

《传热学》数值计算大作业

二维稳态导热问题的解析解与数值解

姓 名： 刘铭
学 号： 2017151613
班 级： 20171516
任课教师： 谭思超

哈尔滨工程大学
核科学与技术学院
2019 年 10 月 9 日

第一题

问题描述

例 2.1 图 2.3 所示是一个长矩形柱体的横截面,边长分别为 L_1 和 L_2 ,材料为常物性。由于柱体很长,且边界上的换热条件与坐标 z 几乎无关,因而可看作二维稳态导热,边界条件如图中所示,试求柱体内温度场的表达式。

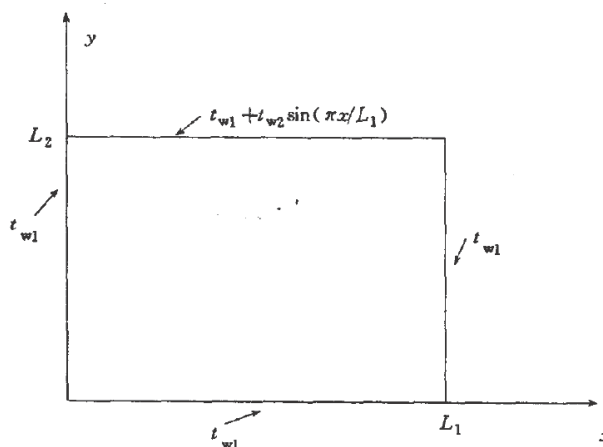


图 2.3 矩形柱体内稳态导热(例 2.1)

设图中 $t_{w1} = 25.0^\circ\text{C}$, $t_{w2} = 5.0^\circ\text{C}$, $L1=100.0$, $L2=80.0$

一、建立控制方程及定解条件

对上述问题的微分方程及其边界条件为: $\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$

$$x=0, T=T_1=tw1$$

$$x=L1, T=T_1=tw1$$

$$y=0, T=T_1=tw1$$

$$y=L2, T=T_2=tw1+tw2*\sin(\pi*x/L1)$$

该问题的解析解:

$$\frac{T-T_1}{T_2-T_1} = \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{1-(-1)^n}{n} \sin\left(\frac{n\pi}{L} \cdot x\right) \frac{\text{sh}\left(\frac{n\pi}{L} \cdot y\right)}{\text{sh}\left(\frac{n\pi}{L} \cdot W\right)}$$

通过附件一中 C++代码编译运行,将解出的数据通过绘图软件绘制。

二、数值离散

区域离散 x 方向总节点数为 N, y 方向总节点数为 M, 区域内任一节点用 i,j 表示。

三、建立代数方程

对于图中所有的内部节点方程可写为: $\left(\frac{\partial^2 t}{\partial x^2}\right)_{i,j} + \left(\frac{\partial^2 t}{\partial y^2}\right)_{i,j} = 0$

用 i,j 节点的二阶中心差分代替上式中的二阶导数, 得:

$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = 0$$

上式整理成迭代形式:

$$T_{i,j} = \frac{\Delta y^2}{2(\Delta x^2 + \Delta y^2)} (T_{i+1,j} + T_{i-1,j}) + \frac{\Delta x^2}{2(\Delta x^2 + \Delta y^2)} (T_{i,j+1} + T_{i,j-1})$$

(i=2,3,...,N-1), (j=2,3,...,M-1)

