

发动机内流场数值分析基础

大作业报告

成员及分工：

于 洋：Fluent 仿真

赵明潇：公式推导，隐式编程计算

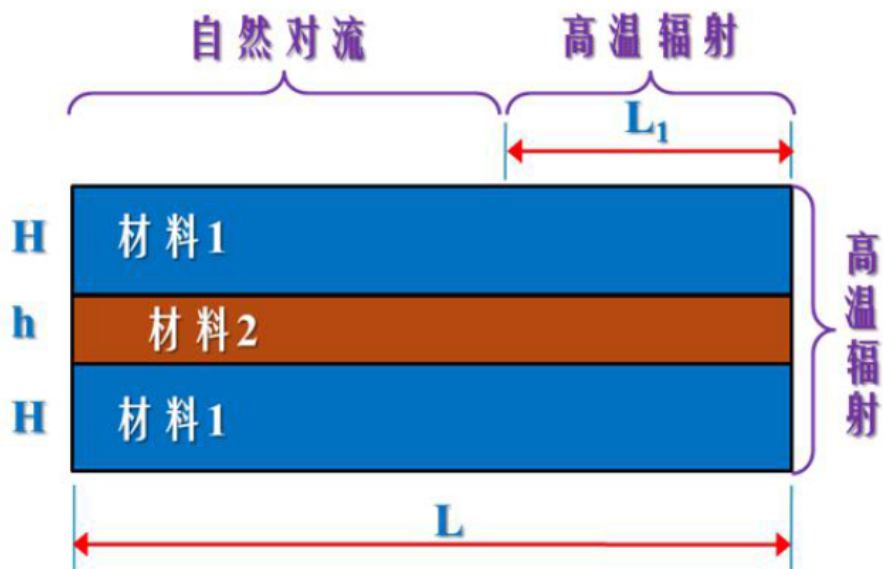
谭 力：显式编程计算

高 屹：半隐式编程计算

肖良丰：结果分析

郭思佳：参数确定，结果分析

一、 问题分析



求解不同时刻多层结构温度场

材料 1 和材料 2 组成三明治结构，右端处于 923 k 的燃气中，下边界和左边界为绝热边界，上边界左段处于大气中，大气温度为 293 k。求某一时刻计算域内温度分布。

二、 参数确定

1. 系统发射率

1) 燃气发射率

气体辐射和吸收与固体和液体不同，固体和液体的辐射和吸收具有在表面上进行的特点，而气体的辐射和吸收则是在整个容积中进行的，因此必须说明气体所处容积的形状和容积的大小。我们完善本题中的物理模型如下图。

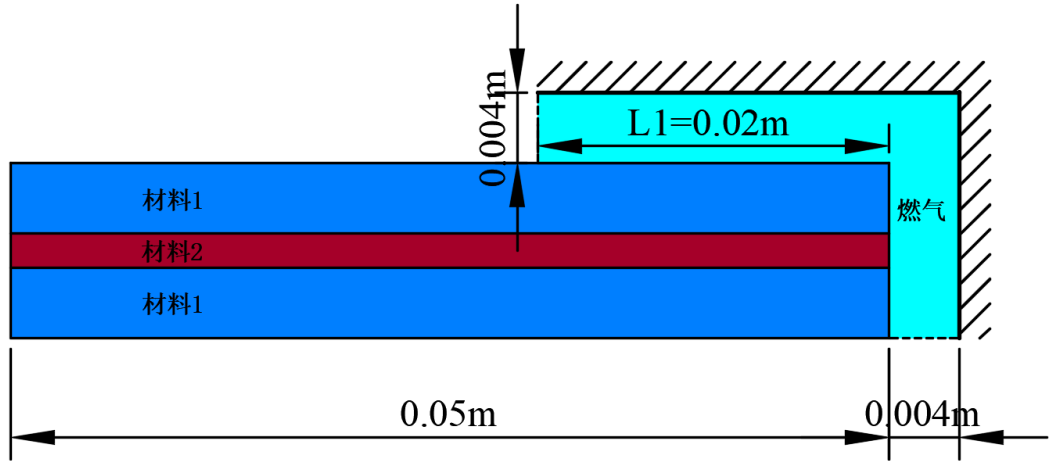


图 1 气体容积物理模型

假设燃气分布在距离材料 $d = 0.004m$ 的区域内，本题中的辐射只考虑右边界和右上边界，可简化为燃气分布在上图绿色区域。查阅文献【1】，右边界和右上边界的气体容积均选取平行平板模型，则平均射线路长为 $s = 1.8d$ 。

在火箭发动机燃烧产物的辐射中，起着主要作用的气体组分是 H_2O, CO_2, NO, HF ，其中三原子气体比起双原子气体辐射更为强烈。查阅文献【2】，对于含 C, H, O, N 的推进剂燃烧产物，热辐射主要取决于其中三原子水蒸气和二氧化碳的含量，燃气发射率的计算公式为

$$\varepsilon_g = \varepsilon_{H_2O} + \varepsilon_{CO_2} - \Delta\varepsilon$$

这里取 $\Delta\varepsilon = \varepsilon_{H_2O} \cdot \varepsilon_{CO_2}$

定义燃气参数：燃气温度 $T_g = 923k$ ， $P_{\text{总}} = 10MPa$ ，选取 85 系列推进剂热力计算得水蒸气和二氧化碳分压

$$P_{H_2O} = 0.2404 \times 10MPa = 24atm, P_{CO_2} = 0.08367 \times 10MPa = 8.3atm$$

$$P_{CO_2} \cdot s = 8.3 \times 1.8 \times 0.004 = 0.05976 atm \cdot m$$

$$P_{H_2O} \cdot s = 24 \times 1.8 \times 0.004 = 0.1728 atm \cdot m$$

查文献【1】p427 中气体发射率经验确定曲线得

$$\begin{cases} \varepsilon_{CO_2}^* = 0.1 \\ C_{CO_2} = 1.2 \\ \varepsilon_{CO_2} = C_{CO_2} \cdot \varepsilon_{CO_2}^* = 0.12 \end{cases}$$

$$\begin{cases} \varepsilon_{0,H_2O} = 0.18 \\ n = 1 + K_{H_2O} P_{H_2O} = 8 \\ \varepsilon_{H_2O} = 1 - (1 - \varepsilon_{0,H_2O})^n = 0.7956 \end{cases}$$

由上，可得燃气发射率

$$\varepsilon_g = \varepsilon_{H_2O} + \varepsilon_{CO_2} - \Delta\varepsilon = 0.82$$

2) 系统发射率

材料 1 的密度为 7830 kg/m^3 ，与钢接近，查阅【1】得 $\varepsilon_{\text{wall1}} = \varepsilon_{\text{钢}} = 0.82$ ；材料 2 为推进剂，大部分非金属材料的发射率都很高，一般在 0.85~0.95 之间，且与表面状况关系不大，在缺乏资料时，近似取 $\varepsilon_{\text{wall2}} = 0.90$ 。

因此，燃气与材料 1 之间的系统发射率为

$$\varepsilon_1 = \frac{1}{\frac{1}{\varepsilon_{\text{wall1}}} + \frac{1}{\varepsilon_g} - 1} = 0.6949$$

燃气与材料 2 之间的系统发射率为

$$\varepsilon_2 = \frac{1}{\frac{1}{\varepsilon_{\text{wall2}}} + \frac{1}{\varepsilon_g} - 1} = 0.7515$$

2. 对流换热系数

对流换热发生在左上边界，且其数值与壁面和大气温差有关。考虑到本题左上壁面温度随时间变化，故推导对流换热系数与壁面温度函数关系。选用文献【3】中均匀壁温边界条件的大空间自然对流实验关联式（水平热面向上的情形），则对流换热系数表达式如下，

$$h = Nu \frac{K_{\text{air}}}{l}, Nu = 0.54(Gr \cdot Pr)^{0.25}, Gr = \frac{ga_v(T_{\text{wall}} - T_{\text{air}})l^3}{\nu^2}$$

取定性温度 $t_m = \frac{t_{\text{wall}} + t_{\text{air}}}{2} (\text{°C})$ ，特征长度 $l = 0.03 \text{ m}$ ， $g = 9.8 \text{ m/s}^2$ ，

$$a_v = \frac{1}{273 + t_m},$$

应用常压下干空气的热物理性质表拟合出 $20\text{°C} \sim 180\text{°C}$ 范围内空气的普朗克数 Pr 和运动粘度 a_v 随定性温度 t_m 的变化关系，

$$Pr = 4 \times 10^{-14} t_m^6 - 3 \times 10^{-11} t_m^5 + 6 \times 10^{-9} t_m^4 - 7 \times 10^{-7} t_m^3 + 4 \times 10^{-5} t_m^2 - 0.0012 t_m + 0.7163$$

$$\nu = 9 \times 10^{-11} t_m^2 + 9.13 \times 10^{-8} t_m + 1.3175 \times 10^{-5}$$

$$Gr = \frac{ga_v(T_{\text{wall}} - 293) \times 0.03^3}{\nu^2}$$

$$Nu = 0.54(Gr \cdot Pr)^{0.25} \quad 10^4 \leq Gr \cdot Pr \leq 10^7$$

利用以上各式求出努塞尔数，代入下式求出对流换热系数表达式，用于编程计算和 UDF 函数编写，

$$h = Nu \frac{K_{\text{air}}}{l} = Nu \times \frac{2.59}{3}$$

三、 公式推导

1. 控制方程

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + S_u$$

初始条件: $T|_{t=0} = 293 \text{ K}$

边界条件: W: $\frac{\partial T}{\partial x}|_{x=0} = 0$

S: $\frac{\partial T}{\partial y}|_{y=0} = 0$

NL: $q_{NL} = h(T_{air} - T) = k \frac{\partial T}{\partial y}|_{y=2H+h, 0 \leq x \leq L-L_1}$

NR: $q_{NR} = \varepsilon \sigma (T_{gas}^4 - T^4) = k \frac{\partial T}{\partial y}|_{y=2H+h, L-L_1 \leq x \leq L}$

E: $q_E = \varepsilon \sigma (T_{gas}^4 - T^4) = k \frac{\partial T}{\partial x}|_{x=L}$

2. 生成离散网格

总共分为 25 种情况，包括 4 个角点，左上侧自然对流边界，右上侧辐射换热边界，右侧辐射换热边界（材料 1 内部、材料 2 内部、材料 12 交界处、材料 21 交界处共 6 种），左侧绝热边界（材料 1 内部、材料 2 内部、材料 12 交界处、材料 21 交界处共 6 种），下侧绝热边界，内部节点（材料 1 内部、材料 2 内部、材料 12 交界处、材料 21 交界处共 6 种）。

8	9	10	11
2	20		13
7	25		18
6	24		17
5	23		16
4	22		15
3	21		14
2	20		13
1	19		12

图 2 离散网格的 25 种情况

3. 构造离散方程

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + S_u$$

两侧同时对控制体及时间积分，

$$\begin{aligned} \int_{\Delta V} \left[\int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t} dt \right] dV &= \int_{\Delta V} \left[\int_t^{t+\Delta t} \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) dt \right] dV + \int_{\Delta V} \left[\int_t^{t+\Delta t} \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) dt \right] dV + \int_{\Delta V} \left[\int_t^{t+\Delta t} S_u dt \right] dV \\ \int_s^n \int_w^e \int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t} dx dy dt &= \int_s^n \int_w^e \int_t^{t+\Delta t} \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) dx dy dt + \int_s^n \int_w^e \int_t^{t+\Delta t} \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) dx dy dt + \int_t^{t+\Delta t} \bar{S} \cdot \Delta V dt \end{aligned}$$

对时间后差，对空间中心差分，同时对温度加权处理，有

$$\begin{aligned} \frac{\partial T}{\partial t} &= \frac{T_p - T_p^0}{\Delta t} \\ \left(\frac{\partial T}{\partial x} \right)_e &= \frac{T_E - T_p}{\delta x_{EP}}, \quad \left(\frac{\partial T}{\partial x} \right)_w = \frac{T_p - T_w}{\delta x_{PW}}, \quad \left(\frac{\partial T}{\partial y} \right)_n = \frac{T_N - T_p}{\delta y_{NP}}, \quad \left(\frac{\partial T}{\partial y} \right)_s = \frac{T_p - T_S}{\delta y_{PS}} \\ T &= \theta T + (1 - \theta) T^0 \end{aligned}$$

则可得

$$\begin{aligned} \rho c (T_p - T_p^0) \frac{\Delta x \Delta y}{\Delta t} &= \theta \left[\left(k_e \frac{T_E - T_p}{\delta x_{EP}} \right) - \left(k_w \frac{T_p - T_w}{\delta x_{PW}} \right) \right] \Delta y + (1 - \theta) \left[\left(k_e \frac{T_E^0 - T_p^0}{\delta x_{EP}} \right) - \left(k_w \frac{T_p^0 - T_w^0}{\delta x_{PW}} \right) \right] \Delta y \\ &+ \theta \left[\left(k_n \frac{T_N - T_p}{\delta y_{NP}} \right) - \left(k_s \frac{T_p - T_S}{\delta y_{PS}} \right) \right] \Delta x + (1 - \theta) \left[\left(k_n \frac{T_N^0 - T_p^0}{\delta y_{NP}} \right) - \left(k_s \frac{T_p^0 - T_S^0}{\delta y_{PS}} \right) \right] \Delta x \\ &+ S_u + S_p [\theta T_p + (1 - \theta) T_p^0] \end{aligned}$$

按节点整理系数，则有：

$$\begin{aligned} a_p T_p &= a_E [\theta T_E + (1 - \theta) T_E^0] + a_w [\theta T_w + (1 - \theta) T_w^0] + a_N [\theta T_N + (1 - \theta) T_N^0] + a_S [\theta T_S + (1 - \theta) T_S^0] \\ &+ [a_p^0 - (1 - \theta) a_E - (1 - \theta) a_w - (1 - \theta) a_N - (1 - \theta) a_S + (1 - \theta) S_p] T_p^0 + S_u \end{aligned}$$

其中，

$$\begin{aligned} a_E &= \frac{k_e \Delta y}{\delta x_{EP}}, \quad a_w = \frac{k_w \Delta y}{\delta x_{PW}}, \quad a_N = \frac{k_n \Delta x}{\delta y_{NP}}, \quad a_S = \frac{k_s \Delta x}{\delta y_{PS}} \\ a_p &= a_p^0 + \theta (a_E + a_w + a_N + a_S - S_p) \end{aligned}$$

依据界面上热流密度连续，可得

$$\frac{\delta y_{NP}}{k_n} = \frac{\delta y_{nP}}{k_p} + \frac{\delta y_{Nn}}{k_N}$$

对于两种材料交界处，若 $\delta y_{nP} = \delta y_{Nn} = \frac{1}{2} \delta y_{NP}$ ，则有 $k_n = \frac{2k_p k_N}{k_p + k_N}$

同理，有 $k_s = \frac{2k_p k_S}{k_p + k_S}$ 。

下面以显式格式（ $\theta = 0$ ）为例。

1) 左下角节点

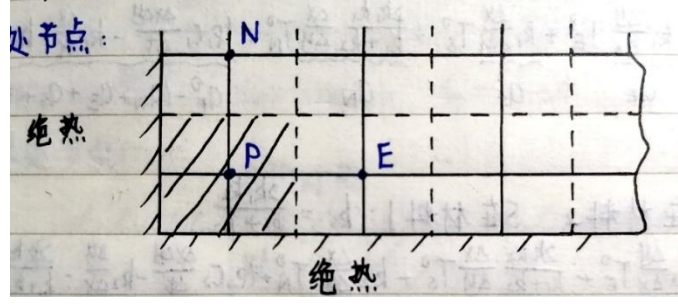


图 3 左下角节点处空间离散

绝热边界: $q_w = k_w \frac{T_P^0 - T_W^0}{\Delta x} = 0$, $q_s = k_s \frac{T_P^0 - T_S^0}{\Delta y} = 0$

$$\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_P^0}_{a_P} = \underbrace{0 \cdot T_W^0}_{a_W} + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{0 \cdot T_S^0}_{a_S} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} - k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta x}{\Delta y} \right) T_P^0$$

$$a_P = a_P^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

W 边界处节点

绝热边界: $q_w = k_w \frac{T_P^0 - T_W^0}{\Delta x} = 0$

$$\rho c \frac{\Delta x \Delta y}{\Delta t} T_P^0 = 0 \cdot T_W^0 + k_e \frac{\Delta y}{\Delta x} T_E^0 + k_s \frac{\Delta x}{\Delta y} T_S^0 + k_n \frac{\Delta x}{\Delta y} T_N^0 + \left(\rho c \frac{\Delta x \Delta y}{\Delta t} - k_e \frac{\Delta y}{\Delta x} - k_s \frac{\Delta x}{\Delta y} - k_n \frac{\Delta x}{\Delta y} \right) T_P^0$$

2) W 边界处节点, 对材料 1 内部:

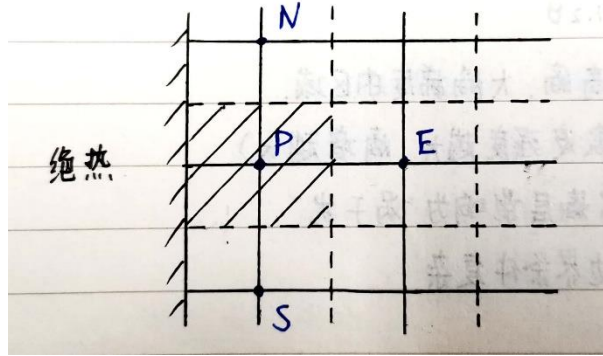


图 4 W 边界材料内部空间离散

$$\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_P^0}_{a_P} = \underbrace{0 \cdot T_W^0}_{a_W} + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} - k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta x}{\Delta y} - k_1 \frac{\Delta x}{\Delta y} \right) T_P^0$$

$$a_P = a_P^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

3) W 边界处节点, 对材料 2 内部:

空间离散如图 4。

$$\underbrace{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_P}_{a_P} = \underbrace{0 \cdot T_W^0}_{a_W} + \underbrace{k_2 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{k_2 \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{k_2 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}}_{a_P^0} - \underbrace{k_2 \frac{\Delta y}{\Delta x} - k_2 \frac{\Delta x}{\Delta y} - k_2 \frac{\Delta x}{\Delta y}}_{(a_W + a_E + a_S + a_N - S_P)} \right) T_P^0$$

$$a_P = a_P^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

4) W 边界处节点，对材料 1、2 交界处，情况一：

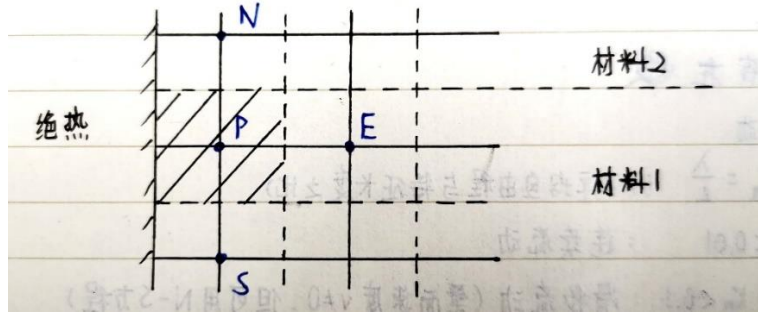


图 5 W 边界材料 1、2 交界处情况一

P、E、S 点在材料 1，N 点在材料 2， $k_n = \frac{2k_1 k_2}{k_1 + k_2}$

$$\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_P}_{a_P} = \underbrace{0 \cdot T_W^0}_{a_W} + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_P^0} - \underbrace{k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta x}{\Delta y} - \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}}_{(a_W + a_E + a_S + a_N - S_P)} \right) T_P^0$$

$$a_P = a_P^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

5) W 边界处节点，对材料 1、2 交界处，情况二：

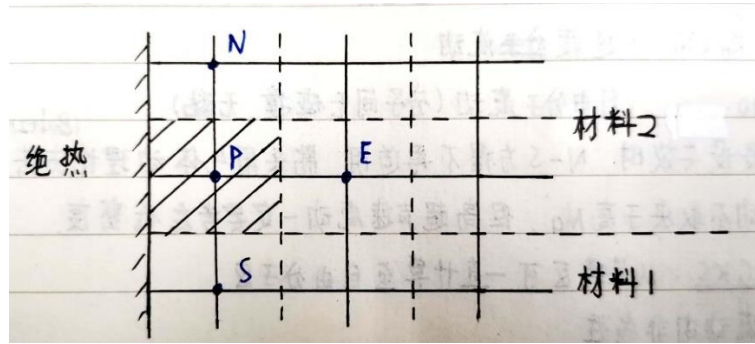


图 6 W 边界材料 1、2 交界处情况二

S 点在材料 1，N、P、E 点在材料 2， $k_s = \frac{2k_1 k_2}{k_1 + k_2}$

$$\underbrace{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_P}_{a_P} = \underbrace{0 \cdot T_W^0}_{a_W} + \underbrace{k_2 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{k_2 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}}_{a_P^0} - \underbrace{k_2 \frac{\Delta y}{\Delta x} - \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} - k_2 \frac{\Delta x}{\Delta y}}_{(a_W + a_E + a_S + a_N - S_P)} \right) T_P^0$$

$$a_P = a_P^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

$$a_p = a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

6) W 边界处节点，对材料 2、1 交界处，情况一：

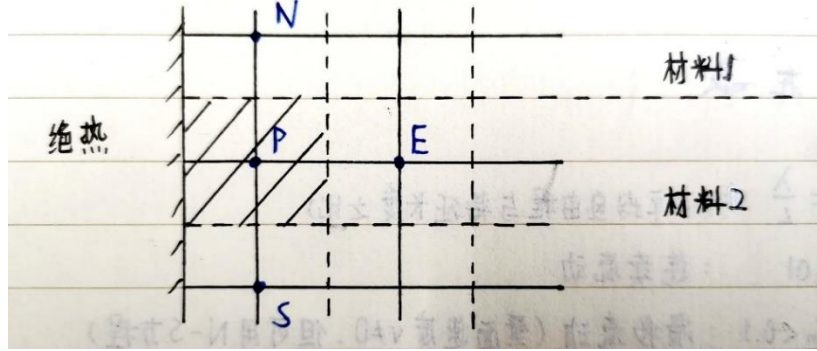


图 7 W 边界材料 2、1 交界处情况一

P、E、S 点在材料 2，N 点在材料 1， $k_n = \frac{2k_1 k_2}{k_1 + k_2}$

$$\underbrace{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}}_{a_p} T_p = \underbrace{0}_{a_w} \cdot T_w^0 + \underbrace{k_2 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{k_2 \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}}_{a_p^0} - \underbrace{k_2 \frac{\Delta y}{\Delta x}}_{a_w} - \underbrace{k_2 \frac{\Delta x}{\Delta y}}_{a_E} - \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}}_{a_N} - \underbrace{S_p}_{S_p} \right) T_p^0$$

$$a_p = a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

7) W 边界处节点，对材料 2、1 交界处，情况二：

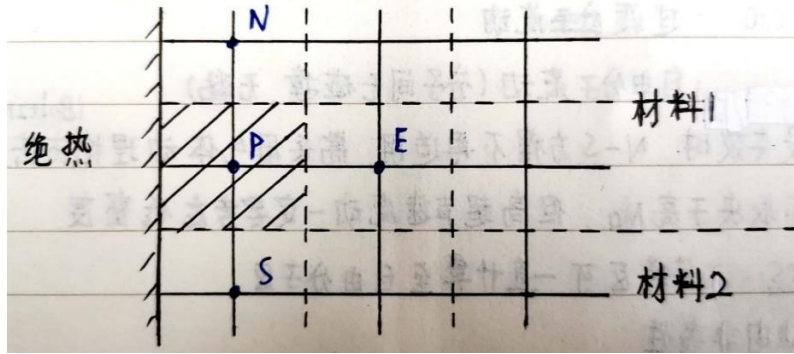


图 8 W 边界材料 2、1 交界处情况二

S 点在材料 2，N、P、E 点在材料 1， $k_s = \frac{2k_1 k_2}{k_1 + k_2}$

$$\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_p} T_p = \underbrace{0}_{a_w} \cdot T_w^0 + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_p^0} - \underbrace{k_1 \frac{\Delta y}{\Delta x}}_{a_E} - \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}}_{a_S} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_N} - \underbrace{S_p}_{S_p} \right) T_p^0$$

$$a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

8) 左上角处节点

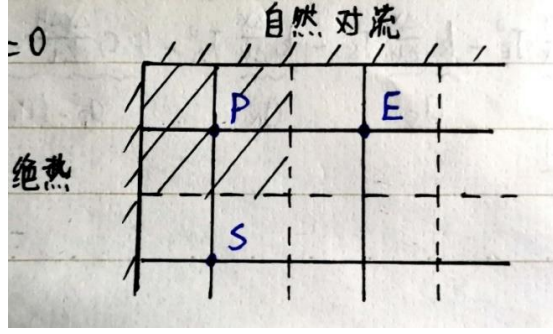


图 9 左上角处节点空间离散

绝热边界: $q_w = k_w \frac{T_P^0 - T_W^0}{\Delta x} = 0$

自然对流边界: $q_{NL} = h(T_{air} - T_P^0) = k_n \frac{T_N^0 - T_P^0}{\Delta y}$

$$\begin{aligned} \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_P = & \underbrace{0}_{a_P} \cdot T_W^0 + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{0 T_N^0}_{a_N} + \left(\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_P^0} - \underbrace{k_1 \frac{\Delta y}{\Delta x}}_{a_W} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_S} - \underbrace{h \Delta x}_{S_p} \right) T_P^0 + \underbrace{h \Delta x T_{air}}_{S_u} \\ & a_P \quad a_W \quad a_E \quad a_S \quad a_N \quad a_P^0 - (a_W + a_E + a_S + a_N - S_p) \quad S_u \\ & a_P = a_P^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = h \Delta x T_{air}, S_p = -h \Delta x \end{aligned}$$

