数值试验一 赵衍智 **2020511041**

第一部分: 数值试验的目的

1 : 将矩形区域上的泊松问题转化为线性代数方程组的求解问题
2 : 用不同的迭代求解线性代数方程的近似解
3 : 分析不同迭代的近似解、迭代次数、近似解与精确解之间的误差

第二部分: 问题的提出(详见《数值分析习题》)

考虑矩形区域上的泊松问题,通过差分法将泊松问题的求解转化为线性代数方程组的求解

第三部分: 数值试验结果

一 : 迭代理论

- 设E是$(N-1)^2$阶的单位矩阵,D是L_h的对角阵,L是L_h的下三角阵,U是L_h的上三角阵
- Jacobi迭代

$$L_h u^h = h^2 f^h \Rightarrow u^{k+1} = J u^k + g$$

$$J = E - D^{-1}L_h = \begin{pmatrix} \frac{1}{4}C & \frac{1}{4}I & & & \\ \frac{1}{4}I & \frac{1}{4}C & \frac{1}{4}I & & \\ & \ddots & \ddots & \ddots & \\ & & \frac{1}{4}I & \frac{1}{4}C & \frac{1}{4}I \\ & & & \frac{1}{4}I & \frac{1}{4}C \end{pmatrix}_{(N-1)\times(N-1)}, \quad g = D^{-1}h^2 f^h = \frac{1}{4}Eh^2 f^h = \begin{pmatrix} \frac{1}{4}h^2 f_1^h \\ \frac{1}{4}h^2 f_2^h \\ \vdots \\ \frac{1}{4}h^2 f_{N-1}^h \end{pmatrix}$$

- Gauss-Seidel迭代

$$L_h u^h = h^2 f^h \Rightarrow u^{(k+1)} = S u^k + g$$
$$S = -(D+L)^{-1}U, g = (D+L)^{-1}h^2 f^h$$

- SOR迭代

第一类: $L_h u^h = h^2 f^h \Rightarrow u^{(k+1)} = S_\omega u^{(k)} + g$
$S_\omega = (E + \omega D^{-1}L)^{-1}[(1-\omega)E - \omega D^{-1}U], g = \omega(E+\omega D^{-1}L)^{-1}D^{-1}h^2 f^h$
等价地有第二类表示 : $S_\omega = (D+\omega L)^{-1}((1-\omega)D - \omega U), g = \omega(D+\omega L)^{-1}h^2 f^h$
在数值试验中使用第一类SOR迭代
当$\omega = 1$时,SOR迭代变为Gauss-Seidel迭代

二 : 步长$h = 0.1$

- $$\text{取初始迭代值为} u_0 = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}_{(N-1)^2 \times 1}$$

- 精确解u^*(u^*为81×1列向量,为了方便展示,取u^*的转置,如下数组所示)

列 1 至 11

0.0955	0.1816	0.2500	0.2939	0.3090	0.2939	0.2500	0.1816	0.0955	0.1816	0.3455

列 12 至 22

0.4755	0.5590	0.5878	0.5590	0.4755	0.3455	0.1816	0.2500	0.4755	0.6545	0.7694

列 23 至 33

0.8090	0.7694	0.6545	0.4755	0.2500	0.2939	0.5590	0.7694	0.9045	0.9511	0.9045

列 34 至 44

0.7694	0.5590	0.2939	0.3090	0.5878	0.8090	0.9511	1.0000	0.9511	0.8090	0.5878

列 45 至 55

0.3090	0.2939	0.5590	0.7694	0.9045	0.9511	0.9045	0.7694	0.5590	0.2939	0.2500

列 56 至 66

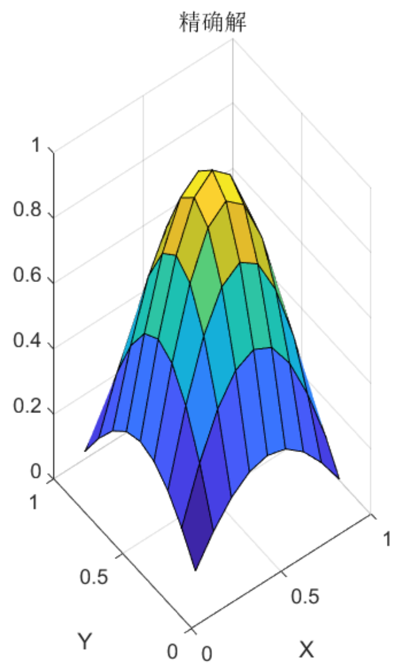
0.4755	0.6545	0.7694	0.8090	0.7694	0.6545	0.4755	0.2500	0.1816	0.3455	0.4755

