長安大學

《科学计算设计实践（二）》课程设计实践训练

偏微分方程数值算法及程序设计

学　　院：　理学院

专　　业：　信息与计算科学

姓　　名：　刘通源

学　　号：　2021905058

指导教师：　高普阳

完成时间：二〇二四年六月

目 录

Problem1

将$u''=-u(0\leq t\leq 1),\ u(0)=0,\ u'(0)=1$ 化成一阶方程组, 并用 Euler 法和改进的

Euler 法求解，步长 h = 0.1, 0.05

(1) 画出 Euler 法,Euler 改进法以及精确解的数值结果图形;

(2) 列表对比 Euler 法,Euler 改进法以及精确解, 判断哪种方法的精度更高。

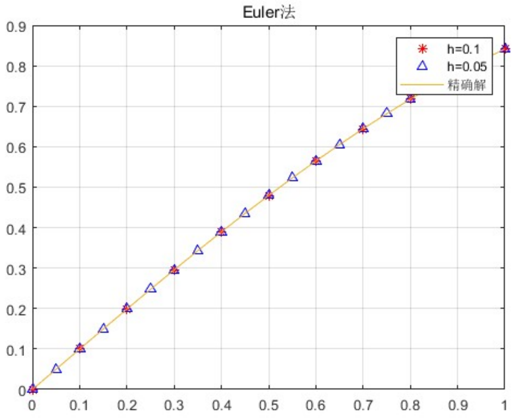
数值方法

欧拉法：$u_{n+1}=u_n+hf(t_n,u_n),\quad n=0,1,\cdots,N-1,$
改进欧拉法：$u_{n+1}^{[k+1]}=u_n+\frac{h}{2}\left[f(t_n,u_n)+f(t_{n+1},u_{n+1}^{[k]})\right],\quad k=0,1,\cdots.$
一阶方程组：

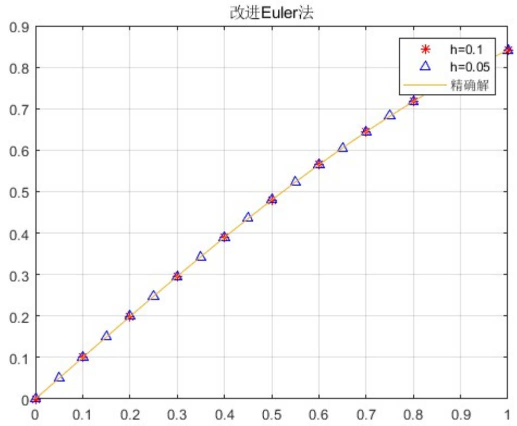
$$\begin{cases} v=u' \\ \frac{dv}{dt}=-u \\ \frac{du}{dt}=v \end{cases}$$

初值：v(0) = 1, u(0) = 1

解答



Euler 法



改进 Euler 法

h=0.1 时数值解比较

位置	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Euler	0	0.1000	0.2000	0.2990	0.3960	0.4900	0.5801	0.6652	0.7446	0.8173	0.8825
Improved Euler	0	0.0998	0.1985	0.2953	0.3891	0.4791	0.5642	0.6438	0.7169	0.7829	0.8410

h=0.05 时数值解比较

位置	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4	0.45	0.5
Euler	0	0.0500	0.1000	0.1499	0.1995	0.2488	0.2975	0.3456	0.3930	0.4395	0.4851
Improve	0	0.0500	0.0998	0.1494	0.1986	0.2474	0.2955	0.3428	0.3893	0.4349	0.4793
精确解	0	0.0500	0.0998	0.1494	0.1987	0.2474	0.2955	0.3429	0.3894	0.4350	0.4794
位置	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	1	
Euler	0.5295	0.5727	0.6147	0.6551	0.6941	0.7314	0.7669	0.8007	0.8325	0.8623	
Improve	0.5226	0.5645	0.6051	0.6441	0.6815	0.7172	0.7512	0.7832	0.8133	0.8414	
精确解	0.5227	0.5646	0.6052	0.6442	0.6816	0.7174	0.7513	0.7833	0.8134	0.8415	

