Name: Senthamilaruvi Moorthy

CWID: 10722805

HW : LAB 8

TABLE OF CONTENTS

|  |  |
| --- | --- |
| Content | Page Number |
| 1. Summary Problem 1 | 1 |
| 2. Summary Problem 2 | 5 |
| 1. Summary Problem 3a, 3b and 3c | 7 |
| 1. Program Problem 1 | 10 |
| 1. Program Problem 2 | 12 |
| 1. Program Problem 3 | 15 |

1

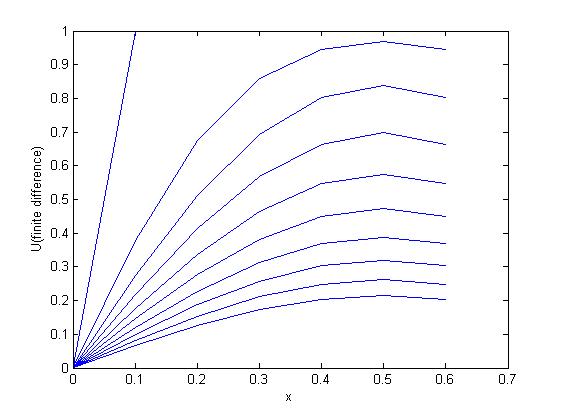
1. Solution for the given differential equation was calculated using explicit finite difference approach
2. The initial and boundary conditions were fed and the corresponding row and columns of solution matrix (U ) were populated.
3. We know that the standard equation for finite difference method is

U(I,j+1) = r\*U(i+1,j +(1-2r)\*U(I,j)+r\*U(i-1,j)

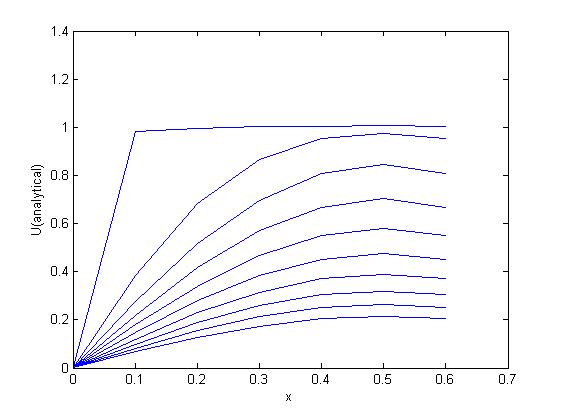
1. The other elements of solution matrix were populated using above equation.
2. Analytical solution was obtained using the given solution.

The Solution for U by finite difference method was plotted against x =0,0.1,0.2 , 0.3,0.4 and 0.5

For the time inter t = 0 to t = 0.200 as shown below . The graph was plotted for selected values of t for clarity



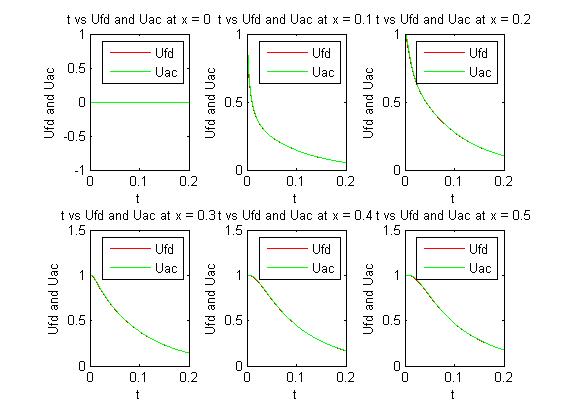
The analytical solution was obtained and plotted in the same way.



The error plot was made between analytical solution and finite difference solution (shown below) and found the error was minimum.

Ufd – Finite difference solution

Uac- Analytical solution



2

1. Solution for the given differential equation was calculated using implicit finite difference approach
2. The initial and boundary conditions were fed and the corresponding row and columns of solution matrix (U) were populated.
3. We know that the standard equation for Implicit finite difference method is of the form

AX = B

1. Where A is of the form. The size of A matrix differs based on the number of nodes along x direction

A = 2\*(1+r) -r 0 0

-r 2\*(1+r) -r 0

0 -r 2\*(1+r) -r

0 0 -r 2\*(1+r)