列 67 至 77

0.5590	0.5878	0.5590	0.4755	0.3455	0.1816	0.0955	0.1816	0.2500	0.2939	0.3090

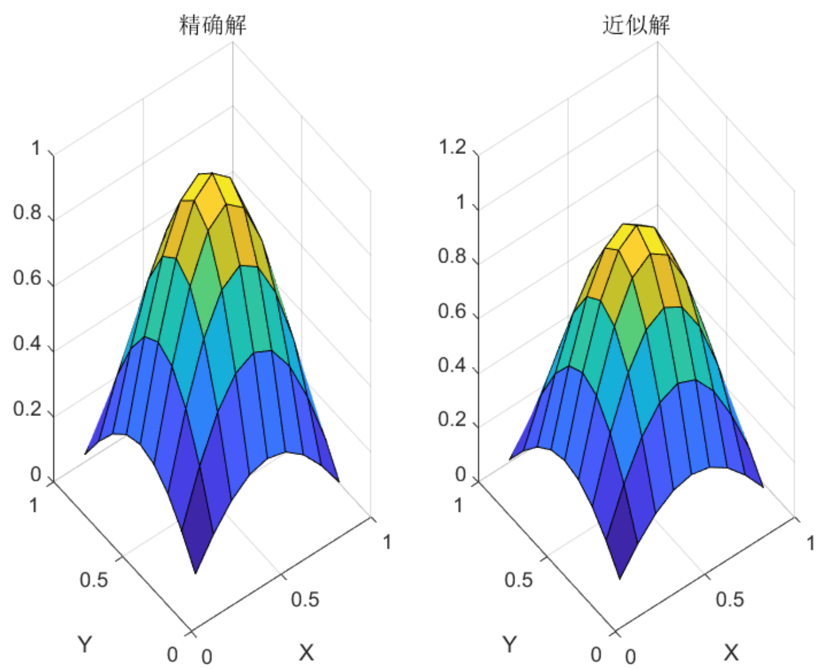
列 78 至 81

0.2939	0.2500	0.1816	0.0955



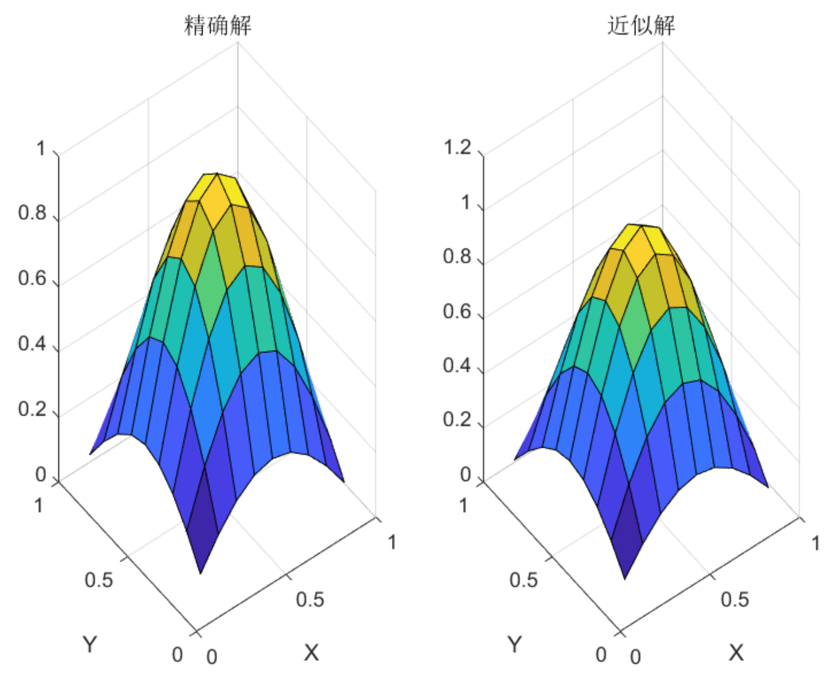
- Jacobi迭代近似解(数组为近似解$u^{(k+1)}$,左侧图像为精确解u^*,右侧图像为近似解$u^{(k+1)}$)

```
列 1 至 11

  0.0963   0.1831   0.2521   0.2963   0.3116   0.2963   0.2521   0.1831   0.0963   0.1831   0.3483

列 12 至 22

  0.4794   0.5636   0.5926   0.5636   0.4794   0.3483   0.1831   0.2521   0.4794   0.6599   0.7758

列 23 至 33

  0.8157   0.7758   0.6599   0.4794   0.2521   0.2963   0.5636   0.7758   0.9120   0.9589   0.9120

列 34 至 44

  0.7758   0.5636   0.2963   0.3116   0.5926   0.8157   0.9589   1.0082   0.9589   0.8157   0.5926

列 45 至 55

  0.3116   0.2963   0.5636   0.7758   0.9120   0.9589   0.9120   0.7758   0.5636   0.2963   0.2521

列 56 至 66

  0.4794   0.6599   0.7758   0.8157   0.7758   0.6599   0.4794   0.2521   0.1831   0.3483   0.4794

列 67 至 77

  0.5636   0.5926   0.5636   0.4794   0.3483   0.1831   0.0963   0.1831   0.2521   0.2963   0.3116

列 78 至 81

  0.2963   0.2521   0.1831   0.0963
```



- Gauss-Seidel迭代近似解(数组为近似解$u^{(k+1)}$,左侧图像为精确解u^*,右侧图像为近似解$u^{(k+1)}$)

```
列 1 至 11

  0.0963    0.1831    0.2521    0.2963    0.3116    0.2963    0.2521    0.1831    0.0963    0.1831    0.3483

列 12 至 22

  0.4795    0.5636    0.5926    0.5636    0.4795    0.3483    0.1831    0.2521    0.4795    0.6599    0.7758

列 23 至 33

  0.8157    0.7758    0.6599    0.4795    0.2521    0.2963    0.5636    0.7758    0.9120    0.9589    0.9120

列 34 至 44

  0.7758    0.5636    0.2963    0.3116    0.5926    0.8157    0.9589    1.0083    0.9589    0.8157    0.5926

列 45 至 55

  0.3116    0.2963    0.5636    0.7758    0.9120    0.9589    0.9120    0.7758    0.5636    0.2963    0.2521

列 56 至 66

  0.4795    0.6599    0.7758    0.8157    0.7758    0.6599    0.4795    0.2521    0.1831    0.3483    0.4795

列 67 至 77

  0.5636    0.5926    0.5636    0.4795    0.3483    0.1831    0.0963    0.1831    0.2521    0.2963    0.3116

列 78 至 81

  0.2963    0.2521    0.1831    0.0963
```



