Contents

Heat_equation

```
%- Program to solve the diffusion equation
% using the Backward Euler method
```

Parameters

```
clear all;close all;clc;
saveFigurePath = ['/Users/kevin/SkyDrive/KTH Work/' ...
'Period 3 2014/DN2255/Homework/1/Heat Equation/Figures/'];
N = 100; % Number of grid points
L = 1; % The system extends from (x)=(0) to (x)=(L)
h = L/N;
i = 1:(N); % 1:N
```

```
[x,y] = meshgrid(h/2:h:L,h/2:h:L);
w = 0.2;
xs = 0.5;
ys = 0.5;
tfinal = .5;
tau = .005*h;
coeff = tau/h^2;
tsteps = ceil(tfinal/tau);
time = linspace(0,tfinal,tsteps);
```

Initialize Source function

```
xExponent = (x-xs).^2;
yExponent = (y-ys).^2;
S = exp(-(xExponent)/w^2).*exp(-yExponent/w^2);
deltaFunction = zeros(N);
deltaFunction(round(N/2),round(N/2))=2;
S = deltaFunction;
S = reshape(S,[N^2,1]);
```

Initialize Q-matrix

```
Q = zeros(N^2,1);
```

Compute matrix A

```
TN = 2*eye(N) - diag(ones(N-1,1),1) - diag(ones(N-1,1),-1);
TNxN = kron(eye(N),TN) + kron(TN,eye(N));
dM = eye(N^2) + coeff*TNxN;
sparsedM = sparse(dM);
```

Initialize loop and plot variables

```
Qplot(:,1) = Q; % initial value
stepNumber=round(.25/tau);
```

Main loops

```
for iter=1:stepNumber
    Q = sparsedM\Q + S;
    Qplot(:,iter+1) = Q(:);
end
% Loop after source is gone
for iter2=(iter+2):tsteps
    Q = sparsedM\Q;
    Qplot(:,iter2) = Q(:);
end
```

Reshape Q for plotting

```
Qresh = reshape(Qplot,[N,N,tsteps])
```

look at dx*dy*Qij for Conservation

```
cons(1:tsteps,1) = h^2*sum(sum(Qresh(:,:,1:end)));
figure(1);
plot(tau*(1:length(cons)),cons);
tL=title('\Deltax \Deltay Q_{i,j} vs time');
xL = xlabel('time (sec)');
yL = ylabel('\Deltax \Deltay Q_{i,j}');
    set( gca                       , ...
```

```
        'FontName'   , 'Helvetica' );
    set([tL,xL,yL], ...
        'FontName'   , 'AvantGarde');
    set( gca             , ...
        'FontSize'   , 8           );
    set([xL,yL]  , ...
        'FontSize'   , 10          );
    set( tL                    , ...
        'FontSize'   , 12          , ...
        'FontWeight' , 'bold'      );
    set(gca, ...
        'Box'         , 'off'         , ...
        'TickDir'     , 'out'         , ...
        'TickLength'  , [.02 .02]     , ...
        'XMinorTick'  , 'on'          , ...
        'YMinorTick'  , 'on'          , ...
        'XColor'      , [.3 .3 .3]    , ...
        'YColor'      , [.3 .3 .3]    , ...
        'LineWidth'   , 1             );
    printYesNo1 = 1;
if printYesNo1 == 1
set(figure(1), 'PaperPositionMode', 'auto');
print('-depsc2', [saveFigurePath ...
    sprintf('deltaConservationPlot')]);
end
```

figure(1);clf; for i = 1:tsteps surf(x,y,Qresh(:,:,i)) title(sprintf('%g',time(i))); axis([0 1 0 1 0 max(max(Qplot))]); hold off; pause(0.02) end

Print Plots

```
fin = length(Qplot(1,:));
maxZ = max(max(max(Qresh)));
%     n1 = [1 ceil(.25/(2*tau)) ceil(.25/tau)...
%        ceil(.25/tau)+5 ceil(.25/tau)+20 ceil(.25/tau)+50];
```

```
n1 = [1 floor(fin*6/24) floor(fin*12/24)...
    floor(fin*13/24) floor(fin*14/24) floor(fin*15/24)];
plotMyFigure(L, N, Q, Qplot, Qresh, S, TN,  ...
TNxN, coeff, cons, dM,  ...
deltaFunction, fin, h, i, iter, iter2,  ...
maxZ, n1, sparsedM, stepNumber, tau,  ...
tfinal, time, tsteps, w, x, xExponent,  ...
xL, xs, y, yExponent, yL, ys)
```

Plot 1

```
printYesNo = 0;
if printYesNo == 1
set(figure(2), 'PaperPositionMode', 'auto');
print('-depsc2', [saveFigurePath ...
    sprintf('deltaFunctionPlot')]);
end
```

Published with MATLAB® R2013a