Department of Applied Mathematics and Statistics
COLORADO SCHOOL OF MINES
MATH484: Capstone - Mathematical and Computational Modeling

Assignment #3, Spring 2015
SOLUTIONS

1. Friedman & Littman, p.34, Problem **2.4.1**
For this problem use $dt = 0.001$ and $dx = 0.1$ in order to guarantee stability and convergence of the method. For the spatial interval use $[-2, 200]$, fix the y axis in your plots to $[0, 5]$, and create a 5×1 matrix of plots on the same figure using the `subplot(m,n,position)` command.

Using the forward-time/backward-space (FTBS) scheme from the text, we find that the maximum concentrations are approximately $4.3495, 2.5465, 1.9170, 1.4636$, and 1.2043, respectively. The concentration appears to decrease (due to dissipation in the numerical method) and be pushed further to the right as the wind velocity is increased. Simulations are produced by the code below.

MATLAB Code

```
clear; clc;
U = [1, 5, 10, 20, 40];
dx = 1e-1;
dt = 1e-3;
a = -2;
b = 200;
T = 4;

for vel = 1:5
    x = a:dx:b;
    n = T/dt;
    I = find(abs(x) < 1);
    c(1,:) = zeros(length(x), 1);
    c(1,I) = 5;

    for i = 1:n
        for j = 2:length(x)
            c(i+1, j) = c(i,j) - U(vel)*(dt/dx)*(c(i,j) - c(i, j-1));
        end
```

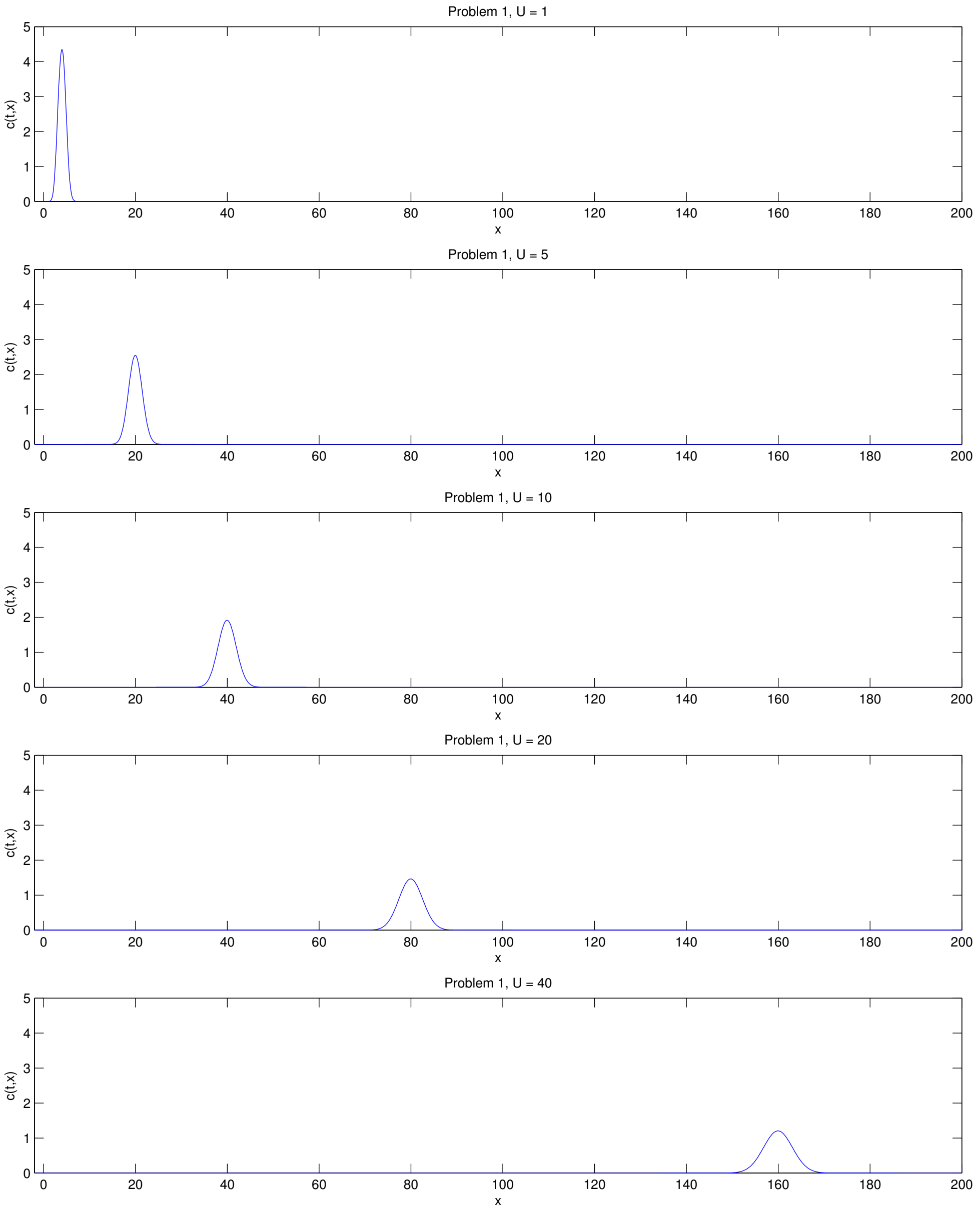

Figure 1: Graphs for Problem 1

```
    end
    subplot(5,1, vel);
    plot(x, c(n+1, :)), title(['Problem 1, U = ', num2str(U(vel))]),
    axis([-2,200,0,5]), xlabel('x'), ylabel('c(t,x)')

    %Maximum concentration at t=4
    M = max(c(end, :))
end
```

```
M =

    4.3495


M =

    2.5465


M =

    1.9170


M =

    1.4636


M =

    1.2043
```

2. Friedman & Littman, p.34, Problem **2.4.2**
Again, use $dt = 0.001$ and $dx = 0.1$, but for the spatial interval use $[-5, 15]$, fix the y axis in your plots to $[0, 0.5]$, and create a 5×2 matrix of plots on the same figure using the `subplot(m,n,position)` command.