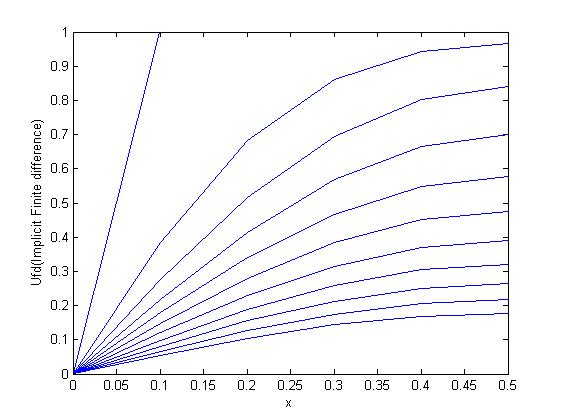
And values of column matrix B is given by

B = B(i,j)= r\*U(i,j)+2\*(1- r)\*U(i+1,j)+r\*Ufd(i+2,j);

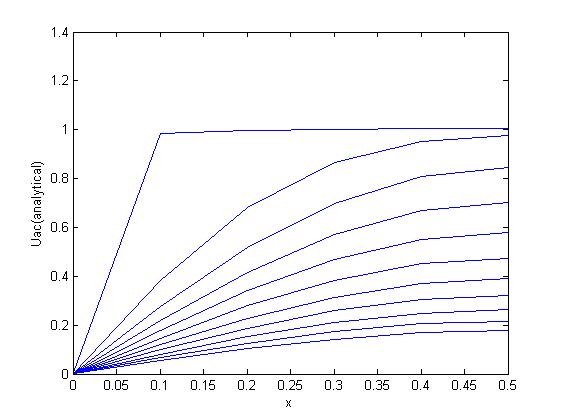
1. The solution matrix X was obtained column by column and stored in final solution matrix of U
2. Analytical solution was obtained using the given solution.

The Solution for U by finite difference method was plotted against x =0,0.1,0.2 , 0.3,0.4 and 0.5

For the time inter t = 0 to t = 0.200 as shown below . The graph was plotted for selected values of t for clarity



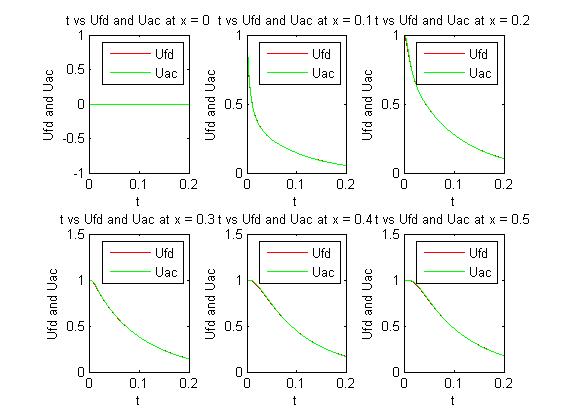
The analytical solution was obtained and plotted in the same way.



The error plot was made between analytical solution and finite difference solution (shown below) and found the error was minimum.

Ufd – Finite difference solution

Uac- Analytical solution



3.

1. Solution for the given differential equation was calculated using explicit finite difference approach
2. The initial condition was fed and the corresponding row and columns of solution matrix (U ) were populated.
3. We know that the standard equation for finite difference method is

U(I,j+1) = r\*U(i+1,j +(1-2r)\*U(I,j)+r\*U(i-1,j) which holds good for domain.

1. Since the boundary conditions are different the solution for U at the line x =0 and x = N were given by the following equations

At x = 0

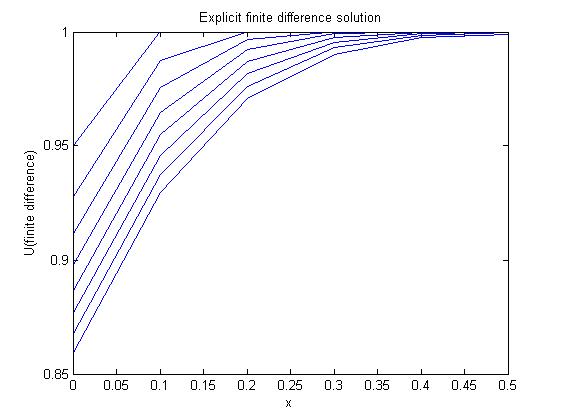
U(i,j+1) = U(i,j) + 2\*r\*(U(i+1,j)-(1+h)\*U(i,j));

At x = N

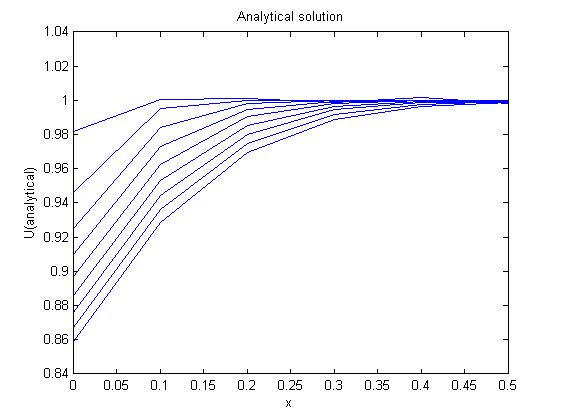
U(i,j+1) = U(i,j) + 2\*r\*(U(i-1,j)-(1+h)\*U(i,j));

1. The solution of U was obtained by the above three equations
2. Analytical solution was obtained using the given solution.