四、设立迭代初场

补充四个边界上的第一类边界条件得: $T_{1,j} = T_1 \quad (j=1,2,3,...,M)$

$T_{N,j} = T_1 \quad (j=1,2,3,...,M)$

$T_{i,j} = T_1 \quad (i=1,2,3,...,N)$

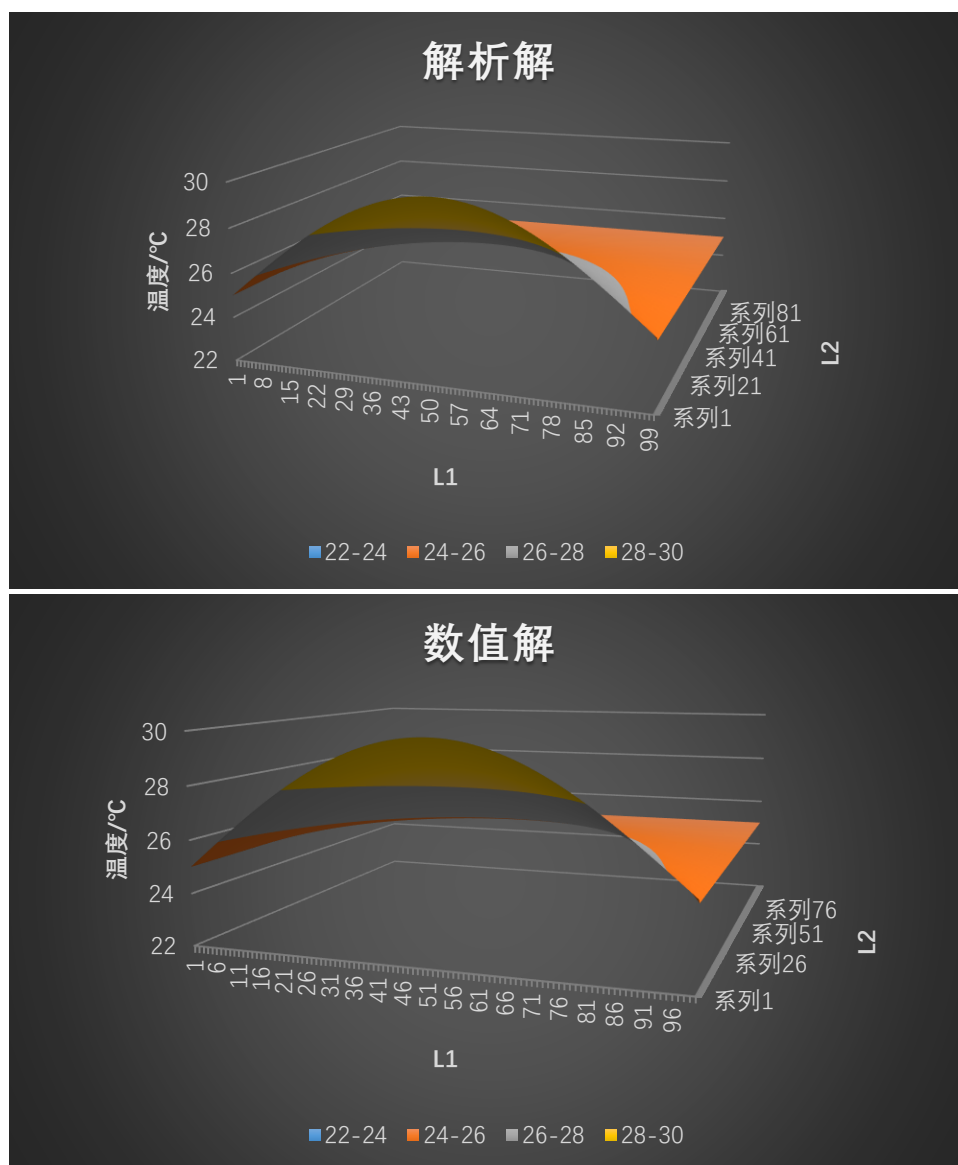
$T_{i,M} = T_2 \quad (i=1,2,3,...,N)$

五、迭代求解

通过由附件二中 C++编写的迭代程序, 计算 9598 次迭代, 相对偏差小于代码中所设定的误差最大值 10^{-6} . 得出的数据通过绘图软件绘制。

六、解的分析

将解析解的图形与数值解的图形进行对比, 其结果极其相似。



第二题

问题描述

例 2.2 例 2-1 中, $y=L_2$ 处的边界条件变为 $t=t_{w2}$, 其他条件不变, 试求温度场的表达式。

一、解析解

对上述问题的微分方程及其边界条件为: $\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$

$$\begin{aligned}
 x=0, \quad T=T_1=t_{w1} \\
 x=L_1, \quad T=T_1=t_{w1} \\
 y=0, \quad T=T_1=t_{w1} \\
 y=L_2, \quad T=T_2=t_{w2}
 \end{aligned}$$