- SOR($\omega=1.2$)迭代近似解(数组为近似解,左侧图像为精确解u^*,右侧图像为近似解$u^{(k+1)}$)

列 1 至 11

0.0669	0.1272	0.1750	0.2058	0.2164	0.2058	0.1750	0.1272	0.0669	0.1272	0.2419

列 12 至 22

0.3330	0.3914	0.4116	0.3914	0.3330	0.2419	0.1272	0.1750	0.3330	0.4583	0.5387

列 23 至 33

0.5665	0.5387	0.4583	0.3330	0.1750	0.2058	0.3914	0.5387	0.6333	0.6659	0.6333

列 34 至 44

0.5387	0.3914	0.2058	0.2164	0.4116	0.5665	0.6659	0.7002	0.6659	0.5665	0.4116

列 45 至 55

0.2164	0.2058	0.3914	0.5387	0.6333	0.6659	0.6333	0.5387	0.3914	0.2058	0.1750

列 56 至 66

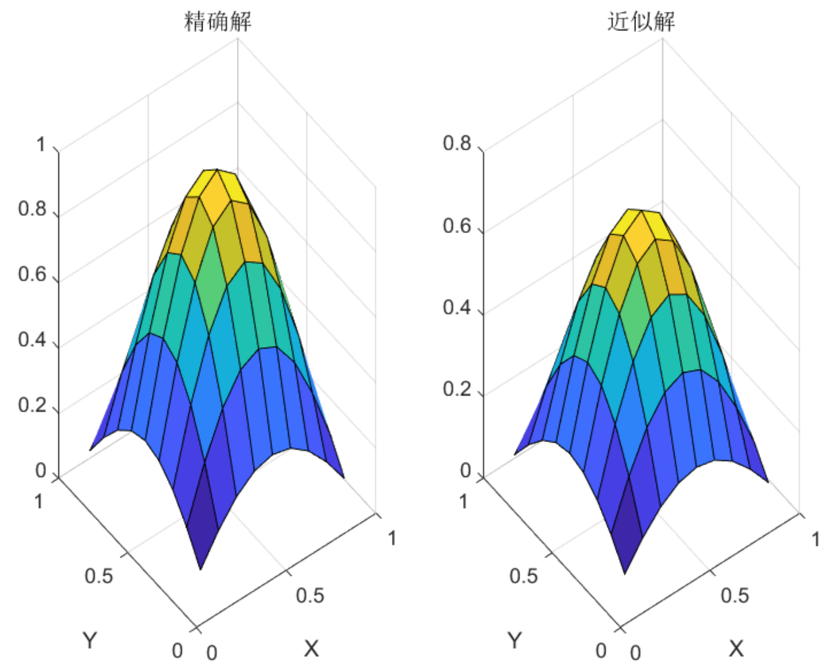
0.3330	0.4583	0.5387	0.5665	0.5387	0.4583	0.3330	0.1750	0.1272	0.2419	0.3330

列 67 至 77

0.3914	0.4116	0.3914	0.3330	0.2419	0.1272	0.0669	0.1272	0.1750	0.2058	0.2164

列 78 至 81

0.2058	0.1750	0.1272	0.0669



- SOR($\omega = 1.3$)迭代近似解(数组为近似解,左侧图像为精确解u^*,右侧图像为近似解$u^{(k+1)}$)

列 1 至 11

0.0570 0.1084 0.1492 0.1753 0.1844 0.1753 0.1492 0.1084 0.0570 0.1084 0.2061

列 12 至 22

0.2837 0.3335 0.3507 0.3335 0.2837 0.2061 0.1084 0.1492 0.2837 0.3905 0.4590

列 23 至 33

0.4827 0.4590 0.3905 0.2837 0.1492 0.1753 0.3335 0.4590 0.5396 0.5674 0.5396

列 34 至 44

0.4590 0.3335 0.1753 0.1844 0.3507 0.4827 0.5674 0.5966 0.5674 0.4827 0.3507

列 45 至 55

0.1844 0.1753 0.3335 0.4590 0.5396 0.5674 0.5396 0.4590 0.3335 0.1753 0.1492

列 56 至 66

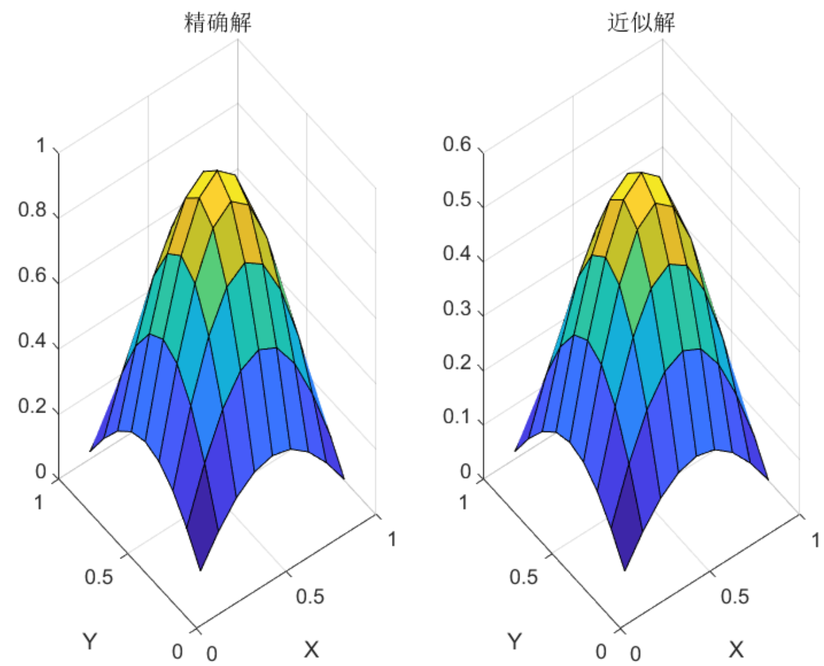
0.2837 0.3905 0.4590 0.4827 0.4590 0.3905 0.2837 0.1492 0.1084 0.2061 0.2837

