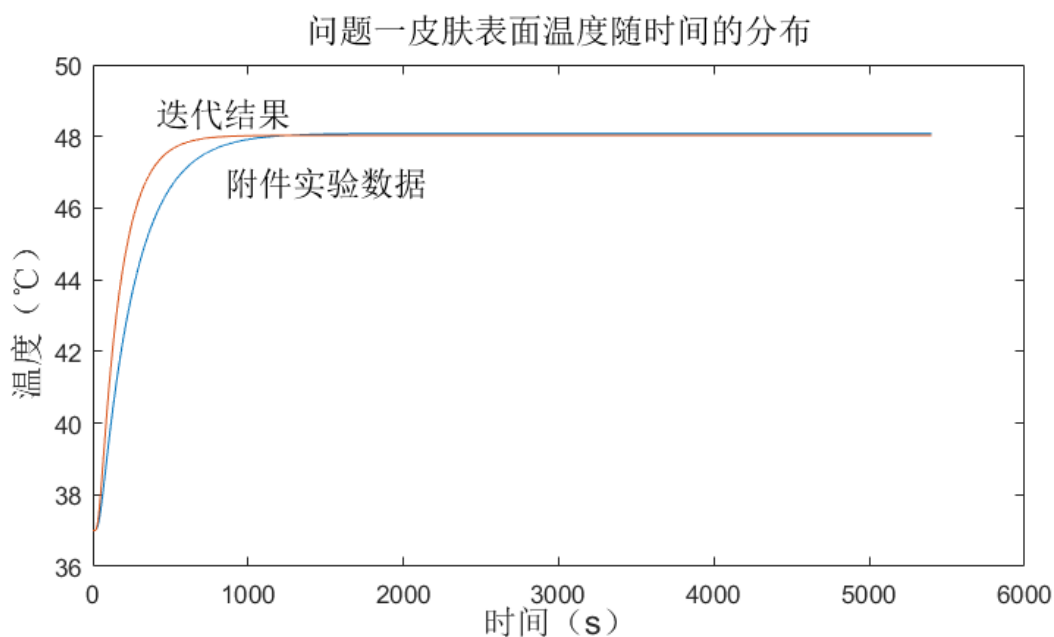


图 5 数值解与附录的温度曲线变化比较

高温服各层材料温度分布在部分时刻的情况如图 6 所示。

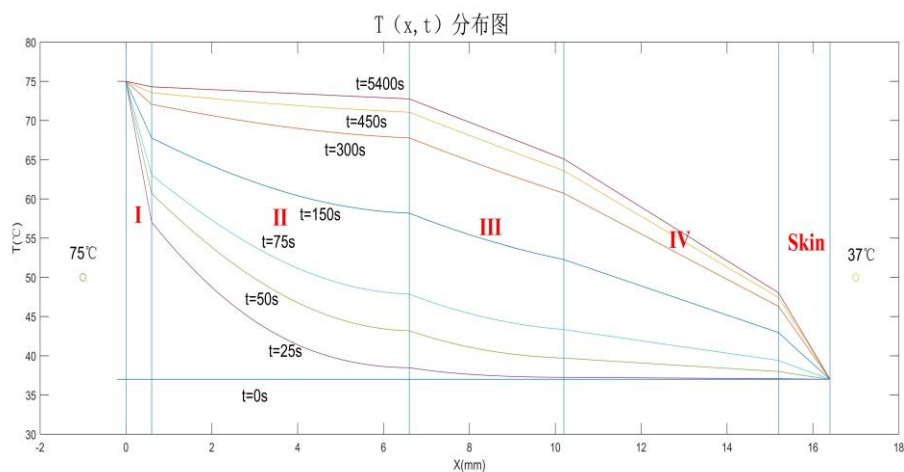


图 6 高温服的各层材料在部分时刻的温度分布情况

从上述结果可以看出，I 层与外界环境接触的边界 x_0 的温度会迅速地升高至 75 摄氏度左右。因为在该问题中 I 层外界环境中的对流系数与 I 层的导热系数之间满足：

$$h_a \gg k_1 \quad (22)$$

根据传热学的理论可知，此时流体给平板壁的传热速度很快，但平板壁向平板内部的传热速度很慢。因此，热量将积聚在平板壁附近的区域，平板壁的温度将接近外界环境中的流体。

5.2 问题二的求解

由于仅有 II 层的厚度需要确定，故皮肤表面的温度与高温服 II 层的厚度有直接的关系，且二者之间是单调的——即 II 层的厚度越厚，皮肤表面达到的最

高温度越低，维持在某一温度以上的时间也越短。所以，高温服的厚度越薄，其所用的材料越少，质量也越轻。故只需求得满足题目条件的最小厚度，即为题中所求的最优厚度。

可以构造一个衡量是否满足题中条件的参量 $value$ ，其取值的情况如下所示：

$$value(h) = \begin{cases} 0, & \text{no less than one restriction is unfitted;} \\ 1, & \text{if both two restrictions are fitted.} \end{cases} \quad (23)$$

两个约束条件如下所示：

$$\begin{aligned} T_4(x_4, 60 \text{ min}) &\leq 47^\circ\text{C} \\ -T_4^{-1}(44^\circ\text{C})\big|_{x=x_4} &\leq -55 \text{ min} \end{aligned} \quad (24)$$

其中 $T_4^{-1}(44^\circ\text{C})\big|_{x=x_4}$ 表示介质边界 x_4 上达到 44°C 的时刻。

显然，此问题即为求因变量 $value$ 的跳变点，与对连续函数求零点的方式类似。二分查询是一种对区间内连续的函数求根的一种常用方法，黄金分割查询与二分查询的原理类似，但其比二分搜索的效率更高，区间收敛得更快。故本文在程序中选用黄金分割搜索的方式确定高温服 II 层厚度的最小值。