该问题的解析解:

$$\frac{t(x,y)-t_{w1}}{t_{w2}-t_{w1}} = \sum_{m=1}^{\infty} \frac{4}{\varepsilon} \sin\left(\varepsilon \frac{x}{L_1}\right) \frac{\sinh(\varepsilon y/L_1)}{\sinh(\varepsilon L_2/L_1)}$$

$$\text{其中: } \varepsilon = (2m-1) * \pi \quad m = 1, 2, 3, \dots$$

通过附件三中 C++ 代码编译运行, 将解出的数据通过绘图软件绘制。

二、数值解

$$\text{其边界条件为: } \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} = 0$$

$$\begin{aligned}
 x=0, \quad T=T_1=t_{w1} \\
 x=L_1, \quad T=T_1=t_{w1} \\
 y=0, \quad T=T_1=t_{w1} \\
 y=L_2, \quad T=T_2=t_{w2}
 \end{aligned}$$

$$\text{对于所有的内部节点方程可写为: } \left(\frac{\partial^2 t}{\partial x^2} \right)_{i,j} + \left(\frac{\partial^2 t}{\partial y^2} \right)_{i,j} = 0$$

用 i, j 节点的二阶中心差分代替上式中的二阶导数, 得:

$$\frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} + \frac{T_{i,j+1} - 2T_{i,j} + T_{i,j-1}}{\Delta y^2} = 0$$

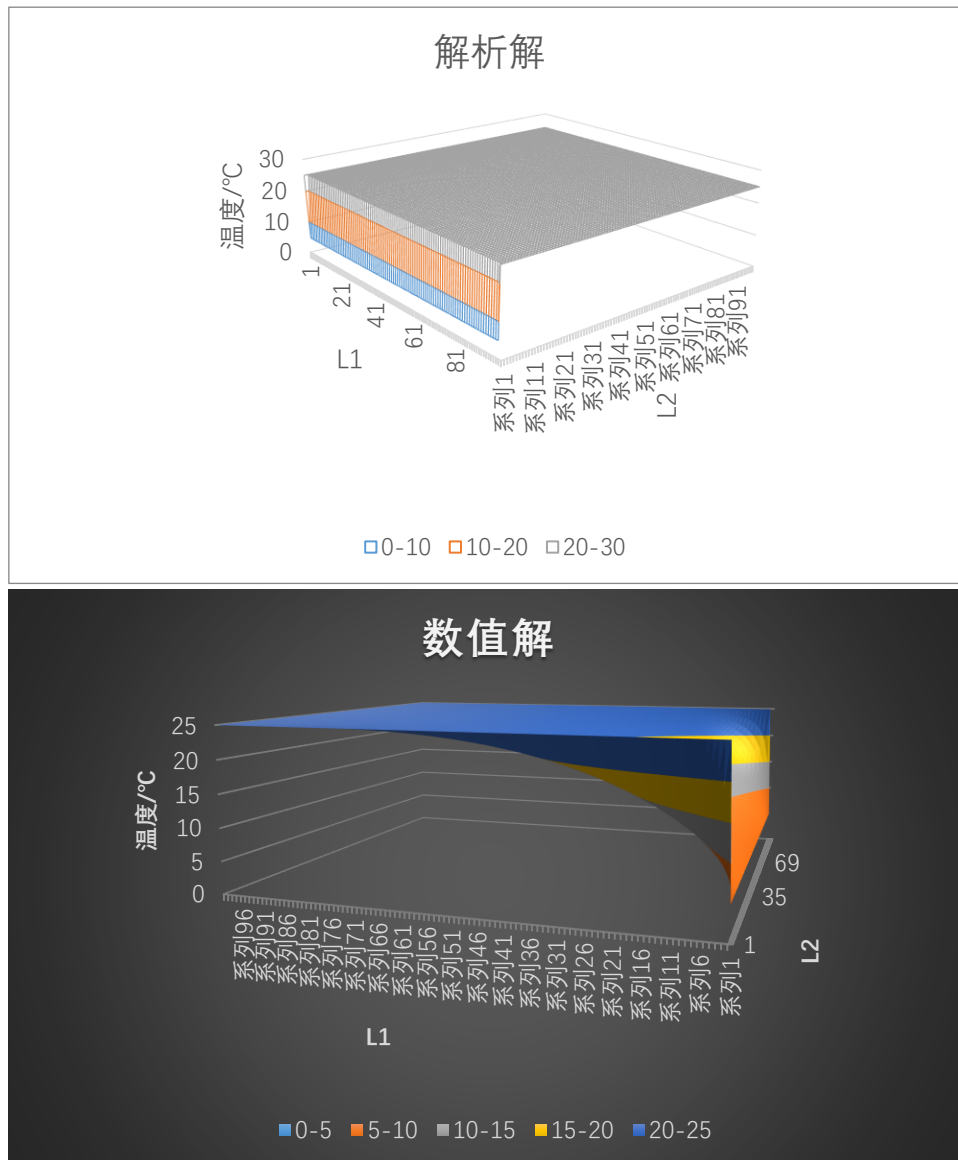
上式整理成迭代形式:

$$T_{i,j} = \frac{\Delta y^2}{2(\Delta x^2 + \Delta y^2)} (T_{i+1,j} + T_{i-1,j}) + \frac{\Delta x^2}{2(\Delta x^2 + \Delta y^2)} (T_{i,j+1} + T_{i,j-1})$$

$$(i=2, 3, \dots, N-1), (j=2, 3, \dots, M-1)$$

通过附件四中 C++ 代码编译运行, 将解出的数据通过绘图软件绘制。

三、解的分析



解析解的值可能是由于数值过小，计算的结果作图效果不好；数值解的图形比较形象。

第三题

问题描述

例 2.3 图 2.4 所示是一个等截面直肋, 肋厚 $b=2\delta$, 肋基温度为 t_0 , 肋表面和肋端与环境间的对流换热系数为 α , 流体温度为 t_f , 试确定肋内温度分布及散热量。

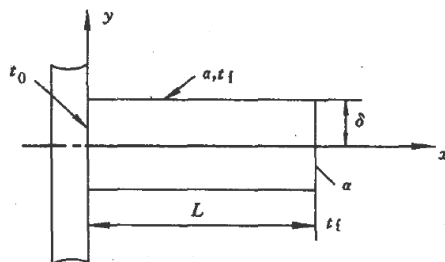


图 2.4 直肋导热

设对流换热系数 $h=10\text{W}/(\text{K}\cdot\text{m}^2)$, 墙壁导热系数 $a=1000\text{W}/(\text{K}\cdot\text{m})$, 壁面温度恒为 50°C , 环境温度为 10°C 。

一、解析解

通过计算, 肋内温度的解析解为:

$$\frac{t(x, y) - t_f}{t_0 - t_f} \approx \frac{\cosh\left(\frac{\sqrt{Bi}(L-x)}{\delta}\right) + \sqrt{Bi}\sinh\left(\frac{\sqrt{Bi}(L-x)}{\delta}\right)}{\cosh\left(\frac{\sqrt{Bi}L}{\delta}\right) + \sqrt{Bi}\sinh\left(\frac{\sqrt{Bi}L}{\delta}\right)}$$

通过附件五中 C++ 代码编译运行, 将解出的数据通过绘图软件绘制。

计算, 得: $Q=-5427.26\text{W/m}$

二、数值解

2.1 区域离散化

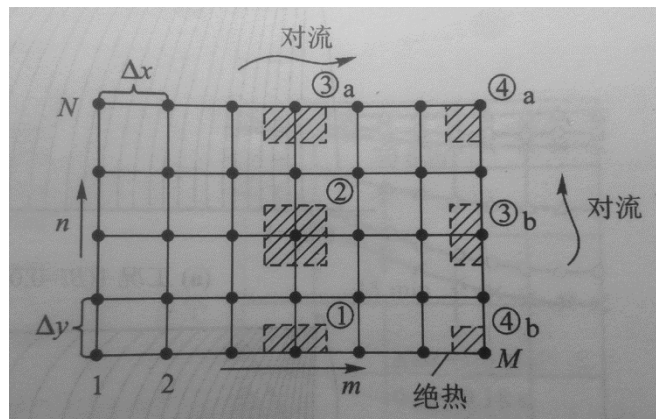
由于对称性, 取一半区域研究即可, 其网格划分示意图如下:

如图所示, 共划分 $M \times N$ 个节点。为后续计算方便, 我们将控制步长相等, 即:

$$\Delta x = \Delta y$$

实现方法是, 设定肋片厚度 ($2 \times \delta$) 及肋片高度 (H) 满足如下条件:

$$\frac{\delta}{N-1} = \frac{H}{M-1}$$



2.2 建立节点离散方程

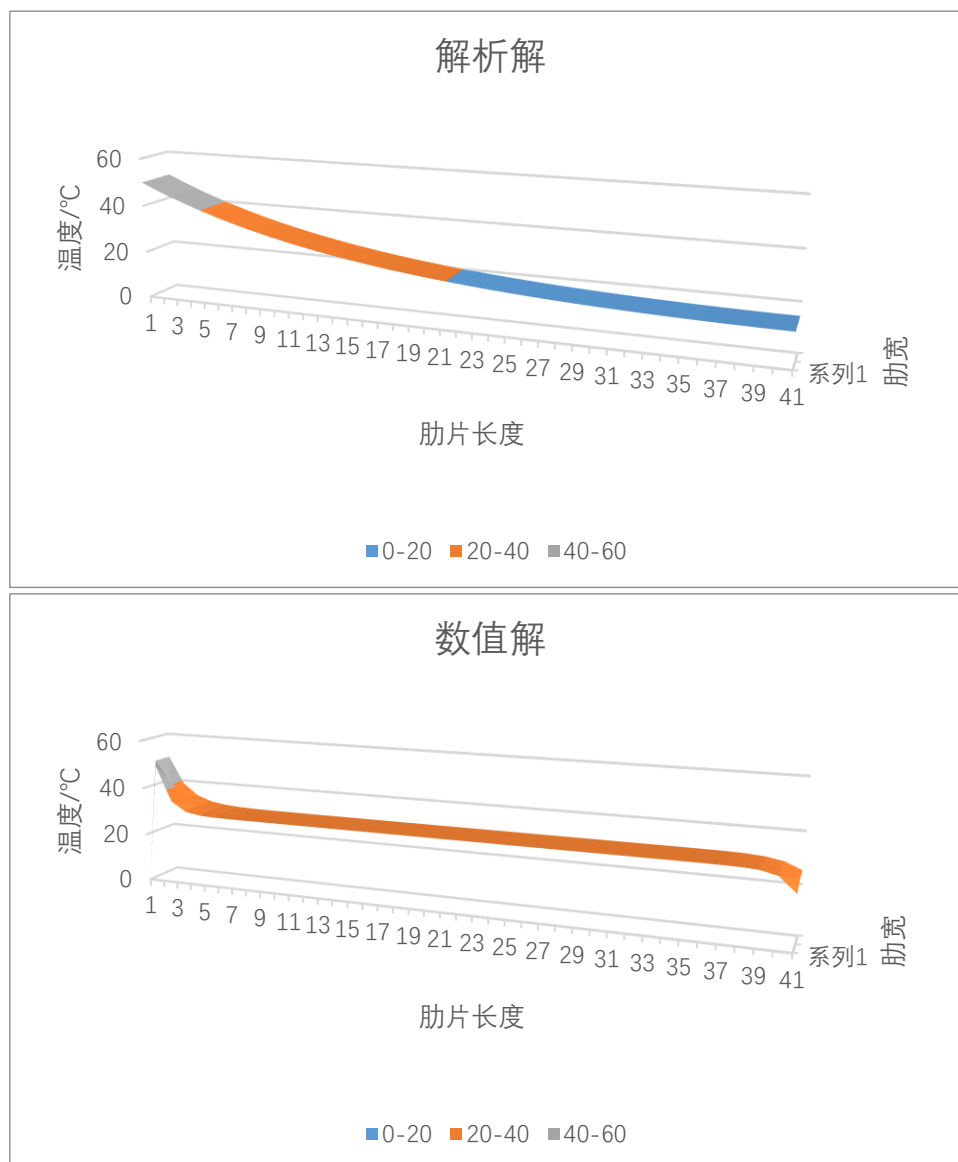
在稳态条件下，我们采用形式简捷的热平衡法对各节点建立离散方程。网格离散化后共形成 5 类节点，即①, ②, ③a, ③b, ④a, ④b，对应各节点的离散方程为：

①	$m = 2, 3, \dots, (M-1)$ $n = 1$	$t_{m,1} = \frac{1}{4} * (t_{m-1,1} + t_{m+1,1} + 2 * t_{m+1,1})$
②	$m = 2, 3, \dots, (M-1)$ $n = 2, 3, \dots, (N-1)$	$t_{m,n} = \frac{1}{4} * (t_{m+1,n} + t_{m-1,n} + t_{m,n+1} + t_{m,n-1})$
③ a	$m = 2, 3, \dots, (M-1)$ $n = N$	$t_{m,N} = \frac{1}{4 + 2 * Bi} * (t_{m-1,N} + t_{m+1,N} + 2 * t_{m,N-1} + 2 * Bi * t_f)$
③ b	$m = M$ $n = 2, 3, \dots, (N-1)$	$t_{M,n} = \frac{1}{4 + 2 * Bi} * (t_{M-1,n} + t_{M,n+1} + 2 * t_{M,n-1} + 2 * Bi * t_f)$
④ a	$m = M$ $n = N$	$t_{M,N} = \frac{1}{2 + 2 * Bi} * (t_{M,N-1} + t_{M-1,N} + 2 * Bi * t_f)$
④ b	$m = M$ $n = 1$	$t_{M,1} = \frac{1}{2 + Bi} * (t_{M-1,1} + t_{M,2} + Bi * t_f)$