列 67 至 77

0.3335 0.3507 0.3335 0.2837 0.2061 0.1084 0.0570 0.1084 0.1492 0.1753 0.1844

列 78 至 81

0.1753 0.1492 0.1084 0.0570



- SOR($\omega = 1.9$)迭代近似解(数组为近似解,左侧图像为精确解u^*,右侧图像为近似解$u^{(k+1)}$)

列 1 至 11

0.0267　0.0507　0.0698　0.0821　0.0863　0.0821　0.0698　0.0507　0.0267　0.0507　0.0965

列 12 至 22

0.1328　0.1561　0.1642　0.1561　0.1328　0.0965　0.0507　0.0698　0.1328　0.1828　0.2149

列 23 至 33

0.2260　0.2149　0.1828　0.1328　0.0698　0.0821　0.1561　0.2149　0.2526　0.2656　0.2526

列 34 至 44

0.2149　0.1561　0.0821　0.0863　0.1642　0.2260　0.2656　0.2793　0.2656　0.2260　0.1642

列 45 至 55

0.0863　0.0821　0.1561　0.2149　0.2526　0.2656　0.2526　0.2149　0.1561　0.0821　0.0698

列 56 至 66

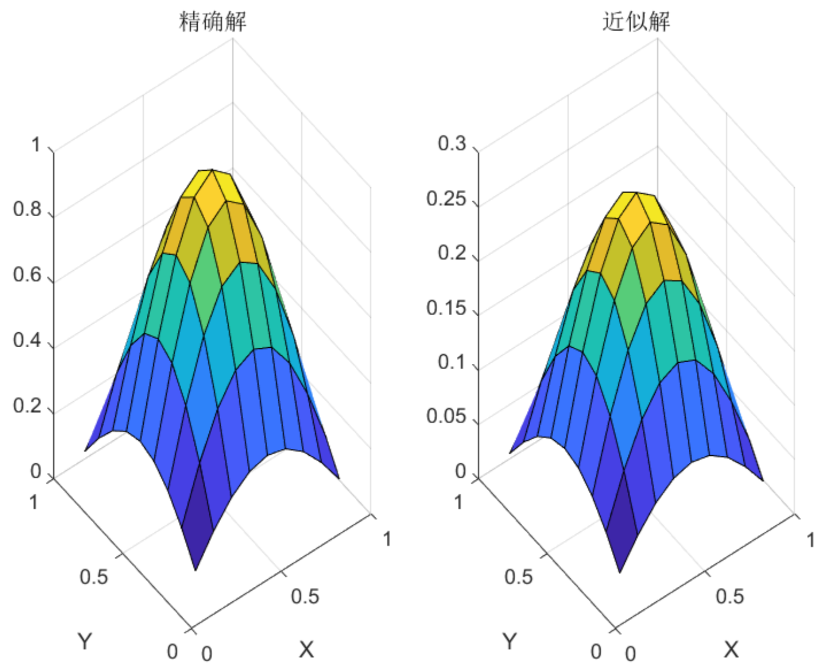
0.1328　0.1828　0.2149　0.2260　0.2149　0.1828　0.1328　0.0698　0.0507　0.0965　0.1328

列 67 至 77

0.1561　0.1642　0.1561　0.1328　0.0965　0.0507　0.0267　0.0507　0.0698　0.0821　0.0863

列 78 至 81

0.0821　0.0698　0.0507　0.0267



- SOR($\omega=0.9$)迭代近似解(数组为近似解,左侧图像为精确解u^*,右侧图像为近似解$u^{(k+1)}$)

