Table of Contents

Jacobi

```
clear all
a=0.1;
b=1;
i0=1;
kap=3;
n=20;
h=(b-a)/(n-1);
k=pi/2/(n-1);
beta=h^2/k^2;
a_mat=zeros((n-1)^2);
b_ary=zeros((n-1)^2,1);
b_ary_1=zeros((n-1)^2,1);
b_ary_2=zeros((n-1)^2,1);
a_var=100;
u_1=ones(n,n-1);
ct_j=0;
err_j=1;
while err_j>=1e-5
    ct_j=ct_j+1;
    count=0;
    for i=1:n
        for j=1:n-1
            count=count+1;
            theta=(i-1)*k;
            r=(j-1)*h+a;
            r_ray(count)=r;
            theta=(i-1)*((pi/2)/(n-1));
            theta_ray(count)=theta;
            b_0=2*(1+beta/r^2);
            b_1=beta/r^2;
            b_2=beta/r^2;
            b_3=(1+h/(2*r));
            b_4=(1-h/(2*r));
            if i==1 %boundry C
                if j==1 %corner BC
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+2*u_1(i,j+1));
                elseif j==(n-1) % corner CD
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_1(i,j-1)+(1+h/(2*r))*(-i0*b*cos(3.*theta)));
                else
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_1(i,j-1)+(1+h/(2*r))*u_1(i,j+1));
                end
            end
            if j==(n-1) %boundry D
                if i==1 %corner CD
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_1(i,j-1)+(1+h/(2*r))*(-i0*b*cos(3.*theta)));
                elseif i== n %corner AD
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*((1-h/(2*r))*u_1(i,j-1)+2*b_1*u_1(i-1,j)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)));
                else
                    u_2(i,j)=(1/b_0)*(b_1*u_1(i
+1,j)+b_2*u_1(i-1,j)+(1-h/(2*r))*u_1(i,j-1)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)));
                end
            end
            if j==1 %boundry B
                if i==1 %corner BC
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+2*u_1(i,j+1));
                elseif i==n %corner AB
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_1(i-1,j)+2*u_1(i,j+1));
                else
                    u_2(i,j)=(1/b_0)*(b_1*u_1(i+1,j)+2*u_1(i,j
+1)+b_2*u_1(i-1,j));
                end
            end
            if i==n %boundry A
                if j==1 %corner AB
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_1(i-1,j)+2*u_1(i,j+1));
                elseif j==(n-1) %corner AD
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*((1-h/(2*r))*u_1(i,j-1)+2*b_1*u_1(i-1,j)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)));
                else
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_1(i-1,j)+(1-h/(2*r))*u_1(i,j-1)+(1+h/(2*r))*u_1(i,j
+1));
                end
            end
            if i~=1 && i~=n && j~=1 && j~=(n-1)
                u_2(i,j)=(1/b_0)*(b_1*u_1(i+1,j)+b_2*u_1(i-1,j)+(1-h/
(2*r))*u_1(i,j-1)+(1+h/(2*r))*u_1(i,j+1));
            end
        end
    end
    err_j=abs(mean(mean(u_2-u_1))/mean(mean(u_2)));
    co_j(ct_j)=err_j;
    lamb_j(ct_j)=mean(mean(u_2-u_1));
    rms_j(ct_j)=rms(rms(abs(u_2-u_1)));
    u_1=u_2;
end
sr_j=mean(lamb_j(2:end)./lamb_j(1:end-1));
con_j=log(co_j(end-1))/log(co_j(end));

[x_ray,y_ray]=pol2cart(theta_ray,r_ray);
x_mat=reshape(x_ray,[n-1,(n)]);
y_mat=reshape(y_ray,[n-1,(n)]);
```

analytic solution

```
analy= @(th,rr) -(rr.^3+0.1.^6.*rr.^-3)./(1+0.1.^6).*cos(3.*th);
ana_u=analy(theta_ray,r_ray);
ana_mat=reshape(ana_u,[n-1,(n)]);
```

contour plot

```
figure()
pcolor(x_mat,y_mat,u_2')
hold on
contour(x_mat,y_mat,u_2')
figure()
pcolor(x_mat,y_mat,ana_mat)
```