问题 2 的程序框图如图 7 所示

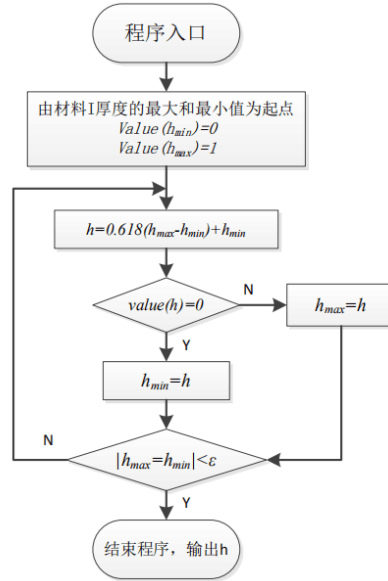


图 7 问题 2 的程序框图

解得因变量 $value$ 的跳变点的范围变化为 $(0.6, 25)$, $(0.6, 17.4)$, $(0.6, 11.0)$, $(7.0, 11.0)$ 。当 h 取不同值时边界的温度变化情况如图 8 所示，可知 II 层厚度材料厚度的最小值为 9.9mm。

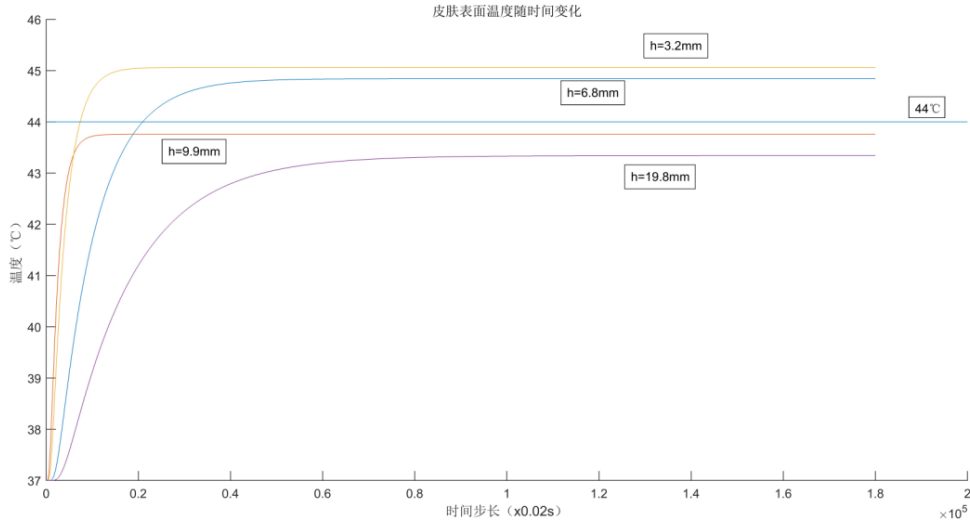


图 8 边界温度随 II 层厚度变化情况

5.3 问题三的求解

与问题二不同的是，此时需要同时确定 II 层与 IV 层的厚度，而优化的目标仍然是使 II 层与 IV 层的厚度总和达到最低。此时，由于 IV 层的厚度可变，其对热传导的速率与热稳态值均有较为显著的影响。

由于要使 II 层与 IV 层的厚度总和达到最小，先将两层总和的厚度表达式写出：

$$d = (x_4 - x_2 - 3.6) + (x_2 - 0.6) = x_4 - 4.2 \quad (25)$$

显然， d 的大小仅与 x_4 的取值有关，且 x_4 越小， d 越小。上述的目标规划问题可以表述为如下形式：

$$\begin{aligned} & \text{Min } x_4 \\ & \text{s.t. } 1.2 \leq x_2 \leq 26.2 \\ & \quad x_2 + 4.2 \leq x_4 \leq x_2 + 10 \\ & \quad T_4(x_4, 30 \text{ min}) \leq 47^\circ\text{C} \\ & \quad -T_4^{-1}(44^\circ\text{C}) \Big|_{x=x_4} \leq -25 \text{ min} \end{aligned} \quad (26)$$

显然，上述问题存在两个非线性的约束条件，属于非线性规划，且 $T_4(x, t)$ 无法用解析函数的形式表示，只能通过正演的形式求出其在特定点的函数值，故采用一般的目标规划方法不易求解。无约束多维极值法在求解的过程中不依赖于函数的导数，通过合适的构造，将各约束条件转化成为一个优化目标，即可用正演的形式确定构造函数的极值。

用因变量 $value$ 表征一个需要求解极值的函数值：

$$value(d_1, d_2) = \begin{cases} (x_4 - x_3) + (x_2 - x_1), & \text{if all four restrictions are fitted;} \\ (x_4 - x_3)_{\max} + (x_2 - x_1)_{\max}, & \text{no less than one restriction is unfitted.} \end{cases} \quad (27)$$

其中 $(x_4 - x_3)_{\max} + (x_2 - x_1)_{\max} = 31.4$ 。因此，其优化目标可表示为：

$$\begin{aligned} & \text{Min value}(d_1, d_2) \\ & x_1, x_2 \in R \end{aligned} \quad (28)$$

此时，因变量 $value$ 的自变量 x_1, x_2 无约束，构造的新问题符合无约束多维极值法的求解条件。

然而此时 x_1, x_2 的选取范围过于宽泛，需要的运算次数过多，故需要将无约束多维极值法进行修改。将两自变量的选取范围用二分法进一步压缩，可得到如图 9 所示的程序框图。