列 1 至 11

0.1189	0.2261	0.3112	0.3658	0.3847	0.3658	0.3112	0.2261	0.1189	0.2261	0.4301

列 12 至 22

0.5919	0.6958	0.7317	0.6958	0.5919	0.4301	0.2261	0.3112	0.5919	0.8147	0.9577

列 23 至 33

1.0070	0.9577	0.8147	0.5919	0.3112	0.3658	0.6958	0.9577	1.1259	1.1838	1.1259

列 34 至 44

0.9577	0.6958	0.3658	0.3847	0.7317	1.0070	1.1838	1.2448	1.1838	1.0070	0.7317

列 45 至 55

0.3847	0.3658	0.6958	0.9577	1.1259	1.1838	1.1259	0.9577	0.6958	0.3658	0.3112

列 56 至 66

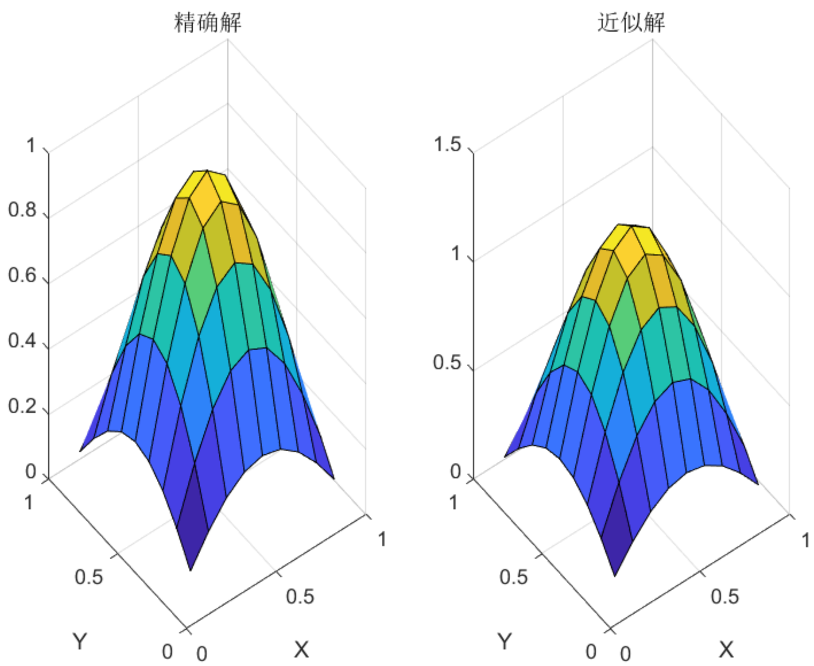
0.5919	0.8147	0.9577	1.0070	0.9577	0.8147	0.5919	0.3112	0.2261	0.4301	0.5919

列 67 至 77

0.6958	0.7317	0.6958	0.5919	0.4301	0.2261	0.1189	0.2261	0.3112	0.3658	0.3847

列 78 至 81

0.3658	0.3112	0.2261	0.1189

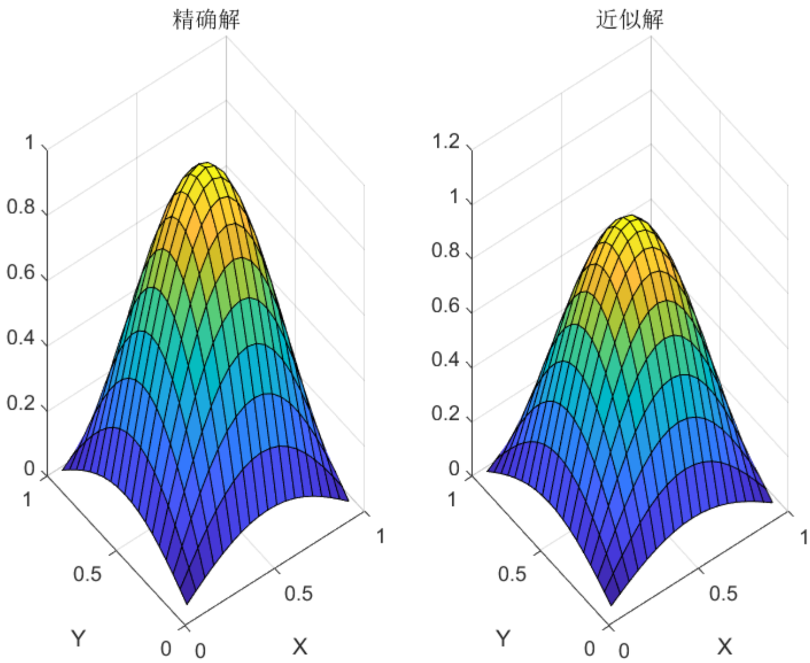


- 迭代次数与误差

$$
\begin{aligned}
&\text{Jacobi迭代：迭代次数} = 217; \|u^{(k+1)} - u^*\|_\infty = 0.0082 \\
&\text{Gauss-Seidel迭代：迭代次数} = 116; \|u^{(k+1)} - u^*\|_\infty = 0.0083 \\
&\text{SOR迭代：}\begin{cases} \omega = 1.2:\text{迭代次数} = 76; \|u^{(k+1)} - u^*\|_\infty = 0.2998 \\ \omega = 1.3:\text{迭代次数} = 61; \|u^{(k+1)} - u^*\|_\infty = 0.4034 \\ \omega = 1.9:\text{迭代次数} = 118; \|u^{(k+1)} - u^*\|_\infty = 0.7207 \\ \omega = 0.9:\text{迭代次数} = 143; \|u^{(k+1)} - u^*\|_\infty = 0.2448 \end{cases}
\end{aligned}
$$

