P72, 17: **A convection-diffusion problem.** Consider the following convection-diffusion problem on the unit square :



Where  is an integer,  . It has the exact solution . Apply the multigrid algorithms discussed in this chapter to this problem. Compare the algorithms and explore how their performance changes for 

In this problem, I use differential discretization to form a linear system, taking n equal divisions in both the x and y directions, then replacing  by a second-order center difference operator, and replacing the  by a first-order center difference operator.

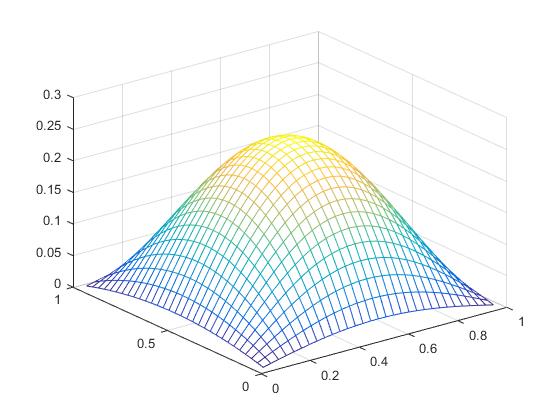
I use the weighted-Jacobi method in a V-cycle scheme on fine grids with n = 32, 64, and 128 points in each direction (three separate experiments). Full weighting and linear interpolation are used. Let e be the error vector, I compute its discrete L2 norm and infinite norm of the vector. Because the error is not available in most problems, a more practical measure, the discrete L2 norm of the residual r, is also computed. I give the table which shows the residual and error norms after each V-cycle.

At last, I will give the matlab code.

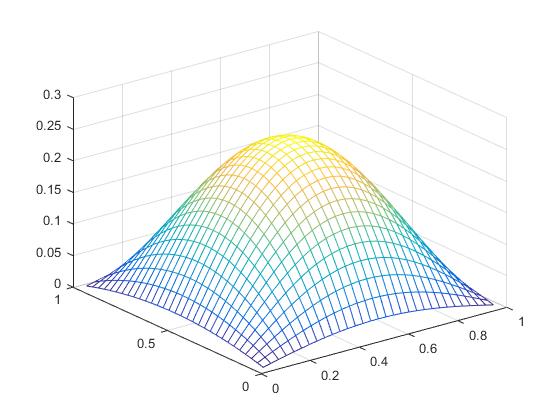
1. 

|  |  |  |  |
| --- | --- | --- | --- |
| Steps of V-cycle |  |  |  |
| 1 | 2.111984e+00 | 1.163057e-01 | 2.329643e-02 |
| 2 | 3.574124e-01 | 1.923691e-02 | 3.954598e-03 |
| 3 | 6.777309e-02 | 3.541326e-03 | 7.660276e-04 |
| 4 | 1.175971e-02 | 1.279108e-03 | 1.385563e-04 |
| 5 | 2.086696e-03 | 1.136819e-03 | 7.374864e-05 |
| 6 | 3.802497e-04 | 1.132452e-03 | 7.386537e-05 |
| 7 | 6.929293e-05 | 1.132455e-03 | 7.376959e-05 |
| 8 | 1.254780e-05 | 1.132477e-03 | 7.374640e-05 |
| 9 | 2.264476e-06 | 1.132480e-03 | 7.374371e-05 |
| 10 | 4.079199e-07 | 1.132481e-03 | 7.374361e-05 |
| 11 | 7.335516e-08 | 1.132481e-03 | 7.374364e-05 |
| 12 | 1.316921e-08 | 1.132481e-03 | 7.374365e-05 |
| 13 | 2.360751e-09 | 1.132481e-03 | 7.374365e-05 |
| 14 | 4.226358e-10 | 1.132481e-03 | 7.374365e-05 |
| 15 | 7.556707e-11 | 1.132481e-03 | 7.374365e-05 |
| 16 | 1.349478e-11 | 1.132481e-03 | 7.374365e-05 |
| 17 | 2.407026e-12 | 1.132481e-03 | 7.374365e-05 |

Exact solution:



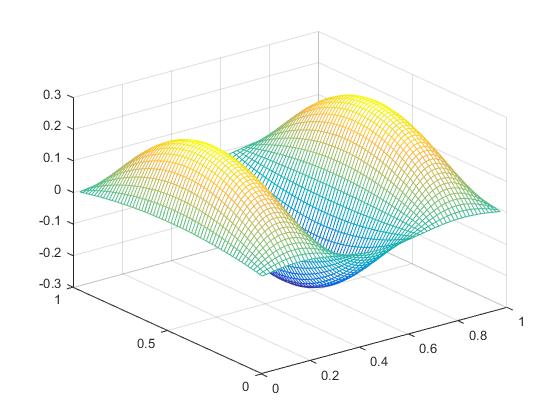
Numerical solution:



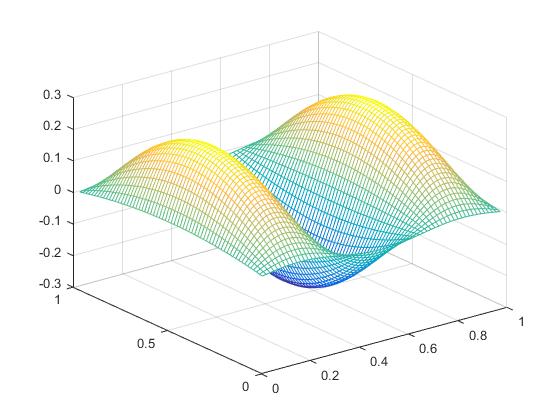
1. 

|  |  |  |  |
| --- | --- | --- | --- |
| Steps of V-cycle |  |  |  |
| 1 | 1.531692e+02 | 9.648222e-02 | 1.903317e-02 |
| 2 | 1.048895e+01 | 1.588883e-02 | 2.005691e-03 |
| 3 | 1.357651e+00 | 1.302887e-02 | 3.983443e-04 |
| 4 | 1.873682e-01 | 1.296126e-02 | 3.977234e-04 |
| 5 | 2.616840e-02 | 1.296003e-02 | 3.977222e-04 |
| 6 | 3.664964e-03 | 1.296004e-02 | 3.977221e-04 |
| 7 | 5.133800e-04 | 1.296004e-02 | 3.977221e-04 |
| 8 | 7.186471e-05 | 1.296004e-02 | 3.977221e-04 |
| 9 | 1.005044e-05 | 1.296004e-02 | 3.977221e-04 |
| 10 | 1.404185e-06 | 1.296004e-02 | 3.977221e-04 |
| 11 | 1.959957e-07 | 1.296004e-02 | 3.977221e-04 |
| 12 | 2.733243e-08 | 1.296004e-02 | 3.977221e-04 |
| 13 | 3.808468e-09 | 1.296004e-02 | 3.977221e-04 |
| 14 | 5.302579e-10 | 1.296004e-02 | 3.977221e-04 |
| 15 | 7.378605e-11 | 1.296004e-02 | 3.977221e-04 |
| 16 | 1.039928e-11 | 1.296004e-02 | 3.977221e-04 |
| 17 | 2.183764e-12 | 1.296004e-02 | 3.977221e-04 |

Exact solution:



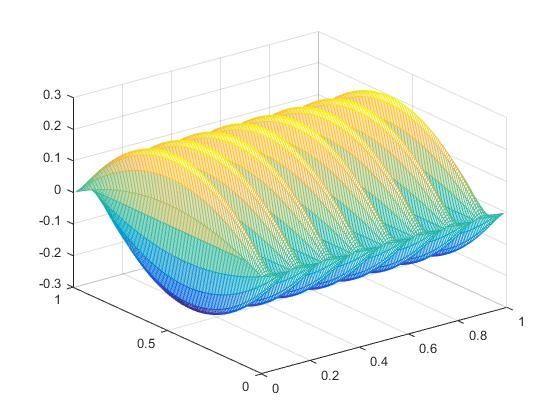
Numerical solution:



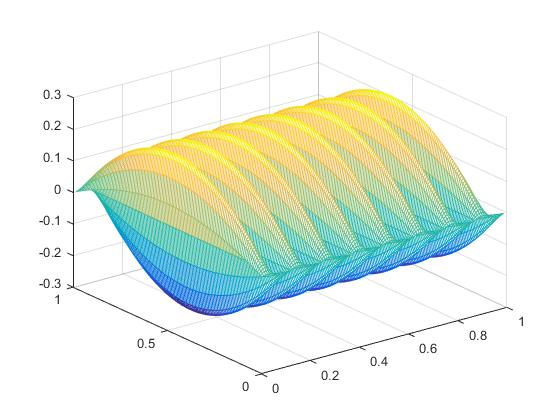
(3) 

|  |  |  |  |
| --- | --- | --- | --- |
| Steps of V-cycle |  |  |  |
| 1 | 4.375139e+04 | 2.002747e+00 | 5.983450e-02 |
| 2 | 6.424325e+03 | 2.126480e-01 | 1.504606e-02 |
| 3 | 1.227640e+03 | 1.927826e-01 | 3.239622e-03 |
| 4 | 2.298221e+02 | 2.103330e-01 | 3.229899e-03 |
| 5 | 4.144119e+01 | 2.127837e-01 | 3.227463e-03 |
| 6 | 7.357173e+00 | 2.130929e-01 | 3.227002e-03 |
| 7 | 1.305112e+00 | 2.131320e-01 | 3.226923e-03 |
| 8 | 2.328579e-01 | 2.131371e-01 | 3.226910e-03 |
| 9 | 4.184703e-02 | 2.131378e-01 | 3.226907e-03 |
| 10 | 7.567577e-03 | 2.131378e-01 | 3.226907e-03 |
| 11 | 1.374884e-03 | 2.131379e-01 | 3.226907e-03 |
| 12 | 2.505814e-04 | 2.131379e-01 | 3.226907e-03 |
| 13 | 4.576408e-05 | 2.131379e-01 | 3.226907e-03 |
| 14 | 8.368914e-06 | 2.131379e-01 | 3.226907e-03 |
| 15 | 1.531702e-06 | 2.131379e-01 | 3.226907e-03 |
| 16 | 2.804867e-07 | 2.131379e-01 | 3.226907e-03 |
| 17 | 5.137927e-08 | 2.131379e-01 | 3.226907e-03 |

Exact solution:



Numerical solution:



Matlab code:

**main.m:**

clc

clear

close all

format long e

epsilon = 0.01;

a = 0.1;

l = 1;

n = 32;

multigrid(epsilon, a, n, l);

fprintf('...............................\n');

%%%%%%%%%%%%%%%%%%%%%%%%

epsilon = 0.1;

a = 1;

l = 3;

n = 64;

multigrid(epsilon, a, n, l);

fprintf('...............................\n');

%%%%%%%%%%%%%%%%%%%%%%%%%

epsilon = 1;

a = 10;

l = 16;

n = 128;

multigrid(epsilon, a, n, l);

fprintf('...............................\n');

**multigrid.m:**

function multigrid(epsilon,a,n,l)

m = n;

u = exactSolution(l, m, n);

v0 = zeros((m-1)\*(n-1),1);

v = v0;

i = 0;

A = totalA(epsilon, a, m, n);

F = rightF(@right\_f, epsilon, a, l, m, n);

while(norm((u-v),inf)>(1e-6))

v = Vcycle(n, F, v0, epsilon, a);

v0 = v;

i = i+ 1;

if i>20

break;

end

fprintf('%e %e %e\n',norm(F-A\*v,2),norm(u-v,2),norm(u-v,inf));

end

plx =zeros(n-1);

ply =zeros(n-1);

for i=1:1:n-1

for j=1:1:n-1

plu(i,j) = u((i-1)\*(n-1)+j);

plv(i,j) = v((i-1)\*(n-1)+j);

ply(:,j) = [1/n:1/n:(n-1)\*1/n];

end

plx(i,:) = [1/n:1/n:(n-1)\*1/n]';

end

figure

mesh(plx, ply, plu);

figure

mesh(plx, ply, plv)

end

**exactSolution.m:**

function [ u ] = exactSolution(l, m, n)

hx = 1/m;

hy = 1/n;

u = zeros((m-1)\*(n-1),1);

for i=1:1:m-1

for j=1:1:n-1

u((i-1)\*(n-1)+j) = i\*hx\*(1-i\*hx)\*sin(l\*pi\*j\*hy);

end

end

end

**rightF.m:**

function [ F ] = rightF(right\_f, eplison, a, l, m, n )

hx = 1/m;

hy = 1/n;