9) NL 边界处节点

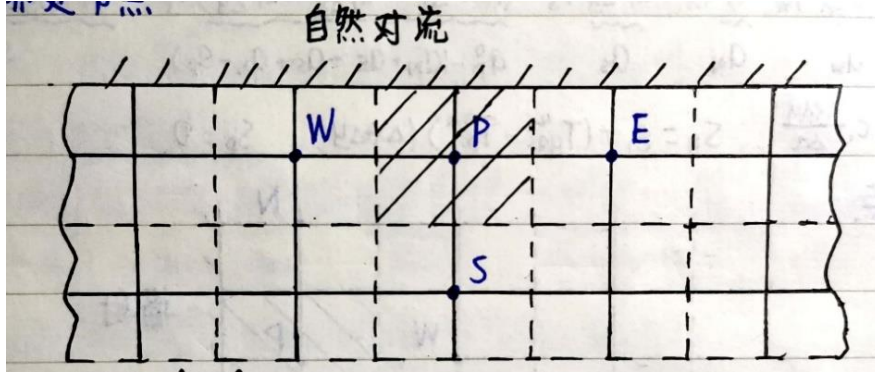


图 10 NL 边界处空间离散

自然对流边界: $q_{NL} = h(T_{air} - T_P^0) = k_n \frac{T_N^0 - T_P^0}{\Delta y}$

$$\begin{aligned} \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_P = & \underbrace{k_1 \frac{\Delta y}{\Delta x} T_W^0}_{a_W} + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_S^0}_{a_S} + \underbrace{0 T_N^0}_{a_N} + \left(\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_P^0} - \underbrace{k_1 \frac{\Delta y}{\Delta x}}_{a_W} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_S} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_N} - \underbrace{h \Delta x}_{S_p} \right) T_P^0 + \underbrace{h \Delta x T_{air}}_{S_u} \\ & a_P \quad a_W \quad a_E \quad a_S \quad a_N \quad a_P^0 - (a_W + a_E + a_S + a_N - S_p) \quad S_u \\ & a_P = a_P^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = h \Delta x T_{air}, S_p = -h \Delta x \end{aligned}$$

10) NL 边界处节点

空间离散如图 10。

辐射边界: $q_{NR} = k_n \frac{T_N^0 - T_P^0}{\Delta y} = \varepsilon \sigma (T_{gas}^4 - T_P^{04})$

$$\begin{aligned}
& \frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_1 \frac{\Delta y}{\Delta x} \cdot T_w^0}{a_w} + \frac{k_1 \frac{\Delta y}{\Delta x} T_E^0}{a_E} + \frac{k_1 \frac{\Delta x}{\Delta y} \cdot T_S^0}{a_S} + \frac{0 \cdot T_N^0}{a_N} + \left(\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}{a_p^0 - (a_w + a_E + a_S + a_N - S_p)} T_p^0 + \frac{\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta x}{S_u} \right) \\
& a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta x, S_p = 0
\end{aligned}$$

11) 右上角处节点

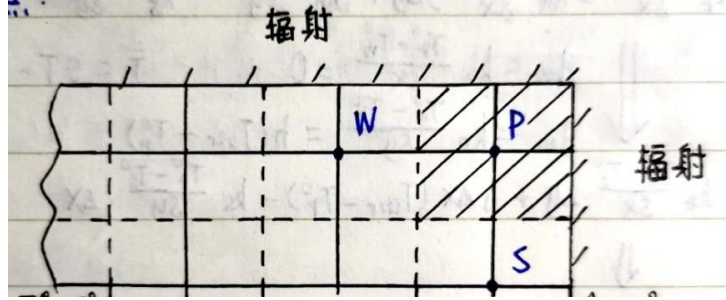


图 11 右上角处空间离散

$$\begin{aligned}
& \text{辐射边界: } q_{NR} = k_n \frac{T_N^0 - T_p^0}{\Delta y} = \varepsilon \sigma (T_{gas}^4 - T_p^{04}), \quad q_E = k_e \frac{T_E^0 - T_p^0}{\Delta x} = \varepsilon \sigma (T_{gas}^4 - T_p^{04}) \\
& \frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_1 \frac{\Delta y}{\Delta x} \cdot T_w^0}{a_w} + \frac{0 \cdot T_E^0}{a_E} + \frac{k_1 \frac{\Delta x}{\Delta y} \cdot T_S^0}{a_S} + \frac{0 \cdot T_N^0}{a_N} + \left(\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}{a_p^0 - (a_w + a_E + a_S + a_N - S_p)} T_p^0 + \frac{\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) (\Delta x + \Delta y)}{S_u} \right) \\
& a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) (\Delta x + \Delta y), S_p = 0
\end{aligned}$$

12) 右下角处节点

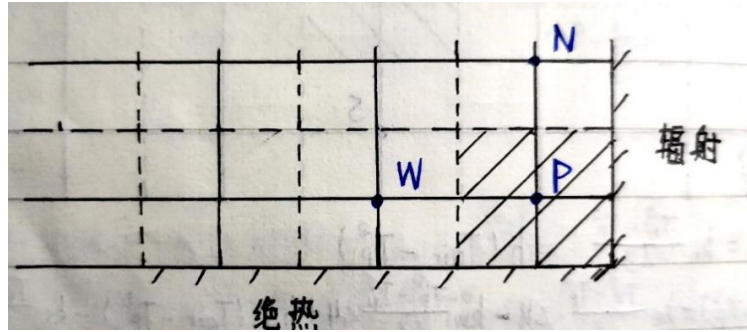


图 12 右下角处空间离散

$$\begin{aligned}
& \text{绝热边界: } q_S = k_s \frac{T_p^0 - T_S^0}{\Delta x} = 0 \\
& \text{辐射边界: } q_E = k_e \frac{T_E^0 - T_p^0}{\Delta x} = \varepsilon \sigma (T_{gas}^4 - T_p^{04}) \\
& \frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_1 \frac{\Delta y}{\Delta x} T_w^0}{a_w} + \frac{0 \cdot T_E^0}{a_E} + \frac{0 \cdot T_S^0}{a_S} + \frac{k_1 \frac{\Delta x}{\Delta y} T_N^0}{a_N} + \left(\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}{a_p^0 - (a_w + a_E + a_S + a_N - S_p)} T_p^0 + \frac{\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta y}{S_u} \right) \\
& a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta y, S_p = 0
\end{aligned}$$

E 边界处节点

$$\text{辐射边界: } q_E = k_e \frac{T_E^0 - T_P^0}{\Delta x} = \varepsilon \sigma (T_{gas}^4 - T_P^{04})$$

$$\rho c \frac{\Delta x \Delta y}{\Delta t} T_P = k_w \frac{\Delta y}{\Delta x} \cdot T_W^0 + 0 \cdot T_E^0 + k_s \frac{\Delta x}{\Delta y} T_S^0 + k_n \frac{\Delta x}{\Delta y} T_N^0 + \left(\rho c \frac{\Delta x \Delta y}{\Delta t} - k_w \frac{\Delta y}{\Delta x} - k_s \frac{\Delta x}{\Delta y} - k_n \frac{\Delta x}{\Delta y} \right) T_P^0 + \varepsilon \sigma (T_{gas}^4 - T_P^{04}) \Delta y$$

13) E 边界处节点，对材料 1 内部：

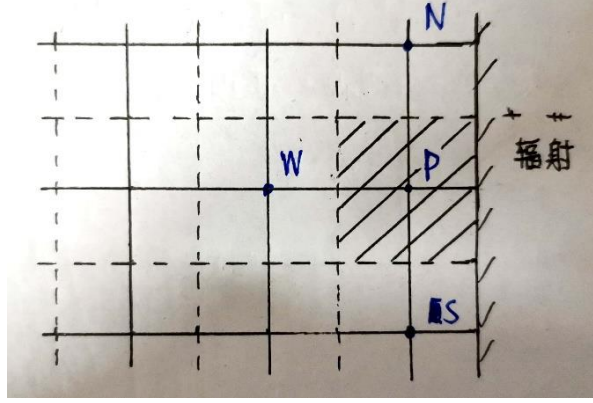


图 13 E 边界材料内部空间离散

$$\begin{aligned} \frac{\rho_1 c_1 \Delta x \Delta y}{\Delta t} T_P &= \frac{k_1 \Delta y}{\Delta x} T_W^0 + 0 \cdot T_E^0 + \frac{k_1 \Delta x}{\Delta y} T_S^0 + \frac{k_1 \Delta x}{\Delta y} T_N^0 + \left(\frac{\rho_1 c_1 \Delta x \Delta y}{\Delta t} - k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta x}{\Delta y} - k_1 \frac{\Delta x}{\Delta y} \right) T_P^0 + \frac{\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y}{S_u} \\ a_P & \quad a_W \quad a_E \quad a_S \quad a_N \quad a_P^0 - (a_W + a_E + a_S + a_N - S_p) \quad S_u \\ a_P &= a_P^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y, S_p = 0 \end{aligned}$$

14) E 边界处节点，对材料 2 内部：

空间离散如图 13。

$$\begin{aligned} \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_P &= k_2 \frac{\Delta y}{\Delta x} T_W^0 + 0 \cdot T_E^0 + k_2 \frac{\Delta x}{\Delta y} T_S^0 + k_2 \frac{\Delta x}{\Delta y} T_N^0 + \left(\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} - k_2 \frac{\Delta y}{\Delta x} - k_2 \frac{\Delta x}{\Delta y} - k_2 \frac{\Delta x}{\Delta y} \right) T_P^0 + \frac{\varepsilon_2 \sigma (T_{gas}^4 - T_P^{04}) \Delta y}{S_u} \\ a_P & \quad a_W \quad a_E \quad a_S \quad a_N \quad a_P^0 - (a_W + a_E + a_S + a_N - S_p) \quad S_u \\ a_P &= a_P^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_2 \sigma (T_{gas}^4 - T_P^{04}) \Delta y, S_p = 0 \end{aligned}$$

15) E 边界处节点，对材料 1、2 交界处，情况一：

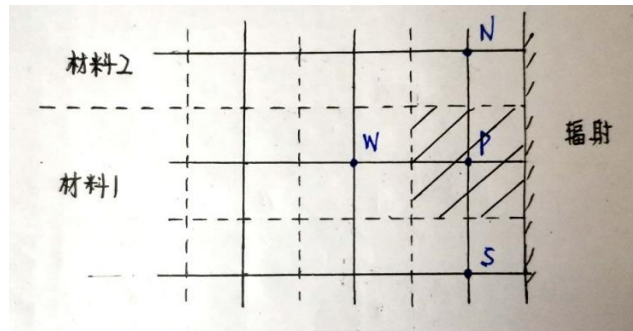


图 14 E 边界材料 1、2 交界处情况一

$$P、W、S \text{ 点在材料 1, } N \text{ 点在材料 2, } k_n = \frac{2k_1 k_2}{k_1 + k_2}$$

$$\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_1 \frac{\Delta y}{\Delta x} T_w^0 + 0 \cdot T_E^0 + k_1 \frac{\Delta x}{\Delta y} \cdot T_S^0 + \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_N^0}{a_w} + \left(\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} - k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta x}{\Delta y} - \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} \right) T_p^0 + \frac{\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta y}{a_p - (a_w + a_E + a_S + a_N - S_p)} S_u$$

$$a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta y, S_p = 0$$

16) E 边界处节点，对材料 1、2 交界处，情况二：

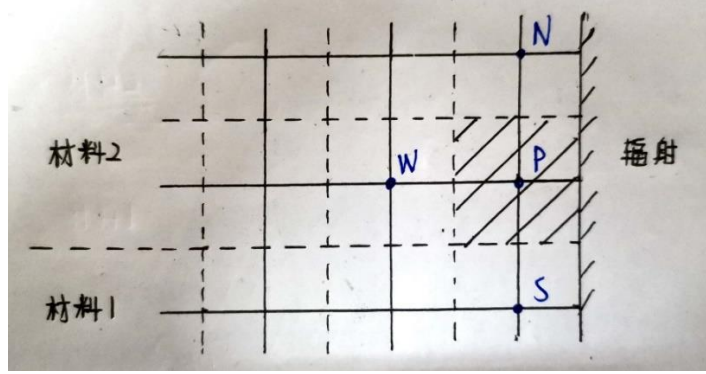


图 15 E 边界材料 1、2 交界处情况二

S 点在材料 1，N、P、W 点在材料 2， $k_s = \frac{2k_1 k_2}{k_1 + k_2}$

$$\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_2 \frac{\Delta y}{\Delta x} T_w^0 + 0 \cdot T_E^0 + \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} \cdot T_S^0 + k_2 \frac{\Delta x}{\Delta y} T_N^0}{a_w} + \left(\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} - k_2 \frac{\Delta y}{\Delta x} - \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} - k_2 \frac{\Delta x}{\Delta y} \right) T_p^0 + \frac{\varepsilon_2 \sigma (T_{gas}^4 - T_p^{04}) \Delta y}{a_p - (a_w + a_E + a_S + a_N - S_p)} S_u$$

$$a_p = a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_2 \sigma (T_{gas}^4 - T_p^{04}) \Delta y, S_p = 0$$

17) E 边界处节点，对材料 2、1 交界处，情况一：

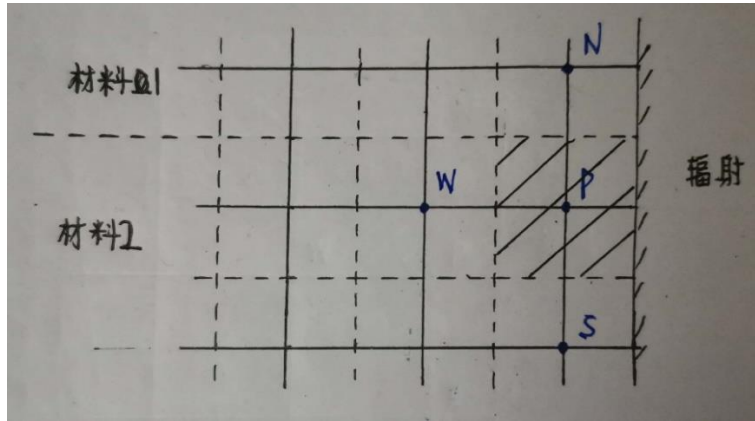


图 16 E 边界材料 2、1 交界处情况一

P、W、S 点在材料 2，N 点在材料 1， $k_n = \frac{2k_1 k_2}{k_1 + k_2}$

$$\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_2 \frac{\Delta y}{\Delta x} T_w^0 + 0 \cdot T_E^0 + k_2 \frac{\Delta x}{\Delta y} \cdot T_S^0 + \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_N^0}{a_w} + \left(\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} - k_2 \frac{\Delta y}{\Delta x} - \frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} - k_2 \frac{\Delta x}{\Delta y} \right) T_p^0 + \frac{\varepsilon_2 \sigma (T_{gas}^4 - T_p^{04}) \Delta y}{a_p - (a_w + a_E + a_S + a_N - S_p)} S_u$$

$$a_p = a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_2 \sigma (T_{gas}^4 - T_p^{04}) \Delta y, S_p = 0$$

18) E 边界处节点，对材料 2、1 交界处，情况二：

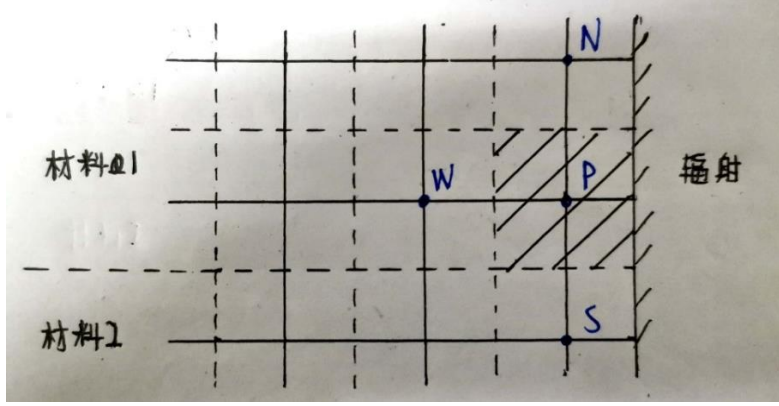


图 17 E 边界材料 2、1 交界处情况二

S 点在材料 2，N、P、W 点在材料 1， $k_s = \frac{2k_1 k_2}{k_1 + k_2}$

$$\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}_{a_p} = \underbrace{k_1 \frac{\Delta y}{\Delta x} T_w^0}_{a_w} + \underbrace{0 \cdot T_E^0}_{a_E} + \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y} T_s^0}_{a_s} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_p^0} - \underbrace{k_1 \frac{\Delta y}{\Delta x}}_{a_w} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_E} - \underbrace{\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}}_{a_s} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_N} - \underbrace{S_p}_{S_p} \right) T_p^0 + \underbrace{\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta y}_{S_u}$$

$$a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = \varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta y, S_p = 0$$

19) S 边界处节点

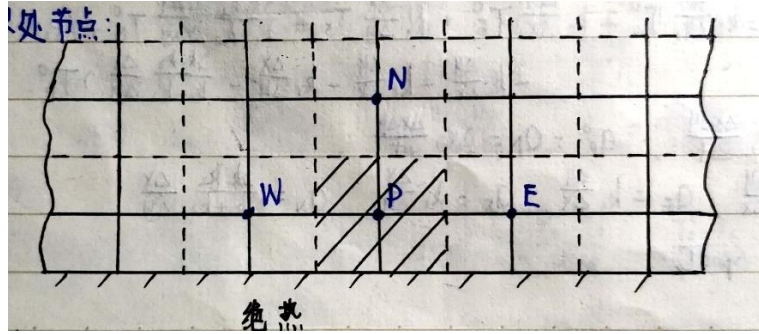


图 18 S 边界处空间离散

绝热边界： $q_s = k_s \frac{T_p^0 - T_s^0}{\Delta x} = 0$

$$\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}_{a_p} = \underbrace{k_1 \frac{\Delta y}{\Delta x} T_w^0}_{a_w} + \underbrace{k_1 \frac{\Delta y}{\Delta x} T_E^0}_{a_E} + \underbrace{0 \cdot T_s^0}_{a_s} + \underbrace{k_1 \frac{\Delta x}{\Delta y} T_N^0}_{a_N} + \left(\underbrace{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}}_{a_p^0} - \underbrace{k_1 \frac{\Delta y}{\Delta x}}_{a_w} - \underbrace{k_1 \frac{\Delta y}{\Delta x}}_{a_E} - \underbrace{k_1 \frac{\Delta x}{\Delta y}}_{a_N} - \underbrace{S_p}_{S_p} \right) T_p^0$$

$$a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

内部节点

$$\rho c \frac{\Delta x \Delta y}{\Delta t} T_p = k_w \frac{\Delta y}{\Delta x} T_w^0 + k_e \frac{\Delta y}{\Delta x} T_e^0 + k_s \frac{\Delta x}{\Delta y} T_s^0 + k_n \frac{\Delta x}{\Delta y} T_n^0 + \left(\rho c \frac{\Delta x \Delta y}{\Delta t} - k_w \frac{\Delta y}{\Delta x} - k_e \frac{\Delta y}{\Delta x} - k_s \frac{\Delta x}{\Delta y} - k_n \frac{\Delta x}{\Delta y} \right) T_p^0$$

20) 内部节点，对材料 1 内部：

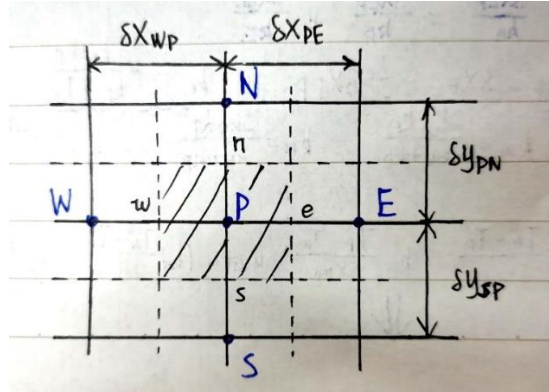


图 19 内部节点空间离散

$$\begin{aligned} \frac{\rho_1 c_1 \Delta x \Delta y}{\Delta t} T_p &= \frac{k_1 \Delta y}{\Delta x} T_w^0 + \frac{k_1 \Delta y}{\Delta x} T_e^0 + \frac{k_1 \Delta x}{\Delta y} T_s^0 + \frac{k_1 \Delta x}{\Delta y} T_n^0 + \left(\frac{\rho_1 c_1 \Delta x \Delta y}{\Delta t} - k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta y}{\Delta x} - k_1 \frac{\Delta x}{\Delta y} - k_1 \frac{\Delta x}{\Delta y} \right) T_p^0 \\ a_p & \quad a_w \quad a_E \quad a_S \quad a_N \quad a_p^0 - (a_w + a_E + a_S + a_N - S_p) \\ a_p &= a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0 \end{aligned}$$

21) 内部节点，对材料 2 内部：

空间离散如图 19。

$$\begin{aligned} \frac{\rho_2 c_2 \Delta x \Delta y}{\Delta t} T_p &= k_2 \frac{\Delta y}{\Delta x} T_w^0 + k_2 \frac{\Delta y}{\Delta x} T_e^0 + k_2 \frac{\Delta x}{\Delta y} T_s^0 + k_2 \frac{\Delta x}{\Delta y} T_n^0 + \left(\frac{\rho_2 c_2 \Delta x \Delta y}{\Delta t} - k_2 \frac{\Delta y}{\Delta x} - k_2 \frac{\Delta y}{\Delta x} - k_2 \frac{\Delta x}{\Delta y} - k_2 \frac{\Delta x}{\Delta y} \right) T_p^0 \\ a_p & \quad a_w \quad a_E \quad a_S \quad a_N \quad a_p^0 - (a_w + a_E + a_S + a_N - S_p) \\ a_p &= a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0 \end{aligned}$$

22) 内部节点，对材料 1、2 交界处，情况一：

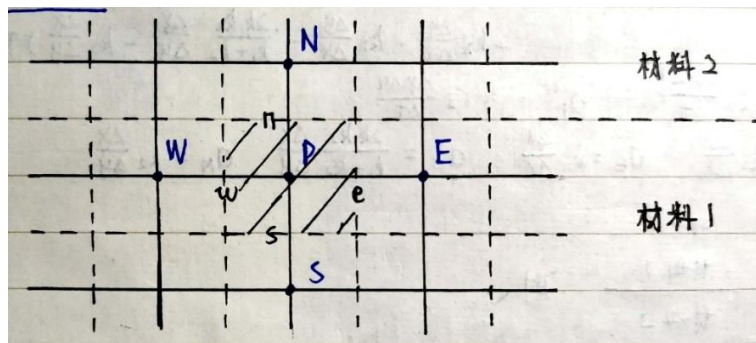


图 20 内部节点材料 1、2 交界处情况一

W、P、E、S 点在材料 1，N 点在材料 2， $k_n = \frac{2k_1 k_2}{k_1 + k_2}$

$$\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_1 \frac{\Delta y}{\Delta x} T_w^0}{a_w} + \frac{k_1 \frac{\Delta y}{\Delta x} T_e^0}{a_e} + \frac{k_1 \frac{\Delta x}{\Delta y} T_s^0}{a_s} + \frac{2k_1 k_2 \frac{\Delta x}{k_1 + k_2} T_N^0}{a_N} + \left(\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p^0}{a_p^0 - (a_w + a_e + a_s + a_N - S_p)} \right) T_p^0$$

$$a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

23) 内部节点，对材料 1、2 交界处，情况二：

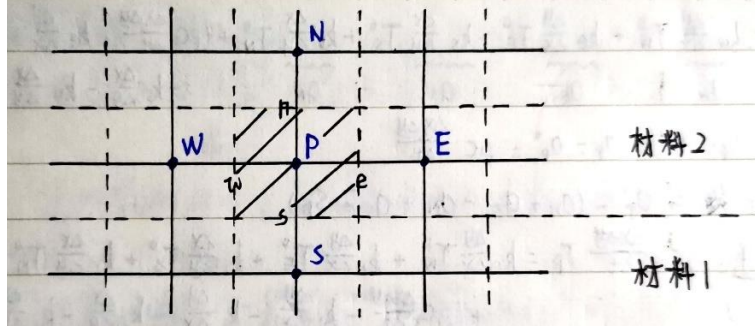


图 21 内部节点材料 1、2 交界处情况二

S 点在材料 1， N、W、P、E 点在材料 2， $k_s = \frac{2k_1 k_2}{k_1 + k_2}$

$$\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_2 \frac{\Delta y}{\Delta x} T_w^0}{a_w} + \frac{k_2 \frac{\Delta y}{\Delta x} T_e^0}{a_e} + \frac{2k_1 k_2 \frac{\Delta x}{k_1 + k_2} T_s^0}{a_s} + \frac{k_2 \frac{\Delta x}{\Delta y} T_N^0}{a_N} + \left(\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_p^0}{a_p^0 - (a_w + a_e + a_s + a_N - S_p)} \right) T_p^0$$

$$a_p = a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

24) 内部节点，对材料 2、1 交界处，情况一：

W、P、E、S 点在材料 2， N 点在材料 1， $k_n = \frac{2k_1 k_2}{k_1 + k_2}$

$$\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_2 \frac{\Delta y}{\Delta x} T_w^0}{a_w} + \frac{k_2 \frac{\Delta y}{\Delta x} T_e^0}{a_e} + \frac{k_2 \frac{\Delta x}{\Delta y} T_s^0}{a_s} + \frac{2k_1 k_2 \frac{\Delta x}{k_1 + k_2} T_N^0}{a_N} + \left(\frac{\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t} T_p^0}{a_p^0 - (a_w + a_e + a_s + a_N - S_p)} \right) T_p^0$$

$$a_p = a_p^0 = \rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

25) 内部节点，对材料 2、1 交界处，情况二：

S 点在材料 2， N、W、P、E 点在材料 1， $k_s = \frac{2k_1 k_2}{k_1 + k_2}$

$$\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p}{a_p} = \frac{k_1 \frac{\Delta y}{\Delta x} T_w^0}{a_w} + \frac{k_1 \frac{\Delta y}{\Delta x} T_e^0}{a_e} + \frac{2k_1 k_2 \frac{\Delta x}{k_1 + k_2} T_s^0}{a_s} + \frac{k_1 \frac{\Delta x}{\Delta y} T_N^0}{a_N} + \left(\frac{\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t} T_p^0}{a_p^0 - (a_w + a_e + a_s + a_N - S_p)} \right) T_p^0$$

$$a_p = a_p^0 = \rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}, S_u = 0, S_p = 0$$

4. 边界系数

1) 显式格式:

$$a_p T_p = a_E T_E^0 + a_W T_W^0 + a_N T_N^0 + a_S T_S^0 + \left[a_p^0 - (a_E + a_W + a_N + a_S - S_p) \right] T_p^0 + S_u$$

其中, $a_p = a_p^0$

2) 半隐式格式:

$$a_p T_p = a_E \frac{T_E + T_E^0}{2} + a_W \frac{T_W + T_W^0}{2} + a_N \frac{T_N + T_N^0}{2} + a_S \frac{T_S + T_S^0}{2} + \left[a_p^0 - \frac{1}{2} (a_E + a_W + a_N + a_S - S_p) \right] T_p^0 + S_u$$