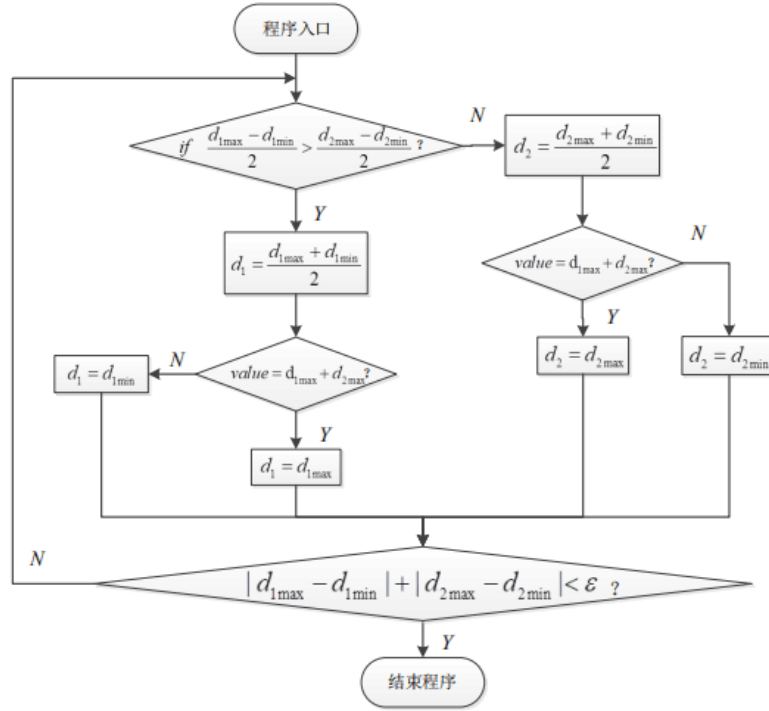


图 9 问题 3 采用的算法的程序框图

采用算法得到了一组较优的解，如图 10 所示。由程序运行的结果可知，当 d_1 （即 II 层材料的厚度）为 15.7mm， d_2 （即 IV 层材料的厚度）为 3.5mm 时，两层材料的总厚度将达到最小值，为 19.2mm。

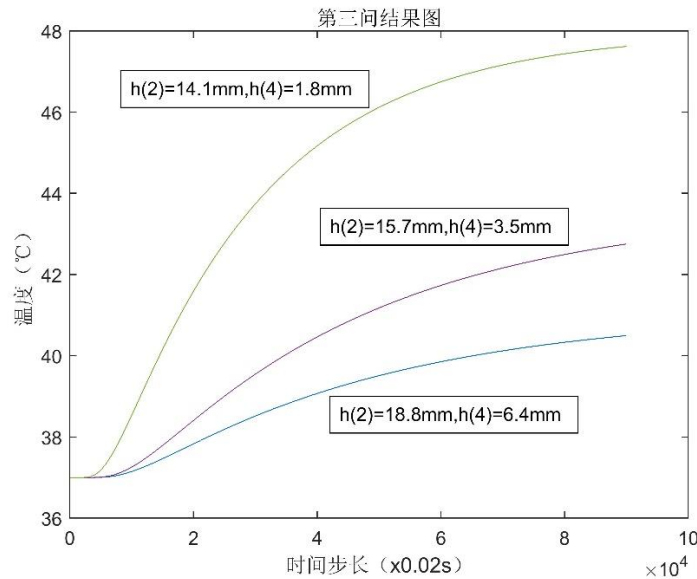


图 10 边界温度随 II 层与 IV 层厚度变化情况

六、 模型的评价

6.1 模型的优点

- 1 模型通过热传导方程确定任意时刻高温服与皮肤的温度分布的精确情况。
- 2 模型能够与附件二所给出的数据拟合较好，能较为准确地描述 IV 层与 V 层间的边界温度随时间的变化情况。
- 3 模型给出了在采用了有限差分法时时间与空间的迭代精度所要满足的关系——即网格傅里叶数与网格毕渥数之间的不等约束关系。在一定程度上避免求解的温度分布出现不收敛的情况。
- 4 模型的算法采用黄金比例查询与改进的无约束多维极值法能够较快地收敛，以求得最优解。

6.2 模型的不足

- 1 人体的皮肤是一个多层的介质，每层介质的热参数都存在相应的差异。且皮肤周围分布着血管，会对表皮的温度分布造成一定的影响。为了简化模型，这些方面在分析的过程中被忽略了，从而造成第一问中的表皮的温度分布与实际的测量结果存在一些差异。
- 2 层间的空气也存在一定的对流，会造成热量分布的变化。而由于在实际测量中是采用的是假人，且处于静止状态，故这方面引起的热量变化能够忽略不计。
- 3 在考虑材料表面的对流情况时引入了表征对流强度的热参数 h_a 。在模型中假设环境是密闭的，故选取 $h_a=1$ 。从稳态分析可知，最后各层介质的稳态温度分布受 h 的的影响较大，故模型对热参数 h_a 的值较为敏感。
- 4 构造的偏微分方程组的数值解受时间与空间上的迭代精度取值影响较大，若迭代精度选取得不合适，可能会不满足网格傅里叶数与网格毕渥数之间的不等关系，最终导致温度分布不收敛的情况出现。

七、模型的改进与推广

7.1 模型的改进

1、考虑引入 PENNES 方程中对于血管流动与皮肤代谢项的影响，将 $\omega_b \rho_b c_b (T_a - T_s)$ 与 Q_m 的加入皮肤层的偏微分方程中。

2、考虑 PENNES 多层皮肤模型，使皮肤中各层的温度分布更加精确。

3、假人的皮肤表面会模拟人体出汗，汗蒸发后会造成 IV 层间的气压升高，出现内外层的湿流变化。IV 层空气气压升高后会使该层的湿流由内向外扩散，减小 IV 层空气温度升高的速率。