- **结论：** 由表对比可知步长相同时，改进 Euler 法相较于 Euler 法更精准。

代码

Listing 1 problem1.m

```
clear;clc;
% 初始化条件和参数
h1 = 0.1;
h2 = 0.05;

t1 = 0:h1:1;
```

```
t2 = 0:h2:1;

N1 = 1/h1;
N2 = 1/h2;

u1 = zeros(1,N1+1);
v1 = zeros(1,N1+1);
u2 = zeros(1,N2+1);
v2 = zeros(1,N2+1);
u_exact = zeros(1,N2+1);

%% Euler
disp('Euler:');
% h1
u1(1) = 0;
v1(1) = 1;
for n=1:N1
    u1(n+1) = u1(n) + h1*v1(n);
    v1(n+1) = v1(n) - h1*u1(n);
end
result1 = [t1',u1'];
disp('h=0.1的结果')
disp(result1);

% h2
u2(1) = 0;
v2(1) = 1;
for n=1:N2
    u2(n+1) = u2(n) + h2*v2(n);
    v2(n+1) = v2(n) - h2*u2(n);
end
disp('h=0.05的结果')
result2 = [t2',u2'];
disp(result2);

%% improved euler
disp('improved Euler:');
% h1
for n=1:N1
    u1(n+1) = ((1-h1^2/4)*u1(n)+h1*v1(n))/(1+h1^2/4);
    v1(n+1) = ((1-h1^2/4)*v1(n)-h1*u1(n))/(1+h1^2/4);
end
result1 = [t1’,u1’];
disp(’h=0.1的结果’)
disp(result1);
%h2
for n=1:N2
    u2(n+1) = ((1-h2^2/4)*u2(n)+h2*v2(n))/(1+h2^2/4);
    v2(n+1) = ((1-h2^2/4)*v2(n)-h2*u2(n))/(1+h2^2/4);
end
disp(’h=0.05的结果’)
result2 = [t2’,u2’];
disp(result2);

%精确解
for n=1:N2+1
    u_exact(n) = sin(t2(n));
end
disp(’精确解的结果’)
result3 = [t2’,u_exact’];
disp(result3);

%% 绘图
% Euler
figure(1);
plot(t1,u1,"r*");
hold on;
plot(t2,u2,"b^");
plot(t2,u_exact);
hold off;
grid on;
title(’Euler法’);
legend(’h=0.1’,’h=0.05’,’精确解’);
% Improved Euler
figure(2);
plot(t1,u1,"r*");
hold on;
plot(t2,u2,"b^");
plot(t2,u_exact);
hold off;
```

```
        grid on;
        title('改进Euler法');
        legend('h=0.1','h=0.05','精确解');
```

Problem2

用四级四阶 Runge-Kutta 法计算初值问题:

$$\begin{aligned} u' &= 4tu^{\frac{1}{2}}, \quad 0 \le t \le 2, \\ u(0) &= 1. \end{aligned}$$

取 h=0.2,0.4,0.5.

精确解为$u(t) = (1+t^2)^2$.

数值方法

$$\begin{aligned}
&u_{n+1} = u_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4), \\
&k_1 = f(t_n, u_n), \\
&k_2 = f\left(t_n + \frac{1}{2}h, u_n + \frac{1}{2}hk_1\right), \\
&k_3 = f\left(t_n + \frac{1}{2}h, u_n + \frac{1}{2}hk_2\right), \\
&k_4 = f(t_n + h, u_n + hk_3).
\end{aligned}$$

解答

精准解

0	0.2000	0.4000	0.6000	0.8000	1.0000	1.2000	1.4000	1.6000	1.8000	2.0000
1.0000	1.0816	1.3456	1.8496	2.6896	4.0000	5.9536	8.7616	12.6736	17.9776	25.0000

h=0.2 时数值解

0	0.2000	0.4000	0.6000	0.8000	1.0000	1.2000	1.4000	1.6000	1.8000	2.0000
1.0000	1.0816	1.3456	1.8495	2.6895	3.9998	5.9532	8.7611	12.6728	17.9765	24.9986

h=0.4 时数值解

0	0.4000	0.8000	1.2000	1.6000	2.0000
1.0000	1.3452	2.6880	5.9488	12.6632	24.9813

h=0.5 时数值解

0	0.5000	1.0000	1.5000	2.0000
1.0000	1.5611	3.9932	10.5427	24.9580

代码

Listing 2 problem2.m

```
  clear;clc;

% 初始化条件和参数
h1 = 0.2;
h2 = 0.4;
h3 = 0.5;

t1 = 0:h1:2;
t2 = 0:h2:2;
t3 = 0:h3:2;

[N1, N2, N3] = deal(2/h1, 2/h2, 2/h3);
[u1, u2, u3, u_exact] = deal(zeros(1, N1+1), zeros(1, N2+1),
   zeros(1, N3+1), zeros(1, N1+1));

[u1(1), u2(1), u3(1), u_exact(1)] = deal(1);

% 定义函数 f(t, u)
f = @(t, u) 4 * t * sqrt(u);

% h1
disp('h=0.2的结果')
for n = 1:N1
    k1 = f(t1(n), u1(n));
    k2 = f(t1(n) + 0.5 * h1, u1(n) + 0.5 * h1 * k1);
    k3 = f(t1(n) + 0.5 * h1, u1(n) + 0.5 * h1 * k2);
    k4 = f(t1(n) + h1, u1(n) + h1 * k3);
    u1(n+1) = u1(n) + h1 / 6 * (k1 + 2 * k2 + 2 * k3 + k4);
end
disp([t1', u1']);

% h2
```

```
disp('h=0.4的结果')
for n = 1:N2
    k1 = f(t2(n), u2(n));
    k2 = f(t2(n) + 0.5 * h2, u2(n) + 0.5 * h2 * k1);
    k3 = f(t2(n) + 0.5 * h2, u2(n) + 0.5 * h2 * k2);
    k4 = f(t2(n) + h2, u2(n) + h2 * k3);
    u2(n+1) = u2(n) + h2 / 6 * (k1 + 2 * k2 + 2 * k3 + k4);
end
disp([t2', u2']);

% h3
disp('h=0.5的结果')
for n = 1:N3
    k1 = f(t3(n), u3(n));
    k2 = f(t3(n) + 0.5 * h3, u3(n) + 0.5 * h3 * k1);
    k3 = f(t3(n) + 0.5 * h3, u3(n) + 0.5 * h3 * k2);
    k4 = f(t3(n) + h3, u3(n) + h3 * k3);
    u3(n+1) = u3(n) + h3 / 6 * (k1 + 2 * k2 + 2 * k3 + k4);
end
disp([t3', u3']);

% 精确解
disp('精确解的结果')
for n = 1:N1+1
    u_exact(n) = (1 + t1(n)^2)^2;
end
disp([t1', u_exact']);
```

Problem3

求解边值问题

$\Delta u = 2\pi^2 e^{\pi(x+y)}(\sin\pi x\cos\pi y + \cos\pi x\sin\pi y), \quad (x,y)\in G=(0,1)\times(0,1)$

$u=0, \quad (x,y)\in\partial G$

(1)取步长$h=k=1/16; 1/32; 1/64$进行数值求解, 并列表给出不同网格中$(1/4,1/4)$, $(1/2,1/2), (3/4,3/4)$点处的数值解和解析解。

(2)就$h=1/32$，画出$y=1/4, 1/2$和$3/4$及$i=0,1,2,...,32$时的差分解曲线.