其中, $a_p = a_p^0 + \frac{1}{2} (a_E + a_W + a_N + a_S - S_p)$

3) 隐式格式:

$$a_p T_p = a_E T_E + a_W T_W + a_N T_N + a_S T_S + a_p^0 T_p^0 + S_u$$

其中, $a_p = a_p^0 + (a_E + a_W + a_N + a_S - S_p)$

表 1 显式、半隐式、隐式格式边界系数表

节点	a_p^0	a_W	a_E	a_S	a_N	S_u	S_p
1	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_1 \frac{\Delta y}{\Delta x}$	0	$k_1 \frac{\Delta x}{\Delta y}$	0	0
2	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	$k_1 \frac{\Delta x}{\Delta y}$	0	0
3	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	0	0
4	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_2 \frac{\Delta y}{\Delta x}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$k_2 \frac{\Delta x}{\Delta y}$	0	0
5	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta x}{\Delta y}$	$k_2 \frac{\Delta x}{\Delta y}$	0	0
6	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta x}{\Delta y}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	0	0
7	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_1 \frac{\Delta y}{\Delta x}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$k_1 \frac{\Delta x}{\Delta y}$	0	0
8	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	0	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	0	$h \Delta x T_{air}$	$-h \Delta x$
9	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	0	$h \Delta x T_{air}$	$-h \Delta x$
10	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	0	$\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) \Delta x$	0
11	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	0	$k_1 \frac{\Delta x}{\Delta y}$	0	$\varepsilon_1 \sigma (T_{gas}^4 - T_p^{04}) (\Delta x + \Delta y)$	0

12	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	0	0	$k_1 \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
13	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	0	$k_1 \frac{\Delta x}{\Delta y}$	$k_1 \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
14	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	0	$k_1 \frac{\Delta x}{\Delta y}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
15	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	$k_2 \frac{\Delta y}{\Delta x}$	0	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$k_2 \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
16	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	$k_2 \frac{\Delta y}{\Delta x}$	0	$k_2 \frac{\Delta x}{\Delta y}$	$k_2 \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
17	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	$k_2 \frac{\Delta y}{\Delta x}$	0	$k_2 \frac{\Delta x}{\Delta y}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
18	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	0	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$k_1 \frac{\Delta x}{\Delta y}$	$\varepsilon_1 \sigma (T_{gas}^4 - T_P^{04}) \Delta y$	0
19	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta y}{\Delta x}$	0	$k_1 \frac{\Delta x}{\Delta y}$	0	0
20	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	$k_1 \frac{\Delta x}{\Delta y}$	0	0
21	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta x}{\Delta y}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	0	0
22	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta y}{\Delta x}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$k_2 \frac{\Delta x}{\Delta y}$	0	0
23	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta x}{\Delta y}$	$k_2 \frac{\Delta x}{\Delta y}$	0	0
24	$\rho_2 c_2 \frac{\Delta x \Delta y}{\Delta t}$	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta y}{\Delta x}$	$k_2 \frac{\Delta x}{\Delta y}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	0	0
25	$\rho_1 c_1 \frac{\Delta x \Delta y}{\Delta t}$	$k_1 \frac{\Delta y}{\Delta x}$	$k_1 \frac{\Delta y}{\Delta x}$	$\frac{2k_1 k_2}{k_1 + k_2} \frac{\Delta x}{\Delta y}$	$k_1 \frac{\Delta x}{\Delta y}$	0	0

5. 求解方程组

$$\begin{pmatrix}
 a_p[0][0] & -a_e[0][0] & 0 & 0 & \cdots & -a_n[0][0] & 0 & 0 & \cdots \\
 -a_w[0][1] & a_p[0][1] & -a_e[0][1] & 0 & \cdots & 0 & -a_n[0][1] & 0 & \cdots \\
 0 & -a_w[0][2] & a_p[0][2] & -a_e[0][2] & \cdots & 0 & 0 & -a_n[0][2] & \cdots \\
 \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
 -a_s[1][0] & 0 & 0 & 0 & \cdots & a_p[1][0] & -a_e[1][0] & 0 & \cdots \\
 0 & -a_s[1][1] & 0 & 0 & \cdots & -a_w[1][1] & a_p[1][1] & -a_e[1][1] & \cdots \\
 \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots
 \end{pmatrix}
 \begin{pmatrix}
 T[0][0] \\
 T[0][1] \\
 T[0][2] \\
 \cdots \\
 T[1][0] \\
 T[1][1] \\
 \cdots
 \end{pmatrix}
 =
 \begin{pmatrix}
 a_p^0[0][0]T_p^0[0][0] + Su[0][0] \\
 a_p^0[0][1]T_p^0[0][1] + Su[0][1] \\
 a_p^0[0][2]T_p^0[0][2] + Su[0][2] \\
 \cdots \\
 a_p^0[1][0]T_p^0[1][0] + Su[1][0] \\
 a_p^0[1][1]T_p^0[1][1] + Su[1][1] \\
 \cdots
 \end{pmatrix}$$

四、编程计算思路

1. 时间步长与空间步长的关系

1) 时间步长选取原则:

为进行计算机数值求解, 需要求解矩阵的系数能够满足有界性条件, 即方程的各项系数大于 0。

2) 显式格式:

网格数目为 20×20 时 $\Delta x = 0.0025 \text{ m}$, $\Delta y = 0.0005 \text{ m}$

由有界性条件: $a_p^0 = a_p - (a_w + a_e + a_s + a_n - S_p) > 0$

以及: $a_e = k_e \frac{\Delta y}{\delta x_{EP}}$ $a_w = k_w \frac{\Delta y}{\delta x_{PW}}$ $a_n = k_n \frac{\Delta x}{\delta y_{NP}}$ $a_s = k_s \frac{\Delta x}{\delta y_{PS}}$ $a_p = \rho C \frac{\Delta x \Delta y}{\Delta t}$

得到:

$$\Delta t_{\max} = (\rho C)_{\min} \frac{\Delta x \Delta y}{k_{\max} \frac{\Delta y}{\delta x_{EP}} + k_{\max} \frac{\Delta y}{\delta x_{PW}} + k_{\max} \frac{\Delta x}{\delta y_{NP}} + k_{\max} \frac{\Delta x}{\delta y_{PS}}}$$
$$= 1500 \times 1465 \times \frac{0.0025 \times 0.0005}{53.6 \times \frac{0.0005}{0.0025} \times 2 + 53.6 \times \frac{0.0025}{0.0005} \times 2} = 0.004928$$

实际计算时取: $\Delta t_{\max} = 0.004$

显式格式的计算时间步长的最大值给出了一个相当严格的限制。此外 Δt 与成 $\Delta x \Delta y$ 正比关系, 若将网格缩小, 时间步长会大幅度减小, 导致实际计算时为提高计算精度需花费巨大的代价。但是, 显式格式的线性方程组每行仅含一个未知量, 在求解线性方程组时计算量相对隐式更少, 这是一个优点。

3) 半隐格式 (C-N 格式):

网格数目为 20×20 时 $\Delta x = 0.0025 \text{ m}$, $\Delta y = 0.0005 \text{ m}$

由有界性条件: $a_p^0 = a_p - 1/2 \times (a_w + a_e + a_s + a_n - S_p) > 0$

以及: $a_e = k_e \frac{\Delta y}{\delta x_{EP}}$ $a_w = k_w \frac{\Delta y}{\delta x_{PW}}$ $a_n = k_n \frac{\Delta x}{\delta y_{NP}}$ $a_s = k_s \frac{\Delta x}{\delta y_{PS}}$ $a_p = \rho C \frac{\Delta x \Delta y}{\Delta t}$

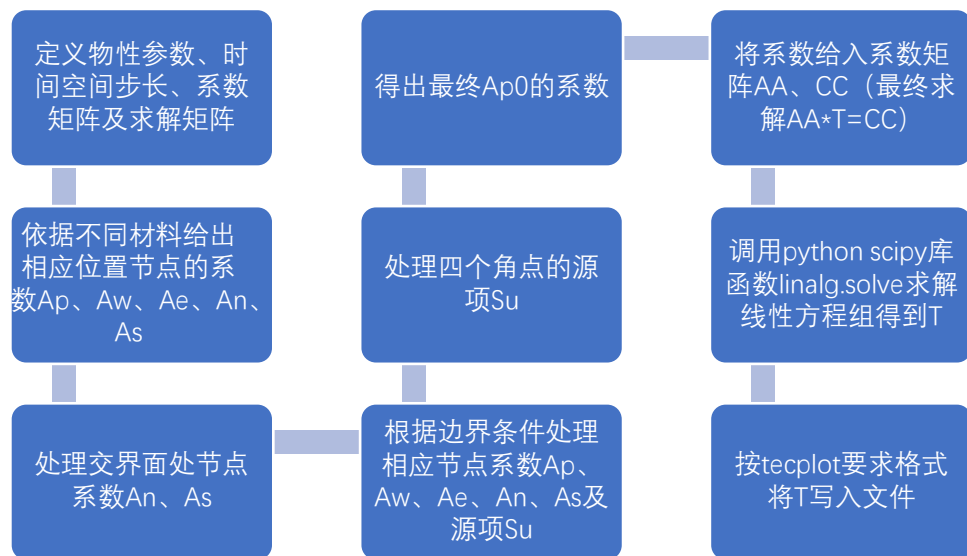
得到:

$$\Delta t_{\max} = 2 \times (\rho C)_{\min} \frac{\Delta x \Delta y}{k_{\max} \frac{\Delta y}{\delta x_{EP}} + k_{\max} \frac{\Delta y}{\delta x_{PW}} + k_{\max} \frac{\Delta x}{\delta y_{NP}} + k_{\max} \frac{\Delta x}{\delta y_{PS}}}$$
$$= 0.009856$$

4) 隐式格式:

有界性条件自动满足, 无最大时间步长的要求, 但时间步长的选取关乎计算结果的精确度。当时间步长取 0.02s 时能得到较好结果, 当时间步长取 0.2s 时计算结果偏差较大。

2. 编程整体思路（以显式为例）



3. 迭代求解方法（半隐式）

采用亚松弛迭代的方法，以半隐式方法为例，迭代公式为 $T_{i,j} = T_{i,j}^0 + m \cdot \left\{ \left(a_e \cdot \left(\frac{T_{i+1,j}^0 + T_{i,j+1}^0}{2} \right) + a_w \cdot \left(\frac{T_{i-1,j}^0 + T_{i,j-1}^0}{2} \right) + a_s \cdot \left(\frac{T_{i,j-1}^0 + T_{i,j+1}^0}{2} \right) + a_n \cdot \left(\frac{T_{i,j+1}^0 + T_{i,j+1}^0}{2} \right) + \left[a_{p0} - \frac{1}{2} \cdot (a_e + a_w + a_s + a_n - s_p) \right] \cdot T_{i,j}^0 + s_u \right\} \cdot \frac{1}{a_p} - T_{i,j}^0$ ，其中 m 为松弛因子。以计算网格数为 Nx=100、Ny=20（Nx 表示 x 方向的网格数量，Ny 同理）例，当计算时间 t=30 秒时，当松弛因子分别为 0.1、0.5、0.9 时，计算得到的云图分别为：

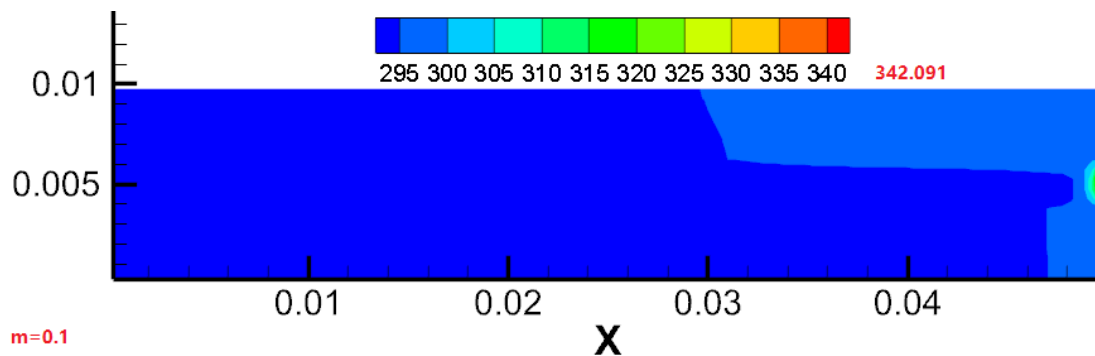


图 22 松弛因子=0.1

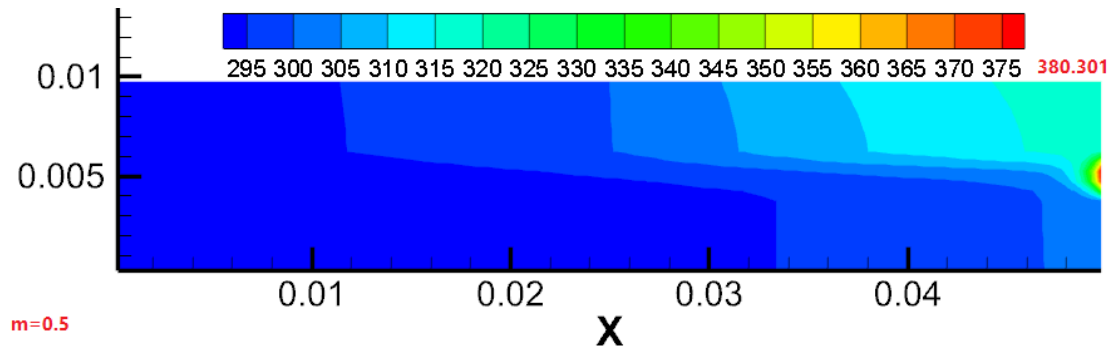


图 23 松弛因子=0.5

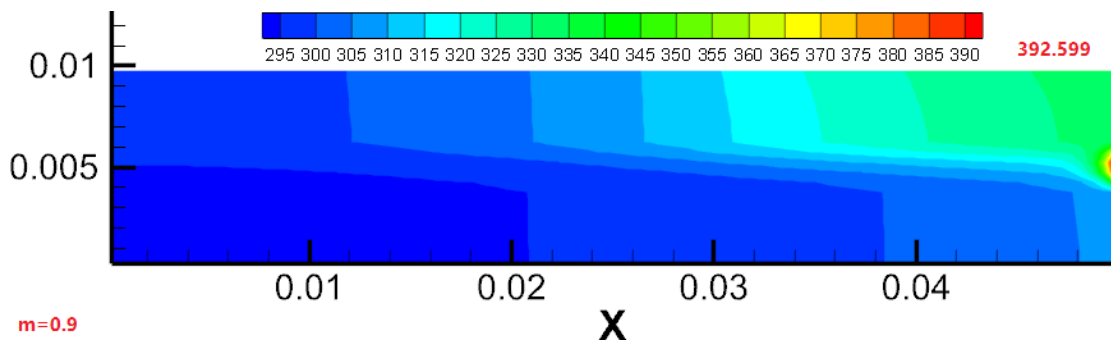


图 24 松弛因子=0.9

分析可以得到，结果逐渐收敛，最终通过计算得到最终的松弛因子为 0.93，最终得到的云图为：

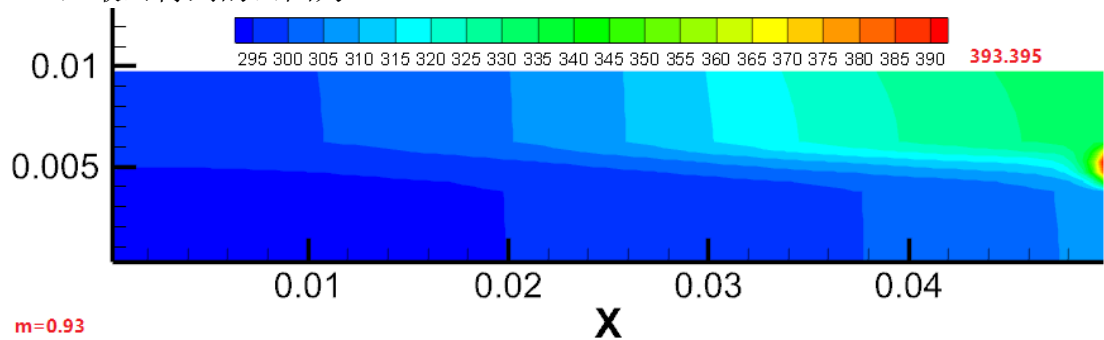


图 25 松弛因子=0.93

通过这种计算和分析方法可以得到最终收敛稳定的不同网格数量、不同时刻的数据。

4. 隐式求解方法

在 C++ 中引入 Eigen 库，对系数矩阵进行 QR 分解，求解每步时间步长时空间各点温度。但由于运算速度较慢，因此没有尝试 100×20 网格的计算。

五、 Fluent 仿真

1. SpaceClaim 建模

建模时要注意：

- 1.由于 Fluent 中有两个计算域，几何模型要划分成两个表面；
- 2.由于上边界水平线要分段处理边界条件，为便于在划分网格时分块与关联，水平方向的线都要分段画出，突出交点。



图 26 几何模型

2. ICEM 划分网格

1) 创建 part B1~B6，方便 Fluent 设置边界条件

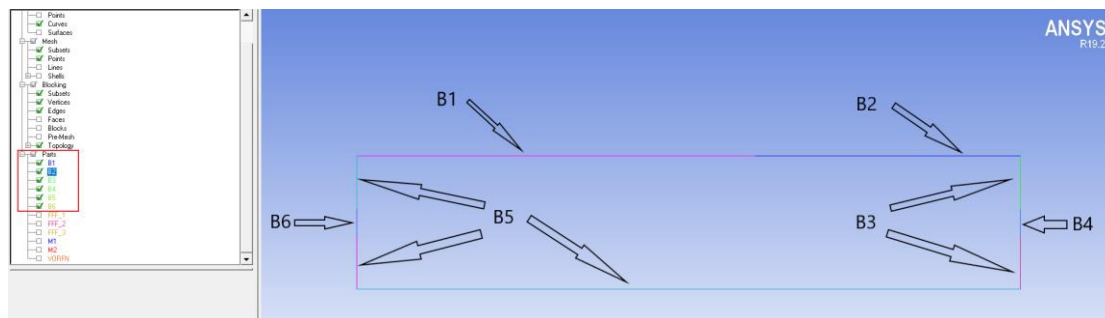


图 27 创建边界 part

2) 创建块 block，并进行分割

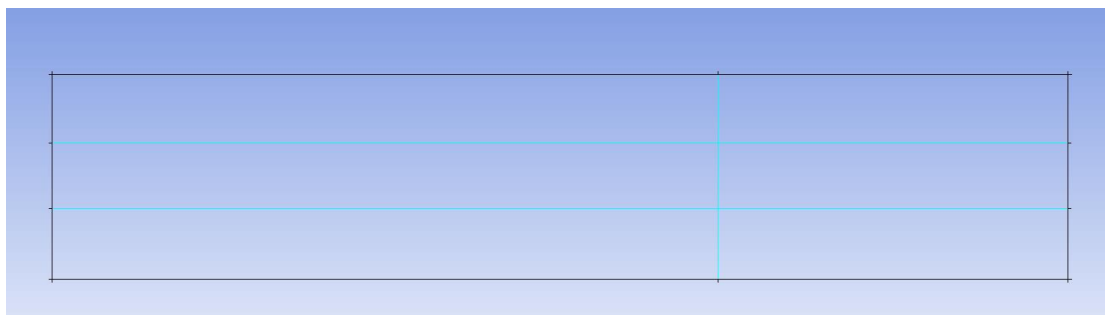


图 28 块的分割

3) 将块与几何模型进行点与线的关联

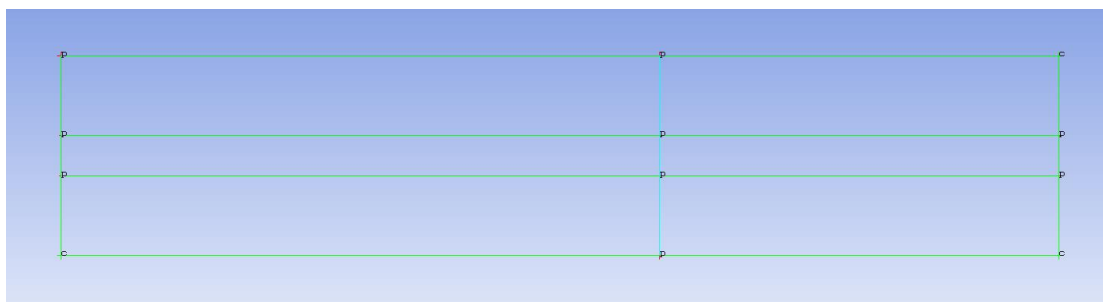


图 29 关联效果图

4) 设置各边节点数

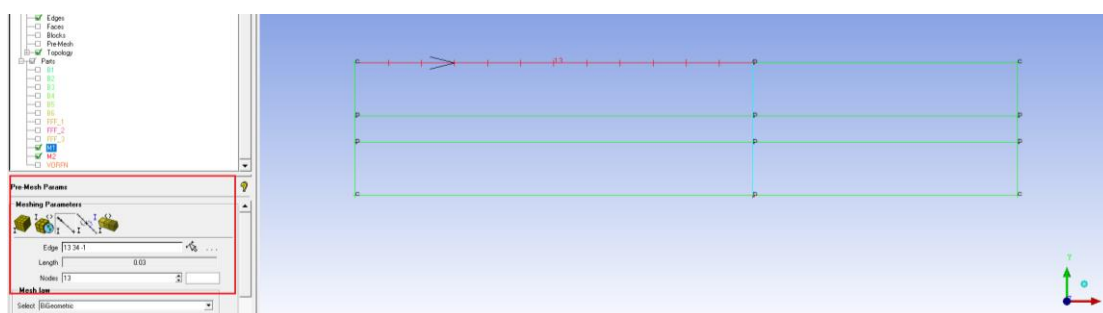


图 30 设置各边节点数目

5) 创建 part M1 与 M2，将块分成两个计算域



图 31 划分计算域

6) Pre-Mesh，转成非结构网格，导入 Fluent

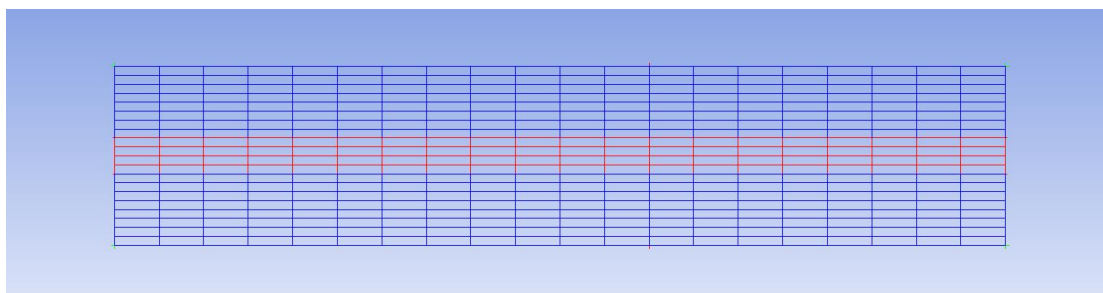


图 32 网格 (20×20)

3. Fluent 处理

1) 模型选择

General 中选择瞬态计算；由于计算域内为固体的热量传递问题，因此只需要开一个能量方程。

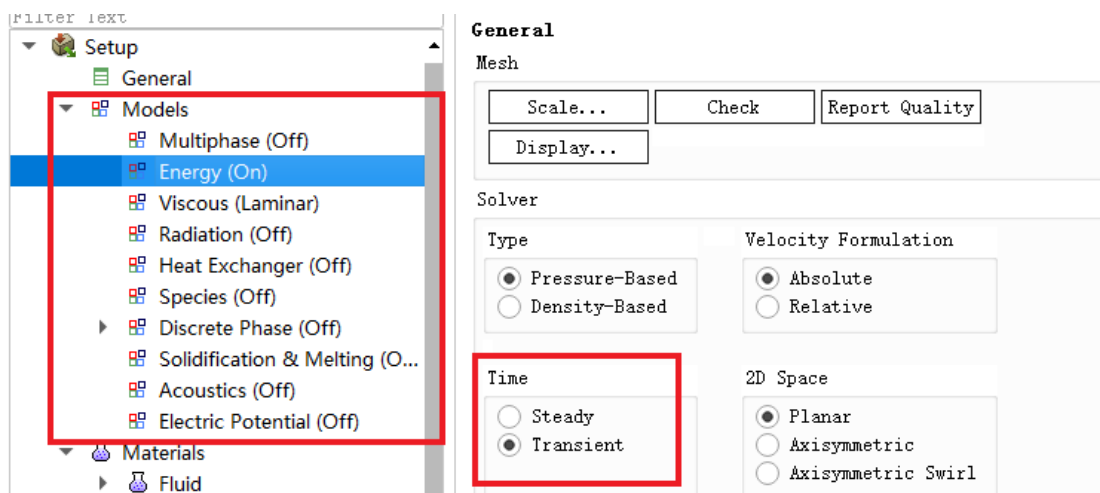


图 33 模型选择

2) 设置材料参数

按题目所给条件，给定两个计算域的材料及参数：

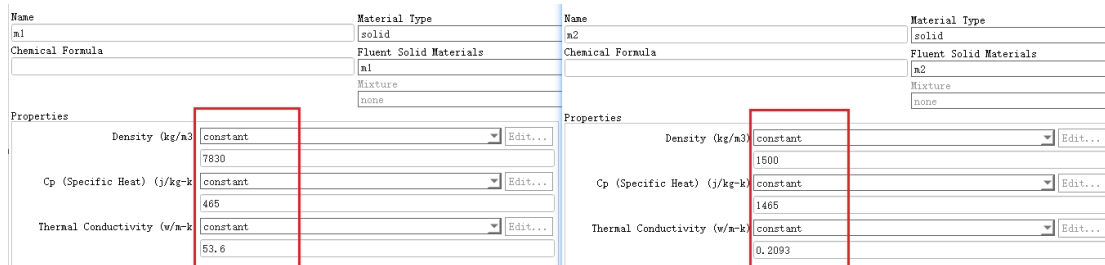


图 34 材料参数设置

3) 边界条件处理

由题目中物理模型及参数，设置各边对流、辐射或绝热等边界条件。

要注意的是，b1 边界对流换热系数需要满足实验关联式，因此需要编写 UDF 进行处理。

```
#include "udf.h"
DEFINE_PROFILE(user_h, thread, pos)
{
    double tem, tm, beta, Pr, v, Gr, Nu, h;
    face_t f;
    begin_f_loop(f, thread)
    {
        tem = F_T(f, thread);
        tm = (tem - 253) / 2;
        beta = 2 / (tem + 293);
        Pr = 4 * (1 - 14) * pow(tm, 6) - 3 * (1 - 11) * pow(tm, 5) + 6 * (1 - 9) * pow(tm, 4) - 7 * (1 - 7) * pow(tm, 3) + 4 * (1 - 5) * pow(tm, 2) - 0.0012 * tm + 0.7163;
        v = 9 * (1 - 11) * pow(tm, 2) + 9 * (1 - 8) * tm + 1.3175 * (1 - 5);
        Gr = 9.8 * beta * fabs(tem - 293) * pow(0.03, 3) / pow(v, 2);
        Nu = 0.54 * pow(Gr * Pr, 0.25);
        h = Nu * 2.59 / 3;
        F_PROFILE(f, thread, pos) = h;
    }
    end_f_loop(f, thread)
}
```