7.2 模型的推广

若给定外界环境的初始温度，则稳态下的初始温度分布可以得到。忽略血液流动对传热系统的影响，设初始稳态时各分界面的温度分别为 $t_0, t_1, t_2, t_3, t_4, t_5$ ，外界环境的温度为 t_{out} ，如图 11 所示。

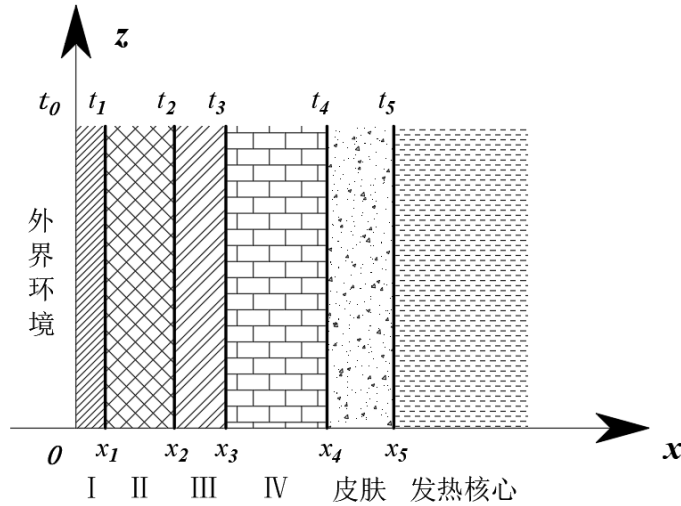


图 11 各层皮肤稳态温度的分布

传热系统稳定时，各层传递的热量 q 相等，由傅里叶定律的稳态形式，有：

$$q = \frac{\lambda_i}{x_i} (t_i - t_{i-1}) = \frac{(t_i - t_{i-1})}{R_A^i} \quad (i = 1, 2, 3, 4) \quad (29)$$

其中 R_A^i 为第 i 层热阻的大小。对于外界的对流，有：

$$q = h^a (t_0 - t_{out}) \quad (30)$$

由上述表达式可以解出各层介质的初始温度的分布：

$$\begin{aligned}
T_1(x, 0) &= \frac{t_1 - t_0}{x_1} x + t_0 \quad (0 \leq x \leq x_1) \\
T_2(x, 0) &= \frac{t_2 - t_1}{x_2} (x - x_1) + t_1 \quad (x_1 \leq x \leq x_1 + x_2) \\
T_3(x, 0) &= \frac{t_3 - t_2}{x_3} (x - x_1 - x_2) + t_2 \quad (x_1 + x_2 \leq x \leq x_1 + x_2 + x_3) \\
T_4(x, 0) &= \frac{t_4 - t_3}{x_4} (x - x_1 - x_2 - x_3) + t_3 \quad (x_1 + x_2 + x_3 \leq x \leq x_1 + x_2 + x_3 + x_4) \\
T_5(x, 0) &= T^1 = \text{const}
\end{aligned} \tag{31}$$

本题中热流密度是在 x 轴的方向上进行传播的一维热传导问题，且各层介质是均匀的。而实际上，若高温服在 x 轴的方向上的介质是不均匀的，即 k_i , c_i , ρ_i 均为 x 的函数，则偏微分方程可以改写为：

$$\frac{\partial T_i}{\partial t} = \nabla \cdot \left(\frac{k_i}{c_i \rho_i} \nabla T_i \right) = \frac{\partial}{\partial x} \left(\frac{k_i}{c_i \rho_i} \right) \cdot \frac{\partial T_i}{\partial x} + \frac{k_i}{c_i \rho_i} \frac{\partial^2 T_i}{\partial x^2} \quad (x_{i-1} < x < x_i) \quad (i=1, 2, 3, 4, 5) \tag{32}$$

等式左侧利用四阶龙格-库塔法的离散形式与前文一致，而右侧等式离散化的结果如下所示：

$$\frac{1}{\Delta x} \cdot \left(\frac{k_i^{j+1}}{c_i^{j+1} \rho_i^{j+1}} - \frac{k_i^j}{c_i^j \rho_i^j} \right) \cdot \frac{T_i^{j+1} - T_i^j}{\Delta x} + \frac{k_i^j}{c_i^j \rho_i^j} \cdot \frac{T_i^{j+1} - 2T_i^j + T_i^{j-1}}{(\Delta x)^2} \quad (i=1, 2, 3, 4, 5) \tag{33}$$

边界条件与前文所述的形式类似，只需将相应各边界中条件中的 k , c , ρ 的值修改为这些变量在边界上的值即可。

对于初值条件，由于介质在 x 轴上不均匀，故每一层的面积热阻值需要重新计算。

$$R_A^i = \int_{x_{i-1}}^{x_i} \frac{1}{k_i(\delta)} d\delta \quad (i=1, 2, 3, 5) \tag{34}$$

$$R_A^4 = \frac{x_4 - x_3}{k_4} \tag{35}$$

其中 x_0 的值为 0。传递的热流密度为：

$$q = \frac{t_5(x_5) - t_{out}}{\left(\sum_{k=1}^5 R_A^k + \frac{1}{h^a} \right)} \tag{36}$$