Gauss_Seidel

```
clearvars -except ct_j lamb_j sr_j rms_j con_j
a=0.1;
b=1;
i0=1;
kap=3;
n=20;
h=(b-a)/(n-1);
k=pi/2/(n-1);
beta=h^2/k^2;
a_mat=zeros((n-1)^2);
b_ary=zeros((n-1)^2,1);
b_ary_1=zeros((n-1)^2,1);
b_ary_2=zeros((n-1)^2,1);
a_var=100;
u_1=ones(n,n-1);
ct_gs=0;
err_gs=1;
while err_gs>=1e-5
    ct_gs=ct_gs+1;
    count=0;
    for i=1:n
        for j=1:n-1
            count=count+1;
            theta=(i-1)*k;
            r=(j-1)*h+a;
            r_ray(count)=r;
            theta=(i-1)*((pi/2)/(n-1));
            theta_ray(count)=theta;
            b_0=2*(1+beta/r^2);
            b_1=beta/r^2;
            b_2=beta/r^2;
            b_3=(1+h/(2*r));
            b_4=(1-h/(2*r));
            if i==1 %boundry C
                if j==1 %corner BC
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+2*u_1(i,j+1));
                elseif j==(n-1) % corner CD
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*(-i0*b*cos(3.*theta)));
                else
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*u_1(i,j+1));
                end
            end
            if j==(n-1) %boundry D
                if i==1 %corner CD
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*(-i0*b*cos(3.*theta)));
                elseif i== n %corner AD
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*((1-h/(2*r))*u_2(i,j-1)+2*b_1*u_2(i-1,j)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)));
                else
                    u_2(i,j)=(1/b_0)*(b_1*u_1(i
+1,j)+b_2*u_2(i-1,j)+(1-h/(2*r))*u_2(i,j-1)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)));
                end
            end
            if j==1 %boundry B
                if i==1 %corner BC
                    u_2(i,j)=(1/b_0)*(2*b_1*u_1(i+1,j)+2*u_1(i,j+1));
                elseif i==n %corner AB
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_2(i-1,j)+2*u_1(i,j+1));
                else
                    u_2(i,j)=(1/b_0)*(b_1*u_1(i+1,j)+2*u_1(i,j
+1)+b_2*u_2(i-1,j));
                end
            end
            if i==n %boundry A
                if j==1 %corner AB
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_2(i-1,j)+2*u_1(i,j+1));
                elseif j==(n-1) %corner AD
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*((1-h/(2*r))*u_2(i,j-1)+2*b_1*u_2(i-1,j)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)));
```

```
                else
                    u_2(i,j)=(1/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_2(i-1,j)+(1-h/(2*r))*u_2(i,j-1)+(1+h/(2*r))*u_1(i,j
+1));
                end
            end
            if i~=1 && i~=n && j~=1 && j~=(n-1)
                u_2(i,j)=(1/b_0)*(b_1*u_1(i+1,j)+b_2*u_2(i-1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*u_1(i,j+1));
            end
        end
    end
    err_gs=abs(mean(mean(u_2-u_1))/mean(mean(u_2)));
    co_gs(ct_gs)=err_gs;
    lamb_gs(ct_gs)=mean(mean(u_2-u_1));
    rms_gs(ct_gs)=rms(rms(abs(u_2-u_1)));
    u_1=u_2;
end
sr_gs=mean(lamb_gs(2:end)./lamb_gs(1:end-1));
con_gs=log(co_gs(end-1))/log(co_gs(end));

[x_ray,y_ray]=pol2cart(theta_ray,r_ray);
x_mat=reshape(x_ray,[n-1,(n)]);
y_mat=reshape(y_ray,[n-1,(n)]);
```

analytic solution

```
analy= @(th,rr) -(rr.^3+0.1.^6.*rr.^-3)./(1+0.1.^6).*cos(3.*th);
ana_u=analy(theta_ray,r_ray);
ana_mat=reshape(ana_u,[n-1,(n)]);
```

contour plot

```
figure()
pcolor(x_mat,y_mat,u_2')
hold on
contour(x_mat,y_mat,u_2')
figure()
pcolor(x_mat,y_mat,ana_mat)
```