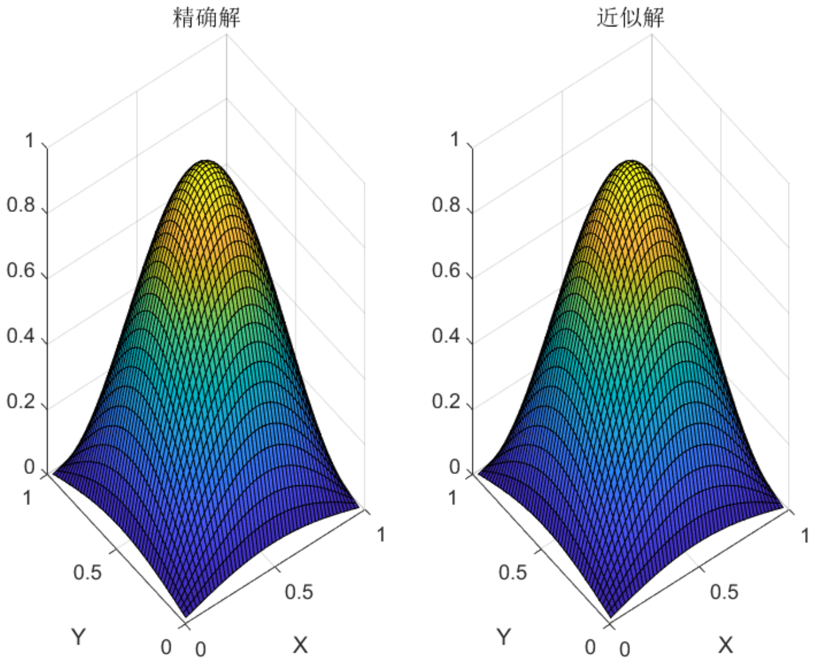
三：步长$h = 0.05$

$$
\text{Jacobi迭代：迭代次数} = 762; \|u^{(k+1)} - u^*\|_\infty = 0.0020
$$
$$
(\text{左侧图像为精确解}u^*, \text{右侧图像为近似解}u^{(k+1)})
$$



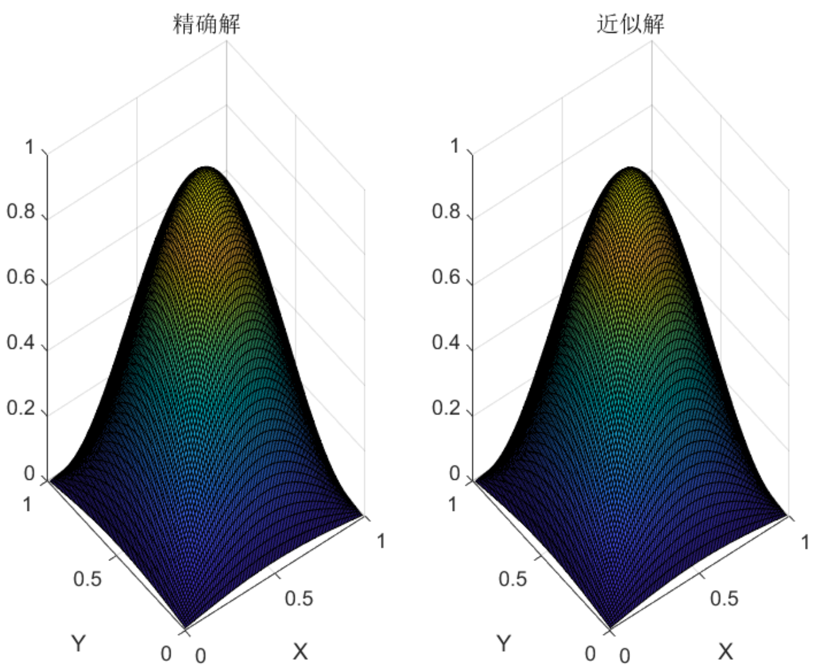
四：步长$h = 0.02$

$$
\text{Jacobi迭代：迭代次数} = 3843; \|u^{(k+1)} - u^*\|_\infty = 1.7620 \times 10^{-4}
$$
$$
(\text{左侧图像为精确解}u^*, \text{右侧图像为近似解}u^{(k+1)})
$$



五：步长$h = 0.01$

$$\text{Jacobi迭代：迭代次数} = 12566; \|u^{(k+1)} - u^*\|_\infty = 0.0019$$
$$(\text{左侧图像为精确解}u^*, \text{右侧图像为近似解}u^{(k+1)})$$



第四部分：数值结果的分析总结

一：步长$h = 0.1$的迭代次数、收敛速度分析

- Jacobi迭代

$$\begin{aligned}&\text{根据数值试验的结果,迭代次数为217}\\&\text{谱半径}\rho(J_{h=0.1})=0.9511,\text{收敛速度为}-\ln\rho(J_{h=0.1})=0.0501\end{aligned}$$

- Gauss-Seidel迭代

$$\begin{aligned}&\text{根据数值试验的结果,迭代次数为116}\\&\text{谱半径}\rho(S)=0.9045,\text{收敛速度为}-\ln\rho(S)=0.1004\end{aligned}$$

- SOR迭代

$$\begin{aligned}&\text{根据数值试验的结果:}\\&\text{当}\omega=1.2\text{时,迭代次数为76;}\ \text{当}\omega=1.3\text{时,迭代次数为61;}\\&\text{当}\omega=1.9\text{时,迭代次数为118;}\ \text{当}\omega=0.9\text{时,迭代次数为143}\\&\omega=1.2:\text{迭代次数为76,谱半径}\rho(S_{1.2})=0.8557,\text{收敛速度为}-\ln\rho(S_{1.2})=0.1558\\&\omega=1.3:\text{谱半径}\rho(S_{1.3})=0.8187,\text{收敛速度为}-\ln\rho(S_{1.3})=0.2\\&\omega=1.9:\text{谱半径}\rho(S_{1.9})=0.9,\text{收敛速度为}-\ln\rho(S_{1.9})=0.1054\\&\omega=0.9:\text{谱半径}\rho(S_{0.9})=0.9218,\text{收敛速度为}-\ln\rho(S_{0.9})=0.0814\end{aligned}$$