数值方法

五点差分：$-\Delta_h u_{ij} = -\left[\frac{u_{i+1,j}-2u_{ij}+u_{i-1,j}}{h_1^2} + \frac{u_{i,j+1}-2u_{ij}+u_{i,j-1}}{h_2^2}\right] = f_{ij}$,
并写出如下格式的矩阵：

$$A = \begin{bmatrix} D_1 & C_1 & & 0 & \\ B_1 & D_2 & C_2 & & \\ & B_2 & \ddots & \ddots & \\ & 0 & \ddots & \ddots & C_{m-1} \\ & & & B_{m-1} & D_m \end{bmatrix}$$

解答

- ▶ **精确解：**

$(1/4,1/4):$	$2.4052,$
$(1/2,1/2):$	$23.1407,$
$(3/4,3/4):$	$55.6589,$

- ▶ **数值解：**

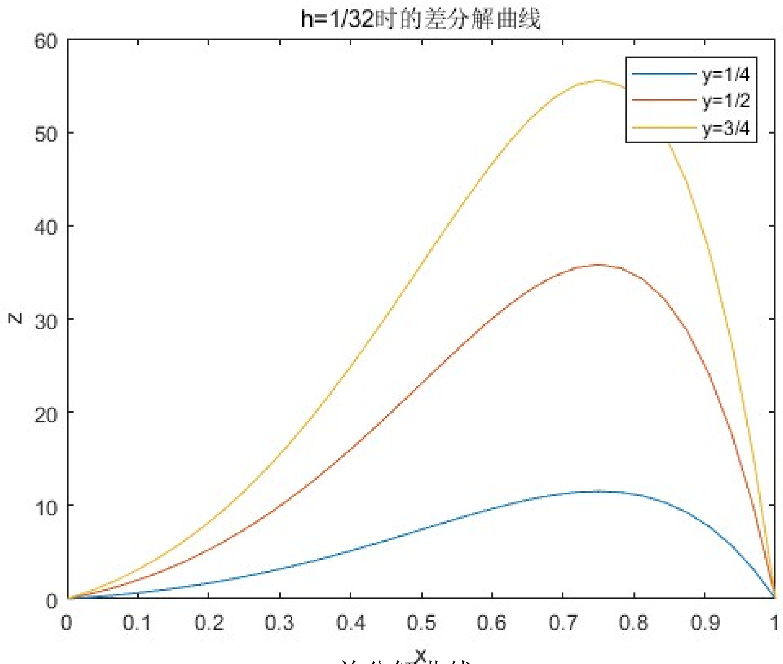
$$h=\frac{1}{16}:$$

$(1/4,1/4):$	2.2905
$(1/2,1/2):$	22.7574
$(3/4,3/4):$	55.2767

$$h=\frac{1}{32}:$$

$(1/4,1/4):$	2.3765
$(1/2,1/2):$	23.0449
$(3/4,3/4):$	55.5637

$$
\begin{aligned}
&h=\frac{1}{64}:\\
&(1/4,1/4):\quad 2.4052\\
&(1/2,1/2):\quad 23.1407\\
&(3/4,3/4):\quad 55.6589
\end{aligned}
$$