F =zeros((m-1)\*(n-1), 1);

for i = 1: 1: m-1

for j = 1: 1: n-1

F((n-1)\*(i-1)+j) = right\_f(eplison, a, l, i\*hx, j\*hy);

end

end

end

function f = right\_f(eplison, a, l, x, y)

C1 = eplison\*l^2\*pi^2-2\*a;

C2 = -eplison\*l^2\*pi^2;

C0 = a+2\*eplison;

f = sin(l\*pi\*y)\*(C2\*x^2+C1\*x+C0);

end

**Vcycle.m:**

function [ v ] = Vcycle( n, F, v0 ,epsilon, a)

m = n;

A = totalA(epsilon, a, m, n);

v = dampedJacobi(A, F, v0, 5);

if n==4

return;

end

if n==8

r = F-A\*v;

r2h = I\_h\_2h(n)\*r;

v0 = zeros((m/2-1)\*(n/2-1),1);

e2h = Vcycle(n/2, r2h, v0,epsilon,a);

e = get\_I\_2h\_to\_h(n/2)\*e2h;

v = v+e;

end

if n==16

r = F-A\*v;

r2h = I\_h\_2h(n)\*r;

v0 = zeros((m/2-1)\*(n/2-1),1);

e2h = Vcycle(n/2, r2h, v0,epsilon,a);

e2h = Vcycle(n/2, r2h, e2h,epsilon,a);

e = get\_I\_2h\_to\_h(n/2)\*e2h;

v = v+e;

end

if n==64

r = F-A\*v;

r2h = I\_h\_2h(n)\*r;

v0 = zeros((m/2-1)\*(n/2-1),1);

e2h = Vcycle(n/2, r2h, v0,epsilon,a);

e2h = Vcycle(n/2, r2h, e2h,epsilon,a);

e = get\_I\_2h\_to\_h(n/2)\*e2h;

v = v+e;

end

r = F-A\*v;

r2h = I\_h\_2h(n)\*r;

v0 = zeros((m/2-1)\*(n/2-1),1);

e2h = Vcycle(n/2, r2h, v0,epsilon,a);

e = get\_I\_2h\_to\_h(n/2)\*e2h;

v = v+e;

end

function [ Ih ] = I\_h\_2h( n )

m = n;

Ih = zeros((m/2-1)\*(n/2-1), (m-1)\*(n-1));

for i=1: 1: n/2-1

for j=1: 1: n/2-1

Ih((n/2-1)\*(i-1)+j,((n-1)\*(2\*i-2)+2\*j)-1:((n-1)\*(2\*i-2)+2\*j)+1)=1/16\*[1 2 1];

Ih((n/2-1)\*(i-1)+j,((n-1)\*(2\*i-1)+2\*j)-1:((n-1)\*(2\*i-1)+2\*j)+1)=1/16\*[2 4 2];

Ih((n/2-1)\*(i-1)+j,((n-1)\*(2\*i)+2\*j)-1:((n-1)\*(2\*i)+2\*j)+1)=1/16\*[1 2 1];

end

end

end

function I\_2h\_to\_h = get\_I\_2h\_to\_h(n)

n1 = n - 1;

m1 = 2\*n-1;