Optimal SOR

```
clearvars -
except ct_j lamb_j sr_j ct_gs lamb_gs sr_gs rms_j rms_gs con_j con_gs
w=2/(1+sqrt(1-sr_gs));
%w=1;
a=0.1;
b=1;
i0=1;
kap=3;
n=20;
h=(b-a)/(n-1);
```

```
k=pi/2/(n-1);
beta=h^2/k^2;
a_mat=zeros((n-1)^2);
b_ary=zeros((n-1)^2,1);
b_ary_1=zeros((n-1)^2,1);
b_ary_2=zeros((n-1)^2,1);
a_var=100;
u_1=ones(n,n-1);
ct_sor=0;
err_sor=1;
while err_sor>=1e-5
    ct_sor=ct_sor+1;
    count=0;
    for i=1:n
        for j=1:n-1
            count=count+1;
            theta=(i-1)*k;
            r=(j-1)*h+a;
            r_ray(count)=r;
            theta=(i-1)*((pi/2)/(n-1));
            theta_ray(count)=theta;
            b_0=2*(1+beta/r^2);
            b_1=beta/r^2;
            b_2=beta/r^2;
            b_3=(1+h/(2*r));
            b_4=(1-h/(2*r));
            if i==1 %boundry C
                if j==1 %corner BC
                    u_2(i,j)=(w/b_0)*(2*b_1*u_1(i+1,j)+2*u_1(i,j
+1))+(1-w)*u_1(i,j);
                elseif j==(n-1) % corner CD
                    u_2(i,j)=(w/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*(-i0*b*cos(3.*theta)))+(1-w)*u_1(i,j);
                else
                    u_2(i,j)=(w/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*u_1(i,j+1))+(1-w)*u_1(i,j);
                end
            end
            if j==(n-1) %boundry D
                if i==1 %corner CD
                    u_2(i,j)=(w/b_0)*(2*b_1*u_1(i+1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*(-i0*b*cos(3.*theta)))+(1-w)*u_1(i,j);
                elseif i== n %corner AD
                    u_2(i,j)=(w/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*((1-h/(2*r))*u_2(i,j-1)+2*b_1*u_2(i-1,j)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)))+(1-w)*u_1(i,j);
                else
                    u_2(i,j)=(w/b_0)*(b_1*u_1(i
+1,j)+b_2*u_2(i-1,j)+(1-h/(2*r))*u_2(i,j-1)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)))+(1-w)*u_1(i,j);
                end
            end
            if j==1 %boundry B
                if i==1 %corner BC
```

```
          u_2(i,j)=(w/b_0)*(2*b_1*u_1(i+1,j)+2*u_1(i,j
+1))+(1-w)*u_1(i,j);
        elseif i==n %corner AB
          u_2(i,j)=(w/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_2(i-1,j)+2*u_1(i,j+1))+(1-w)*u_1(i,j);
        else
          u_2(i,j)=(w/b_0)*(b_1*u_1(i+1,j)+2*u_1(i,j
+1)+b_2*u_2(i-1,j))+(1-w)*u_1(i,j);
        end
      end
      if i==n %boundry A
        if j==1 %corner AB
          u_2(i,j)=(w/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_2(i-1,j)+2*u_1(i,j+1))+(1-w)*u_1(i,j);
        elseif j==(n-1) %corner AD
          u_2(i,j)=(w/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*((1-h/(2*r))*u_2(i,j-1)+2*b_1*u_2(i-1,j)+(1+h/(2*r))*(-
i0*b*cos(3.*theta)))+(1-w)*u_1(i,j);
        else
          u_2(i,j)=(w/(2*(1+beta/r^2)+2*k*a_var*beta/
r^2))*(2*b_1*u_2(i-1,j)+(1-h/(2*r))*u_2(i,j-1)+(1+h/(2*r))*u_1(i,j
+1))+(1-w)*u_1(i,j);
        end
      end
      if i~=1 && i~=n && j~=1 && j~=(n-1)
        u_2(i,j)=(w/b_0)*(b_1*u_1(i+1,j)+b_2*u_2(i-1,j)+(1-h/
(2*r))*u_2(i,j-1)+(1+h/(2*r))*u_1(i,j+1))+(1-w)*u_1(i,j);
      end
    end
  end
  err_sor=abs(mean(mean(u_2-u_1))/mean(mean(u_2)));
  co_sor(ct_sor)=err_sor;
  lamb_sor(ct_sor)=mean(mean(u_2-u_1));
  rms_sor(ct_sor)=rms(rms(abs(u_2-u_1)));
  u_1=u_2;
end
sr_sor=mean(lamb_sor(2:end)./lamb_sor(1:end-1));
con_sor=log(co_sor(end-1))/log(co_sor(end));

[x_ray,y_ray]=pol2cart(theta_ray,r_ray);
x_mat=reshape(x_ray,[n-1,(n)]);
y_mat=reshape(y_ray,[n-1,(n)]);
```

analytic solution

```
analy= @(th,rr) -(rr.^3+0.1.^6.*rr.^-3)./(1+0.1.^6).*cos(3.*th);
ana_u=analy(theta_ray,r_ray);
ana_mat=reshape(ana_u,[n-1,(n)]);
```

contour plot

```
figure()
pcolor(x_mat,y_mat,u_2')
```

```
hold on
contour(x_mat,y_mat,u_2')
figure()
pcolor(x_mat,y_mat,ana_mat)
```

RMS plot

```
figure()
semilogy(rms_j,'linewidth',2)
hold on
semilogy(rms_gs,'linewidth',2)
semilogy(rms_sor,'linewidth',2)
title('RMS error Low A, N=40','fontsize',25)
xlabel('Iteration number','fontsize',20)
ylabel('Error','fontsize',20)
legend({'Jacobi','Gauss seidel','SOR'},'fontsize',20)
hold off
```

Published with MATLAB® R2015a