设 $R_A^0 = 0$ ，各层介质中任意一个位置的温度 $t_i(x)$ 满足：

$$\frac{t_5(x_5) - t_{out}}{\left(\sum_{k=0}^5 R_A^k + \frac{1}{h^a}\right)} = \frac{t_i(x) - t_{out}}{\left(\sum_{k=0}^{i-1} R_A^k + \int_{x_{i-1}}^x \frac{1}{k_i(\delta)} d\delta + \frac{1}{h^a}\right)} (i=1,2,3) \quad (37)$$

$$\frac{t_5(x_5) - t_{out}}{\left(\sum_{k=0}^5 R_A^k + \frac{1}{h^a}\right)} = \frac{t_4(x) - t_{out}}{\left(\sum_{k=0}^3 R_A^k + \frac{x - x_3}{k_4} + \frac{1}{h^a}\right)} \quad (38)$$

$$\frac{t_5(x_5) - t_{out}}{\left(\sum_{k=0}^5 R_A^k + \frac{1}{h^a}\right)} = \frac{t_5(x) - t_{out}}{\left(\sum_{k=0}^4 R_A^k + \int_{x_4}^{x_5} \frac{1}{k_i(\delta)} d\delta + \frac{1}{h^a}\right)} \quad (39)$$

至此，对于材质非均匀的高温服的情况，也给出各层介质中任意时刻的温度分布。

参考文献

- [1] 姜启源, 谢金星, 叶俊.数学模型 (第四版) [M].北京: 高等教育出版社, 2011:27-29.
- [2] 杨世铭, 陶文铨.传热学 (第四版) [M].北京: 高等教育出版社, 2006: 1-50.
- [3] 张昭华, 王云仪, 李俊.衣下空气层厚度对着装人体热传递的影响[J].纺织学报, 2010, 12.
- [4] 迪丽娜·马合木提, 江世臣, 张学学.皮肤组织传热过程研究[J].西安交通大学学报, 2006, 40 (3): 369-372.
- [5] 付明, 翁文国, 韩雪峰.高温下防护服热阻和湿阻的暖体假人实验[J].清华大学学报 (自然科学版), 2017, 57 (3): 281-285+292.
- [6] 杨杰.基于人体-服装-环境的高温人体热反应模拟与实验研究[D].清华大学, 2016.
- [7] PENNES H H. Analysis of tissue and arterial blood temperature in resting human forearm [J]. Journal of Applied Physiology, 1948,1(2):93-122.
- [8] PENNES H H. Analysis of tissue and arterial blood temperature in resting human forearm [J]. Journal of Applied Physiology, 1948,1(2):93-122.
- [9] Oliveira A, Gaspar A, Quintela D. Measurements of clothing insulation with a thermal manikin operating under the thermal comfort regulation mode: Comparative analysis of the calculation methods [J]. European Journal of Applied Physiology, 2008,104(4):679-688.
- [10] Torvi D A, Dale J D. A finite element model of skin subjected to a flash fire[J]. Journal of Biomechanical Engineering, 1994,116(5):250-255.
- [11] Diller K R, Hayes L J. A finite element model of burn injury in blood perfused skin [J]. Journal of Biomechanical Engineering, 1983,105(6):300-307.
- [12] WANG Yunyi, JINHUAN P, SUYAN L. Study on the relation between the air gap size under clothing and the comfort sensations [J]. Journal of Donghua University: English Edition, 2006,23(5):101-107.
- [13] <https://wenku.baidu.com/view/ec97422fbd64783e09122b3e.html>

附录

问题一程序：

```
%-----  
%                               温度分布求解程序  
%-----  
  
% T: 计算总时长, 1800s=30min  
% dx: x轴步长  
% dt: t轴步长  
% ux: 划分的微小区间数目  
% tem: 记录当前温度分布的矩阵, 大小为1*ux  
% temn: 记录所有时刻温度分布的矩阵, 大小为T*(ux+1)  
  
clc;clear all  
tic  
global x  
global Ti  
  
%迭代参数  
T=1800;  
t0=0;tn=0;%记录当前时刻  
tt=0.02;%rk迭代精度  
dt=0.02;  
dx=0.2*10-3;          %mm换算成m单位, 须能除尽  
ux=((0.6+6+3.6+5+1.2)*10-3)/dx;    %离散划分数目  
%tem=InitFun(40);      %初始分布  
%tem=[tem, zeros(1, uint64(0.045/dx))+37];  
temn=zeros(T, uint64(ux+2));  
mmm=size(temn);  
tem=ones(1, mmm(2))*37;  
  
%计算过程所需参数  
k=[0.082 0.37 0.045 0.028 0.26];  
Tout=75;  
h=1;  
thickness=[0.6 6 3.6 5 1.2];  
thickness=thickness.*10-3;  
  
while(tn<=T)  
t0=tn;tn=tn+dt;ts=[t0, tn];  
time=uint64(tn/dt);
```

```

temn(time,1)=tn;

for i=1:thickness(1)/dx-1+0.1      %1层
    x=1;
    Ti=(tem(i+2)-2*tem(i+1)+tem(i))/(dx)^2;
%    dT = dTdt([], []);
%    temn(time,i+2)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(i+1),tt);
    temn(time,i+2)=w(length(t));
end
for i=1:thickness(2)/dx-1+0.1      %2层
    x=2;
    n=thickness(1)/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%    dT = dTdt([], []);
%    temn(time,n+1)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(n),tt);
    temn(time,n+1)=w(length(t));
end
for i=1:thickness(3)/dx-1+0.1      %3层
    x=3;
    n=(thickness(1)+thickness(2))/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%    dT = dTdt([], []);
%    temn(time,n+1)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(n),tt);
    temn(time,n+1)=w(length(t));
end
for i=1:(thickness(4)/dx)/5-1+0.1    %4层
    dx1=dx*5;
    x=4;
    n=(thickness(1)+thickness(2)+thickness(3))/dx+1+i*5;
    Ti=(tem(n+5)-2*tem(n)+tem(n-5))/(dx1)^2;
    [t,w] = rk4sys(@dTdt,ts,tem(n),tt);
    temn(time,n+1)=w(length(t));
%    dT = dTdt([], []);
%    temn(time,n+1)=tem(i+1)+dT*dt;
end
for i=1:thickness(5)/dx-1+0.1      %5层
    x=5;
    n=(thickness(1)+thickness(2)+thickness(3)+thickness(4))/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%    dT = dTdt([], []);