I\_2h\_to\_h = zeros(m1\*m1,n1\*n1);

for p1 = 1:m1

for p2 = 1:m1

for q1 = 1:n1

for q2 = 1:n1

if p1/2 == q1 && p2/2 == q2

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1)\*n1+q2 ) = 1;

end

%%

if (p1-1)/2 == q1 && p2/2 == q2

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1 )\*n1+q2 ) = 1/2;

if q1 ~= n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1+1-1)\*n1+q2 ) = 1/2;

end

elseif (p1-1)/2 == 0 && p2/2 == q2

I\_2h\_to\_h( (p1-1)\*m1+p2, (0 +1-1)\*n1+q2 ) = 1/2;

end

%%

if p1/2 == q1 && (p2-1)/2 == q2

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1)\*n1+q2 ) = 1/2;

if q2 ~= n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1)\*n1+q2+1 ) = 1/2;

end

elseif p1/2 == q1 && (p2-1)/2 == 0

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1)\*n1+0 +1 ) = 1/2;

end

%%

if (p1-1)/2 == q1 && (p2-1)/2 == q2

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1 )\*n1+q2 ) = 1/4;

if q1 ~= n1 && q2 ~= n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1 )\*n1+q2+1) = 1/4;

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1+1-1)\*n1+q2 ) = 1/4;

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1+1-1)\*n1+q2+1) = 1/4;

elseif q2 ~= n1 && q2 == n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1+1-1)\*n1+q2 ) = 1/4;

elseif q1 == n1 && q2 ~= n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1-1 )\*n1+q2+1) = 1/4;

end

elseif (p1-1)/2 == 0 && (p2-1)/2 == q2

I\_2h\_to\_h( (p1-1)\*m1+p2, (0 +1-1 )\*n1+q2 ) = 1/4;

if q2 ~= n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (0 +1-1 )\*n1+q2+1) = 1/4;

end

elseif (p1-1)/2 == q1 && (p2-1)/2 == 0

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1 -1 )\*n1+0 +1) = 1/4;

if q1 ~= n1

I\_2h\_to\_h( (p1-1)\*m1+p2, (q1+1-1 )\*n1+0 +1) = 1/4;

end

elseif (p1-1)/2 == 0 && (p2-1)/2 == 0

I\_2h\_to\_h( (p1-1)\*m1+p2, (0 +1-1 )\*n1+0 +1) = 1/4;

end

end

end

end

end

I\_2h\_to\_h = sparse(I\_2h\_to\_h);

end

**totalA.m:**

function [ A ] = totalA(epsilon, a, m, n)

hx = 1/m;

hy = 1/n;

h = hx;

B\_a = -epsilon/(h^2); B\_b = 4\*epsilon/(h^2); B\_c = -epsilon/(h^2);

C\_a = -epsilon/(h^2)-a/(2\*h);

D\_a = -epsilon/(h^2)+a/(2\*h);

B = diagonalB(B\_a, B\_b, B\_c, n);

C = lowBlockC(C\_a, n);

D = upBlockD(D\_a, n);

A = zeros((m-1)\*(n-1));

for i = 2: 1: m-2

row = -(i-1)\*(n-1)-1+ i\*(n-1);

column = -(i-2)\*(n-1)-1+ (i+1)\*(n-1);

A((i-1)\*(n-1)+1: i\*(n-1), (i-2)\*(n-1)+1: (i+1)\*(n-1)) = [C B D];

end

A(1: (n-1), 1: 2 \* (n-1)) = [B, D];

A((m-2)\*(n-1)+1: (m-1)\*(n-1), (m-3)\*(n-1)+1: (m-1)\*(n-1))=[C B];

end

function [ D ] = upBlockD( a, n )

D = a \* eye(n-1);

end

function [ C ] = lowBlockC( a, n )

C = a \* eye(n-1);

end

function [ B ] = diagonalB( a, b, c, n )

B=zeros(n-1,n-1);

for i = 2: 1: n-2

B(i, i-1: i+1) = [a, b, c];

end

B(1, 1: 2) = [b c];

B(n-1, n-2: n-1) = [a, b];

end

**dampedJacobi.m:**

function [ v ] = dampedJacobi( A, f, v0, k)

%[v,flag,relres,iter,resvec] = bicgstab(A,f,1e-6,k,[],[],v0);

[row, column] = size(A);

D = diag(diag(A));

L = tril(A,-1);

U = triu(A,1);

H = inv(D);

Rj = H\*(-L-U);

w = 2/3;

G = (1-w)\*eye(row);

for i = 1: 1: k

v = (G+w\*Rj)\*v0+w\*H\*f;

v0 = v;

end

end