图 35 UDF 代码

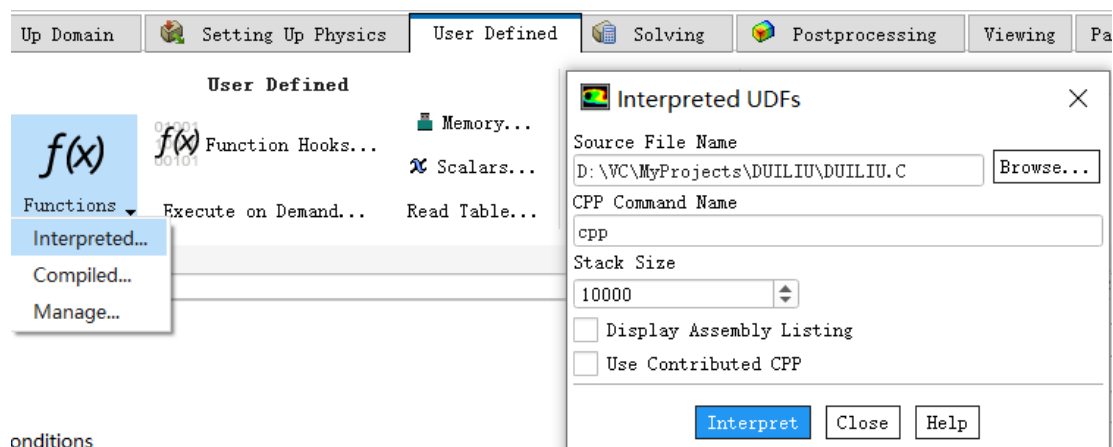


图 36 解释 UDF

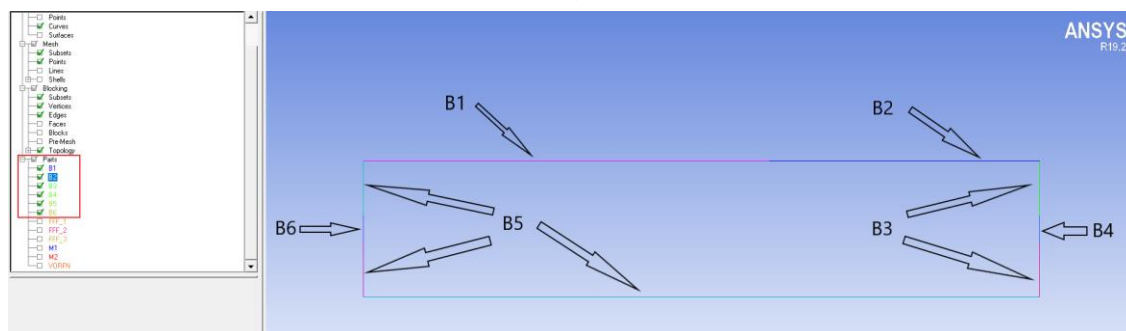


图 37 边界代号含义

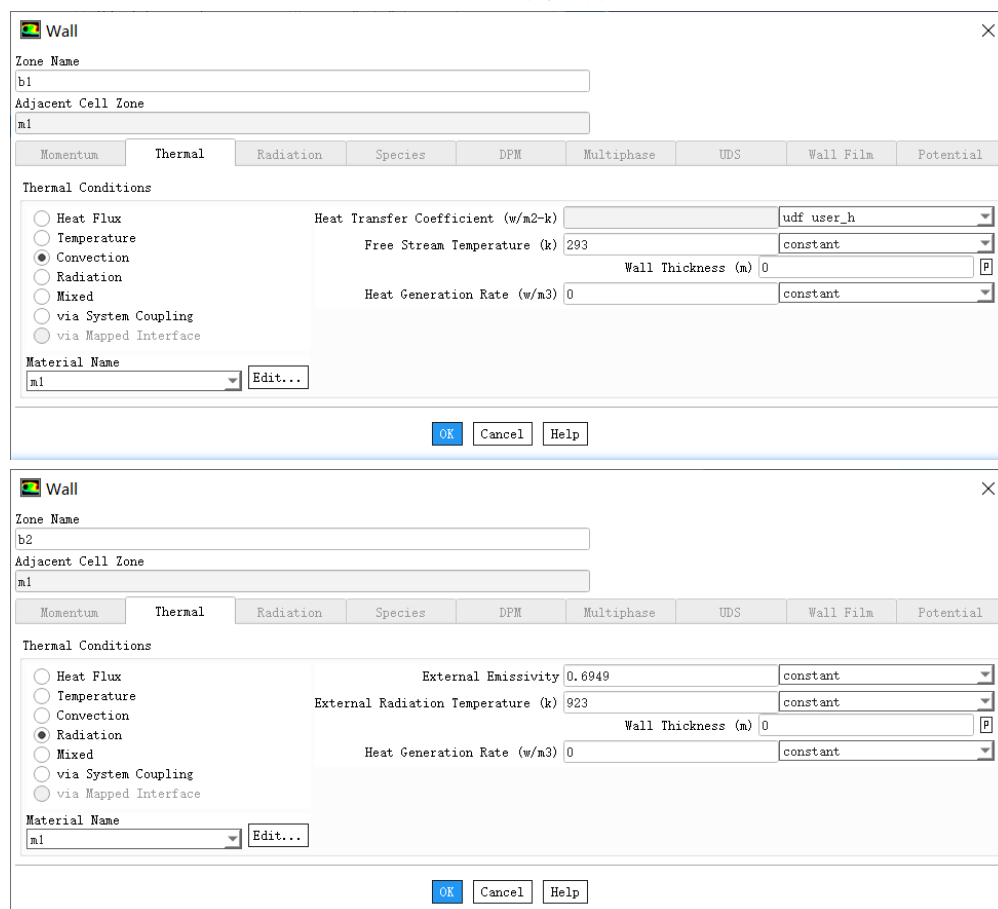


图 38 上边界条件

Wall

Zone Name

b3

Adjacent Cell Zone

m1

Momentum

Thermal

Radiation

Species

DPM

Multiphase

UDS

Wall Film

Potential

Thermal Conditions

☐ Heat Flux
 ☐ Temperature
 ☐ Convection
 ☒ Radiation
 ☐ Mixed
 ☐ via System Coupling
 ☐ via Mapped Interface

External Emissivity

0.6949

constant

External Radiation Temperature (k)

923

constant

Wall Thickness (m)

0

P

Heat Generation Rate (w/m3)

0

constant

Material Name

m1

Edit...

OK

Cancel

Help

Wall

Zone Name

b4

Adjacent Cell Zone

m2

Momentum

Thermal

Radiation

Species

DPM

Multiphase

UDS

Wall Film

Potential

Thermal Conditions

☐ Heat Flux
 ☐ Temperature
 ☐ Convection
 ☒ Radiation
 ☐ Mixed
 ☐ via System Coupling
 ☐ via Mapped Interface

External Emissivity

0.7515

constant

External Radiation Temperature (k)

923

constant

Wall Thickness (m)

0

P

Heat Generation Rate (w/m3)

0

constant

Material Name

m2

Edit...

OK

Cancel

Help

图 39 右边界条件

Wall

Zone Name

b5

Adjacent Cell Zone

m1

Momentum

Thermal

Radiation

Species

DPM

Multiphase

UDS

Wall Film

Potential

Thermal Conditions

☒ Heat Flux
 ☐ Temperature
 ☐ Convection
 ☐ Radiation
 ☐ Mixed
 ☐ via System Coupling
 ☐ via Mapped Interface

Heat Flux (w/m2)

0

constant

Wall Thickness (m)

0

P

Heat Generation Rate (w/m3)

0

constant

Material Name

m1

Edit...

OK

Cancel

Help

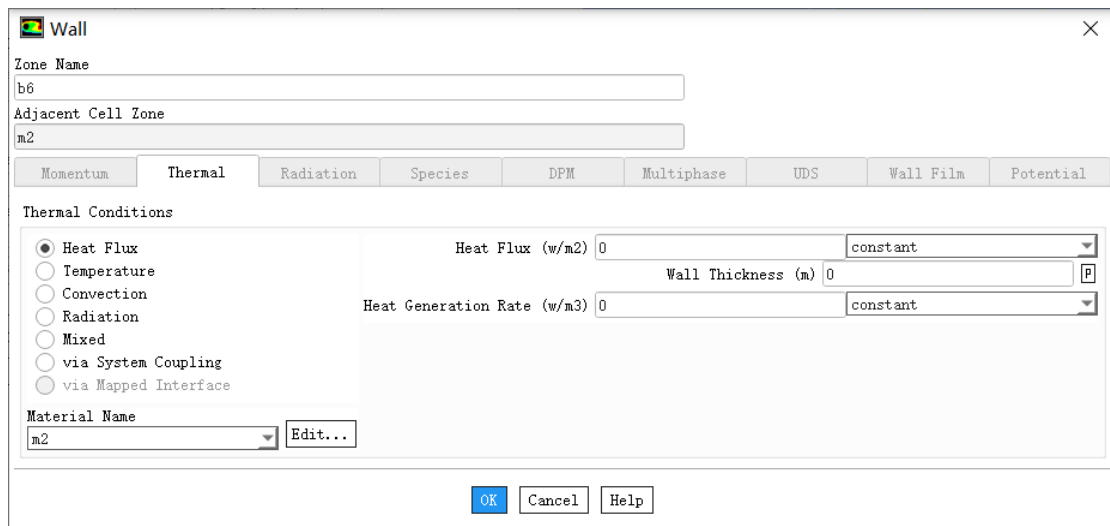


图 40 下边界及左边界条件

4) 进行初始化，设置时间步长并计算

初始化计算从全域开始，初温为 293K；时间步长与隐式格式相一致，取 0.02 s，计算 1500 个时间步，总计 30 s。

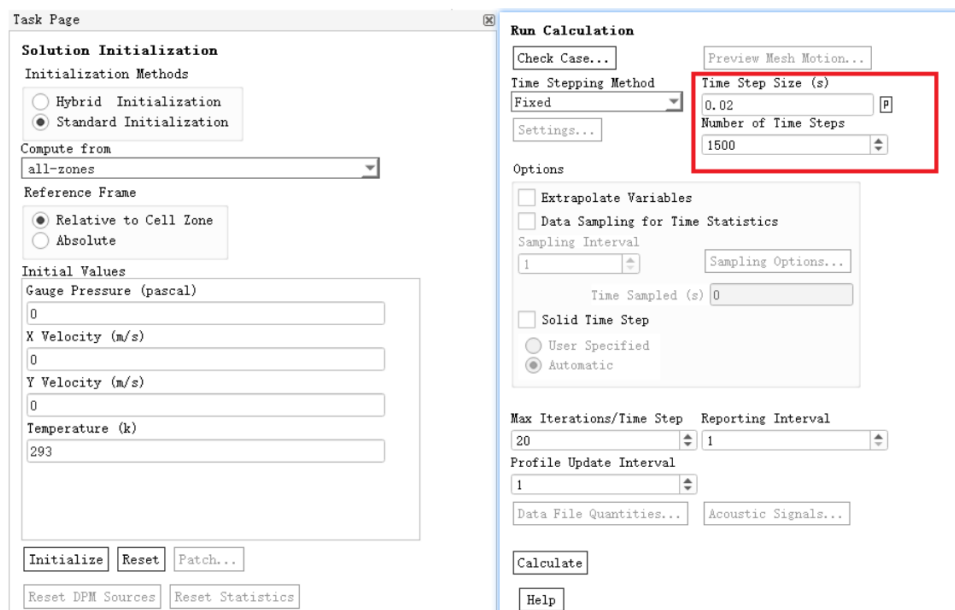


图 41 初始化及时间步长

六、 结果分析

1. 温度分布

由题目所给参数，材料一的密度大、定压比热容大、导热系数大；材料二的密度小、定压比热容小、导热系数小。两种材料对热量的吸收和热量传导会有较大的差别。由于 5s 时间太短，所以选取 30s 的情况下 fluent 计算结果进行分析。

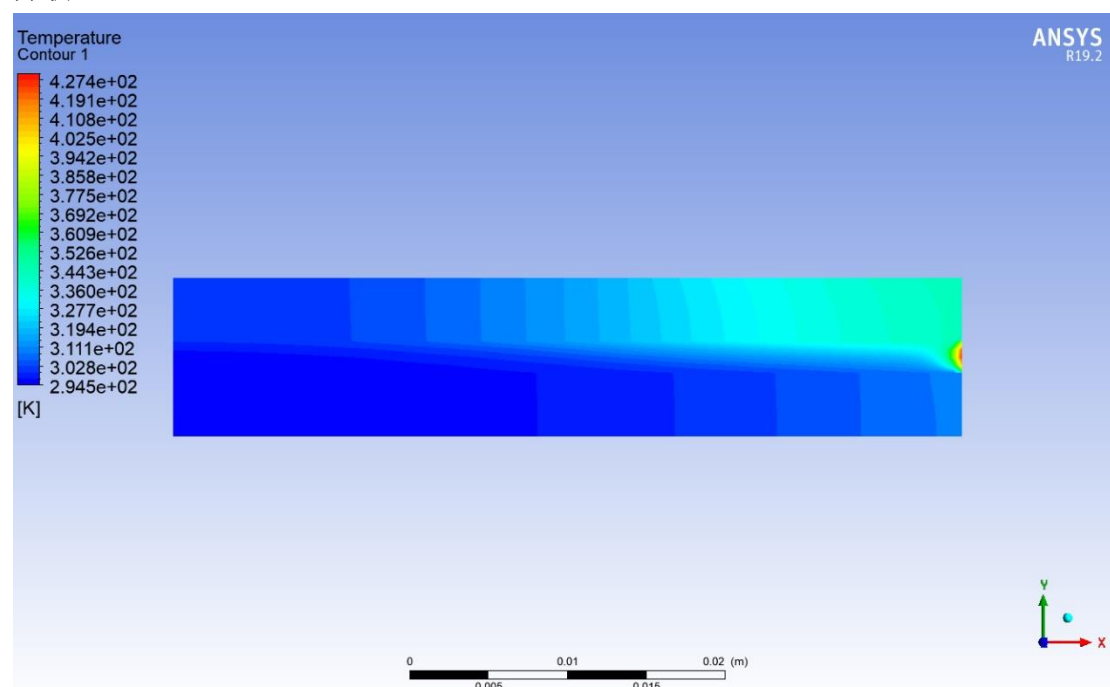


图 42 30sFluent 云图

在右端和右上部分受高温辐射的情况下，材料 2 右侧出现温度集中区，材料 1 的温度分比较均匀。计算两材料的热扩散系数：

$$a_1 = \frac{k_1}{\rho_1 c_{p1}} = 1.47 \times 10^{-5} \quad a_2 = \frac{k_2}{\rho_2 c_{p2}} = 9.52 \times 10^{-8}$$

从计算结果，材料 1 的热扩散系数大，内部热量传递快，温度分布较均匀，材料 2 热导热系数极低，使得热量聚集在右端。材料 2 的定压比热容大，在相同时间内吸收的热量多，同时材料 2 的导热系数极低，使得热量无法快速向左侧低温区传递。

由于材料 2 所处区域受右侧及上侧高温区影响，且材料 2 热扩散系数小，扯平内部温度的能力较小，所以出现了向左倾斜的温度等值线。

由于材料热量都来自与右侧和右上部分，材料 1 左侧上部分为自然对流，散热量非常少，由此材料 1 中的左侧温度梯度小，而右侧温度梯度较大。

2. 不同求解方式的对比

1) 网格为 20×20

时间为 5s

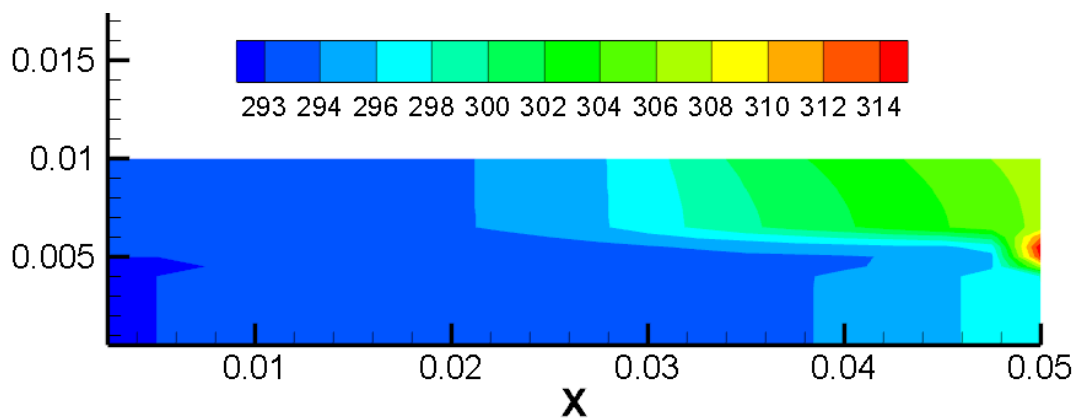


图 43 显式 5s

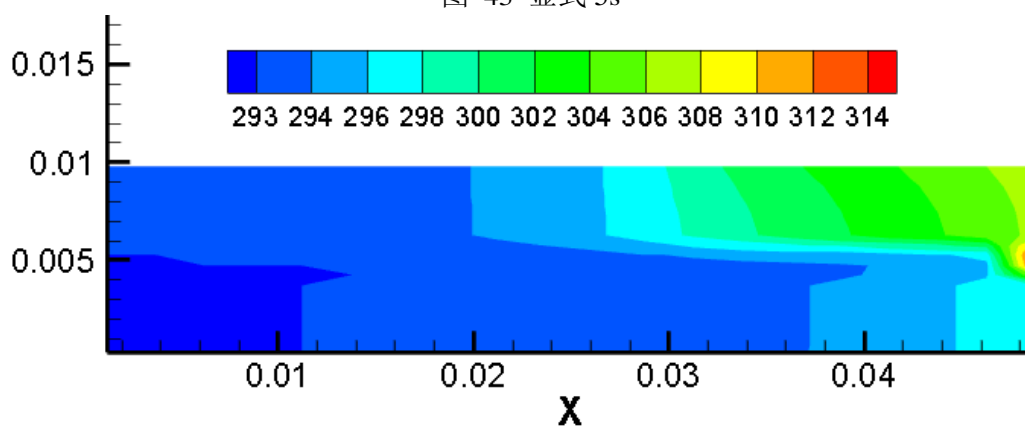


图 44 隐式 5s

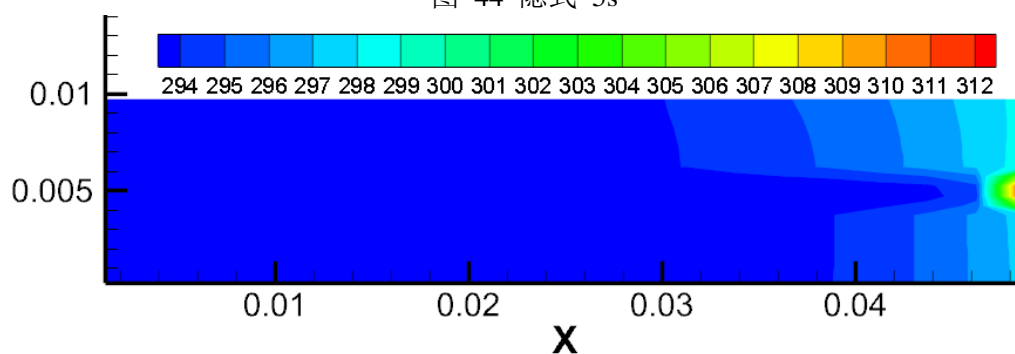


图 45 半隐式 5s



图 46 Fluent 5s

分析：显式和隐式的求解结果接近，最高温在 314K 左右，半隐式和 Fluent 的求解结果存在较大的偏差，其中 Fluent 的云图中，最高温达到了 480K，热量集中在材料 2 右端无法向右端低温区传导。

时间为 30s

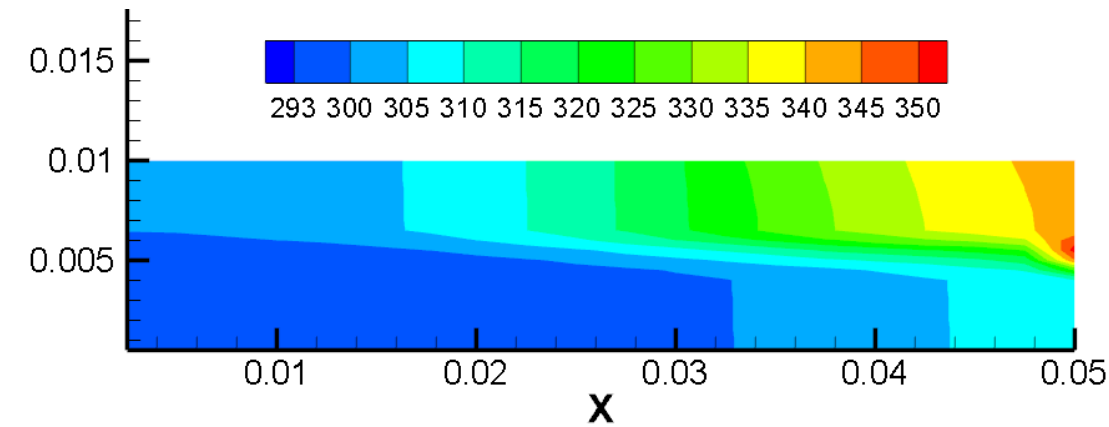


图 47 显式 30s

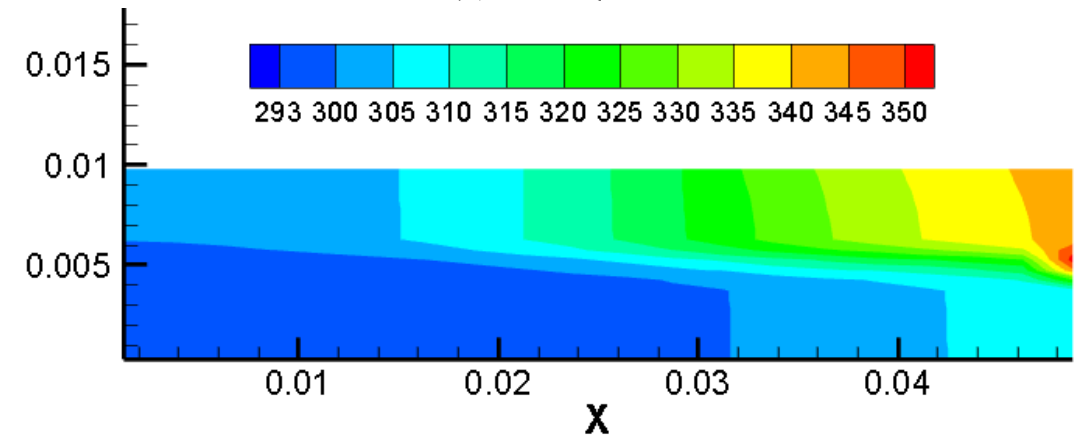


图 48 隐式 30s

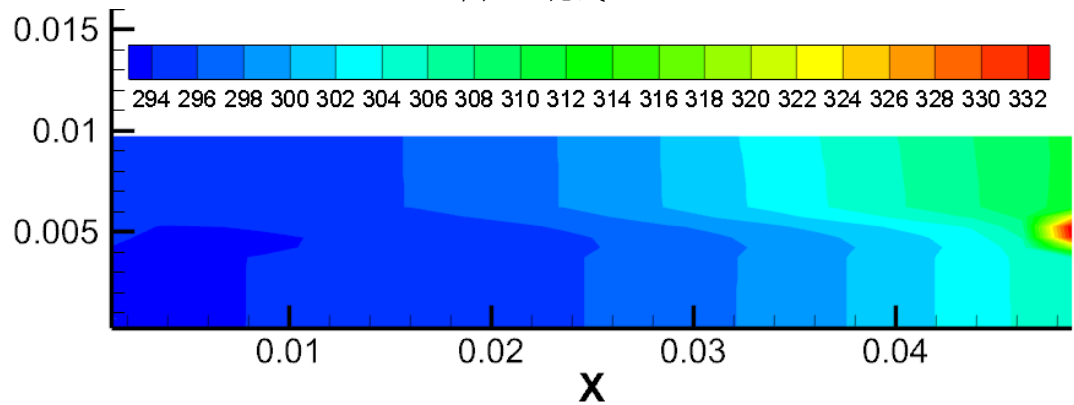


图 49 半隐式 30s

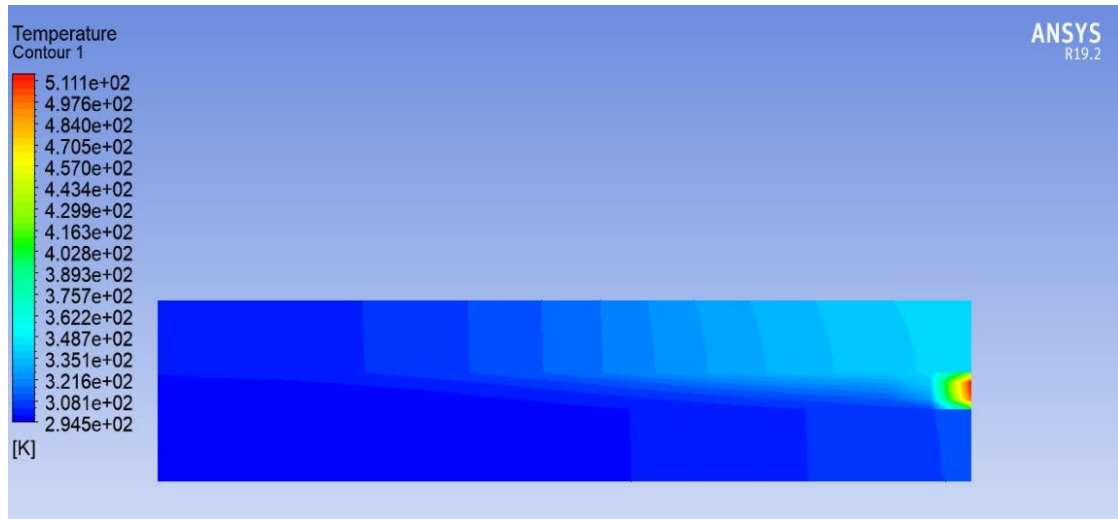


图 50 Fluent 30s

分析：显式与隐式的求解结果相差不大，最高温在 350K 左右，高温集中区向右上方移动，Fluent 的求解结果中高温集中区依然在右端材料二的位置，且最高温达到了 510K。