差分解曲线

代码

Listing 3 problem3.m

```
clear;clc;close all;

% 网格步长
H = [1/16,1/32,1/64];
for t =1:3
    h = H(t);

    % 计算节点数
    N = 1/h + 1;

    % 初始化参数
    x = linspace(0, 1, N);
    y = linspace(0, 1, N);
```

```
% 右端项的函数定义
f = @(x,y)
   2*pi^2*exp(pi*(x+y))*(sin(pi*x)*cos(pi*y)+cos(pi*x)*sin(pi*y));

% 精确解函数定义
u_exact = @(x,y) exp(pi*(x+y))*sin(pi*x)*sin(pi*y);

% 内部网格点数
n = N - 2;  % 不包括边界点

% 初始化系数矩阵和右端项向量
A = zeros(n*n, n*n);
b = zeros(n*n, 1);

% 右端向量
for i = 1:n
    for j = 1:n
        K = (j-1)*n + i;
        b(K) = f(x(i+1), y(j+1));
    end
end

% 系数矩阵
for i = 1:n
    for j = 1:n
        K = (j-1)*n + i;

        A(K, K) = -4;  % 中心点
        if i > 1
            A(K, K-1) = 1;  % 左边点
        end
        if i < n
            A(K, K+1) = 1;  % 右边点
        end
        if j > 1
            A(K, K-n) = 1;  % 下边点
        end
        if j < n
            A(K, K+n) = 1;  % 上边点
        end
    end
end
g_s = @(A, b, tol, max_iter)  gauss_seidel(A, b, tol,
   max_iter);

% 高斯赛德尔计算较慢，调试时用\计算。
u_vector = g_s(A,h^2*b,1e-4,1000);
% u_vector = A \ (h^2 * b);

% 将解向量转换回网格方便下面绘图
u = zeros(N, N);
for i = 1:n
    for j = 1:n
        K = (j-1)*n + i;
        u(i+1, j+1) = u_vector(K);
    end
end

% 计算精确解在节点处的值
U_exact = zeros(N, N);
for i = 1:N
    for j = 1:N
        U_exact(i,j) = u_exact(x(i), y(j));
    end
end

% 输出指定位置的数值解，解析解。设(x，y)位于矩阵（i,j)处，则
% i = x/h +1 （+1因为第一列为0）。纵坐标同理
fprintf("h= %s 时：\n",rats(h));
fprintf("(1/4,1/4)数值解：%f",u(1/(4*h)+1,1/(4*h)+1));
fprintf(" 解析解： %f \n",U_exact(1/(4*h)+1,1/(4*h)+1));
fprintf("(1/2,1/2)数值解：%f",u(1/(2*h)+1,1/(2*h)+1));
fprintf(" 解析解： %f \n",U_exact(1/(2*h)+1,1/(2*h)+1));
fprintf("(3/4,3/4)数值解：%f",u(3/(4*h)+1,3/(4*h)+1));
fprintf(" 解析解： %f \n",U_exact(3/(4*h)+1,3/(4*h)+1));

% 绘制数值解图,h=1/32
if h==1/32
```

```
        figure;

        % y =1/4
        plot(x,u(:,9));
        hold on;

        % y =1/2
        plot(x,u(:,17));

        % y =3/4
        plot(x,u(:,25));

        legend('y=1/4','y=1/2','y=3/4');
        xlabel('x');
        ylabel('z');
        title("h=1/32时的差分解曲线");
    end

end

%% 比较数值解 精确解
[X, Y] = meshgrid(x, y);
figure;
surf(X, Y, u);
title('Gauss\_sidel求解泊松方程的数值解');
xlabel('x');
ylabel('y');
zlabel('u');

figure;
surf(X, Y, U_exact);
title('泊松方程的精确解');
xlabel('x');
ylabel('y');
zlabel('u\_exact');

function [x, iter, error] = gauss_seidel(A, b, tol, max_iter)
    % 高斯赛德尔迭代法
    % A: 系数矩阵
    % b: 常数项向量
    % tol：误差容限
    % max_iter：最大迭代次数
    % 返回值：
    % x：解向量
    % iter：实际迭代次数
    % error：最终误差

    % 输入检查
    if nargin < 4
        error('需要四个输入参数：A，b，tol，max_iter');
    end

    [m, n] = size(A);
    if m ~= n || length(b) ~= n
        error('A必须是方阵，且A和b的维度必须匹配');
    end

    % 初始化
    x = zeros(n, 1);  % 初始解向量
    x_old = x;        % 前一次迭代的解向量
    inv_diag = 1 ./ diag(A);  % 预先计算对角线元素的倒数

    for iter = 1:max_iter
        for i = 1:n
            % 计算当前行的新的 x 值
            sum1 = A(i, 1:i-1) * x(1:i-1);
            sum2 = A(i, i+1:n) * x_old(i+1:n);
            x(i) = (b(i) - sum1 - sum2) * inv_diag(i);
        end

        % 检查收敛性
        error = norm(x - x_old, inf);
        if error < tol
            fprintf('迭代收敛于第 %d 次迭代。\n', iter);
            return;
        end

        % 更新旧的解向量
        x_old = x;
    end
```

```
    warning('达到最大迭代次数 %d，未收敛。最终误差：%e',
       max_iter, error);
end
```

Problem4

求解一维抛物方程的初边值问题：

$$\begin{aligned} &\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \sin t, && 0 < x < 1, \quad t > 0, \\ &u_x(0,t) = u_x(1,t) = 0, && t > 0, \\ &u(x,0) = \cos \boldsymbol{\pi} x, && 0 < x < 1. \end{aligned}$$

(精确解为：$u = e^{-\pi^2 t}\cos \pi x + (1 - \cos t)$)
分别采用向后以及六点对称格式，通过以下两种方案：
(1) h=1/40, $\tau = 1/1600$; (2) h=1/80, $\tau = 1/3200$
计算到时间 t=1 。
并列表给出该时刻, 两种数值方法以及解析解在x=1/5,2/5,3/5,4/5,1处的值。

数值方法

向后差分：$-ru_{j+1}^{n+1} + (1+2r)u_j^{n+1} - ru_{j-1}^{n+1} = u_j^n + \tau f_j$.

$$\begin{bmatrix} 1+2r & -r & \\ -r & \ddots & -r \\ & -r & 1+2r \end{bmatrix} \begin{bmatrix} u_1^{n+1} \\ \vdots \\ u_{J-1}^{n+1} \end{bmatrix} = \begin{bmatrix} u_1^n \\ \vdots \\ u_{J-1}^n \end{bmatrix} + \tau \begin{bmatrix} f_1 \\ \vdots \\ f_{J-1} \end{bmatrix}$$