```

```

%      temn(time,n+1)=tem(i+1)+dT*dt;
      [t,w] = rk4sys(@dTdt,ts,tem(n),tt);
      temn(time,n+1)=w(length(t));

end
x=0;
n=[1,thickness(1)/dx+1,(thickness(1)+thickness(2))/dx+1,...
   (thickness(1)+thickness(2)+thickness(3))/dx+1,...
   (thickness(1)+thickness(2)+thickness(3)+thickness(4))/dx+1,...
   (thickness(1)+thickness(2)+thickness(3)+thickness(4)+thickness(5))/dx+1 ];
n=uint32(n);
temn(time,n(1)+1)=(h*(tem(1)-Tout)*dx)/k(1)+Tout; %边界一
temn(time,n(2)+1)=(k(1)*temn(time,n(2))+k(2)*temn(time,n(2)+2))/(k(1)+k(2)); %边界二
temn(time,n(3)+1)=(k(2)*temn(time,n(3))+k(3)*temn(time,n(3)+2))/(k(2)+k(3)); %边界三
temn(time,n(4)+1)=(k(3)*temn(time,n(4))+(k(4)/5)*temn(time,n(4)+6))/(k(3)+k(4)/5); %边界四
temn(time,n(5)+1)=(k(4)*temn(time,n(5)-4)+(k(5)/5)*temn(time,n(5)+2))/(k(4)+k(5)/5); %边界五
temn(time,n(6)+1)=37;
tem=temn(time,2:mmm(2));
toc
end

```

```

xlswrite('C:\Users\ZPJ\Desktop\result.xlsx',temn);

```

```

function [tp,yp] = rk4sys(dydt,tspan,y0,h,varargin)
%rk4sys:四阶龙格-库塔方法解ODES
%input:
% dydt——描述ODES的M-file name
% tspan——[ti,tf],起始和结束的微分变量值;或者[t0 t1 t2 ...],指定特定的输出微分变量点
% y0——初始化自变量值矩阵
% h——步长
% varargin——额外变量值
%output
% tp——自变量矩阵
% yp——因变量矩阵

if nargin<4,error('at least 4 input arguments required'),end
if any(diff(tspan)<=0),error('tspan not ascending order'),end
n=length(tspan);
ti=tspan(1);tf=tspan(n);
if n==2
    t=(ti:h:tf)';n=length(t);
    if t(n)<tf %例: 0:2:7=[0 2 4 6],此处再增成[0 2 4 6 7]

```

```

        t(n+1)=tf;
        n=n+1;
    end
else
    t=tspan;
end
tt=ti;y(1,:)=y0;
np=1;tp(np)=tt;yp(np,:)=y(1,:);    %np用来记录指定输出点的序号
i=1;
while(1)
    tend=t(np+1);
    hh=t(np+1)-t(np);
    if hh>h,hh=h;end    %即使指定自变量输出点，也要按照步长来计算
    while(1)    %用来计算相邻指定输出点之间的过程，若是起止点方式，则while只循环一次
        if tt+hh>tend,hh=tend-tt;end    %也是调整临时步长，避免指定点不被步长整除的问题
        k1=dydt(tt,y(i,:),varargin{:})';
        ymid=y(i,:)+k1*hh/2;
        k2=dydt(tt+hh/2,ymid,varargin{:})';
        ymid=y(i,:)+k2*hh/2;
        k3=dydt(tt+hh/2,ymid,varargin{:})';
        yend=y(i,:)+k3*hh;
        k4=dydt(tt+hh,yend,varargin{:})';
        phi=(k1+2*(k2+k3)+k4)/6;
        y(i+1,:)=y(i,:)+phi*hh;
        tt=tt+h;
        i=i+1;
        if tt>=tend,break,end
    end
    np=np+1;tp(np)=tt;yp(np,:)=y(i,:);
    if tt>=tf,break,end
end

```

```

function dT = dTdt(t,T)
%储存微分方程的函数
global x
global Ti
%计算过程所需参数
k=[0.082 0.37 0.045 0.028];
c=[1377 2100 1726 1005];
rho=[300 862 74.2 1.18];

lambda=0.26;
rho_sk=500;

```

```

c_sk=2500;
wb=0.005;
ta=36.5;

if (x==1)
    dT=k(1)/(c(1)*rho(1))*Ti;
elseif (x==2)
    dT=k(2)/(c(2)*rho(2))*Ti;
elseif (x==3)
    dT=k(3)/(c(3)*rho(3))*Ti;
elseif (x==4)
    dT=k(4)/(c(4)*rho(4))*Ti;
elseif (x==5)
    %dT=lambda/(rho_sk*c_sk)*Ti+wb*(ta-T);
    dT=lambda/(rho_sk*c_sk)*Ti;
else error(' something may be wrong!'),end

end

%-----
%                               绘图程序
%-----

xxx=[0:1:52, 57, 62, 67, 72, 77:83];
xxx=(xxx-1)*0.2;
for i=1:3
yyy=[75, temn(((150*i)/0.02), 2:53), temn(((150*i)/0.02), 58), temn(((150*i)/0.02), 63),...
    temn(((150*i)/0.02), 68), temn(((150*i)/0.02), 73), temn(((150*i)/0.02), 78:84)];
plot(xxx, yyy);
hold on
end
for i=1:3
yyy=[75, temn(((25*i)/0.02), 2:53), temn(((25*i)/0.02), 58), temn(((25*i)/0.02), 63),...
    temn(((25*i)/0.02), 68), temn(((25*i)/0.02), 73), temn(((25*i)/0.02), 78:84)];
plot(xxx, yyy);
hold on
end
yyy=[75, temn((5400/0.02), 2:53), temn((5400/0.02), 58), temn((5400/0.02), 63),...
    temn((5400/0.02), 68), temn((5400/0.02), 73), temn((5400/0.02), 78:84)];
plot(xxx, yyy);
hold on
yyy=37*ones(1, 64);
plot(xxx, yyy);