2) 网格为 100×20

时间为 5s

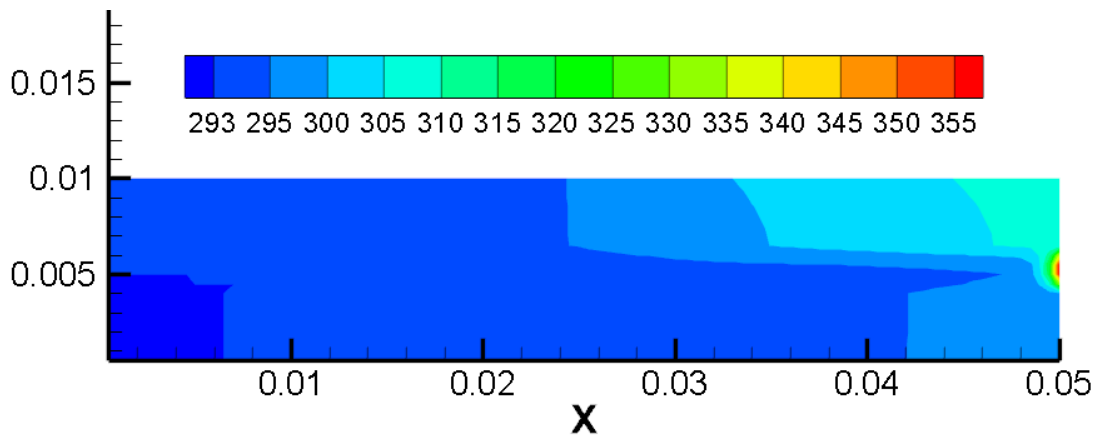


图 51 显式 5s

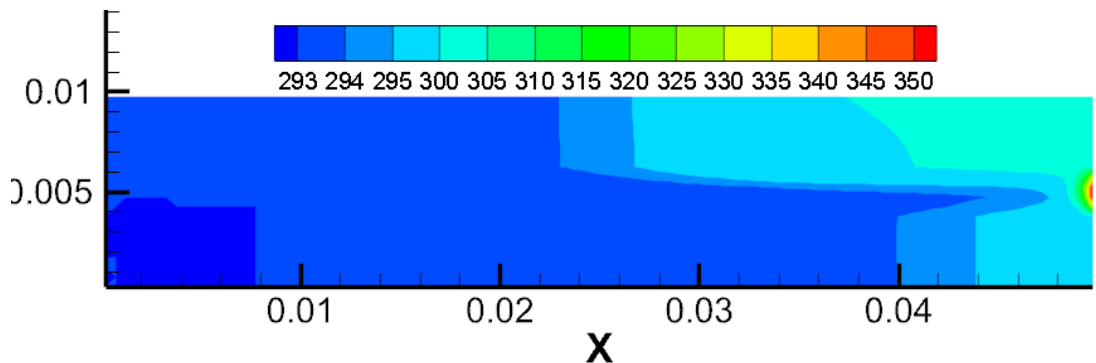


图 52 半隐式 5s



图 53 Fluent 5s



图 54 Mechanical 5s

分析：三种求解方式下，显式情况下的最高温在 355K 左右，半隐式情况下的最高温在 350K 左右，Fluent 的情况下最高温 390K 左右，三种情况的求解温度更加接近。材料 1 中的温度梯度分布接近。

时间为 30s

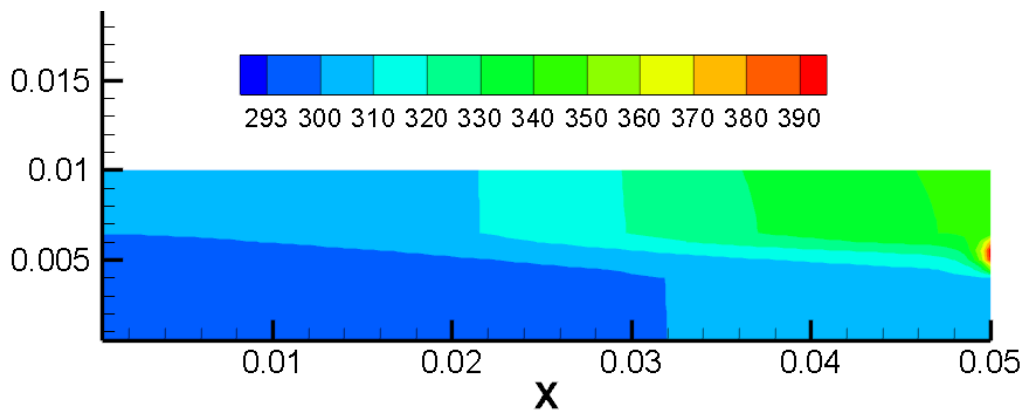


图 55 显式 30s

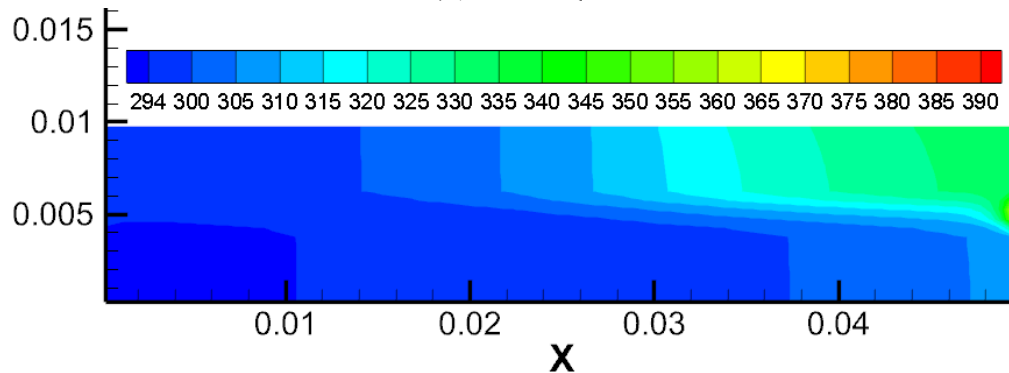


图 56 半隐式 30s

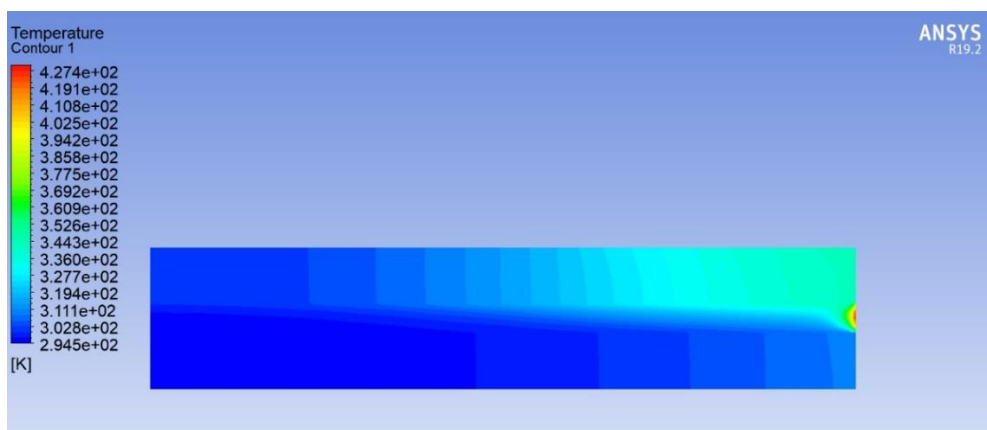


图 57 Fluent 30s

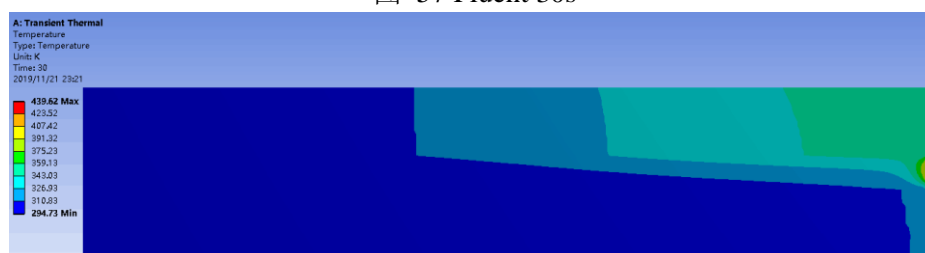


图 58 Mechanical 30s

分析：显式与半隐式求解结果最高温为 390K，材料中的温度梯度分布相差不大，Fluent 的求解结果中最高温为 430K，与显式与半隐式的计算结果差距缩小。

3. 空间温度分布对比（轴线 $y=0.005m$ ，网格数 20×20 ）

我们取材料 2 的中轴线即 $y = 0.005m$ 处温度分布进行对比，在网格数目均为 20×20 情况下应用 5 种方法对中轴线温度场进行求解，结果如下

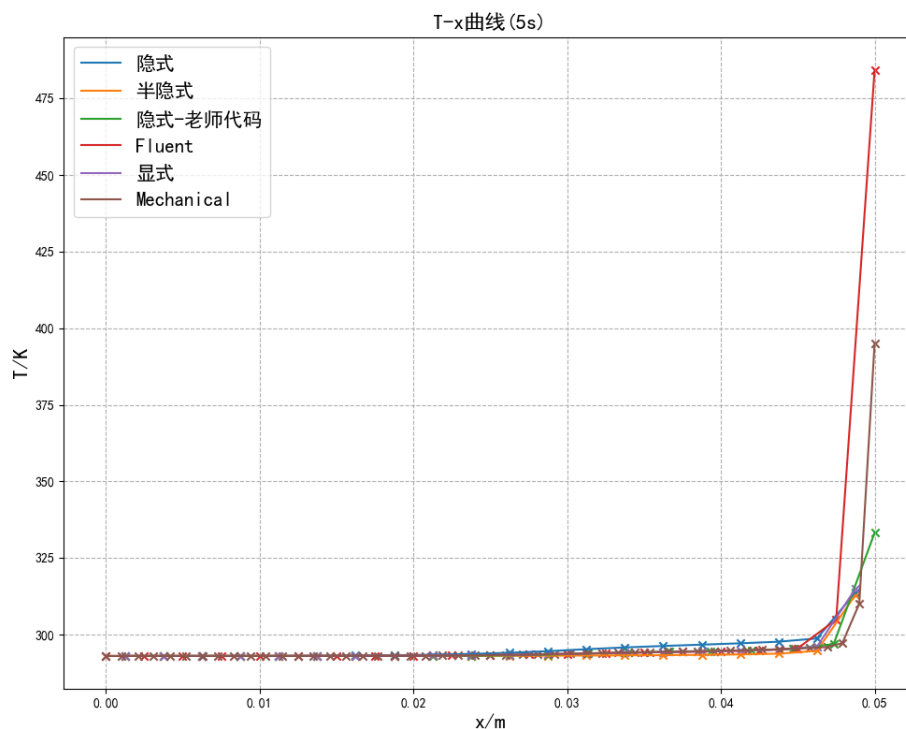


图 59 5s 时不同解法得到的曲线对比

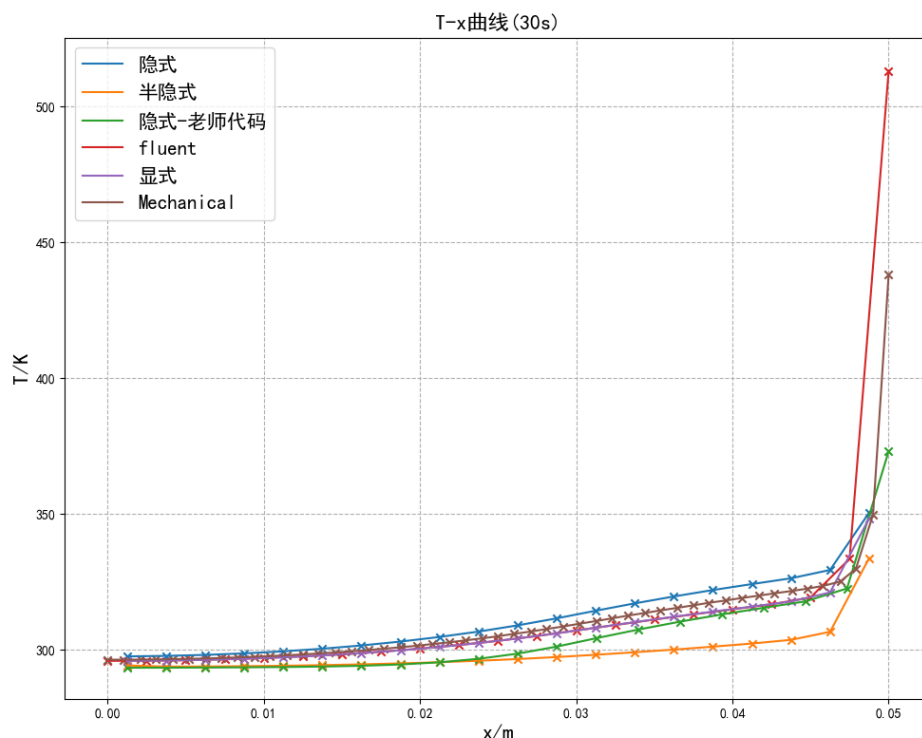


图 60 30s 时不同解法得到的曲线对比

由图分析，商业软件计算结果在 $x = 0.045$ 之前与代码计算结果吻合良好，但是在接近高温辐射壁面处（右壁面）出现较大误差。由模型所处的物理情形可知，在右壁面附件温度梯度较大，出现此种误差可能是因为网格 Δx 过大。

经验证，发现当网格数逐渐增大时，商业软件计算出的右壁面温度逐渐变小，代码计算结果则逐渐变大，两者最终都接近 $420\text{k} \sim 430\text{k}$ 。因此认为出现上图原因是网格数目过少。

4. 网格无关性分析（时间 $t=30\text{s}$ ）

1) 显式格式

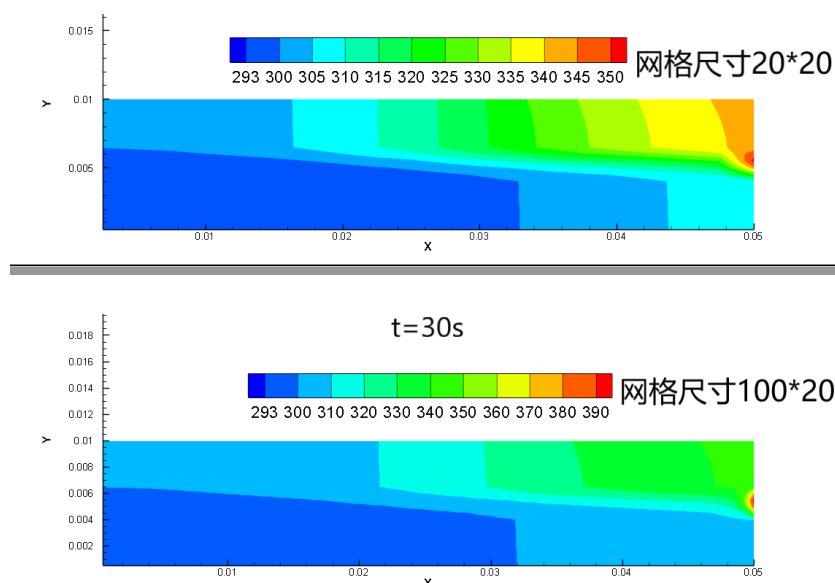


图 61 网格 20×20 和 100×20 结果对比

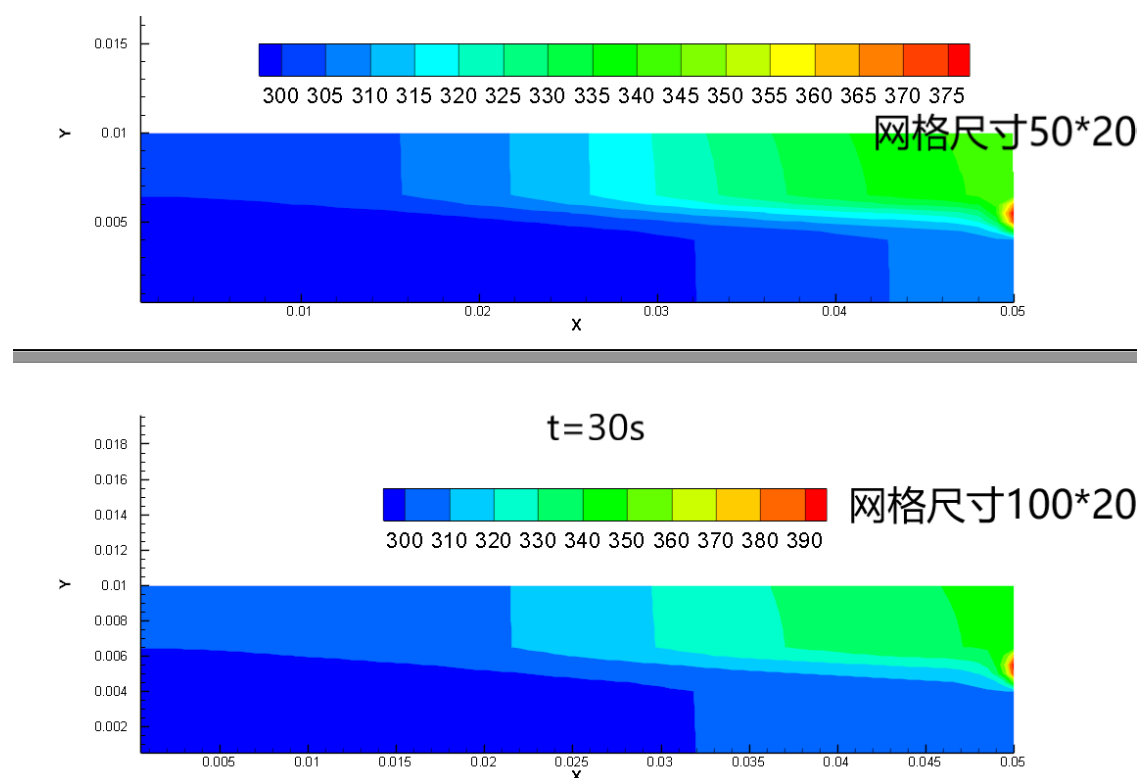


图 62 网格 50×20 和 100×20 结果对比

由网格尺寸 20×20 、 50×50 、 100×20 的对比云图可知， x 方向网格尺寸会影响最后的温度分布，其中最突出的是，随着 x 方向网格变密，最高温度会不断上升，网格尺寸 100×20 的最高温度比 20×20 的情况高 40°C ，因此，计算结果与网格尺寸有关，经过验证主要与 x 方向的疏密程度有关。

2) 半隐格式

以 30 秒为例，分析各种数量网格计算出的结果之间的差异，分析对比的内容包括云图、轴线 $T-x$ 曲线，之后分析 30s 内特征点 $T-t$ 曲线。

云图对比

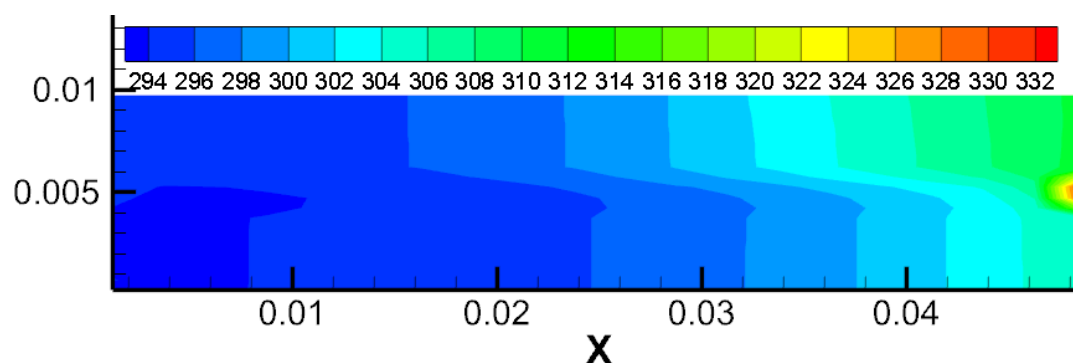


图 63 20×20

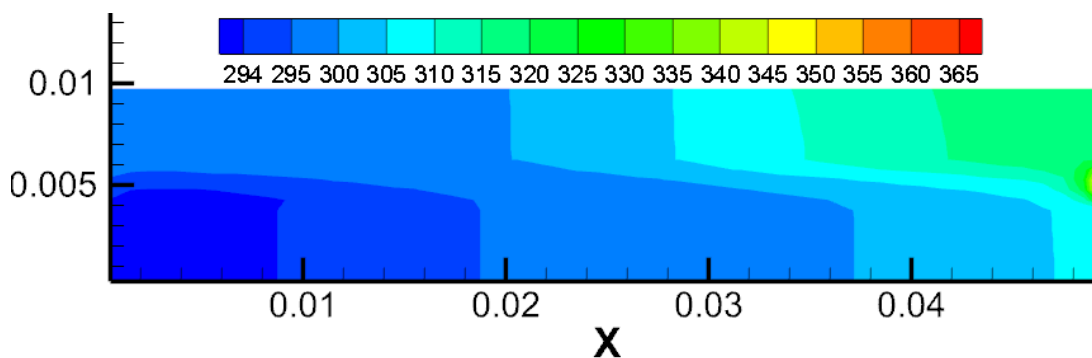


图 64 50×20

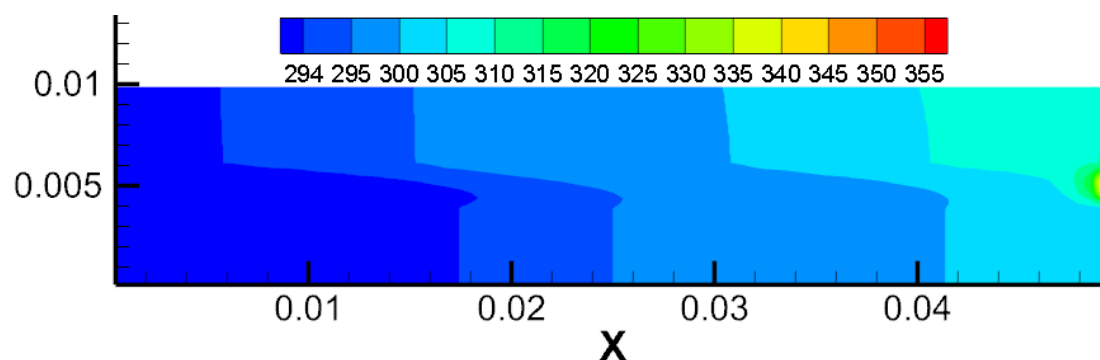


图 65 50×40

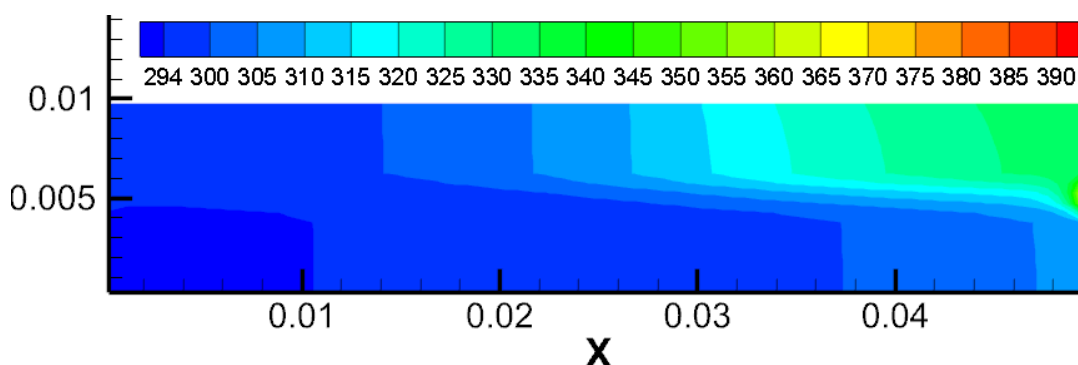


图 66 100×20

由以上四张云图可得，随着 X 方向网格数的增加，最高温度逐渐增大，294K 等温线所包围的左下区域的面积逐渐减小，并且当网格数增加到一定程度时，这些差异逐渐减小，故当网格 X 方向网格数 100 时，结果逐渐趋于稳定，计算结果与网格数量无关。但是对比网格分别为 50×20 与 50×40 的云图可以发现，Y 方向网格数目增加时，最高温度值减小，294K 等温线所包围的区域的面积表达，故 X 方向的网格数目对结果的影响有待验证。