六点对称 (CN)：$-\frac{r}{2}u_{j+1}^{n+1} + (1+r)u_j^{n+1} - \frac{r}{2}u_{j-1}^{n+1} = \frac{r}{2}u_{j+1}^n + (1-r)u_j^n + \frac{r}{2}u_{j-1}^n + \tau f_j$,

$$\begin{bmatrix} 1+r & -\frac{r}{2} & & \\ -\frac{r}{2} & \ddots & \ddots & \\ & \ddots & \ddots & -\frac{r}{2} \\ & & -\frac{r}{2} & 1+r \end{bmatrix} \begin{bmatrix} u_1^{n+1} \\ \vdots \\ u_{J-1}^{n+1} \end{bmatrix} = \tau \begin{bmatrix} 1-r & \frac{r}{2} & & \\ \frac{r}{2} & \ddots & \ddots & \\ & \ddots & \ddots & \frac{r}{2} \\ & & \frac{r}{2} & 1-r \end{bmatrix} + \begin{bmatrix} f_1 \\ \vdots \\ \vdots \\ f_{J-1} \end{bmatrix}$$

解答

- ▶ **结论**：由表对比可知步长相同时，CN 法精度更高。

x	方案 1(BD)	方案 1(CN)	方案 2(BD)	方案 2(CN)	解析解
0.2	0.460004	0.459740	0.459872	0.459740	0.459740
0.4	0.459977	0.459714	0.459845	0.459714	0.459714
0.6	0.459944	0.459682	0.459813	0.459682	0.459682
0.8	0.459917	0.459656	0.459787	0.459656	0.459656
1.0	0.459907	0.459646	0.459777	0.459646	0.459646

代码

Listing 4 problem4.m

```
clear;clc;

% 函数定义
solve_heat_equation_BD = @(h, tau)
   heat_equation_solver_backward_difference(h, tau);
solve_heat_equation_CN = @(h, tau)
   heat_equation_solver_crank_nicolson(h, tau);

% 方案1: h=1/40,  =1/1600
[x1_BD, u1_BD] = solve_heat_equation_BD(1/40, 1/1600);
[x1_CN, u1_CN] = solve_heat_equation_CN(1/40, 1/1600);

% 方案2: h=1/80,  =1/3200
[x2_BD, u2_BD] = solve_heat_equation_BD(1/80, 1/3200);
[x2_CN, u2_CN] = solve_heat_equation_CN(1/80, 1/3200);

% 解析解
x_eval = [1/5, 2/5, 3/5, 4/5, 1];
T = 1;
analytical = exp(-pi^2*T) * cos(pi * x_eval) + (1 - cos(T));

% 直接从计算结果中提取所需点的值
indices1 = round(x_eval / (1/40)) + 1;  % 方案1的索引
indices2 = round(x_eval / (1/80)) + 1;  % 方案2的索引

u1_BD_eval = u1_BD(indices1);
u2_BD_eval = u2_BD(indices2);
u1_CN_eval = u1_CN(indices1);
u2_CN_eval = u2_CN(indices2);

% 显示结果，保留六位小数。
fprintf('时间 t = %f 时的结果:\n', T);
fprintf('x\t\t方案1(BD)\t方案1(CN)\t方案2(BD)\t方案2(CN)\t解析解\n');
for i = 1:5
```

```
    fprintf('%.1f\t\t%.6f\t%.6f\t%.6f\t%.6f\t%.6f\n', x_eval(i),
       u1_BD_eval(i), u1_CN_eval(i), u2_BD_eval(i), u2_CN_eval(i),
       analytical(i));
end

%% 向后差分
function [x, u] = heat_equation_solver_backward_difference(h, tau)
    L = 1;
    T = 1;
    a = 1;        % 热系数

    % 参数设置
    Nx = L / h;
    Nt = T / tau;


    r = a * tau / h^2;

    % 初始条件
    x = linspace(0, L, Nx+1)';
    t = linspace(0, T, Nt+1)';
    u = cos(pi * x);

    % 系数矩阵A
    A = diag((1 + 2*r) * ones(Nx+1, 1)) + diag(-r * ones(Nx, 1), 1)
       + diag(-r * ones(Nx, 1), -1);

    % Neumann边界条件的调整
    A(1,2) = -2*r;
    A(end,end-1) = -2*r;

    % 右端函数
    f = @(t) sin(t);

    % 求解
    for n = 2:Nt+1
        b = u + tau * f(t(n));
        u = A \ b;
    end
end
```

```
%% CN法
function [x, u] = heat_equation_solver_crank_nicolson(h, tau)
    L = 1;
    T = 1;
    a = 1;         % 热系数

    % 参数设置
    Nx = L / h;
    Nt = T / tau;
    r = a * tau / h^2;

    % 初始条件
    x = linspace(0, L, Nx+1)';
    t = linspace(0, T, Nt+1)';
    u = cos(pi * x);

    % 系数矩阵
    A = zeros(Nx+1, Nx+1);
    B = zeros(Nx+1, Nx+1);
    A(1,1:2) = [1+r, -r];
    B(1,1:2) = [1-r, r];
    for i = 2:Nx
        A(i,i-1:i+1) = [-r/2, 1+r, -r/2];
        B(i,i-1:i+1) = [r/2, 1-r, r/2];
    end
    A(Nx+1,Nx:Nx+1) = [-r, 1+r];
    B(Nx+1,Nx:Nx+1) = [r, 1-r];

    % 右端函数
    f = @(t) sin(t);

    % 求解
    for n = 2:Nt+1
        b = B * u + 0.5 * tau * (f(t(n)) + f(t(n-1)));
        u = A \ b;
    end
end
```