通过附件六中 C++代码编译运行，将解出的数据通过绘图软件绘制。

计算，得： **$Q = -5429.07 \text{ W/m}$**

三、解的分析



数值解的误差较大，得出的图形与解析解计算的图形差异较大！

说明

具体数据以及代码已上传至 [Github](<https://github.com/iuming/Heat-Transfer-Learning---Numerical-Calculation>)

附件一、问题一解析解代码 (C++)

```
1. #include<bits/stdc++.h>
2. #define pi 3.1415926535
3. #define N 100
4. #define M 100
5. using namespace std;
6. int main()
7. {
8.     ofstream fout;
9.     double L1=100,L2=80,x=0,y=0;
10.    double tw1=25,tw2=5;
11.    float len1=L1/(N-1),len2=L2/(M-1);
12.    double t[M][N],T[M][N];
13.    memset(t,0,sizeof(t));
14.    memset(T,0,sizeof(T));
15.    for(int i=0;i<N;i++)
16.    {
17.        t[i][0]=tw1;
18.        t[i][N-1]=tw1+tw2*sin(pi*(i-1)/(N-1));
19.    }
20.    for(int j=0;j<M;j++)
21.    {
22.        t[0][j]=tw1;
23.        t[M-1][j]=tw1;
24.    }
25.    for (int i=1;i<N-1;i++)
26.        for (int j=1;j<M-1;j++)
27.            t[i][j]=tw1+tw2*sin(pi*(j-1)/(N-1))*sinh(pi*(i-1)/(N-1))/sinh(pi*L2/L1);
28.    for(int j=M-1;j>=0;j--)
29.    {
30.        for(int i=0;i<N;i++)
31.            fout<<t[i][j]<<',';
32.        fout<<endl;
33.    }
34.    fout.close();
35.    return 0;
36. }
```

附件二、问题一数值解代码 (C++)

```
1. #include<bits/stdc++.h>
```

```
2. #define N 100
3. #define M 100
4. #define pi 3.1415926535
5. using namespace std;
6. int main()
7. {
8.     ofstream fout;
9.     fout.open("Problem one data(Numerical Solution).csv");
10.    int i,j,l,num=1;
11.    float cha,x,y,x2,y2;
12.    float t[N][M],a[N][M];
13.    memset(t,0,sizeof(t));
14.    for(j=0;j<M;j++)
15.    {
16.        t[0][j]=25;
17.        t[M-1][j]=25;
18.    }
19.    for(i=0;i<N;i++)
20.    {
21.        t[i][0]=25;
22.        t[i][N-1]=25+5*sin(pi*i/N);
23.    }
24.    x=1.0/(N-1);
25.    y=1.0/(M-1);
26.    cha=1;
27.    while(cha>1e-6)
28.    {
29.        for(i=0;i<N;i++)
30.        {
31.            for(j=0;j<M;j++)
32.            {
33.                a[i][j]=t[i][j];
34.            }
35.            for(i=1;i<N-1;i++)
36.            {
37.                for(j=1;j<M-1;j++)
38.                {
39.                    x2=pow(x,2);y2=pow(y,2);
40.                    t[i][j]=0.5*y2*(t[i+1][j]+t[i-1][j])/(x2+y2)+0.5*x2*(t[i][j+1]+t[i][j-1])/(x2+y2);
41.                }
42.            }
43.            cha=0;
44.            for(i=0;i<N;i++)
45.            {
46.                for(j=0;j<M;j++)
47.                {
48.                    cha=cha+abs(a[i][j]-t[i][j]);
49.                }
50.            }
51.            cha=cha/(N*M);
52.            cout<<num++<< ' ' <<cha<<endl;
53.        }
54.    }
```

```
45.     l=0;
46.     for(j=M-1;j>=0;j--)
47.     {
48.         for(i=0;i<N;i++)
49.             fout<<t[i][j]<<',';
50.         fout<<endl;
51.     }
52.     fout.close();
53.     return 0;
54. }
55.
```