T-x 曲线对比

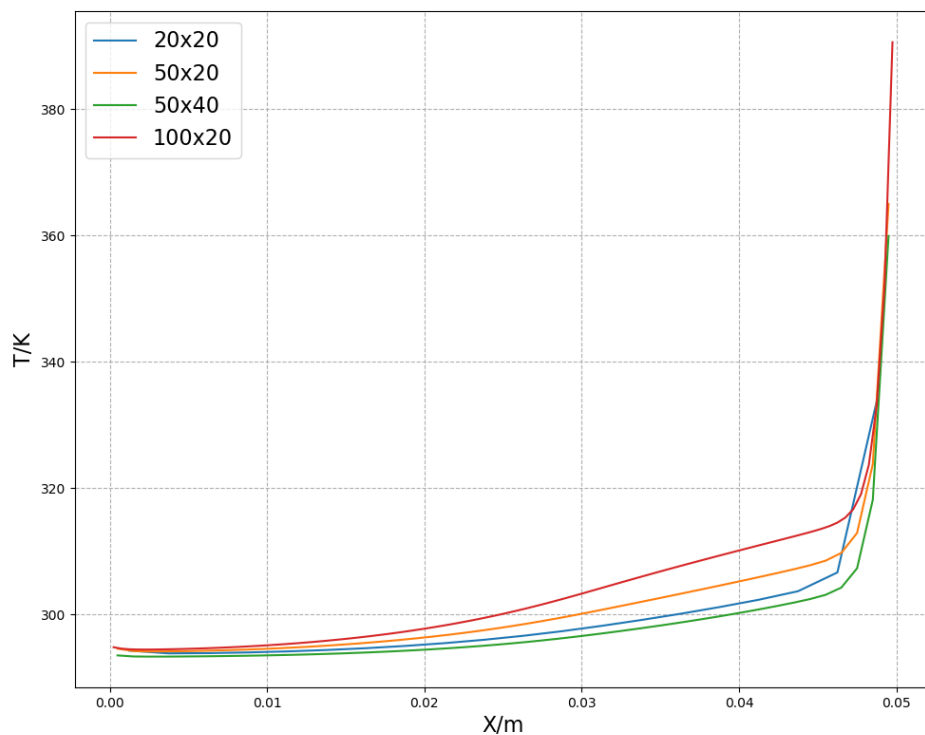


图 67 T-x 曲线

由 $T-x$ 曲线图可得，网格越密，中轴线上的温度均比其余网格数量都要高，这与云图分析出来的结果类似。

T-t 曲线对比

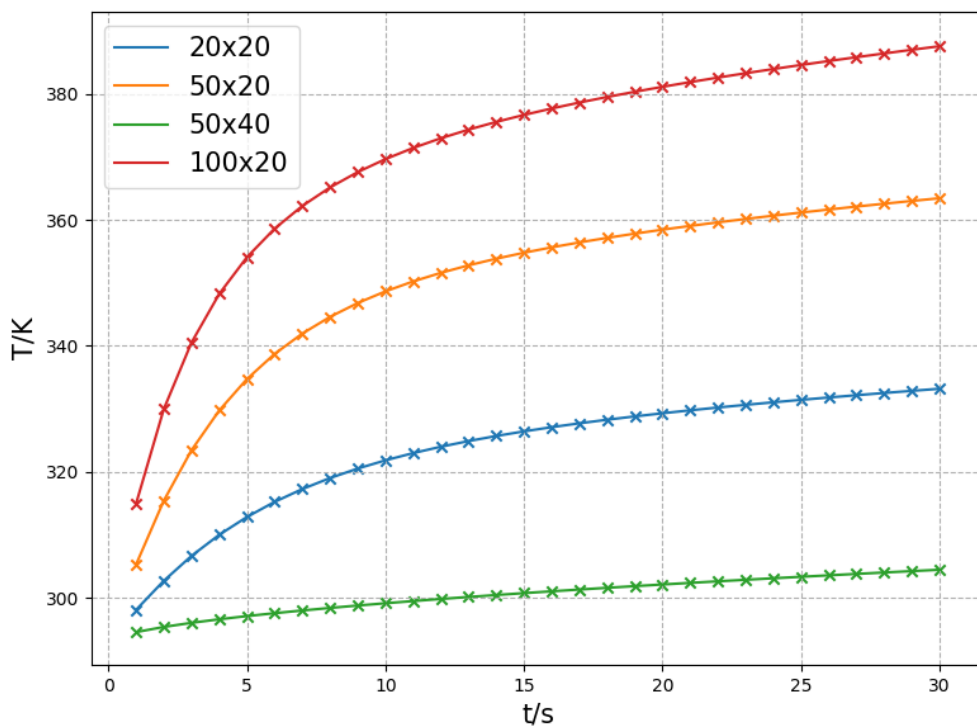


图 68 T-t 曲线

特征点取最右边中轴线靠上的点，具体坐标与网格数量有关，在此不给

出。

分析 $T-t$ 曲线可得，对于该点而言其平均大小顺序为： $100 \times 20 > 50 \times 20 > 20 \times 20$ ，而 50×40 的温度平均值最小，与云图分析类似，Y方向的网格数目对于考察某点温度的时间变化较为重要，需要进行进一步验证；而X方向的网格数量的增加对于温度的影响逐渐减小，最终趋于稳定，即当X方向的网格加密到一定数量时，再增大网格数量，其不再对温度产生，即验证了计算结果与网格无关。

3) Fluent 仿真

云图对比

(1) 5 s 时不同网格规格温度对比

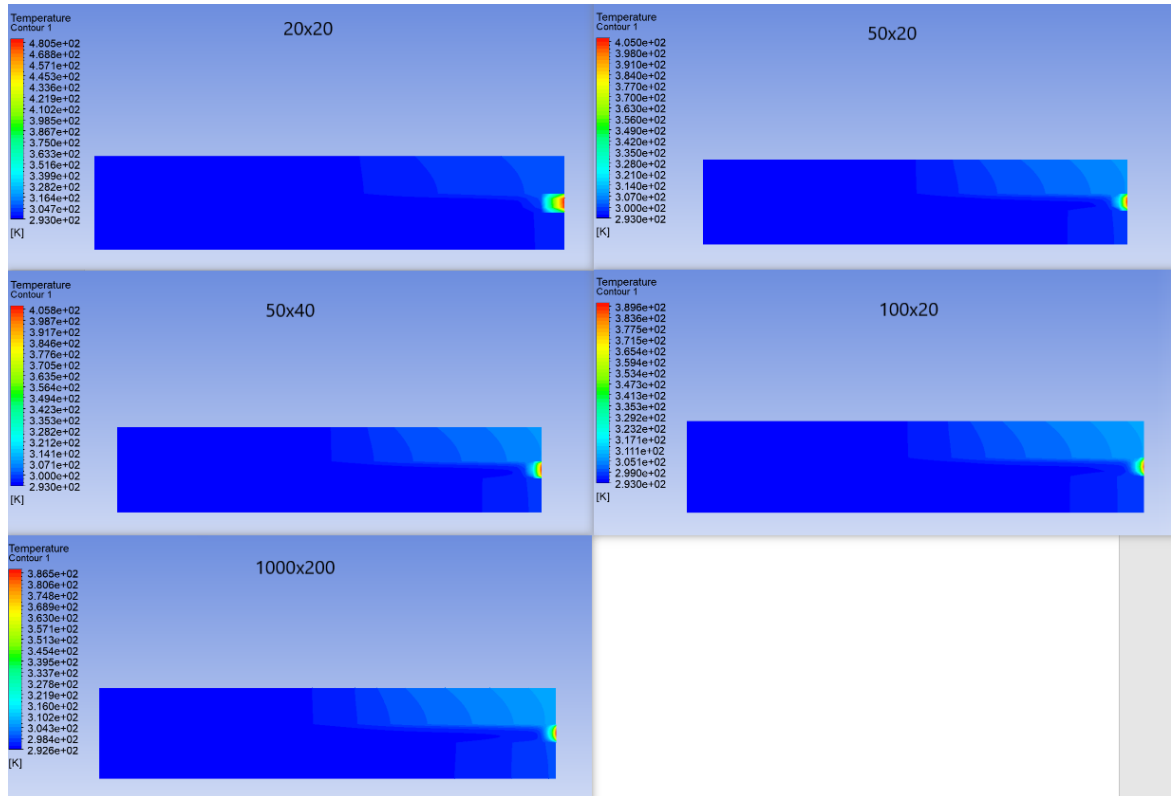
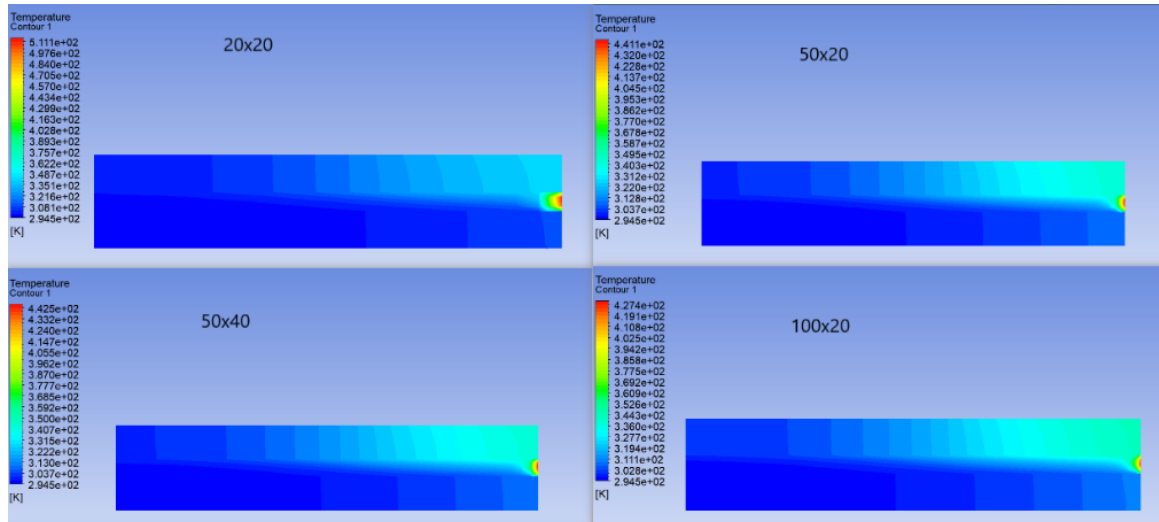


图 69 5 s 时温度对比

(2) 30 s 时不同网格规格温度对比



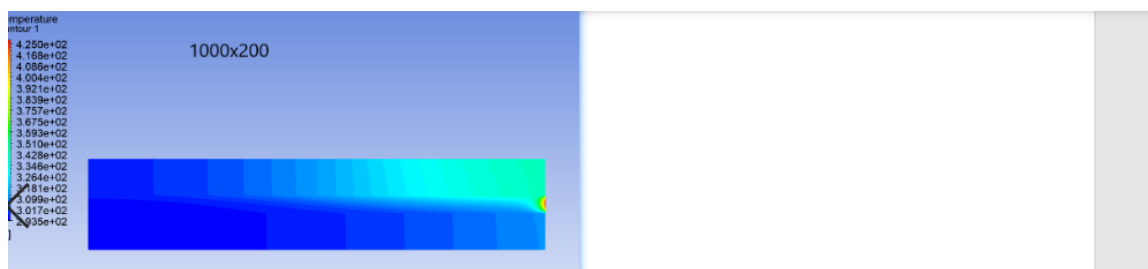


图 70 30 s 时温度对比

由 5 s 和 30 s 的云图可以明显看出不同网格规格最高温度的差别，为了更直观具体的表示，列出不同规格的辐射边界中点（即温度最高点）的温度。

表 2 不同网格规格温度最大值对比

网格规格	20×20	50×20	50×40	100×20	1000×200
5 s 时温度	484.41 K	407.38 K	408.11 K	391.6 K	388.45 K
30 s 时温度	513.23 K	444.20 K	444.57 K	430.14 K	426.99 K

由上表反映的规律，我猜想是，随网格数量增加，最高温度逐渐降低，结果趋于 427 K；但要注意，50×40 网格数量是 50×20 的 2 倍，但与其温度十分接近，且略有增大，说明 x 方向网格数的增多才是温度降低的关键。

轴线温度对比

单靠最高温度点的数据缺乏说服力，因此选取膏体发动机轴线上各点的温度进行对比。

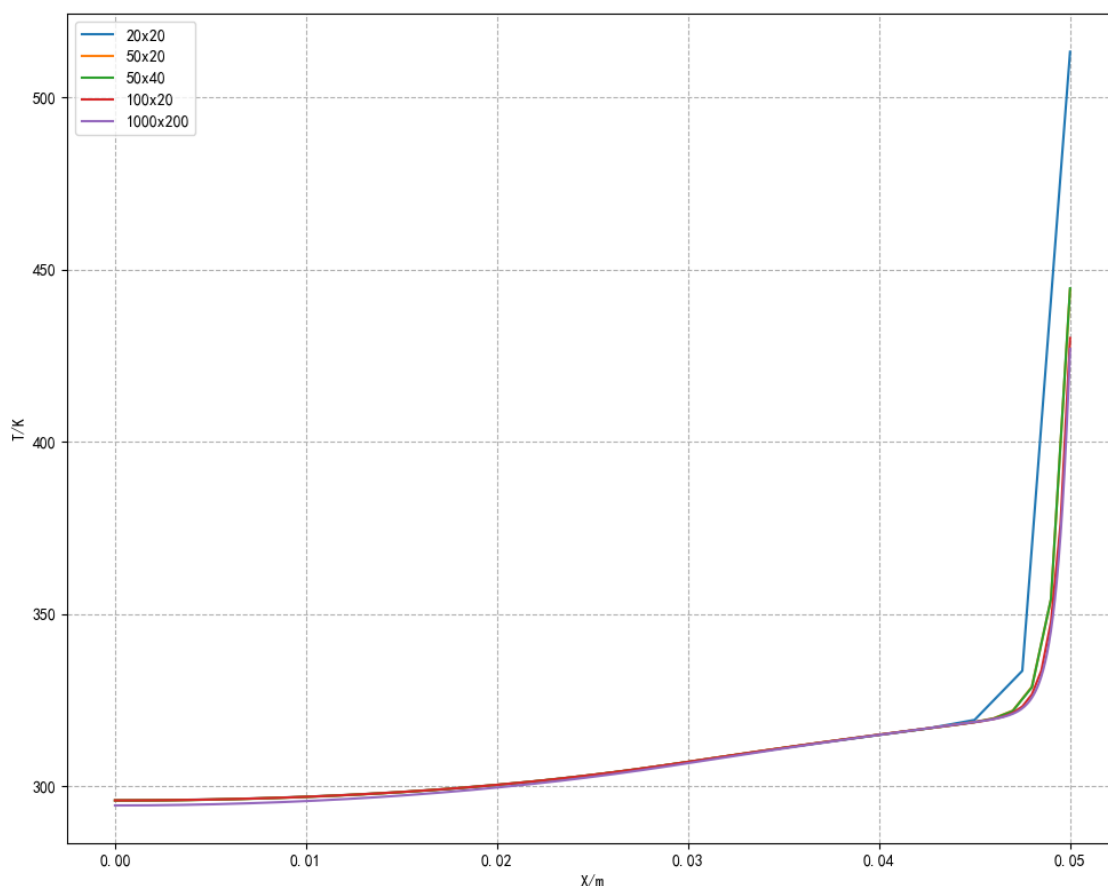


图 71 30 s 时不同网格规格轴线温度对比

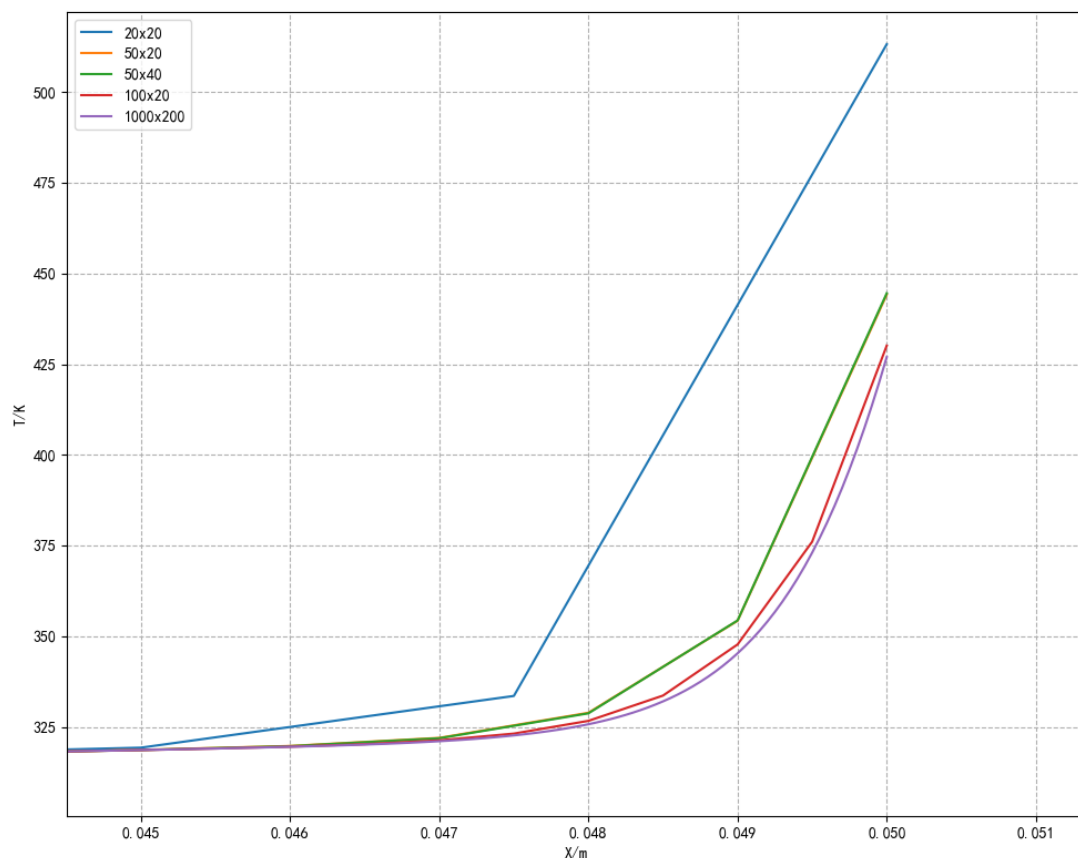


图 72 30 s 时不同网格规格轴线温度对比细节图

随网格数量增加，最高温度逐渐降低，结果趋于 427 K；但要注意， 50×40 与 50×20 的温度十分接近，说明 x 方向网格数的增多才是温度降低的关键。轴线数据较好的吻合了上述猜想。

5. 最终结果

随着网格的加密，计算结果渐渐趋于稳定， 100×20 与 1000×200 结果已经相当接近，计算结果较为可靠，所以最终结果取 100×20 网格的数据。

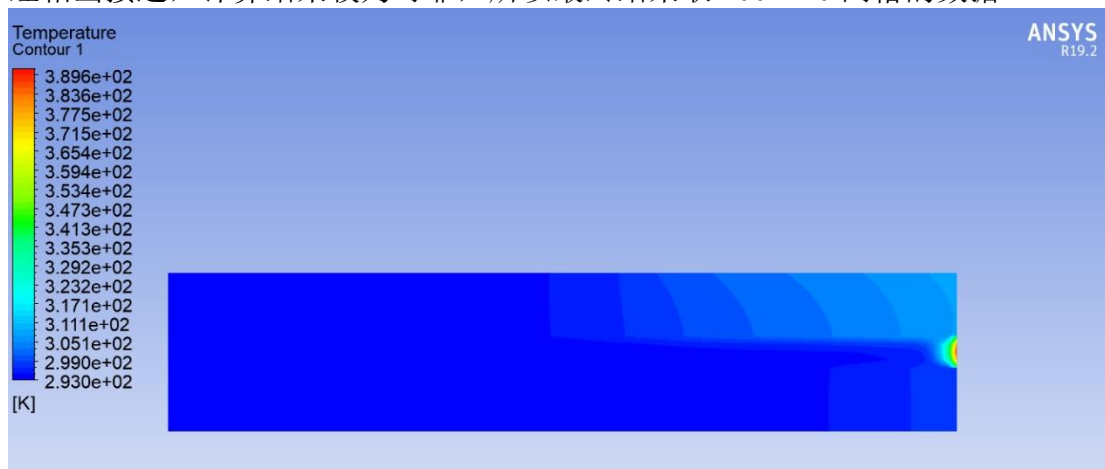


图 73 5sFluent 仿真

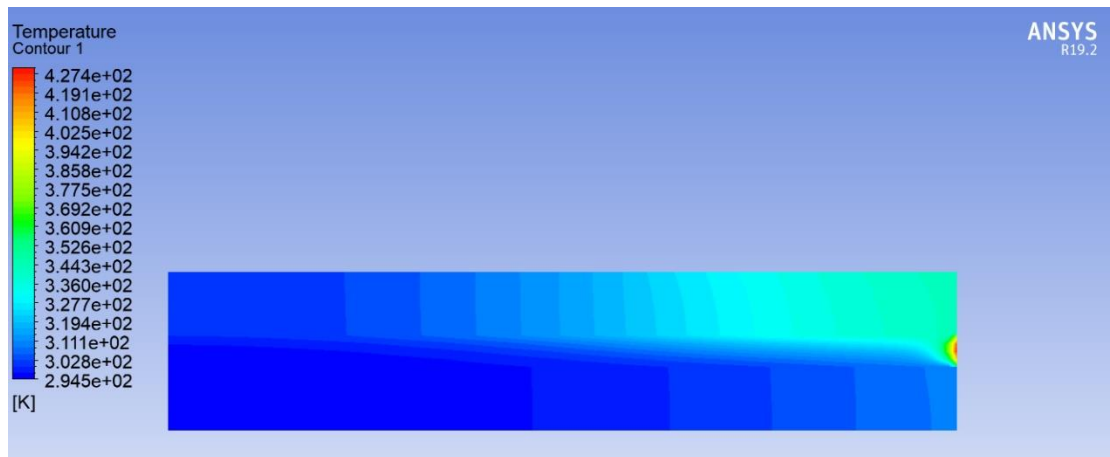


图 74 30sFluent 仿真

七、 参考文献 2

- [1]杨世铭、陶文铨. (2006). 传热学-第 4 版.
- [2]郑亚. (2006). 固体火箭发动机传热学. 北京航空航天大学出版社等.
- [3]杨世铭. 细长竖圆柱外及竖圆管内的自然对流传热[J]. 西安交通大学学报, 1980(03):119-135.

八、 程序代码

1. 显式代码

```
1. # 二维非稳态导热问题的有限体积数值解法#
2. from scipy import linalg
3. import os
4. # 定义求解函数,形参为网格数目,求解时间,eff1,eff2
5. def solve(N_x_grid, N_y_grid ,t,eff1,eff2):
6.     # 物性参数: 发动机长度, 高温辐射区域长度, 两层材料的宽度、热传导系数、密度、比
       热容, 燃气温度, 初始温度,玻尔兹曼常数
7.     L = 0.05
8.     L1 = 0.02
9.     HH = 0.004
10.    hh = 0.002
11.    k1 = 53.6
12.    k2 = 0.2093
13.    k0 = 0.41697
14.    rou1 = 7830
15.    rou2 = 1500
16.    Cp1 = 465
17.    Cp2 = 1465
18.    Tair= 293
19.    Tgas = 923
20.    T0 = 293 # 初始温度
21.    sigma = 5.67 * 10 ** -8
22.
23.    # 网格尺寸:
24.    delta_x = L / N_x_grid # 20 个网格为 0.0025
25.    delta_y = (2 * HH + hh) / N_y_grid # 20 个网格为 0.0005
26.
27.    # 时间步长及迭代次数:
28.    delta_t = 0.004*(20/N_x_grid)*(20/N_y_grid)
29.    cal_num = int(t / delta_t)
30.
31.    # 定义求解线性方程组 AA*T=CC 的动态数组 AA、CC:
32.    AA = np.zeros((N_x_grid * N_y_grid, N_x_grid * N_y_grid))
33.    CC = np.zeros((N_x_grid * N_y_grid))
34.
35.    # 定义初始 0 矩阵:
36.    a_w = np.zeros((N_x_grid, N_y_grid))
37.    a_e = np.zeros((N_x_grid, N_y_grid))
38.    a_n = np.zeros((N_x_grid, N_y_grid))
39.    a_s = np.zeros((N_x_grid, N_y_grid))
40.    sp = np.zeros((N_x_grid, N_y_grid))
```

```

41.     su = np.zeros((N_x_grid, N_y_grid))
42.     a_p = np.zeros((N_x_grid, N_y_grid))
43.     a_p0 = np.zeros((N_x_grid, N_y_grid))
44.     T = np.zeros((N_x_grid, N_y_grid))
45.     X = np.zeros((N_x_grid, N_y_grid))
46.     Y = np.zeros((N_x_grid, N_y_grid))
47.
48.     # 离散方程系数计算
49.     for k in range(0, cal_num):
50.         # 给定初始温度场
51.         if k== 0:
52.             for i in range(0, N_x_grid):
53.                 for j in range(0, N_y_grid):
54.                     T[i][j] = T0
55.                     Y[i][j] = delta_y * (j+1)
56.                     X[i][j] = delta_x * (i+1)
57.
58.         else:
59.             for i in range(0, N_x_grid):
60.                 for j in range(0, N_y_grid):
61.                     # 结点位置:
62.                     Y[i][j] = delta_y * (j + 1)
63.                     X[i][j] = delta_x * (i + 1)
64.
65.                     #对流换热系数计算:
66.                     tm=(T[i][j]-253)/2
67.                     beta=2/(T[i][j]+293)
68.                     Pr=4*10**(-14)*tm**6-3*10**(-11)*tm**5+6*10**(-9)*tm**4-
7*10**(-7)*tm**3+4*10**(-5)*tm**2-0.0012*tm+0.7163
69.                     v=9*10**(-11)*tm**2+9.13*10**(-8)*tm+1.3175*10**(-5)
70.                     Gr=9.8*beta*(T[i][j]-293)*0.03**3/v/v
71.                     if Gr<0:
72.                         print(i,j,Gr,T[i][j],Tair)
73.                         break
74.                     Nu=0.54*(Gr*Pr)**0.25
75.                     h=2.59/3*Nu
76.
77.                     # 1.判断结点属于哪个材料, 给出 a_e,a_w,a_s,a_n,a_p 相应系
数
78.                     if j <= 0.4 * N_y_grid - 1 or j >= 0.6 * N_y_grid: # 20
个网格对应 0—7, 12—19
79.                         a_w[i][j] = k1 / delta_x * delta_y # 10.72
80.                         a_e[i][j] = k1 / delta_x * delta_y # 10.72
81.                         a_s[i][j] = k1 / delta_y * delta_x # 268

```

```

82.             a_n[i][j] = k1 / delta_y * delta_x # 268
83.             a_p[i][j] = rou1 * Cp1 * delta_x * delta_y / delta_t

84.         else: # 20 个网格对应 8—11
85.             a_w[i][j] = k2 / delta_x * delta_y
86.             a_e[i][j] = k2 / delta_x * delta_y
87.             a_s[i][j] = k2 / delta_y * delta_x
88.             a_n[i][j] = k2 / delta_y * delta_x
89.             a_p[i][j] = rou2 * Cp2 * delta_x * delta_y / delta_t

90.
91.         # 2. 交界面处系数
92.         if j == 0.6 * N_y_grid - 1 or j == 0.4 * N_y_grid - 1:
93.             # 7, 11
94.             a_n[i][j] = k0 / delta_y * delta_x
95.             if j == 0.6 * N_y_grid or j == 0.4 * N_y_grid: # 8,
96.                 12
97.             a_s[i][j] = k0 / delta_y * delta_x
98.
99.         # 3. 边界条件
100.        if i == 0: # 左边界为绝热边界
101.            a_w[i][j] = 0.0
102.            su[i][j] = 0.0
103.            sp[i][j] = 0.0
104.        if i == N_x_grid - 1: # 右边界为辐射换热边界
105.            a_e[i][j] = 0.0
106.            if j <= 0.4 * N_y_grid - 1 or j >= 0.6 * N_y_grid:
107.                #材料 1 辐射换热系数 eff1
108.                su[i][j] = eff1 * sigma * (Tgas ** 4 - (T[i][j]
109.                ) ** 4) * delta_y
110.            else : #材料 2 辐射换热系数 eff2
111.                su[i][j] = eff2 * sigma * (Tgas ** 4 - (T[i][j]
112.                ) ** 4) * delta_y
113.            sp[i][j] = 0.0
114.        if j == N_y_grid - 1: # 上边界为混合边界
115.            if i <= N_x_grid * 0.6 - 1: # 上边界左侧为对流换热
116.                a_n[i][j] = 0.0
117.                su[i][j] = h * (Tair - T[i][j]) * delta_x
118.                sp[i][j] = 0
119.            else: #改动 1: 上边界右侧为辐射换热
120.                a_n[i][j] = 0.0
121.                su[i][j] = eff1 * sigma * (Tgas ** 4 - (T[i][j]
122.                ) ** 4) * delta_x
123.                sp[i][j] = 0.0