Problem5

求解二维抛物方程的初边值问题：

$$\begin{aligned}
\frac{\partial u}{\partial t} &= 4^{-2}\left(u_{xx}+u_{yy}\right), (x,y)\in G=(0,1)\times(0,1), t>0,\\
u(0,y,t) &= u(1,y,t)=0, 0<y<1, t>0,\\
u_y(x,0,t) &= u_y(x,1,t)=0, 0<x<1, t>0,\\
u(x,y,0) &= \sin(\pi x)\cos(\pi y).
\end{aligned}$$

(精确解为：$u=\sin\pi x\cos\pi y\exp\left(-\frac{\pi^2}{8}t\right)$)
采用 ADI 法，通过以下方案：h=1/40,$\tau=1/1600$, 计算到时间 t=1。
(1) 画出 t=0.3,0.5,0.8 这三个时刻数值解和精确解的对比图；
(2) 列表给出 t=1 时刻，在节点 $(x_j,y_k)=\left(\frac{j}{4},\frac{k}{4}\right), j,k=1,2,3$ 处的数值结果

数值方法

ADI:

$$\frac{u_{jk}^{n+\frac{1}{2}}-u_{jk}^{n}}{\tau/2}=\frac{1}{h^2}\left(\delta_x^2 u_{jk}^{n+\frac{1}{2}}+\delta_y^2 u_{jk}^{n}\right), \frac{u_{jk}^{n+1}-u_{jk}^{n+\frac{1}{2}}}{\tau/2}=\frac{1}{h^2}\left(\delta_x^2 u_{jk}^{n+\frac{1}{2}}+\delta_y^2 u_{jk}^{n+1}\right),$$

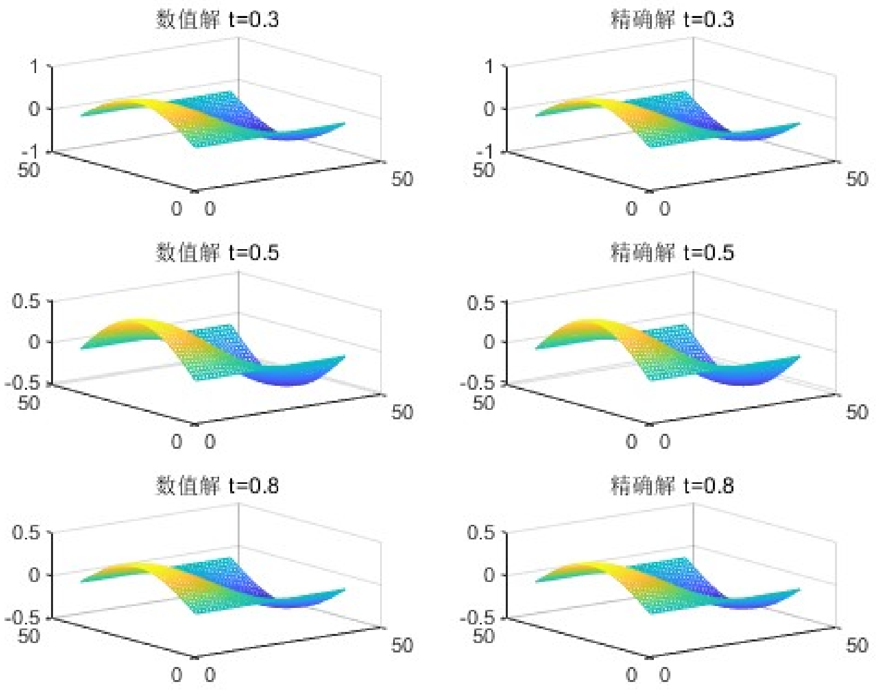
$$\delta_x^2 u_{jk}^n=u_{j+1,k}^n-2u_{j,k}^n+u_{j-1,k}^n, \delta_y^2 u_{jk}^n=u_{j,k+1}^n-2u_{j,k}^n+u_{j,k-1}^n,$$

利用 Neumann 条件对 y 边界处的差分进行处理：

$$u_{j,k+1}^n=u_{j,k}^n u_{j,-1}^n=u_{j,1}^n$$

解答

x	0.25	0.25	0.25	0.50	0.50	0.50	0.75	0.75	0.75
y	0.25	0.50	0.75	0.25	0.50	0.75	0.25	0.50	0.75
数值解	0.142283	0.000000	-0.142283	0.201218	0.000000	-0.201218	0.142283	0.000000	-0.142283