附件三、问题二解析解代码 (C++)

```
1. #include<bits/stdc++.h>
2. #define pi 3.1415926535
3. #define N 100
4. #define M 100
5. using namespace std;
6. int main()
7. {
8.     ofstream fout;
9.     fout.open("Problem two date(Analytical Solution).csv");
10.    double L1=3.0,L2=2.0,f=0.0,s=0.0;
11.    double tw1=25,tw2=5;
12.    float len1=L1/(N-1),len2=L2/(M-1);
13.    double t[M][N],T[M][N];
14.    memset(t,0,sizeof(t));
15.    memset(T,0,sizeof(T));
16.    for(int i=0;i<N;i++)
17.    {
18.        t[i][0]=tw1;
19.        t[i][N-1]=tw2;
20.    }
21.    for(int j=0;j<M;j++)
22.    {
23.        t[0][j]=tw1;
24.        t[M-1][j]=tw1;
25.    }
26.    for (int i=1;i<N-1;i++)
27.        for (int j=1;j<M-1;j++)
28.        {
29.            int sum=0;
30.            for (int k=1;k<=99;k++)
```

```
31.         {
32.             s=(2*k-1)*pi;
33.             f=4/s*sin(s*(i-1)*len1/L1)*sinh(s*(j-
34.             1)*len2/L1)/sinh(s*L2/L1);
35.             sum+=f;
36.         }
37.         t[j][i]=tw1+(tw1-tw2)*sum;
38.     }
39.     {
40.         for(int i=0;i<N;i++)
41.             fout<<t[i][j]<<',';
42.         fout<<endl;
43.     }
44.     fout.close();
45.     return 0;
46. }
```

附件四、问题二数值解代码 (C++)

```
1. #include<bits/stdc++.h>
2. #define N 100
3. #define M 100
4. #define pi 3.1415926535
5. using namespace std;
6. int main()
7. {
8.     ofstream fout;
9.     fout.open("Problem two data(Numerical Solution).csv");
10.    int i,j,l,num=1;
11.    float cha,x,y,x2,y2;
12.    float t[N][M],a[N][M];
13.    for(i=0;i<N;i++)
14.        for(j=0;j<M;j++)
15.            t[i][j]=0;
16.    for(j=0;j<M;j++)
17.    {
18.        t[0][j]=25;
19.        t[M-1][j]=25;
20.    }
21.    for(i=0;i<N;i++)
22.    {
23.        t[i][0]=25;
24.        t[i][N-1]=5;
```

```
25.     }
26.     x=1.0/(N-1);
27.     y=1.0/(M-1);
28.     cha=1;
29.     while(cha>1e-6)
30.     {
31.         for(i=0;i<N;i++)
32.             for(j=0;j<M;j++)
33.                 a[i][j]=t[i][j];
34.         for(i=1;i<N-1;i++)
35.             for(j=1;j<M-1;j++)
36.             {
37.                 x2=pow(x,2);y2=pow(y,2);
38.                 t[i][j]=0.5*y2*(t[i+1][j]+t[i-1][j])/(x2+y2)+0.5*x2*(t[i][j+1]+t[i][j-1])/(x2+y2);
39.             }
40.         cha=0;
41.         for(i=0;i<N;i++)
42.             for(j=0;j<M;j++)
43.                 cha=cha+abs(a[i][j]-t[i][j]);
44.         cha=cha/(N*M);
45.         cout<<num++<< ' ' << cha<< endl;
46.     }
47.     for(j=M-1;j>=0;j--)
48.     {
49.         for(i=0;i<N;i++)
50.             fout<<t[i][j]<<',';
51.         fout<<endl;
52.     }
53.     fout.close();
54.     return 0;
55. }
```