```



```

line([0,0],[80,30]);
line([0.6,0.6],[80,30]);
line([6.6,6.6],[80,30]);
line([10.2,10.2],[80,30]);
line([15.2,15.2],[80,30]);
line([16.4,16.4],[80,30]);
scatter(-1,50);
scatter(17,50);

```

问题二程序:

```

function [x,fval] = HJ(a,b,pre)
%-----
% 黄金分割法求第2层最优厚度
%-----
while abs(b-a)>pre
    lambda=fix((1-0.618)*((b-a)/0.2))*0.2+a;
    value=resolution2(lambda);
    if(value==0)
        a=lambda;
    elseif(value==1)
        b=lambda;
    else error('something may be wrong!'),end
    disp([num2str(a) ' ---- ' num2str(b)]);
end
x=(a+b)/2;
fval=value;
end

function [temn] = resolution2(th2)
%function [value] = resolution2(th2)
% 第二层最优厚度求解子函数

% T: 计算总时长, 3600s=60min
% dx: x轴步长
% dt: t轴步长
% ux: 划分的微小区间数目
% tem: 记录当前温度分布的矩阵, 大小为1*ux
% temn: 记录所有时刻温度分布的矩阵, 大小为T*(ux+1)
global x
global Ti

```

```

%迭代参数
T=3600;
t0=0;tn=0;%记录当前时刻
tt=0.02;%rk迭代精度
dt=0.02;
dx=0.2*10^(-3);          %mm换算成m单位, 须能除尽
ux=((0.6+th2+3.6+5.5+1.2)*10^(-3))/dx;    %离散划分数目
tem=InitFun(40);          %初始分布
tem=[tem,zeros(1,uint64(0.045/dx))+37];
temn=zeros(T,uint64(ux+2));
mmm=size(temn);
tem=ones(1,mmm(2))*37;

%计算过程所需参数
k=[0.082 0.37 0.045 0.028 0.26];
Tout=65;
h=1;
thickness=[0.6 th2 3.6 5.5 1.2];
thickness=thickness.*10^(-3);
countt=0;

while(tn<=T)
t0=tn;tn=tn+dt;ts=[t0,tn];
time=uint64(tn/dt);
temn(time,1)=tn;

for i=1:thickness(1)/dx-1+0.1      %1层
    x=1;
    Ti=(tem(i+2)-2*tem(i+1)+tem(i))/(dx)^2;
    %    dT = dTdt([],[]);
    %    temn(time,i+2)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(i+1),tt);
    temn(time,i+2)=w(length(t));
end
for i=1:uint64(thickness(2)/dx-1+0.1)      %2层
    x=2;
    n=thickness(1)/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
    %    dT = dTdt([],[]);
    %    temn(time,n+1)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(n),tt);
    temn(time,n+1)=w(length(t));

```

```

end
for i=1:uint64(thickness(3)/dx-1+0.1)      %3层
    x=3;
    n=(thickness(1)+thickness(2))/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%     dT = dTdt([], []);
%     temn(time, n+1)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt, ts, tem(n), tt);
    temn(time, n+1)=w(length(t));
end
for i=1:uint64((thickness(4)/dx)/5-1+0.1)    %4层
    dx1=dx*5;
    x=4;
    n=(thickness(1)+thickness(2)+thickness(3))/dx+1+i*5;
    fff=(thickness(1)+thickness(2)+thickness(3))/dx+1+i*5;
    Ti=(tem(n+5)-2*tem(n)+tem(n-5))/(dx1)^2;
    [t,w] = rk4sys(@dTdt, ts, tem(n), tt);
    temn(time, n+1)=w(length(t));
%     dT = dTdt([], []);
%     temn(time, n+1)=tem(i+1)+dT*dt;
end
for i=1:uint64(thickness(5)/dx-1+0.1)      %5层
    x=5;
    n=(thickness(1)+thickness(2)+thickness(3)+thickness(4))/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%     dT = dTdt([], []);
%     temn(time, n+1)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt, ts, tem(n), tt);
    temn(time, n+1)=w(length(t));
end
x=0;
n=[1, thickness(1)/dx+1, (thickness(1)+thickness(2))/dx+1, ...
    (thickness(1)+thickness(2)+thickness(3))/dx+1, ...
    (thickness(1)+thickness(2)+thickness(3)+thickness(4))/dx+1, ...
    (thickness(1)+thickness(2)+thickness(3)+thickness(4)+thickness(5))/dx+1 ];
n=uint32(n);
temn(time, n(1)+1)=(h*(tem(1)-Tout)*dx)/k(1)+Tout; %边界一
temn(time, n(2)+1)=(k(1)*temn(time, n(2))+k(2)*temn(time, n(2)+2))/(k(1)+k(2)); %边界二
temn(time, n(3)+1)=(k(2)*temn(time, n(3))+k(3)*temn(time, n(3)+2))/(k(2)+k(3)); %边界三
temn(time, n(4)+1)=(k(3)*temn(time, n(4))+k(4)/5*temn(time, n(4)+6))/(k(3)+k(4)/5); %边界四
temn(time, n(5)+1)=(k(4)*temn(time, uint64(fff+1))+k(5)/5*temn(time, n(5)+2))/(k(4)+k(5)/5);
%边界五
temn(time, n(6)+1)=37;