对比图

代码

Listing 5 problem5.m

```
clear;clc;close all;

% 参数设置
L = 1;
T = 1;

h = 1/40;
tau = 1/1600;

Nx = 1/h;
Ny = 1/h;
Nt = 1/tau;

x = linspace(0,L,Nx+1);
y = linspace(0,L,Ny+1);
t = linspace(0,T,Nt+1);
r = tau/h^2;
```

```
% 结果矩阵
U = zeros(Nx+1,Ny+1,Nt+1);

% 初始条件
for i = 1:Nx+1
    for j = 1:Ny+1
        U(i,j,1) = sin(pi*x(i))*cos(pi*y(j));
    end
end

% 边界条件
U(1,:,:) = 0; % 左边界
U(Nx+1,:,:) = 0; % 右边界

% 系数矩阵1 (x方向)
A1 = zeros(Nx+1,Nx+1);
A1(1,1) = 1;
A1(Nx+1,Nx+1) = 1;
for i = 2:Nx
    A1(i,i-1:i+1) = [-r/32, 1+r/16, -r/32];
end

% 系数矩阵2 (y方向)
A2 = zeros(Ny+1,Ny+1);
A2(1,1:2) = [1+r/16, -r/16];
A2(Ny+1,Ny:Ny+1) = [-r/16, 1+r/16];
for i = 2:Ny
    A2(i,i-1:i+1) = [-r/32, 1+r/16, -r/32];
end

for n = 1:Nt
    % 时间对偶点n+1/2处值
    U_half = zeros(Nx+1,Ny+1);

    for j = 2:Ny
        b = zeros(Nx+1,1);
        for i = 2:Nx
            b(i) = (r/32)*(U(i,j+1,n) + U(i,j-1,n)) +
               (1-r/16)*U(i,j,n);
        end
        U_half(:,j) = A1\b;
    end

    % 处理y方向的边界条件
    U_half(:,1) = U_half(:,2);
    U_half(:,Ny+1) = U_half(:,Ny);

    % 计算n+1时刻
    for i = 2:Nx
        b = zeros(Ny+1,1);
        for j = 1:Ny+1
            b(j) = (r/32)*(U_half(i+1,j) + U_half(i-1,j)) +
               (1-r/16)*U_half(i,j);
        end
        U(i,:,n+1) = (A2\b)';
    end
end

% 对比图
figure;

t1 = 0.3/tau + 1;
t2 = 0.5/tau + 1;
t3 = 0.8/tau + 1;

subplot(3,2,1); mesh(U(:,:,t1)); title('数值解 t=0.3');
subplot(3,2,2); mesh(sin(pi*x)' * cos(pi*y) * exp(-pi^2*0.3/8));
   title('精确解 t=0.3');

subplot(3,2,3); mesh(U(:,:,t2)); title('数值解 t=0.5');
subplot(3,2,4); mesh(sin(pi*x)' * cos(pi*y) * exp(-pi^2*0.5/8));
   title('精确解 t=0.5');

subplot(3,2,5); mesh(U(:,:,t3)); title('数值解 t=0.8');
subplot(3,2,6); mesh(sin(pi*x)' * cos(pi*y) * exp(-pi^2*0.8/8));
   title('精确解 t=0.8');

% 数值结果
disp('t = 1 时刻的数值结果：')
disp(' x  y  数值解')
```

```
for j = 1:3
    for k = 1:3
        x_idx = j*10 + 1;
        y_idx = k*10 + 1;
        val = U(x_idx, y_idx, end);
        fprintf('%.2f  %.2f  %.6f\n', j/4, k/4, val)
    end
end
```

Problem6

求解：

$$
\begin{aligned}
&u_t - u_x = 0, && x \in (0,1),\\
&u(x,0) = \sin^{40}(\pi x), && x \in [0,1],\\
&u(0,t) = u(1,t), && t \geqslant 0.
\end{aligned}
$$

(精确解为：$u(x,t) = \sin^{40}\pi(x+t)$)
用迎风格式计算。
(a) 取 h=0.05, $\tau = 0.04$ 算出 t=0,0.12,0.2,0.8 的值；
(b) 画出 (a) 的图形，观察峰值位置的变化，与精确解峰值位置比较。

数值方法

利用 a<0 的差分格式求解：

$$
\frac{u_j^{n+1} - u_j^n}{\tau} + a\frac{u_{j+1}^n - u_j^n}{h} = 0,
$$

利用周期边值条件 $u(0,t) = u(1,t)$.

$$
\begin{bmatrix} u_0^{n+1} \\ \vdots \\ u_J^{n+1} \end{bmatrix}
=
\begin{bmatrix} 1+r & -r & \\ & \ddots & -r \\ 1+r & -r & \dots 0 \end{bmatrix}
\begin{bmatrix} u_0^{n} \\ \vdots \\ u_J^{n} \end{bmatrix}
$$

解答

t=0

x	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4
u	0	5.93331839654999e-33	3.97461038954230e-21	1.91368562288649e-14	5.87147764647043e-10	9.53674316406247e-07	0.000208116150133719	0.00989088979069069	0.134354748960890
x	0.45	0.5	0.55	0.60	0.65	0.70	0.75	0.8	0.85
u	0.609252167050786	1	0.609252167050783	0.134354748960891	0.00989088979069074	0.000208116150133719	9.53674316406253e-07	5.87147764647047e-10	1.91368562288651e-14
x	0.9	0.95	1						
u	3.97461038954235e-21	5.93331839655016e-33	0						