The Solution for U by finite difference method was plotted against x =0,0.1,0.2 , 0.3,0.4 and 0.5. For the time interval t = 0 to t = 0.020 as shown below . The graph was plotted for all time steps in interval



The analytical solution was obtained and plotted in the same way.



Solution in tabular form

|  |  |
| --- | --- |
| t | U |
| 0.0000 | 1.0000 |
| 0.0025 | 1.0000 |
| 0.0050 | 1.0000 |
| 0.0075 | 0.9969 |
| 0.0100 | 0.9923 |
| 0.0125 | 0.9872 |
| 0.0150 | 0.9818 |
| 0.0175 | 0.9762 |
| 0.0200 | 0.9708 |
| 1.000 | 0.1786 |

3 B Solution obtained by finite difference method

At x = 0.00 At x =0.10 At x =0.20

|  |  |
| --- | --- |
| t | U |
| 0.0000 | 1.0000 |
| 0.0025 | 0.9500 |
| 0.0050 | 0.9275 |
| 0.0075 | 0.9111 |
| 0.0100 | 0.8978 |
| 0.0125 | 0.8864 |
| 0.0150 | 0.8764 |
| 0.0175 | 0.8673 |
| 0.0200 | 0.8590 |
| 1.0000 | 0.1534 |

|  |  |
| --- | --- |
| t | U |
| 0.0000 | 1.0000 |
| 0.0025 | 1.0000 |
| 0.0050 | 0.9875 |
| 0.0075 | 0.9756 |
| 0.0100 | 0.9648 |
| 0.0125 | 0.9549 |
| 0.0150 | 0.9459 |
| 0.0175 | 0.9375 |
| 0.0200 | 0.9296 |
| 1.0000 | 0.167 |

At x = 0.30 At x = 0.40 At x = 0.50

|  |  |
| --- | --- |
| t | U |
| 0.000 | 1.0000 |
| 0.003 | 1.0000 |
| 0.005 | 1.0000 |
| 0.008 | 1.0000 |
| 0.010 | 0.9992 |
| 0.013 | 0.9977 |
| 0.015 | 0.9956 |
| 0.018 | 0.9931 |
| 0.020 | 0.9902 |
| 1.000 | 0.1867 |

|  |  |
| --- | --- |
| t | U |
| 0.000 | 1.0000 |
| 0.003 | 1.0000 |
| 0.005 | 1.0000 |
| 0.008 | 1.0000 |
| 0.010 | 1.0000 |
| 0.013 | 0.9998 |
| 0.015 | 0.9993 |
| 0.018 | 0.9985 |
| 0.020 | 0.9974 |
| 1.000 | 0.1917 |

|  |  |
| --- | --- |
| t | U |
| 0.000 | 1.0000 |
| 0.003 | 1.0000 |
| 0.005 | 1.0000 |
| 0.008 | 1.0000 |
| 0.010 | 1.0000 |
| 0.013 | 1.0000 |
| 0.015 | 0.9999 |
| 0.018 | 0.9996 |
| 0.020 | 0.9991 |
| 1.000 | 0.1933 |

3 C. Comparison of analytical and finite difference solution at x=0.20

|  |  |  |
| --- | --- | --- |
| t | Uac | Ufd |
| 0.0000 | 1.0010 | 1.0000 |
| 0.0025 | 0.9999 | 1.0000 |
| 0.0050 | 0.9984 | 1.0000 |
| 0.0075 | 0.9950 | 0.9969 |
| 0.0100 | 0.9905 | 0.9923 |
| 0.0125 | 0.9855 | 0.9872 |
| 0.0150 | 0.9802 | 0.9818 |
| 0.0175 | 0.9748 | 0.9762 |
| 0.0200 | 0.9695 | 0.9708 |