```

```

118.             if j == 0: # 下边界为绝热边界
119.                 a_s[i][j] = 0.0
120.                 su[i][j] = 0.0
121.                 sp[i][j] = 0.0
122.
123.             # 4.四个角点系数
124.             if i == 0 and j == 0: # 左下角点
125.                 su[i][j] = 0.0
126.             if i == 0 and j == N_y_grid - 1: # 左上角点
127.                 su[i][j] = h * (Tair - T[i][j]) * delta_x
128.             if i == N_x_grid - 1 and j == N_y_grid - 1: # 右上角
点
129.                 su[i][j] = eff1 * sigma * (Tgas ** 4 - (T[i][j]) **
4) * delta_x + eff1 * sigma * (Tgas ** 4 - (T[i][j]) ** 4) * delta_y
130.             if i == N_x_grid - 1 and j == 0: # 右下角点
131.                 su[i][j] = eff1 * sigma * (Tgas ** 4 - (T[i][j]) **
4) * delta_y
132.
133.             # 5.最终 a_p0 系数
134.             a_p0[i][j] = a_p[i][j] - (a_w[i][j] + a_e[i][j] + a_s[i
][j] + a_n[i][j] - sp[i][j])
135.
136.             # 将系数给入系数矩阵 AA,CC
137.             AA[i * N_y_grid + j][i * N_y_grid + j] = a_p[i][j]
138.             CC[i * N_y_grid + j] = su[i][j] + a_p0[i][j] * T[i][j]
139.
140.             if j != 0:
141.                 CC[i * N_y_grid + j] = CC[i * N_y_grid + j] + a_s[i
][j] * T[i][j - 1]
142.             if j != N_y_grid - 1:
143.                 CC[i * N_y_grid + j] = CC[i * N_y_grid + j] + a_n[i
][j] * T[i][j + 1]
144.             if i != 0:
145.                 CC[i * N_y_grid + j] = CC[i * N_y_grid + j] + a_w[i
][j] * T[i - 1][j]
146.             if i != N_x_grid - 1:
147.                 CC[i * N_y_grid + j] = CC[i * N_y_grid + j] + a_e[i
][j] * T[i + 1][j]
148.
149.             #调用 scipy 库求解线性方程组
150.             result=linalg.solve(AA, CC)
151.             for i in range(0, N_x_grid):
152.                 for j in range(0, N_y_grid):
153.                     if result[i * N_y_grid + j]>293:

```

```

153.             T[i][j] = result[i * N_y_grid + j]
154.         if k*delta_t==5 or k*delta_t==29.96:
155.             for i in range(0,N_x_grid):
156.                 print((T[i][9]+T[i][10])/2)
157.
158. #####
159.     #建立文件夹及写入温度计算数据，导入 tecplot 进行后处理
160.     # 判断是否存在文件夹，不存在则建立
161.     if not os.path.exists('G:\python project\python project1\内流场大
        作业代码上边界对流 eff={0}-{1}-n_x={2}-
        n_y={3}' .format(eff1,eff2,N_x_grid,N_y_grid)):
162.         os.mkdir('G:\python project\python project1\内流场大作业代码上
        边界对流 eff={0}-{1}-n_x={2}-n_y={3}' .format(eff1,eff2,N_x_grid,N_y_grid))
163.
164.         if k%(0.02/delta_t)==0 or k==cal_num :#每隔 0.02 秒输出一个 plt 文件，
            避免文件太多导入 tecplot 太慢
165.             with open('G:\python project\python project1\内流场大作业代码
                上边界对流 eff={3}-{4}-n_x={5}-n_y={6}\T-2d-Nx={0}-Ny={1}-
                t={2}.plt' .format(N_x_grid, N_y_grid, k * delta_t,eff1,eff2,N_x_grid, N_y_g
                rid), 'w', encoding='UTF-8') as fp1:
166.                 fp1.write("VARIABLES = X, Y, T, ap,an,as,aw,ae,sp,su\n") #
                    按计算节点数输出结果
167.                 fp1.write("ZONE I=%d,J=%d, F=POINT,t=\"%3f\"\n" % (N_x_gri
                    d, N_y_grid,k*delta_t))
168.                 for j in range(N_y_grid - 1, -1, -1):
169.                     for i in range(0, N_x_grid):
170.                         fp1.write(
171.                             "%6.5Lf %6.5Lf %6.3Lf %6.3Lf %6.3Lf %
                                6.3Lf %6.3Lf %6.3Lf %6.3Lf %6.3Lf\n" % (
172.                                 X[i][j], Y[i][j], T[i][j], a_p[i][j], a_n[i
                                    ][j], a_s[i][j], a_w[i][j],
173.                                 a_e[i][j], sp[i][j], su[i][j]))
174.
175.
176. # 调用函数的主程序
177. import numpy as np
178. #N_x_grid,N_y_grid,t,eff1,eff2=input("请输入 x 方向网格数量, y 方向网格数量, 计
        算时间, 材料 1 辐射率, 材料 2 辐射率").split()
179. #solve(int(N_x_grid),int(N_y_grid),int(t),float(eff1),float(eff2))#N_xgrid=
        20,N_ygrid=20,t=50,eff1=0.6949,eff2=0.7515
180. solve(20,20,30,0.6949,0.7515)#N_xgrid=20,N_ygrid=20,t=50,eff1=0.6949,eff2=0
        .7515

```


2. 半隐式代码

```
1. """author:gaoyi"""
2. import numpy as np
3. from time import time,sleep
4.
5. #####
   #####
6. """参数"""
7. rho1 = 7830 #材料 1 的密度
8. rho2 = 1500 #材料 2 的密度
9. c1 = 465 #材料 1 的比热
10. c2 = 1465.4 #材料 2 的比热
11. k1 = 53.6 #材料 1 的导热系数
12. k2 = 0.2093 #材料 2 的导热系数
13. k3 = 2*k1*k2/(k1+k2) #导热系数 -> 界面
14. T_air = 293 #大气温度,K
15. T_gas = 923 #燃气温度,K
16.
17. """辐射边界相关参数"""
18. sigma = 5.67*10**-8
19. # epsilon_eff = 0.7
20.
21. """几何参数"""
22. H1 = 0.004
23. H2 = 0.002
24. H3 = 0.004
25. H = H1+H2+H3
26.
27. NX = 50 #x 方向单元体数量
28. NY = 40 #y 方向单元体数量
29.
30. NY_1 = NY*(H1/H)
31. NY_2 = NY*(H2/H)
32. NY_3 = NY*(H3/H)
33.
34. L = 0.05
35.
36. """初始化矩阵"""
37. a_e = np.zeros((NX, NY))
38. a_w = np.zeros((NX, NY))
39. a_n = np.zeros((NX, NY))
40. a_s = np.zeros((NX, NY))
41. a_p = np.zeros((NX, NY))
```

```

42. CN = np.zeros((NX, NY))
43. T_0 = np.zeros((NX+2, NY+2))
44. T = np.zeros((NX+2, NY+2))
45.
46. delta_x = L / NX
47. delta_y = (H1 + H2 + H3) / NY
48.
49. s_u = np.zeros((NX, NY))
50. s_p = np.zeros((NX, NY))
51.
52. m = float(input("请输入迭代系数:"))    #迭代系数
53. print("迭代系数为:{}".format(m))
54.
55. ap0 = np.zeros((NX, NY))
56.
57. # delta_t1 = rho1*c1/k1*(delta_y**2)
58. # delta_t2 = rho2*c2/k2*(delta_y**2)
59. # delta_t3 = rho1*c1/k1*(delta_x**2)
60. # delta_t4 = rho2*c2/k2*(delta_x**2)
61.
62. delta_t = 2*min(rho1*c1, rho2*c2)/max(k1, k2)*delta_x*delta_y/(2*delta_y/delta_x+2*delta_x/delta_y)
63.
64. # delta_t = "%.6f" % min(delta_t1, delta_t2, delta_t3, delta_t4)    #时间步长
65. # delta_t = float(delta_t)
66. print("时间步长为:{}".format(delta_t))
67. delta_t = float(input("请输入手动修正后的时间步长:\n"))
68.
69. sec = 30    #计算秒数
70.
71. #####
72. def cal_h(T):
73.     """计算自然对流换热系数"""
74.     if T > T_air:
75.         tm = (T-253)/2
76.         beta = 2/(T+293)
77.         Pr = 4*10**-14*tm**6-3*10**-11*tm**5+6*10**-9*tm**-4-7*10**-7*tm**3+4*10**-5*tm**2-0.0012*tm+0.7163
78.         v = 9*10**-11*tm**2+9.13*10**-8*tm+1.3175*10**-5
79.         Gr = 9.8*beta*(T-293)*0.03**2/(v**2)
80.         Nu = 0.54*(Gr*Pr)**(1/4)
81.         h = Nu*2.59/3

```

```

82.     else:
83.         h = 0
84.     return h
85.
86.
87. #####
88. #####
89. def cal_epsilon(j):
90.     if 0 <= j <= NY_1-1:
91.         e = 0.6949
92.     elif NY_1 <= j <= NY_1+NY_2-1:
93.         e = 0.7515
94.     elif NY_1+NY_2 <= j <= NY-1:
95.         e = 0.6949
96.     return e
97.
98.
99. #####
100. #####
101. def cal_property(i, j):
102.     """计算节点的物性参数"""
103.     """初始化变量"""
104.     k = 0
105.     ks = 0
106.     kn = 0
107.     rho = 0
108.     c = 0
109.     """判断节点位置并赋值"""
110.     if j == NY_1-1:
111.         k = k1
112.         ks = k1
113.         kn = k3
114.         rho = rho1
115.         c = c1
116.
117.
118.     elif j == NY_1:
119.         k = k2
120.         ks = k3
121.         kn = k2
122.         rho = rho2
123.         c = c2

```

```

124.     elif j == NY_1+NY_2-1:
125.         k = k2
126.         ks = k2
127.         kn = k3
128.         rho = rho2
129.         c = c2
130.     elif j == NY_1+NY_2:
131.         k = k1
132.         ks = k3
133.         kn = k1
134.         rho = rho1
135.         c = c1
136.     elif 0 <= j <= (NY_1-2) or (NY_1+NY_2+1) <= j <= (NY-1):
137.         k = k1
138.         ks = k1
139.         kn = k1
140.         rho = rho1
141.         c = c1
142.     elif (NY_1+1) <= j <= (NY_1+NY_2-2):
143.         k = k2
144.         ks = k2
145.         kn = k2
146.         rho = rho2
147.         c = c2
148.     else:
149.         print("error, 无法计算物性参数!!! ")
150.
151.     return [k, ks, kn, rho, c]
152.
153.
154. #####
155. """计算系数矩阵"""
156. def cal_matrix(i, j):
157.
158.     h = cal_h(T[i,j])
159.     [k,ks,kn,rho,c] = cal_property(i,j)
160.     epsilon_eff = cal_epsilon(j)
161.
162.     if j == (NY-1) and i != 0 and i != (NX-1):#上边界 - 自然对流+辐射
163.         if 0 <= i <= (NX*3/5-1):#自然对流部分
164.
165.             a_e[i,j] = k*delta_y/delta_x
166.             a_w[i,j] = k*delta_y/delta_x

```

```

167.         a_s[i,j] = ks*delta_x/delta_y
168.         a_n[i,j] = 0
169.
170.         s_u[i,j] = h*delta_x*T_air
171.         s_p[i,j] = -h*delta_x
172.
173.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
174.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
175.
176.         elif (NX*3/5) <= i <= (NX-1):#辐射部分
177.
178.         a_e[i,j] = k*delta_y/delta_x
179.         a_w[i,j] = k*delta_y/delta_x
180.         a_s[i,j] = ks*delta_x/delta_y
181.         a_n[i,j] = 0
182.
183.         s_u[i,j] = epsilon_eff*sigma*(T_gas**4-T_0[i,j]**4)*delta_y
184.         s_p[i,j] = 0
185.
186.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
187.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
188.
189.         elif i == (NX-1) and j != 0 and j != (NY-1):#右边界 - 辐射
190.
191.         a_e[i,j] = 0
192.         a_w[i,j] = k*delta_y/delta_x
193.         a_s[i,j] = ks*delta_x/delta_y
194.         a_n[i,j] = kn*delta_x/delta_y
195.
196.         s_u[i,j] = epsilon_eff*sigma*(T_gas**4-T_0[i,j]**4)*delta_y
197.         s_p[i,j] = 0
198.
199.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
200.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
201.
202.         elif j == 0 and i != 0 and i != (NX-1):#下边界 - 绝热
203.
204.         a_e[i,j] = k*delta_y/delta_x
205.         a_w[i,j] = k*delta_y/delta_x
206.         a_s[i,j] = 0
207.         a_n[i,j] = kn*delta_x/delta_y

```

```

208.
209.     s_u[i,j] = 0
210.     s_p[i,j] = 0
211.
212.     ap0[i,j] = rho*c*delta_x*delta_y/delta_t
213.     a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
214.
215.     elif i == 0 and j != 0 and j != (NY-1):#左边界 - 绝热
216.
217.         a_e[i,j] = k*delta_y/delta_x
218.         a_w[i,j] = 0
219.         a_s[i,j] = ks*delta_x/delta_y
220.         a_n[i,j] = kn*delta_x/delta_y
221.
222.         s_u[i,j] = 0
223.         s_p[i,j] = 0
224.
225.         ap0[i,j] = rho*c*delta_x*delta_y*delta_t
226.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
227.
228.     elif i == 0 and j == 0:#左下点
229.
230.         a_e[i,j] = k*delta_y/delta_x
231.         a_w[i,j] = 0
232.         a_s[i,j] = 0
233.         a_n[i,j] = kn*delta_x/delta_y
234.
235.         s_u[i,j] = 0
236.         s_p[i,j] = 0
237.
238.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
239.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
240.
241.     elif i == (NX-1) and j == 0:#右下点
242.
243.         a_e[i,j] = 0
244.         a_w[i,j] = k*delta_y/delta_x
245.         a_s[i,j] = 0
246.         a_n[i,j] = kn*delta_x/delta_y
247.
248.         s_u[i,j] = epsilon_eff*sigma*(T_gas**4-T_0[i,j]**4)*delta_y

```

```

249.         s_p[i,j] = 0
250.
251.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
252.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
253.
254.         elif i == (NX-1) and j == (NY-1):#右上点
255.
256.         a_e[i,j] = 0
257.         a_w[i,j] = k*delta_y/delta_x
258.         a_s[i,j] = ks*delta_x/delta_y
259.         a_n[i,j] = 0
260.
261.         s_u[i,j] = epsilon_eff*sigma*(T_gas**4-T_0[i,j]**4)*delta_y
262.         s_p[i,j] = 0
263.
264.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
265.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
266.
267.         elif i == 0 and j == (NY-1):#左上点
268.
269.         a_e[i,j] = k*delta_y/delta_x
270.         a_w[i,j] = 0
271.         a_s[i,j] = ks*delta_x/delta_y
272.         a_n[i,j] = 0
273.
274.         s_u[i,j] = h*delta_x*T_air
275.         s_p[i,j] = -h*delta_x
276.
277.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
278.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
279.
280.         else:#内部
281.
282.         a_e[i,j] = k*delta_y/delta_x
283.         a_w[i,j] = k*delta_y/delta_x
284.         a_s[i,j] = ks*delta_x/delta_y
285.         a_n[i,j] = kn*delta_x/delta_y
286.
287.         s_u[i,j] = 0
288.         s_p[i,j] = 0
289.

```

```

290.         ap0[i,j] = rho*c*delta_x*delta_y/delta_t
291.         a_p[i,j] = ap0[i,j]+ 1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j])
292.
293.
294. #####
        #####
295. def solve():
296.     """迭代法求解一步的温度"""
297.     err = 0#误差
298.
299.     """初始化温度场"""
300.     for i in range(NX+2):
301.         for j in range(NY+2):
302.             T_0[i,j] = T[i,j]
303.
304.     while err < 0.05:
305.         err0 = err
306.         err = 0
307.
308.         """计算系数矩阵"""
309.         for i in range(NX):
310.             for j in range(NY):
311.                 cal_matrix(i,j)
312.
313.         for i in range(NX):
314.             for j in range(NY):
315.                 n = T_0[i,j]+m*((
316.                     a_e[i,j]*(T_0[i+1,j]+T[i+1,j])/2+
317.                     a_w[i,j]*(T_0[i-1,j]+T[i-1,j])/2+
318.                     a_s[i,j]*(T_0[i,j-1]+T[i,j-1])/2+
319.                     a_n[i,j]*(T_0[i,j+1]+T[i,j+1])/2+
320.                     (ap0[i,j]-1/2*(a_e[i,j]+a_w[i,j]+a_s[i,j]+a_n[i,j]-
        s_p[i,j]))*T_0[i,j]
321.                     +s_u[i,j])/a_p[i,j]-T_0[i,j])
322.
323.                 err = err+abs(T_0[i,j]-n)
324.                 #print("%.8f"%(abs(T_0[i,j]-n)))
325.                 T_0[i,j] = n
326.
327.                 #print("err:{ }\n".format("%.8f"%err))
328.
329.     for i in range(NX+2):
330.         for j in range(NY+2):

```



```

331.         T[i, j] = T_0[i, j]
332.
333.
334. #####
    #####
335. t1=time()
336.
337. for i in range(NX+2):
338.     for j in range(NY+2):
339.         T[i,j] = T_air
340.
341.         if i == 0 or i == NX+1 or j == 0 or j == NY+1:
342.             T[i,j] = T_air
343.
344. for i in range(1, sec+1):
345.     for j in range(1, int(1/delta_t+1)):
346.         solve()
347.         t2 = time()
348.         print("正在迭代计算第{}秒的温度场, 请稍
    等...{}min".format("%02d" % i, "%.4f" % ((t2-t1)/60)))
349.
350.     file_path = r"D:\python_project\college_assignment\data_half\data_t={s
    .plt".format(i)
351.     with open(file=file_path, mode="w") as f:
352.         # f.write('TITLE = "m={}\n".format("%.10f" % m))
353.         f.write("VARIABLES = X,Y,T\n")
354.         f.write('ZONE T="ZONE 1",I={},J={},F=POINT,t="{ }\n'.format(NX,NY,i
    ))
355.
356.         for k in range(NY):
357.             yy = delta_y/2+delta_y*k
358.             for l in range(NX):
359.
360.                 xx = delta_x/2+delta_x*l
361.                 f.write("{} {} {} \n".format(
362.                     "%.8f"%xx,
363.                     "%.8f"%yy,
364.                     "%.4f"%T[l, k],
365.                 ))

```

3. 隐式代码

```

1. #include <iostream>
2. #include <math.h>

```

```

3. #include <stdio.h>
4. #include <stdlib.h>
5. #include <cstring>
6. #include <algorithm>
7. #include <Eigen/Dense>
8.
9. #define rou1 7830. //物性参数
10. #define rou2 1500.0
11. #define c1 465.0
12. #define c2 1465.4
13. #define k1 53.6
14. #define k2 0.2093
15. #define Tair 293.0
16. #define Tgas 923.0
17. // #define h 8
18. #define epsilon1 0.6949
19. #define epsilon2 0.7515
20. #define sigma 5.67*pow(10,-8)
21. #define NX 20 //网格尺寸
22. #define NY1 8
23. #define NY2 4
24. #define NY3 8
25. #define NY (NY1+NY2+NY3)
26. #define L 0.05 //模型尺寸
27. #define L1 0.02
28. #define L2 0.03
29. #define H1 0.004
30. #define H2 0.002
31. #define H3 0.004
32. #define del_x (L/NX) // 0.0025
33. #define del_y (H1/NY1) // 0.0005
34. #define del_t 0.02
35.
36. using namespace Eigen; // 改成这样亦
   可 using Eigen::MatrixXd;
37. using namespace std;
38.
39. double AP[NX + 1][NY + 1], AP0[NX + 1][NY + 1],
40. AW[NX + 1][NY + 1], AE[NX + 1][NY + 1], AN[NX + 1][NY + 1], AS[NX + 1][NY +
   1],
41. SU[NX + 1][NY + 1], SP[NX + 1][NY + 1],
42. T0[NX + 2][NY + 2], T[NX + 2][NY + 2],
43. A[(NX + 1)*(NY + 1)][(NX + 1)*(NY + 1)], B[(NX + 1)*(NY + 1)],
44. x[NX + 2], y[NX + 2];

```

```

45.
46. void boundary_matrix(int i,int j)
    //AP,AP0,AW,AE,AN,AS,SU,SP 的确定
47. {
48.     double h = 0.0, tm = 0.0, beta = 0.0, Pr = 0.0, niu = 0.0, Gr = 0.0, Nu
        = 0.0;
49.     if (i == 0 && j == 0)
        //左下角: 1
50.     {
51.         AW[i][j] = 0;
52.         AE[i][j] = k1 * del_y / del_x;
53.         AS[i][j] = 0;
54.         AN[i][j] = k1 * del_x / del_y;
55.         SU[i][j] = 0;
56.         SP[i][j] = 0;
57.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
58.     }
59.     if (j == 0 && ((i > 0 && i < (NY1 - 1)) || (i > (NY1 + NY2) && i < (NY -
        1)))) //材料 1 左边界: 2
60.     {
61.         AW[i][j] = 0;
62.         AE[i][j] = k1 * del_y / del_x;
63.         AS[i][j] = k1 * del_x / del_y;
64.         AN[i][j] = k1 * del_x / del_y;
65.         SU[i][j] = 0;
66.         SP[i][j] = 0;
67.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
68.     }
69.     if (j == 0 && i == (NY1 - 1))
        //材料 1、2 交界处左边界: 3
70.     {
71.         AW[i][j] = 0;
72.         AE[i][j] = k1 * del_y / del_x;
73.         AS[i][j] = k1 * del_x / del_y;
74.         AN[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
75.         SU[i][j] = 0;
76.         SP[i][j] = 0;
77.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
78.     }
79.     if (j == 0 && i == NY1)
        //材料 1、2 交界处左边界: 4
80.     {
81.         AW[i][j] = 0;
82.         AE[i][j] = k2 * del_y / del_x;

```

```

83.      AS[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
84.      AN[i][j] = k2 * del_x / del_y;
85.      SU[i][j] = 0;
86.      SP[i][j] = 0;
87.      AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
88.  }
89.  if (j == 0 && i > NY1 && i < (NY1 + NY2 - 1))
      //材料2左边界: 5
90.  {
91.      AW[i][j] = 0;
92.      AE[i][j] = k2 * del_y / del_x;
93.      AS[i][j] = k2 * del_x / del_y;
94.      AN[i][j] = k2 * del_x / del_y;
95.      SU[i][j] = 0;
96.      SP[i][j] = 0;
97.      AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
98.  }
99.  if (j == 0 && i == (NY1 + NY2 - 1))
      //材料2、1交界处左边界: 6
100.  {
101.      AW[i][j] = 0;
102.      AE[i][j] = k2 * del_y / del_x;
103.      AS[i][j] = k2 * del_x / del_y;
104.      AN[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
105.      SU[i][j] = 0;
106.      SP[i][j] = 0;
107.      AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
108.  }
109.  if (j == 0 && i == (NY1+ NY2))
      //材料1、2交界处左边界: 7
110.  {
111.      AW[i][j] = 0;
112.      AE[i][j] = k1 * del_y / del_x;
113.      AS[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
114.      AN[i][j] = k1 * del_x / del_y;
115.      SU[i][j] = 0;
116.      SP[i][j] = 0;
117.      AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
118.  }
119.  if (j == 0 && i == (NY - 1))
      //左上角: 8
120.  {
121.      tm = (T0[i][j] - 253.0) / 2.0;
122.      beta = 2.0 / (T0[i][j] + 293.0);

```

```

123.      Pr = 4 * pow(10, -14)*pow(tm, 6) - 3 * pow(10, -
      11)*pow(tm, 5) + 6 * pow(10, -9)*pow(tm, 4) - 7 * pow(10, -
      7)*pow(tm, 3) + 4 * pow(10, -5)*pow(tm, 2) - 0.0012*tm + 0.7163;
124.      niu = 9 * pow(10, -11)*pow(tm, 2) + 9.13*pow(10, -
      8)*tm + 1.3175*pow(10, -5);
125.      Gr = 9.8*beta*fabs(T0[i][j] - 293.0)*pow(0.03, 3) / pow(niu, 2);
126.      Nu = 0.54*pow(Gr*Pr, 0.25);
127.      h = 2.59 *Nu / 3.0;
128.      AW[i][j] = 0;
129.      AE[i][j] = k1 * del_y / del_x;
130.      AS[i][j] = k1 * del_x / del_y;
131.      AN[i][j] = 0;
132.      SU[i][j] = h * del_x*Tair;
133.      SP[i][j] = -h * del_x;
134.      AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
135.  }
136.  if (i == (NY - 1) && j > 0 && j <= ((L2 / del_x) - 1))
      //上边界左半部分: 9
137.  {
138.      tm = (T0[i][j] - 253.0) / 2.0;
139.      beta = 2.0 / (T0[i][j] + 293.0);
140.      Pr = 4 * pow(10, -14)*pow(tm, 6) - 3 * pow(10, -
      11)*pow(tm, 5) + 6 * pow(10, -9)*pow(tm, 4) - 7 * pow(10, -
      7)*pow(tm, 3) + 4 * pow(10, -5)*pow(tm, 2) - 0.0012*tm + 0.7163;
141.      niu = 9 * pow(10, -11)*pow(tm, 2) + 9.13*pow(10, -
      8)*tm + 1.3175*pow(10, -5);
142.      Gr = 9.8*beta*fabs(T0[i][j] - 293.0)*pow(0.03, 3) / pow(niu, 2);
143.      Nu = 0.54*pow(Gr*Pr, 0.25);
144.      h = 2.59 *Nu / 3.0;
145.      AW[i][j] = k1 * del_y / del_x;
146.      AE[i][j] = k1 * del_y / del_x;
147.      AS[i][j] = k1 * del_x / del_y;
148.      AN[i][j] = 0;
149.      SU[i][j] = h * del_x*Tair;
150.      SP[i][j] = -h * del_x;
151.      AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
152.  }
153.  if (i == (NY - 1) && j > ((L2 / del_x) -
      1) && j < (NX - 1)) //上边界右半部分: 10
154.  {
155.      AW[i][j] = k1 * del_y / del_x;
156.      AE[i][j] = k1 * del_y / del_x;
157.      AS[i][j] = k1 * del_x / del_y;
158.      AN[i][j] = 0;