附件五、问题三解析解代码 (C++)

```
1. #include<bits/stdc++.h>
2. #define M 42
3. #define N 4
4. #define pi 3.1415926535
5. using namespace std;
6. int main()
7. {
8.     ofstream fout;
9.     fout.open("Problem three date(Analytical Solution).csv");
```

```

10. double L1=20.0,L2=1.0,h=10,a=1000,m=0.5,n=0.5,tf=10,tw=50;
11. double Bi=h*L2/2/a;
12. double t[M][N],T[M][N];
13. memset(t,0,sizeof(t));
14. memset(T,0,sizeof(T));
15. for(int i=2;i<=M;i++)
16.     for(int j=1;j<=3;j++)
17.         t[i][j]=30;
18. for(int j=1;j<=N;j++)
19.     t[1][j]=50;
20. for(int i=2;i<=M;i++)
21.     for(int j=1;j<=N;j++)
22.         t[i][j]=tf+(tw-tf)*((cosh(sqrt(Bi)*(L1-(i-1)*0.5)/L2*2)+sqrt(Bi)*sinh(sqrt(Bi)*(L1-(i-1)*0.5)/L2*2))/(cosh(sqrt(Bi)*L1/L2*2)+sqrt(Bi)*sinh(sqrt(Bi)*L1/L2*2)));
23. double Q=(t[2][1]-t[1][1])/m*n/2*a+(t[2][3]-t[1][3])/m*n/2*a+(t[2][2]-t[1][2])/m*n*a;
24. cout<<Q<<endl;
25. for(int j=1;j<=N-1;j++)
26. {
27.     for(int i=1;i<=M-1;i++)
28.         fout<<t[i][j]<<',';
29.     fout<<endl;
30. }
31. fout.close();
32. return 0;
33. }

```

附件六、问题三数值解代码 (C++)

```

1. #include<bits/stdc++.h>
2. #define M 42
3. #define N 4
4. #define pi 3.1415926535
5. using namespace std;
6. int main()
7. {
8.     ofstream fout;
9.     fout.open("Problem three date(Numerical Solution).csv");
10.    double L1=20.0,L2=1.0,h=10,a=1000,m=0.5,n=0.5,tf=10,tw=50;
11.    double esp=1.0,Bi=h*L2/2/a;
12.    double t[M][N],T[M][N];
13.    memset(t,0,sizeof(t));
14.    memset(T,0,sizeof(T));

```

```

15.     for(int i=2;i<=M;i++)
16.         for(int j=1;j<=3;j++)
17.             t[i][j]=30;
18.     for(int j=1;j<=N;j++)
19.         t[1][j]=50;
20.     while(esp>1e-6)
21.     {
22.         for(int i=0;i<N;i++)
23.             for(int j=0;j<M;j++)
24.                 T[i][j]=t[i][j];
25.         for(int i=2;i<=M-1;i++)
26.         {
27.             t[i][N]=(n*a/(m*2)*(t[i+1][N]+t[i-1][N])+m*a/n*t[i][N-
28.                 1]+h*tf*m)/(n*a/m+m*a/n+h*m);
29.             t[i][1]=(n*a/(m*2)*(t[i+1][1]+t[i-
30.                 1][1])+m*a/n*t[i][2]+h*tf*m)/(n*a/m+m*a/n+h*m);
31.             t[i][2]=0.25*(t[i-1][2]+t[i+1][2]+t[i][1]+t[i][3]);
32.             t[M][2]=(m/2*a*t[M][3]/n+m/2*a*t[M][1]/n+n*a*t[M-
33.                 1][2]/m+n*h*tf)/(m*a/n+n*a/m+n*h);
34.             t[M][N]=(n/2*a*t[M-1][N]/m+m/2*a*t[M][N-
35.                 1]/n+m/2*h*tf+n/2*h*tf)/(n/2*a/m+m/2*a/n+m/2*h+n/2*h);
36.             t[M][1]=(n/2*a/m*t[M-
37.                 1][1]+m/2*a*t[M][2]/n+m/2*h*tf+n/2*h*tf)/(n/2*a/m+m/2*a/n+m/2*h+n/2*h);
38.         }
39.         esp=0;
40.         for(int i=0;i<N;i++)
41.             for(int j=0;j<M;j++)
42.                 esp+=abs(T[i][j]-t[i][j]);
43.         esp/=(N*M);
44.     }
45.
46.     double Q=(t[2][1]-t[1][1])/m*n/2*a+(t[2][3]-t[1][3])/m*n/2*a+(t[2][2]-
47.         t[1][2])/m*n*a;
48.     cout<<Q<<endl;
49.     for(int j=0;j<=N-2;j++)
50.     {
51.         for(int i=1;i<=M-1;i++)
52.             fout<<t[i][j]<<',';
53.         fout<<endl;
54.     }
55.     fout.close();
56.     return 0;
57. }

```