**Program 1**

% This script is to find the approximate solution for differential

% equation using explicit finite difference method and compare the

% solution with actual solution

% Defining r,k,h,x,t and n

clear

clc

r = 0.1;

k = 0.001;

h = sqrt(k/(r));

x = 0:h:1;

t = 0:k:0.200;

c = zeros(1,100);

d = zeros(1,100);

e = zeros(1,100);

Uac = zeros(length(x),length(t));

% Initializing U

Ufd = zeros(length(x),length(t));

Ufd(:,1) = 1;

Ufd(1,:) = 0;

Ufd(11,:) = 0;

%Explicit finite difference solution

for j = 1:200

for i = 2:10

Ufd(i,(j+1)) = (r\*(Ufd((i+1),j)) + r\*(Ufd((i-1),j)) + (1-(2\*r))\*Ufd(i,j));

end

end

% Analytical solution

sum = 0;

for i = 1:7

for j = 1:201

sum = 0;

for n = 0:50

c = (2\*n +1);

sum = sum + (4/pi)\*(1/c)\*exp(-c^2\*(pi^2)\*t(1,j))\*sin(c\*(pi)\*x(1,i));

end

Uac(i,j) = sum;

end

end

figure(01)

%Plotting Ufd and Uac vs t at x = 0

subplot(2,3,1)

j = 1:201;

plot(t,Ufd(1,j),'r-')

hold on

plot(t,Uac(1,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

legend('Ufd','Uac')

title('t vs Ufd and Uac at x = 0')

%Plotting Ufd and Uac vs t at x = 0.1

subplot(2,3,2)

j = 1:201;

plot(t,Ufd(2,j),'r-')

hold on

plot(t,Uac(2,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.1')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.2

subplot(2,3,3)

j = 1:201;

plot(t,Ufd(3,j),'r-')

hold on

plot(t,Uac(3,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.2')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.3

subplot(2,3,4)

j = 1:201;

plot(t,Ufd(4,j),'r-')

hold on

plot(t,Uac(4,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.3')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.4

subplot(2,3,5)

j = 1:201;

plot(t,Ufd(5,j),'r-')

hold on

plot(t,Uac(5,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.4')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.5

subplot(2,3,6)

j = 1:201;

plot(t,Ufd(6,j),'r-')

hold on

plot(t,Uac(6,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.5')

legend('Ufd','Uac')

%Plotting Ufd at different given values of x at different time

figure(02)

x1 = 0:0.1:0.6;

j = 1:7;

for m = 1:20:200

plot(x1,Ufd(j,m))

hold on

end

xlabel('x')

ylabel('U(finite difference)')

figure(03)

for t1 = 1:20:200

plot(x1,Uac(j,t1))

hold on

end

xlabel('x')

ylabel('U(analytical)')

**Program 2**

% This script is for finding approximat solution for differential equation

% using Crank- Nicolson (Implicit) finite difference method

clear

clc

r = 0.1;

k = 0.001;

h = sqrt(k/(r));

x = 0:h:1;

t = 0:k:0.200;

% First and last row of finite difference matrix and the first column

% are known from given boundary conditions so remaining 9 rows and 200

% columns of finite difference matrix should be populated and hence

% the other required matrix are initialized correspondingly

A = zeros(length(x)-2,length(x)-2);

B = zeros(length(x)-2,length(t)-1);

Y = zeros(length(x)-2,1);

Uac = zeros(length(x),length(t));

%Initializing U

Ufd = zeros(length(x),length(t));

Ufd(:,1) = 1;

Ufd(1,:) = 0;

Ufd(11,:) = 0;

for j = 1:9

for i = 1:9

if i == j

A(i,j) = 2\*(1+r);

elseif i == j-1 || i == j+1

A(i,j) = -r;

else

A(i,j) = 0;

end

end

end

% First and last row of finite difference matrix and the first column

% are known from given boundary conditions so remaining 9 rows and 200

% columns of finite difference matrix is populated

for j = 1:200

for i = 1:9

B(i,j) = r\*Ufd(i,j) + 2\*(1-r)\*Ufd(i+1,j) + r\*Ufd(i+2,j);

Y(i,1) = B(i,j);

end

X = A\Y;

for v = 1:9

Ufd(v+1,j+1) = X(v,1);

end

end

% Analytical solution

sum = 0;

for i = 1:6

for j = 1:201

sum = 0;

for n = 0:50

c = (2\*n +1);

sum = sum + (4/pi)\*(1/c)\*exp(-c^2\*(pi^2)\*t(1,j))\*sin(c\*(pi)\*x(1,i));

end

Uac(i,j) = sum;

end

end

figure(01)