```

```

159.         SU[i][j] = epsilon1 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
        x;
160.         SP[i][j] = 0;
161.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
162.     }
163.     if (i == (NY - 1) && j == (NX - 1))
        //右上角: 11
164.     {
165.         AW[i][j] = k1 * del_y / del_x;
166.         AE[i][j] = 0;
167.         AS[i][j] = k1 * del_x / del_y;
168.         AN[i][j] = 0;
169.         SU[i][j] = epsilon1 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* (del
        _x + del_y);
170.         SP[i][j] = 0;
171.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
172.     }
173.     if (i == 0 && j == (NX - 1))
        //右下角: 12
174.     {
175.         AW[i][j] = k1 * del_y / del_x;
176.         AE[i][j] = 0;
177.         AS[i][j] = 0;
178.         AN[i][j] = k1 * del_x / del_y;
179.         SU[i][j] = epsilon1 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
        y;
180.         SP[i][j] = 0;
181.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
182.     }
183.     if (j == (NX - 1) && ((i > 0 && i < (NY1 - 1)) || (i > (NY1 + NY2) && i
        < (NY - 1))))//材料 1 右边界: 13
184.     {
185.         AW[i][j] = k1 * del_y / del_x;
186.         AE[i][j] = 0;
187.         AS[i][j] = k1 * del_x / del_y;
188.         AN[i][j] = k1 * del_x / del_y;
189.         SU[i][j] = epsilon1 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
        y;
190.         SP[i][j] = 0;
191.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
192.     }
193.     if (j == (NX - 1) && i == (NY1 - 1))
        //材料 1、2 交界处右边界: 14
194.     {

```

```

195.      AW[i][j] = k1 * del_y / del_x;
196.      AE[i][j] = 0;
197.      AS[i][j] = k1 * del_x / del_y;
198.      AN[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
199.      SU[i][j] = epsilon1 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
      y;
200.      SP[i][j] = 0;
201.      AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
202.  }
203.  if (j == (NX - 1) && i == NY1)
      //材料 1、2 交界处右边界: 15
204.  {
205.      AW[i][j] = k2 * del_y / del_x;
206.      AE[i][j] = 0;
207.      AS[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
208.      AN[i][j] = k2 * del_x / del_y;
209.      SU[i][j] = epsilon2 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
      y;
210.      SP[i][j] = 0;
211.      AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
212.  }
213.  if (j == (NX - 1) && i > NY1 && i < (NY1 + NY2 - 1))
      //材料 2 右边界: 16
214.  {
215.      AW[i][j] = k2 * del_y / del_x;
216.      AE[i][j] = 0;
217.      AS[i][j] = k2 * del_x / del_y;
218.      AN[i][j] = k2 * del_x / del_y;
219.      SU[i][j] = epsilon2 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
      y;
220.      SP[i][j] = 0;
221.      AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
222.  }
223.  if (j == (NX - 1) && i == (NY1 + NY2 - 1))
      //材料 2、1 交界处右边界: 17
224.  {
225.      AW[i][j] = k2 * del_y / del_x;
226.      AE[i][j] = 0;
227.      AS[i][j] = k2 * del_x / del_y;
228.      AN[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
229.      SU[i][j] = epsilon2 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
      y;
230.      SP[i][j] = 0;
231.      AP0[i][j] = rou2 * c2* del_x *del_y / del_t;

```

```

232.     }
233.     if (j == (NX - 1) && i == (NY1 + NY2))
           //材料 1、2 交界处右边界: 18
234.     {
235.         AW[i][j] = k1 * del_y / del_x;
236.         AE[i][j] = 0;
237.         AS[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
238.         AN[i][j] = k1 * del_x / del_y;
239.         SU[i][j] = epsilon1 * sigma*(pow(Tgas, 4) - pow(T0[i][j], 4))* del_
           y;
240.         SP[i][j] = 0;
241.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
242.     }
243.     if (i == 0 && j > 0 && j < (NX - 1))
           //下边界: 19
244.     {
245.         AW[i][j] = k1 * del_y / del_x;
246.         AE[i][j] = k1 * del_y / del_x;
247.         AS[i][j] = 0;
248.         AN[i][j] = k1 * del_x / del_y;
249.         SU[i][j] = 0;
250.         SP[i][j] = 0;
251.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
252.     }
253.     if (j > 0 && j < (NX - 1) && ((i > 0 && i < (NY1 - 1)) || (i > (NY1 + N
           Y2) && i < (NY - 1))))//材料 1 内部: 20
254.     {
255.         AW[i][j] = k1 * del_y / del_x;
256.         AE[i][j] = k1 * del_y / del_x;
257.         AS[i][j] = k1 * del_x / del_y;
258.         AN[i][j] = k1 * del_x / del_y;
259.         SU[i][j] = 0;
260.         SP[i][j] = 0;
261.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
262.     }
263.     if (j > 0 && j < (NX - 1) && i == (NY1 - 1))
           //材料 1、2 交界处内部: 21
264.     {
265.         AW[i][j] = k1 * del_y / del_x;
266.         AE[i][j] = k1 * del_y / del_x;
267.         AS[i][j] = k1 * del_x / del_y;
268.         AN[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
269.         SU[i][j] = 0;
270.         SP[i][j] = 0;

```



```

271.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
272.     }
273.     if (j > 0 && j < (NX - 1) && i == NY1)
           //材料 1、2 交界处内部: 22
274.     {
275.         AW[i][j] = k2 * del_y / del_x;
276.         AE[i][j] = k2 * del_y / del_x;
277.         AS[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
278.         AN[i][j] = k2 * del_x / del_y;
279.         SU[i][j] = 0;
280.         SP[i][j] = 0;
281.         AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
282.     }
283.     if (j > 0 && j < (NX - 1) && i > NY1 && i < (NY1 + NY2 - 1))
           //材料 2 内部: 23
284.     {
285.         AW[i][j] = k2 * del_y / del_x;
286.         AE[i][j] = k2 * del_y / del_x;
287.         AS[i][j] = k2 * del_x / del_y;
288.         AN[i][j] = k2 * del_x / del_y;
289.         SU[i][j] = 0;
290.         SP[i][j] = 0;
291.         AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
292.     }
293.     if (j > 0 && j < (NX - 1) && i == (NY1 + NY2 - 1))
           //材料 2、1 交界处内部: 24
294.     {
295.         AW[i][j] = k2 * del_y / del_x;
296.         AE[i][j] = k2 * del_y / del_x;
297.         AS[i][j] = k2 * del_x / del_y;
298.         AN[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
299.         SU[i][j] = 0;
300.         SP[i][j] = 0;
301.         AP0[i][j] = rou2 * c2* del_x *del_y / del_t;
302.     }
303.     if (j > 0 && j < (NX - 1) && i == (NY1 + NY2))
           //材料 1、2 交界处内部: 25
304.     {
305.         AW[i][j] = k1 * del_y / del_x;
306.         AE[i][j] = k1 * del_y / del_x;
307.         AS[i][j] = 2.0*k1*k2 / (k1 + k2) * del_x / del_y;
308.         AN[i][j] = k1 * del_x / del_y;
309.         SU[i][j] = 0;
310.         SP[i][j] = 0;

```

```

311.         AP0[i][j] = rou1 * c1* del_x *del_y / del_t;
312.     }
313.     AP[i][j] = AP0[i][j] + AW[i][j] + AE[i][j] + AS[i][j] + AN[i][j] - SP[i
    ][j];
314. }
315.
316. void mesh()
317. {
318.     int i;
319.     FILE *fp;
320.         //导出数据
321.     errno_t err_mesh;
322.     err_mesh = fopen_s(&fp, "mesh.dat", "w");
323.     x[0] = 0.0;
324.     fprintf(fp, "    i  dx[i]      xu[i]:      \n");
325.     fprintf(fp, "    0 %5.5f %5.5f\n", 0.0, x[0]);
326.     for (i = 1; i <= NX; i++)
327.     {
328.         x[i] = x[i - 1] + del_x;
329.         fprintf(fp, "%5d %5.5f %5.5f\n", i, del_x, x[i]);
330.     }
331.     fprintf(fp, "\n");
332.     y[0] = 0.0;
333.     fprintf(fp, "    j  dy[j]      yv[j]:      \n");
334.     fprintf(fp, "    0 %5.5f %5.5f\n", 0.0, y[0]);
335.     for (i = 1; i <= NY; i++) {
336.         y[i] = y[i - 1] + del_y;
337.         fprintf(fp, "%5d %5.5f %5.5f\n", i, del_y, y[i]);
338.     }
339.     fprintf(fp, "\n");
340.     fclose(fp);
341. }
342.
343. int main(int argc, char** argv)
344. {
345.     int i, j;
346.     double k;
347.     MatrixXd coefficient_matrix(((NX + 1)*(NY + 1)), ((NX + 1)*(NY + 1)));
348.     coefficient_matrix = MatrixXd::Zero(((NX + 1)*(NY + 1)), ((NX + 1)*(NY
    + 1)));
349.     mesh();
350.     VectorXd Tv_vector((NX + 1)*(NY + 1));

```

```

351.     VectorXd B_vector((NX + 1)*(NY + 1));
352.     for (i = 0; i <= (NY - 1); i++)
353.     {
354.         for (j = 0; j <= (NX - 1); j++)
355.         {
356.             T0[i][j] = 293;
357.             //初始条件
358.         }
359.     }
360.     for (k = 0; k <= 30; k = k + 0.02)
361.     {
362.         for (i = 0; i <= (NY - 1); i++)
363.         {
364.             for (j = 0; j <= (NX - 1); j++)
365.             {
366.                 boundary_matrix(i,j);
367.                 //AP,AP0,AW,AE,AN,AS,SU,SP 的确定
368.                 coefficient_matrix(i*NX + j, i*NX + j) = AP[i][j];
369.                 //求解系数矩阵//AP
370.                 Tv_vector(i*NX + j) = T[i][j];
371.                 B_vector(i*NX + j) = AP0[i][j] * T0[i][j] + SU[i][j];
372.                 if (i != 0)
373.                     coefficient_matrix(i*NX + j, (i - 1)*NX + j) = -
374.                     AS[i][j]; //AS
375.                 if (i != (NY - 1))
376.                     coefficient_matrix(i*NX + j, (i + 1)*NX + j) = -
377.                     AN[i][j]; //AN
378.                 if (j != 0)
379.                     coefficient_matrix(i*NX + j, i*NX + j - 1) = -
380.                     AW[i][j]; //AW
381.                 if (j != (NX - 1))
382.                     coefficient_matrix(i*NX + j, i*NX + j + 1) = -
383.                     AE[i][j]; //AE
384.             }
385.         }
386.     }
387.     //Tv_vector = coefficient_matrix.householderQr().solve(B_vector);
388.     Tv_vector = coefficient_matrix.colPivHouseholderQr().solve(B_vector);
389.     //QR 分解
390.     cout << "k =" << endl << k << endl;
391.     cout << " " << endl;
392.     //cout << "Tv_vector =" << endl << Tv_vector << endl;

```

```

386.         //cout <<" "<< endl;
387.         FILE *fp;
           //导出数据
388.         errno_t err_T;
389.         err_T = fopen_s(&fp, "T(t=30)(100,20)", "w");
390.         fprintf(
391.             fp,
392.             "TITLE = \"2D Plot temperature\"\nVARIABLES = X, Y, Temp\nZONE
           T=\"ZONE 1\", I=%d, J=%d, F=POINT, t=\"%2.2f\"\n",
393.             NX, NY, k);
394.
395.         for (i = 0; i <= (NY - 1); i++)
396.         {
397.             for (j = 0; j <= (NX - 1); j++)
398.             {
399.                 T0[i][j] = Tv_vector(i*NX + j);
           //把向量再转化为数组
400.                 fprintf(fp, "%4.8f %4.8f %4.2f  \n", 0.5 * (x[j] + x[j + 1
           ]), 0.5 * (y[i] + y[i + 1]), T0[i][j]);
401.             }
402.         }
403.         fclose(fp);
404.     }
405.     system("pause");
406.     return 0;
407. }

```

4. T-x 曲线绘制代码

```

1.  """author:gaoyi"""
2.  import matplotlib.pyplot as plt
3.
4.  fff = 16
5.
6.  X1=[]
7.  T1=[]
8.  file_1 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作业\半
           隐网格无关性验证\T-x\20x20.txt"
9.  with open(file_1, "r") as f1:
10.     for i in f1:
11.         X1.append(float("%.6f" % float(i.split()[0])))
12.         T1.append(float("%.6f" % float(i.split()[1])))
13.
14.  X2=[]

```

```

15. T2=[]
16. file_2 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作业\半
    隐网格无关性验证\T-x\50x20.txt"
17. with open(file_2, "r") as f2:
18.     for i in f2:
19.         X2.append(float("%.6f" % float(i.split()[0])))
20.         T2.append(float("%.6f" % float(i.split()[1])))
21.
22. X3=[]
23. T3=[]
24. file_3 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作业\半
    隐网格无关性验证\T-x\50x40.txt"
25. with open(file_3, "r") as f3:
26.     for i in f3:
27.         X3.append(float("%.6f" % float(i.split()[0])))
28.         T3.append(float("%.6f" % float(i.split()[1])))
29.
30. X4=[]
31. T4=[]
32. file_4 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作业\半
    隐网格无关性验证\T-x\100x20.txt"
33. with open(file_4, "r") as f4:
34.     for i in f4:
35.         X4.append(float("%.6f" % float(i.split()[0])))
36.         T4.append(float("%.6f" % float(i.split()[1])))
37.
38. plt.figure(figsize=(10, 8))
39.
40. plt.plot(X1, T1, label="20x20")
41. # plt.scatter(X1, T1, marker="x")
42.
43. plt.plot(X2, T2, label="50x20")
44. # plt.scatter(X2, T2, marker="x")
45.
46. plt.plot(X3, T3, label="50x40")
47. # plt.scatter(X3, T3, marker="x")
48.
49. plt.plot(X4, T4, label="100x20")
50. # plt.scatter(X4, T4, marker="x")
51.
52. plt.xlabel("X/m", fontsize=fff)
53. plt.ylabel("T/K", fontsize=fff)
54. plt.grid(linestyle="--")
55. plt.legend(loc=2, fontsize=fff)

```

```
56. plt.tight_layout()
57. plt.show()
```

5. T-t 曲线绘制代码

```
1. """author:gaoyi"""
2. import matplotlib.pyplot as plt
3. import numpy as np
4.
5. fff = 15
6.
7. t = np.arange(1, 31, 1)
8.
9. T1 = []
10. for i in t:
11.     file_1 = r"D:\python_project\college_assignment\半隐式网格无关性验证\半隐式
        -数据\20x20_m=0.95\data_t={}.plt".format(i)
12.     with open(file_1, "r") as f:
13.         k=1
14.         for i in f:
15.             if k == 202:
16.                 T1.append(float("%.6f" % float(i.split()[2])))
17.                 k = k+1
18.
19. T2 = []
20. for i in t:
21.     file_2 = r"D:\python_project\college_assignment\半隐式网格无关性验证\半隐式
        -数据\50x20_m=0.95\data_t={}.plt".format(i)
22.     with open(file_2, "r") as f_1:
23.         k1=1
24.         for i in f_1:
25.             if k1 == 502:
26.                 T2.append(float("%.6f" % float(i.split()[2])))
27.                 k1 = k1+1
28.
29.
30. T3 = []
31. for i in t:
32.     file_3 = r"D:\python_project\college_assignment\半隐式网格无关性验证\半隐式
        -数据\50x40_m=0.95\data_t={}.plt".format(i)
33.     with open(file_3, "r") as f2:
34.         k3=1
35.         for i in f2:
36.             if k3 == 502:
```

```

37.             T3.append(float("%.6f" % float(i.split()[2])))
38.             k3 = k3+1
39.
40.
41. T4 = []
42. for i in t:
43.     file_4 = r"D:\python_project\college_assignment\半隐式网格无关性验证\半隐式
        -数据\100x20_m=0.95\data_t={s}.plt".format(i)
44.     with open(file_4, "r") as f3:
45.         k4=1
46.         for i in f3:
47.             if k4 == 1002:
48.                 T4.append(float("%.6f" % float(i.split()[2])))
49.                 k4 = k4+1
50. plt.figure(figsize=(8, 6))
51.
52. plt.plot(t, T1, label="20x20")
53. plt.scatter(t, T1, marker="x")
54.
55. plt.plot(t, T2, label="50x20")
56. plt.scatter(t, T2, marker="x")
57.
58. plt.plot(t, T3, label="50x40")
59. plt.scatter(t, T3, marker="x")
60.
61. plt.plot(t, T4, label="100x20")
62. plt.scatter(t, T4, marker="x")
63.
64. plt.xlabel("t/s", fontsize=fff)
65. plt.ylabel("T/K", fontsize=fff)
66. plt.grid(linestyle="--")
67. plt.legend(loc=2, fontsize=fff)
68. plt.tight_layout()
69. plt.show()

```

6. UDF 代码

```

1. #include "udf.h"
2. DEFINE_PROFILE(user_h, thread, pos)
3. {
4.     double tem, tm, beta, Pr, v, Gr, Nu, h;
5.     face_t f;
6.     begin_f_loop(f, thread)
7.     {

```

```

8. tem=F_T(f,thread);
9. tm=(tem-253)/2;
10. beta=2/(tem+293);
11. Pr=4*(1E-14)*pow(tm,6)-3*(1E-11)*pow(tm,5)+6*(1E-9)*pow(tm,4)-7*(1E-
    7)*pow(tm,3)+4*(1E-5)*pow(tm,2)-0.0012*tm+0.7163;
12. v=9*(1E-11)*pow(tm,2)+9*(1E-8)*tm+1.3175*(1E-5);
13. Gr=9.8*beta*fabs(tem-293)*pow(0.03,3)/pow(v,2);
14. Nu=0.54*pow(Gr*Pr,0.25);
15. h=Nu*2.59/3;
16. F_PROFILE(f,thread,pos)=h;
17. }
18. end_f_loop(f,thread)
19. }

```

7. 其余 Python 代码

```

1. """author:gaoyi"""
2. import numpy as np
3. import matplotlib.pyplot as plt
4. """5 秒 T-x 曲线绘制"""
5. """网格数目为 20x20"""
6.
7. x1 = []
8. T1 = []
9. file_path_1 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作业\赵明潇数据_隐式\t=5s 隐式.txt"
10. with open(file=file_path_1, mode="r") as f1:
11.     for row in f1:
12.         x1.append(float(row.split()[0]))
13.         T1.append(float(row.split()[2]))
14.
15. x2=[0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,0.02125,
    0.02375,0.02625,0.02875,0.03125,
16.     0.03375,0.03625,0.03875,0.04125,0.04375,0.04625,0.04875]
17. T2=[293.0016,293.00015,293.00025,293.00055,293.0011,293.0023,293.00455,293.0
    0895,293.01695,293.03115,293.05535
18.     ,293.09405,293.1481,293.20575,293.27385,293.36985,293.5191,293.7679,294.
    6746,312.9441]
19.
20. x_ls_5s=[0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,0.0
    2125,0.02375,0.02625,0.02874999,0.03133333
21.     ,0.03399999,0.03666666,0.03933333,0.042,0.04466666,0.04733333,0.05]
22. T_ls_5s=[293,293,293,293,293,293,293,293.005,293.02,293.065,293.175,293.41,2
    93.795,294.125,294.395,294.64,294.92,295.325

```



```

23.     ,296.87,333.235]
24.
25. x_yy_5s = [0.0000,0.0025,0.0050,0.0075,0.0100,0.0125,0.0150,0.0175,0.0200,0.
    0225,0.0250,0.0275,0.0300,0.0325,0.0350,0.0375
26.     ,0.0400,0.0425,0.0450,0.0475,0.0500]
27. T_yy_5s = [293.0009,293.0013,293.0024,293.0043,293.0086,293.0169,293.0310,29
    3.0558,293.0987,293.1678,293.2809,293.4550
28.     ,293.6898,293.9368,294.1571,294.3589,294.5748,294.8633,295.5243,304.8741
    ,484.4058 ]
29.
30. x_xs_5s = [0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,0
    .02125,0.02375,0.02625,0.02875,0.03125,
31.     0.03375,0.03625,0.03875,0.04125,0.04375,0.04625,0.04875]
32. T_xs_5s = [293.0009374,293.0014957,293.0028748,293.0056875,293.0110869,293.0
    21089,293.039066,293.0704529,293.1236799
33.     ,293.2111965,293.3497042,293.555334,293.8187676,294.0529783,294.2554055,
    294.4562743,294.6950132,295.0291772
34.     ,296.0560942,314.8997893]
35.
36. x_M_5s = []
37. T_M_5s = []
38. file_path_2 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作
    业\M 商业软件\data_5s.txt"
39. with open(file=file_path_2, mode="r") as f2:
40.     for row in f2:
41.         x_M_5s.append(float(row.split()[0]))
42.         T_M_5s.append(float(row.split()[1]))
43.
44. """绘图"""
45. """解决中文显示问题"""
46. fff = 16     #字体大小
47.
48. plt.rcParams['font.sans-serif']=['SimHei']
49. plt.rcParams['axes.unicode_minus'] = False
50.
51. plt.figure(figsize=(10, 8))
52. plt.title("T-x 曲线(5s)",fontsize=fff)
53.
54. plt.plot(x1, T1, label="隐式")
55. plt.scatter(x1, T1,marker="x")
56.
57. plt.plot(x2,T2,label="半隐式")
58. plt.scatter(x2, T2,marker="x")
59.

```

```

60. plt.plot(x_ls_5s, T_ls_5s, label="隐式-老师代码")
61. plt.scatter(x_ls_5s, T_ls_5s, marker="x")
62.
63. plt.plot(x_yy_5s, T_yy_5s, label="Fluent")
64. plt.scatter(x_yy_5s, T_yy_5s, marker="x")
65.
66. plt.plot(x_xs_5s, T_xs_5s, label="显式")
67. plt.scatter(x_xs_5s, T_xs_5s, marker="x")
68.
69. plt.plot(x_M_5s, T_M_5s, label="Mechanical")
70. plt.scatter(x_M_5s, T_M_5s, marker="x")
71.
72. plt.xlabel("x/m", fontsize=fff)
73. plt.ylabel("T/K", fontsize=fff)
74. plt.grid(linestyle="--")
75. plt.tight_layout()
76. plt.legend(loc=2, fontsize=15)
77. #####
    #####
78. """30s 曲线绘制"""
79. x_y_30s = [0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,0.
    .02125,0.02375,0.02625,0.02875,0.03125
80.     ,0.03375,0.03625,0.03875,0.04125,0.04375,0.04625,0.04875]
81. T_y_30s = [297.6,297.785,298.145,298.705,299.455,300.42,301.61,303.05,304.75
    ,306.73,309.02,311.605,314.41,317.12
82.     ,319.66,322.025,324.225,326.35,329.425,350.535]
83.
84. x_by_30s = [0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,
    0.02125,0.02375,0.02625,0.02875,0.03125
85.     ,0.03375,0.03625,0.03875,0.04125,0.04375,0.04625,0.04875]
86. T_by_30s = [294.2728,293.84555,293.88235,293.99035,294.153,294.37415,294.659
    85,295.01775,295.45645,295.98585,296.61585
87.     ,297.35425,298.1974,299.1153,300.1042,301.17215,302.33455,303.6891,306.6
    6115,333.7786]
88.
89. x_ls_30s = [0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,
    0.02125,0.02375,0.02625,0.02874999,0.03133333
90.     ,0.03399999,0.03666666,0.03933333,0.042,0.04466666,0.04733333,0.05]
91. T_ls_30s = [293.42,293.44,293.49,293.57,293.685,293.86,294.14,294.62,295.42,
    296.71,298.605,301.19,304.355,307.495,310.405
92.     ,313.06,315.49,317.85,322.595,373.17]
93.
94. x_yy_30s=[0.0000,0.0025,0.0050,0.0075,0.0100,0.0125,0.0150,0.0175,0.0200,0.0
    225,0.0250,0.0275,0.0300,0.0325,0.0350,0.0375

```

```

95.     ,0.0400,0.0425,0.0450,0.0475,0.0500]
96. T_yy_30s=[295.9475,296.0095,296.2000,296.5242,296.9882,297.6006,298.3710,299
    .3138,300.4441,301.7757,303.3292,305.1130,307.0928
97.     ,309.1440,311.1511,313.0759,314.9232,316.7640,319.2902,333.5459,513.2305
    ]
98.
99. x_xs_30s = [0.00125,0.00375,0.00625,0.00875,0.01125,0.01375,0.01625,0.01875,
    0.02125,0.02375,0.02625,0.02875,0.03125
100.     ,0.03375,0.03625,0.03875,0.04125,0.04375,0.04625,0.04875]
101. T_xs_30s = [295.9266267,296.0544902,296.3128458,296.7069128,297.2444243,297
    .9355105,298.7925357,299.8298793,301.0635943
102.     ,302.510614,304.1858346,306.0897203,308.1641478,310.2201699,312.198582,
    314.0980879,315.9360367,317.8130201,321.1519307
103.     ,348.4038164]
104.
105. x_M_30s = []
106. T_M_30s = []
107. file_path_3 = r"D:\大学课程\大 4-上\发动机内流场数值分析基础（刘洋、胡春波）\大作业\M商业软件\data_30s.txt"
108. with open(file=file_path_3, mode="r") as f3:
109.     for row in f3:
110.         x_M_30s.append(float(row.split()[0]))
111.         T_M_30s.append(float(row.split()[1]))
112.
113. plt.figure(figsize=(10, 8))
114. plt.title("T-x 曲线(30s)", fontsize=fff)
115.
116. plt.plot(x_y_30s, T_y_30s, label="隐式")
117. plt.scatter(x_y_30s, T_y_30s, marker="x")
118.
119. plt.plot(x_by_30s, T_by_30s, label="半隐式")
120. plt.scatter(x_by_30s, T_by_30s, marker="x")
121.
122. plt.plot(x_ls_30s, T_ls_30s, label="隐式-老师代码")
123. plt.scatter(x_ls_30s, T_ls_30s, marker="x")
124.
125. plt.plot(x_yy_30s, T_yy_30s, label="fluent")
126. plt.scatter(x_yy_30s, T_yy_30s, marker="x")
127.
128. plt.plot(x_xs_30s, T_xs_30s, label="显式")
129. plt.scatter(x_xs_30s, T_xs_30s, marker="x")
130.
131. plt.plot(x_M_30s, T_M_30s, label="Mechanical")
132. plt.scatter(x_M_30s, T_M_30s, marker="x")

```

```
133.  
134. plt.xlabel("x/m", fontsize=fff)  
135. plt.ylabel("T/K", fontsize=fff)  
136. plt.grid(linestyle="--")  
137. plt.tight_layout()  
138. plt.legend(loc=2, fontsize=15)  
139. plt.show()
```