```

```

tem=temn(time,2:mmm(2));
% if (length(tem)>=44)
%     countt=countt+dt;
%     if (countt>5*60+dt)
%         tn=T+dt;
%         value=0;
%     end
% if (tem(length(tem))>=47)
%     tn=T+dt;
%     value=0;
% end
tn
end
end

```

问题三程序：

```

clear all
clc
fun=@resolvment3;
x0=[2,2];
x=patternsearch(fun,x0);

```

```

function [value] = resolvment3(th2,th4)
% 第二层及第四层最优厚度求解子函数

```

```

% T: 计算总时长, 1800s=30min
% dx: x轴步长
% dt: t轴步长
% ux: 划分的微小区间数目
% tem: 记录当前温度分布的矩阵
% temn: 记录所有时刻温度分布的矩阵
tic
global x
global Ti

%迭代参数
T=1800;
t0=0;tn=0;%记录当前时刻
tt=0.02;%rk迭代精度

```

```

dt=0.02;
dx=0.2*10^(-3);          %mm换算成m单位, 须能除尽
ux=((0.6+th2+3.6+th4+1.2)*10^(-3))/dx;    %离散划分数目
%tem=InitFun(40);        %初始分布
%tem=[tem,zeros(1,uint64(0.045/dx))+37];
temn=zeros(T,uint64(ux+2));
mmm=size(temn);
tem=ones(1,mmm(2))*37;

%计算过程所需参数
k=[0.082 0.37 0.045 0.028 0.26];
Tout=75;
h=1;
thickness=[0.6 th2 3.6 th4 1.2];
thickness=thickness.*10^(-3);
countt=0;
value=1;

while(tn<=T)
t0=tn;tn=tn+dt;ts=[t0,tn];
time=uint64(tn/dt);
temn(time,1)=tn;

for i=1:thickness(1)/dx-1+0.1      %1层
    x=1;
    Ti=(tem(i+2)-2*tem(i+1)+tem(i))/(dx)^2;
    %    dT = dTdt([],[]);
    %    temn(time,i+2)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(i+1),tt);
    temn(time,i+2)=w(length(t));
end
for i=1:uint64(thickness(2)/dx-1+0.1)    %2层
    x=2;
    n=thickness(1)/dx+1+i;
    Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
    %    dT = dTdt([],[]);
    %    temn(time,n+1)=tem(i+1)+dT*dt;
    [t,w] = rk4sys(@dTdt,ts,tem(n),tt);
    temn(time,n+1)=w(length(t));
end
for i=1:uint64(thickness(3)/dx-1+0.1)    %3层
    x=3;
    n=(thickness(1)+thickness(2))/dx+1+i;

```

```

        Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%        dT = dTdt([], []);
%        temn(time, n+1)=tem(i+1)+dT*dt;
        [t, w] = rk4sys(@dTdt, ts, tem(n), tt);
        temn(time, n+1)=w(length(t));
    end
    for i=1:uint64((thickness(4)/dx)/5-1+0.1)        %4层
        dx1=dx*5;
        x=4;
        n=(thickness(1)+thickness(2)+thickness(3))/dx+1+i*5;
        fff=(thickness(1)+thickness(2)+thickness(3))/dx+1+i*5;
        Ti=(tem(n+5)-2*tem(n)+tem(n-5))/(dx1)^2;
        [t, w] = rk4sys(@dTdt, ts, tem(n), tt);
        temn(time, n+1)=w(length(t));
%        dT = dTdt([], []);
%        temn(time, n+1)=tem(i+1)+dT*dt;

    end
    for i=1:uint64(thickness(5)/dx-1+0.1)        %5层
        x=5;
        n=(thickness(1)+thickness(2)+thickness(3)+thickness(4))/dx+1+i;
        Ti=(tem(n+1)-2*tem(n)+tem(n-1))/(dx)^2;
%        dT = dTdt([], []);
%        temn(time, n+1)=tem(i+1)+dT*dt;
        [t, w] = rk4sys(@dTdt, ts, tem(n), tt);
        temn(time, n+1)=w(length(t));
    end
    x=0;
    n=[1, thickness(1)/dx+1, (thickness(1)+thickness(2))/dx+1, ...
        (thickness(1)+thickness(2)+thickness(3))/dx+1, ...
        (thickness(1)+thickness(2)+thickness(3)+thickness(4))/dx+1, ...
        (thickness(1)+thickness(2)+thickness(3)+thickness(4)+thickness(5))/dx+1 ];
    n=uint32(n);
    temn(time, n(1)+1)=(h*(tem(1)-Tout)*dx)/k(1)+Tout;    %边界一
    temn(time, n(2)+1)=(k(1)*temn(time, n(2))+k(2)*temn(time, n(2)+2))/(k(1)+k(2));    %边界二
    temn(time, n(3)+1)=(k(2)*temn(time, n(3))+k(3)*temn(time, n(3)+2))/(k(2)+k(3));    %边界三
    temn(time, n(4)+1)=(k(3)*temn(time, n(4))+k(4)/5*temn(time, n(4)+6))/(k(3)+k(4)/5);    %边界四
    temn(time, n(5)+1)=(k(4)*temn(time, uint64(fff+1))+k(5)/5*temn(time, n(5)+2))/(k(4)+k(5)/5);
    %边界五
    temn(time, n(6)+1)=37;
    tem=temn(time, 2:mmm(2));

    if (tem(length(tem)-6)>=44)
        countt=countt+dt;
    end
end

```

```
        if(countt>5*30)
            tn=T+dt;
            value=0;
        end
        if(tem(length(tem)-6)>=47)
            tn=T+dt;
            value=0;
        end
    end
    value=th2+th4;
    toc
end
```