%Plotting x and Ufd

x1 = 0:0.1:0.5;

u = 1:6;

for m = 1:20:201;

plot(x1,Ufd(u,m))

hold on

end

xlabel('x');

ylabel('Ufd(Implicit Finite difference)')

%plotting x and Uac

figure(02)

x1 = 0:0.1:0.5;

u = 1:6;

for m = 1:20:201;

plot(x1,Uac(u,m))

hold on

end

xlabel('x');

ylabel('Uac(analytical)')

figure(03)

%Plotting Ufd and Uac vs t at x = 0

subplot(2,3,1)

j = 1:201;

plot(t,Ufd(1,j),'r-')

hold on

plot(t,Uac(1,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

legend('Ufd','Uac')

title('t vs Ufd and Uac at x = 0')

%Plotting Ufd and Uac vs t at x = 0.1

subplot(2,3,2)

j = 1:201;

plot(t,Ufd(2,j),'r-')

hold on

plot(t,Uac(2,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.1')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.2

subplot(2,3,3)

j = 1:201;

plot(t,Ufd(3,j),'r-')

hold on

plot(t,Uac(3,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.2')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.3

subplot(2,3,4)

j = 1:201;

plot(t,Ufd(4,j),'r-')

hold on

plot(t,Uac(4,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.3')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.4

subplot(2,3,5)

j = 1:201;

plot(t,Ufd(5,j),'r-')

hold on

plot(t,Uac(5,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.4')

legend('Ufd','Uac')

%Plotting Ufd and Uac vs t at x = 0.5

subplot(2,3,6)

j = 1:201;

plot(t,Ufd(6,j),'r-')

hold on

plot(t,Uac(6,j),'g-')

xlabel('t')

ylabel('Ufd and Uac')

title('t vs Ufd and Uac at x = 0.5')

legend('Ufd','Uac')

**Program 3**

% This script is to find the approximate solution for differential

% equation using explicit finite difference method and compare the

% solution with actual solution

% Defining r,k,h,x,t and n

clear

clc

r = 0.25;

h = 0.1;

k = r\*h^2;

x = 0:h:1;

t = 0:k:1;

Uac = zeros(length(x),9);

% Initializing U

Ufd = zeros(length(x),length(t));

Ufd(:,1) = 1;

% Explicit finite difference solution

% The solution differs at boundaries

for j = 1:400

for i = 1:11

if i == 1

Ufd(i,j+1) = Ufd(i,j) + 2\*r\*(Ufd(i+1,j)-(1+h)\*Ufd(i,j));

elseif i == 11

Ufd(i,j+1) = Ufd(i,j) + 2\*r\*(Ufd(i-1,j)-(1+h)\*Ufd(i,j));

else

Ufd(i,(j+1)) = (r\*(Ufd((i+1),j)) + r\*(Ufd((i-1),j)) + (1-(2\*r))\*Ufd(i,j));

end

end

end

%Reporting solution

for i = 1:6

fprintf('At x = %1.2f\n t U\n',x(1,i))

fprintf('\n')

for j = 1:9

fprintf('%1.4f %1.4f\n',t(1,j),Ufd(i,j))

end

fprintf('%1.4f %1.4f\n',t(1,401),Ufd(i,401))

fprintf('\n')

end

%defining coefficient alpha for analytical solution

a = [0.65327 3.2923 6.3616 9.477 12.606 15.7397];

%Analytical solution

sum =0;

for i = 1:11

for j = 1:9

sum = 0;

for n = 1:6

sum = sum + ((1/(3+4\*(a(1,n))^2))\*sec(a(1,n))\*(exp(-4\*((a(1,n))^2)\*t(1,j)))\*cos(2\*a(1,n)\*(x(1,i)-0.5)));

end

Uac(i,j) = 4\*sum;

end

end

%Reporting comparision

fprintf('At x = %1.2f\n t Uac Ufd\n',x(1,3))

for j = 1:9

fprintf('%1.4f %1.4f %1.4f\n',t(1,j),Uac(3,j),Ufd(3,j));

end

%Plotting Ufd at different given values of x at t = 0 - 0.020

figure(01)

x1 = 0:0.1:0.5;

j = 1:6;

for m = 1:1:9

plot(x1,Ufd(j,m))

hold on

end

xlabel('x')

ylabel('U(finite difference)')

title('Explicit finite difference solution')

%Plotting Uac at different given values of x at t = 0 - 0.020

figure(02)

for m = 1:1:9

plot(x1,Uac(j,m))

hold on

end

xlabel('x')

ylabel('U(analytical)')

title('Analytical solution')