- 结论

$$\begin{aligned}&1:\text{迭代次数与收敛速度成负相关}\\&\quad\text{对于任意两种迭代}G_1,G_2:\text{若}G_1\text{的收敛速度大于}G_2,\text{则}G_1\text{的迭代次数小于}G_2\\&2:\text{在本试验中,Gauss-Seidel迭代的收敛速度大于Jacobi迭代,}\\&\quad\text{事实上Gauss-Seidel迭代的收敛速度与Jacobi迭代的收敛速度没有必然联系}\\&3:\text{在本试验中,随着}\omega\text{的增加,SOR迭代的收敛速度先增加后减少}\\&\quad\omega=0.9,1.2,1.3,1.9\text{时均收敛,满足“}0<\omega<2\text{”的必要收敛条件}\end{aligned}$$

二:步长$h=0.05,0.02,0.01$Jacobi迭代的迭代次数、收敛速度分析

- $h=0.05$

$$\begin{aligned}&\text{根据数值试验结果,迭代次数为762}\\&\text{谱半径}\rho(J_{h=0.05})\approx1-\frac{\pi^2h^2}{2}\approx0.9877,\text{收敛速度为}-\ln\rho(J_{h=0.05})=0.0124\end{aligned}$$

- $h=0.02$

$$\begin{aligned}&\text{根据数值试验结果,迭代次数为3843}\\&\text{谱半径}\rho(J_{h=0.02})\approx1-\frac{\pi^2h^2}{2}\approx0.9980,\text{收敛速度为}-\ln\rho(J_{h=0.02})=0.0020\end{aligned}$$

- $h=0.01$

$$\begin{aligned}&\text{根据数值试验结果,迭代次数为12566}\\&\text{谱半径}\rho(J_{h=0.01})\approx1-\frac{\pi^2h^2}{2}\approx0.9995,\text{收敛速度为}-\ln\rho(J_{h=0.02})=5.0013\times10^{-4}\end{aligned}$$

- 结论

$$\begin{aligned}&1:\text{迭代次数与收敛速度成负相关}\\&2:\text{在本试验中,随着步长的减小,收敛速度减小,迭代次数增加}\\&\quad\text{原因:步长减小,迭代矩阵复杂度增加}\end{aligned}$$

三：$u^{(k+1)}$与u^*误差$\|u^{(k+1)}-u^*\|_\infty$的分析

- $h=0.1$

	Jacobi	Gauss-Seidel	$\mathrm{SOR}(\omega:1.2)$	$\mathrm{SOR}(\omega:1.3)$	$\mathrm{SOR}(\omega:1.9)$	$\mathrm{SOR}(\omega:0.9)$
迭代次数	217	116	76	61	118	143
误差	0.0082	0.0083	0.2998	0.4034	0.7207	0.2448

1：误差来源：由于计算机只能将小数保留到特定位数，故在迭代的过程中舍入误差会积累；
误差限ϵ会限制迭代次数，导致误差

2：本试验中，Jacobi和Gauss-Seidel迭代误差较小，SOR迭代误差较大
可能的原因：SOR迭代矩阵的形式为$S_\omega=(E+\omega D^{-1}L)^{-1}[(1-\omega)E-\omega D^{-1}U]$，
复杂度比Jacobi和Gauss-Seidel迭代矩阵更高，故误差更大

- $h=0.05, 0.02, 0.01$

Jacobi	$h=0.05$	$h=0.02$	$h=0.01$
迭代次数	762	3843	12566
误差	0.0020	1.7620×10^{-4}	0.0019

1：误差来源：同理$h=0.1$，迭代的过程中舍入误差会积累，误差限也会导致误差

2：在本试验中，当步长h减小时，迭代次数单调增加，误差先减小后增加
可能的原因：随着h的减小，泊松问题的解会更加精确，h在一定范围内的减小能使误差减小；
当h减小时，迭代次数也在增加，截断误差的累积也越明显，误差最终会增大

第五部分: 程序(MATLAB)

- Jacobi迭代

```matlab
h = 0.02;
N = 1/h;

% 构造 C
C = zeros(N-1,N-1);
C(1,2) = 1;
C(N-1,N-2) = 1;
for i = 1:N-3
    C(i+1,i) = 1;
    C(i+1,i+2) = 1;
end
C_ = (1/4)*C;

% 构造单位矩阵 I
I = zeros(N-1,N-1);
for i = 1:N-1
    I(i,i) = 1;
end
I_ = (1/4)*I;

%% 构造Jacobi迭代矩阵 J
% 先拼接 J 的第一行
J_1 = cat(2,C_,I_);
for i = 3:N-1
    J_1 = cat(2,J_1,zeros(N-1,N-1));
end
J = J_1;
% 再拼接 J 的中间行 J_
for i = 2:N-2
    if i == 2
        J_ = cat(2,I_,C_,I_);
        for j = 4:N-1
            J_ = cat(2,J_,zeros(N-1,N-1));
        end
        J = cat(1,J,J_);
    else
        J_ = cat(2,zeros(N-1,(i-2)*(N-1)),cat(2,I_,C_,I_),zeros(N-1,(N-i-2)*(N-1)));
        J = cat(1,J,J_);
    end
end
% 再拼接 J 的最后一行,完成 J 的构造
J = cat(1,J,cat(2,zeros(N-1,(N-3)*(N-1)),I_,C_));

%%再构造系数矩阵g
f = zeros(N-1,1);
for i = 1:N-1
    f(i,1) = 2*pi^2*sin(pi*i/N)*sin(pi*1/N);
end
for j = 2:N-1
    f_ = zeros(N-1,1);
    for i = 1:N-1
        f_(i,1) = 2*pi^2*sin(pi*i/N)*sin(pi*j/N);
    end
    f = cat(1,f,f_);
end
g = (1/4)*h^2*f;

%% 迭代
% 初始迭代值
u_0 = zeros((N-1)^2,1);
% 准确值
u_e = zeros(N-1,1);
for i = 1:N-1
    u_e(i,1) = sin(pi*i/N)*sin(pi*1/N);
end
for j = 2:N-1
    u_ = zeros(N-1,1);
    for i = 1:N-1
        u_(i,1) = sin(pi*i/N)*sin(pi*j/N);
    end
    u_e = cat(1,u_e,u_);
end
% 开始计算
u = u_0;
n = 0; %迭代次数
ep = 10^(-6);
while n > -1
    u_1 = u;
    u = J*u_1 + g;
    e = norm(u - u_1,"inf");
    n = n+1;
    if e < ep
        break
    end
end
error = norm(u_e-u,"inf");
%% 画图 X:行; Y:列
% 准确值
x = h:h:(N-1)*h;
y = h:h:(N-1)*h;
[X,Y] = meshgrid(x,y);
U_e = sin(pi.*X).*sin(pi.*Y);
subplot(1,2,1);
surf(X,Y,U_e);
title('精确解');
xlabel('X');
ylabel('Y');
% 近似值
U = zeros(N-1,N-1);
for i = 1:N-1
    for j = 1 : N-1
        U(i,j) = u((N-1)*(i-1) + j,1);
    end
end
subplot(1,2,2);
surf(X,Y,U);
title('近似解');
xlabel('X');
ylabel('Y');
```