t=0.12

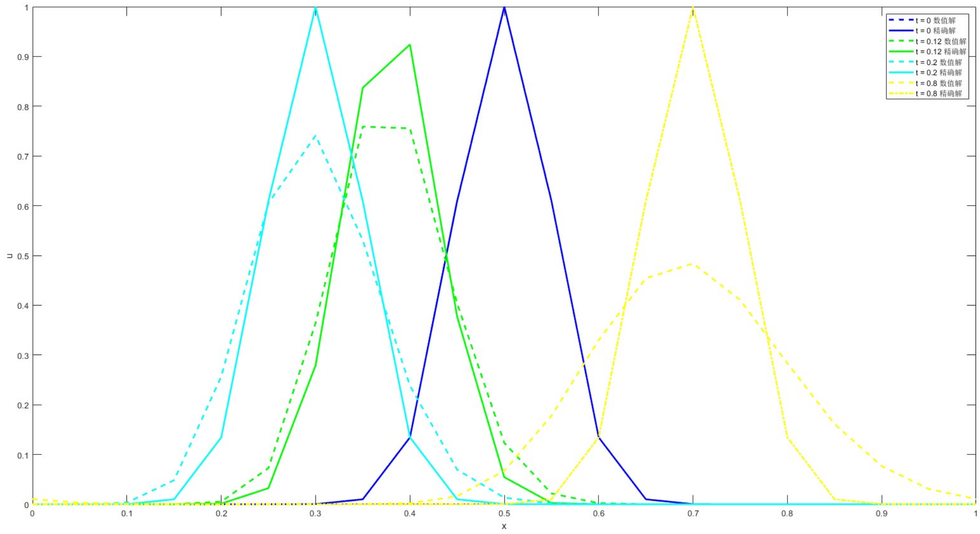
x	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4
u	9.79807191542923e-15	3.00627004052459e-10	4.88506716578761e-07	0.000106921736172303	0.00514414373191654	0.0726077199274082	0.364480523480091	0.758930015166073	0.755500155558563
x	0.45	0.5	0.55	0.60	0.65	0.70	0.75	0.8	0.85
u	0.403616480951883	0.123144567210691	0.0216767303851455	0.00210476829449479	9.94727802955186e-05	1.75670740998445e-06	7.68576806521097e-09	4.69901925690060e-12	1.53095231393518e-16
x	0.9	0.95	1						
u	3.17968831199463e-23	2.03500051944798e-21	9.79807191542923e-15						

t=0.20

x	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4
u	3.12740499643627e-07	6.85862453246592e-05	0.00332648648427038	0.0481193436172014	0.256707771153306	0.605253286017012	0.740956925349828	0.530431798194588	0.238189803141787
x	0.45	0.5	0.55	0.60	0.65	0.70	0.75	0.8	0.85
u	0.0694240281919895	0.0132093881201509	0.00160425764903328	0.000117146314216747	4.54597647457751e-06	7.27307495525701e-08	3.08934506751595e-10	1.88009760770420e-13	6.12512183107579e-18
x	0.9	0.95	1						
u	6.27076667834674e-15	1.92404417976668e-10	3.12740499643627e-07						

t=0.80

x	0	0.05	0.1	0.15	0.2	0.25	0.3	0.35	0.4
u	0.0103648171974659	0.00297699507912104	0.000726845501493071	0.000150775611073033	2.65086802766898e-05	3.94332277913092e-06	2.94321868905239e-06	0.000126212357121167	0.00214786906368297
x	0.45	0.5	0.55	0.60	0.65	0.70	0.75	0.8	0.85
u	0.0161663273640771	0.067121454698917	0.177868907439971	0.329763411190648	0.454117393422128	0.483837948786123	0.410474782848938	0.283111896245817	0.161187260716036
x	0.9	0.95	1						
u	0.0766037453686443	0.0306337143149631	0.0103648171974659						



观察图像可知 t=0 时数值解与精确解重合 (t=0 由初值条件解得)，其它时间点有：

1. 数值解的波峰比精确解的波峰低。
2. 数值解的波形比精确解更加平缓。
3. 数值解的波峰位置略微滞后于精确解。

代码

Listing 6 problem6.m

```
clear;clc;close all;
```

```
% 参数设置
L = 1;
T = 1;
h = 0.05;
tau = 0.04;
a = -1;
r = a*tau/h;
t_idx = [0, 0.12 , 0.2 , 0.8];

% 区间数
Nx = L/h ;
Nt = T/tau;
x = linspace(0,L,Nx+1);
t = linspace(0,T,Nt+1);

% 初始条件
u = zeros(Nx+1,Nt+1);
for i = 1:Nx+1
    u(i,1) = (sin(pi*x(i)))^40;
end

% 求解 u(j,n+1)=(1+r)u(j,n)-ru(j+1,n)
for n = 2:Nt+1
    for i = 1:Nx
        u(i,n) = (1+r)*u(i,n-1)-r*u(i+1,n-1);
    end
    u(end,n) = u(1,n);
end

% 精确解
u_exact = @(x,t) (sin(pi*(x+t))).^40;
U_exact = zeros(Nx+1, Nt+1);
for n = 1:Nt+1
    for i = 1:Nx+1
        U_exact(i,n) = (sin(pi*(x(i) + t(n))))^40;
    end
end

% 绘图
```

```
figure;

% 绘制第一组对比图 (t = 0)
plot(x, u(:, 1), ’b--’, ’LineWidth’, 2);
hold on;
plot(x, u_exact(x, 0), ’b-’, ’LineWidth’, 2);

% 绘制第二组对比图 (t = 0.12)
plot(x, u(:, 4), ’g--’, ’LineWidth’, 2);
plot(x, u_exact(x, 0.12), ’g-’, ’LineWidth’, 2);

% 绘制第三组对比图 (t = 0.2)
plot(x, u(:, 6), ’c--’, ’LineWidth’, 2);
plot(x, u_exact(x, 0.2), ’c-’, ’LineWidth’, 2);

% 绘制第四组对比图 (t = 0.8)
plot(x, u(:, 21), ’y--’, ’LineWidth’, 2);
plot(x, u_exact(x, 0.8), ’y-.’, ’LineWidth’, 2);

legend(’t = 0 数值解’, ’t = 0 精确解’, ’t = 0.12 数值解’, ’t =
   0.12 精确解’, ’t = 0.2 数值解’, ’t = 0.2 精确解’, ’t = 0.8
   数值解’, ’t = 0.8 精确解’);
xlabel(’x’);
ylabel(’u’);
```