- SOR迭代

```matlab
h = 0.1;
N = 1/h;
% 构造 C
C = zeros(N-1,N-1);
C(1,2) = 1;
C(N-1,N-2) = 1;
for i = 1:N-3
    C(i+1,i) = 1;
    C(i+1,i+2) = 1;
end
% 构造单位矩阵 I
I = zeros(N-1,N-1);
for i = 1:N-1
    I(i,i) = 1;
end
%% L_h
% 先拼接 L_h 的第一行
L_1 = cat(2,4*I-C,-I);
for i = 3:N-1
    L_1 = cat(2,L_1,zeros(N-1,N-1));
end
L_h = L_1;
% 再拼接 L_h 的中间行 L_
for i = 2:N-2
    if i == 2
        L_ = cat(2,-I,4*I-C,-I);
        for j = 4:N-1
            L_ = cat(2,L_,zeros(N-1,N-1));
        end
        L_h = cat(1,L_h,L_);
    else
        L_ = cat(2,zeros(N-1,(i-2)*(N-1)),cat(2,-I,4*I-C,-I),zeros(N-1,(N-i-2)*(N-1)));
        L_h = cat(1,L_h,L_);
    end
end
% 再拼接 L_h 的最后一行,完成 L_h 的构造
L_h = cat(1,L_h,cat(2,zeros(N-1,(N-3)*(N-1)),-I,4*I - C));
% 取对角阵, 上下三角阵
D = diag(L_h);
D = diag(D); % 对角阵
L = tril(L_h) - D; %下三角阵
U = triu(L_h) - D; %上三角阵
E = (1/4)*D; % (N-1)^2阶单位矩阵
D_ = 0.25*E; % D的逆矩阵
% 计算系数矩阵b
f = zeros(N-1,1);
for i = 1:N-1
    f(i,1) = 2*pi^2*sin(pi*i/N)*sin(pi*1/N);
end
for j = 2:N-1
    f_ = zeros(N-1,1);
    for i = 1:N-1
        f_(i,1) = 2*pi^2*sin(pi*i/N)*sin(pi*j/N);
    end
    f = cat(1,f,f_);
end
b = h^2*f;
%% 构造迭代矩阵S
w = 1.3;
S = (E + w*(D_*L))\((1-w)*E - w*(D_*U));
% S = (D + w*L)\((1 - w)*D - w*U);
g = w*(E + w*D_*L)\(D_*b);
% g = w*(D + w*L)\b;

%% 迭代
% 初始迭代值
u_0 = zeros((N-1)^2,1);
% 准确值
u_e = zeros(N-1,1);
for i = 1:N-1
    u_e(i,1) = sin(pi*i/N)*sin(pi*1/N);
end
for j = 2:N-1
    u_ = zeros(N-1,1);
    for i = 1:N-1
        u_(i,1) = sin(pi*i/N)*sin(pi*j/N);
    end
    u_e = cat(1,u_e,u_);
end
% 开始计算
u = u_0;
n = 0; %迭代次数
ep = 10^(-6);
while n > -1
    u_1 = u;
    u = S*u_1 + g;
    e = norm(u - u_1,"inf");
    n = n+1;
    if e < ep
        break
    end
end
error = norm(u_e-u,"inf");
%% 画图 X:行; Y:列
% 准确值
x = h:h:(N-1)*h;
y = h:h:(N-1)*h;
[X,Y] = meshgrid(x,y);
U_e = sin(pi.*X).*sin(pi.*Y);
subplot(1,2,1);
surf(X,Y,U_e);
title('精确解');
xlabel('X');
ylabel('Y');
% 近似值
U = zeros(N-1,N-1);
for i = 1:N-1
    for j = 1 : N-1
        U(i,j) = u((N-1)*(i-1) + j,1);
    end
end
subplot(1,2,2);
surf(X,Y,U);
title('近似解');
xlabel('X');
ylabel('Y');
```

- Gauss-Seidel迭代：SOR迭代代码中的ω取1

- 计算矩阵S的谱半径：max(abs(eig(S)))