The profiles for $c(t, x)$ with $t = 1, 2, 3, ..., 10$ and the code to generate these graphs are given below.

MATLAB Code

```
clear; clc; figure;
dx = 1e-1;
dt = 1e-3;
a = -5;
b = 15;
T = 10;

x = a:dx:b;
n = T/dt;
c(1,:) = zeros(length(x), 1);
```

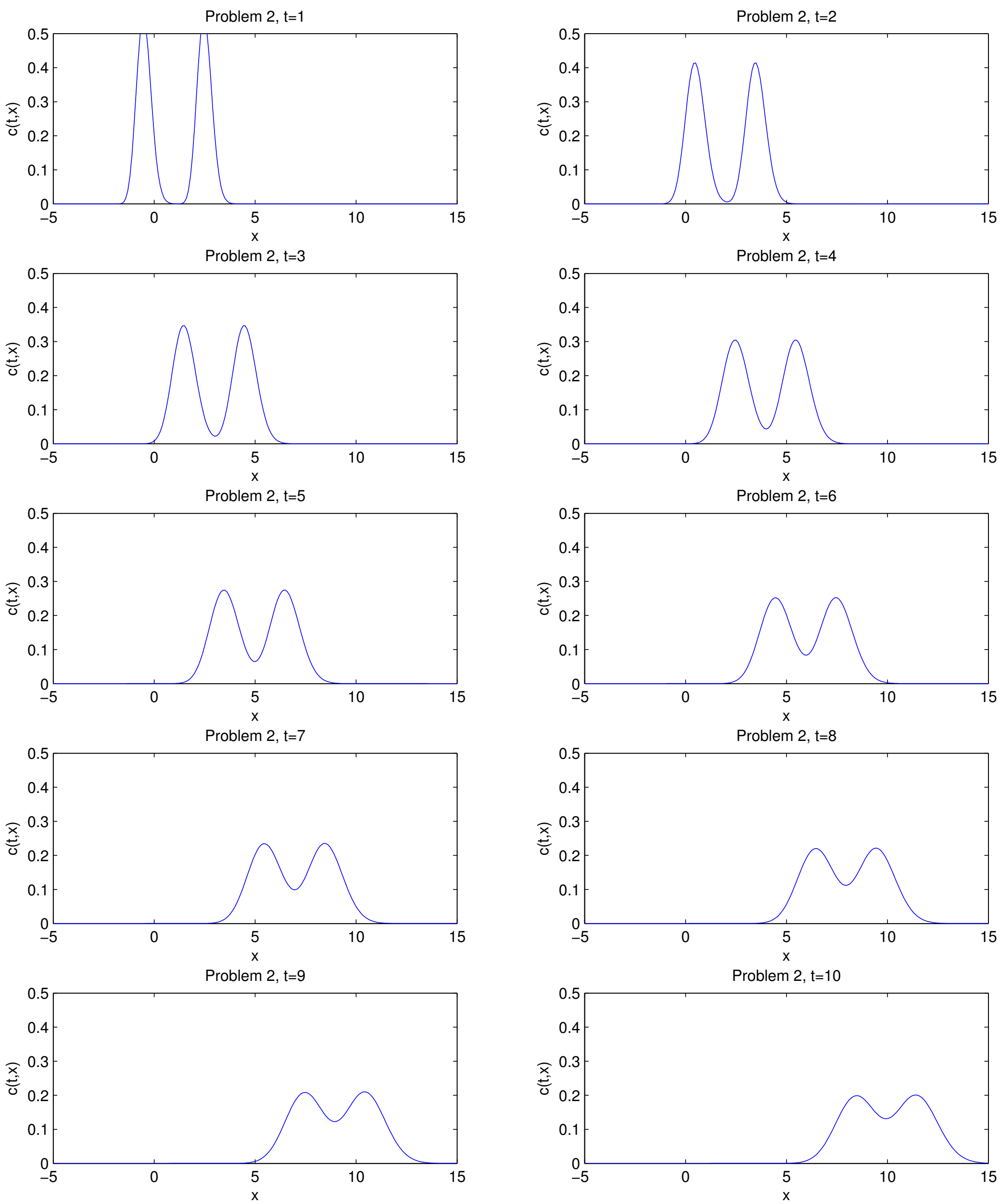

Figure 2: Graphs for Problem 2

```
I = find((x > -2 & x < -1) | (x >= 1 & x <=2));
c(1,I) = (sin(pi*(x(I)+2))).^2;

for i = 1:n
    for j = 2:length(x)
        c(i+1, j) = c(i,j) - (dt/dx)*(c(i,j) - c(i, j-1));
    end
```

```
    if ((i+1)*dt) == floor((i+1)*dt)
        subplot(T/2,2,(i+1)*dt), plot(x, c(i+1, :)),
        title(['Problem 2, t=', num2str((i+1)*dt)]),
        axis([-5,15,0,0.5]), xlabel('x'), ylabel('c(t,x)')
    end
end
```

3. Friedman & Littman, p.34, Problem **2.4.4**
Use $dt = 0.001$ and $dx = 0.1$, but for the spatial interval use $[-2, 15]$, fix the y axis in your plots to $[0, 6]$, and create a 2×2 matrix of plots on the same figure using the `subplot(m,n,position)` command.

As before, the simulations and code are given below.

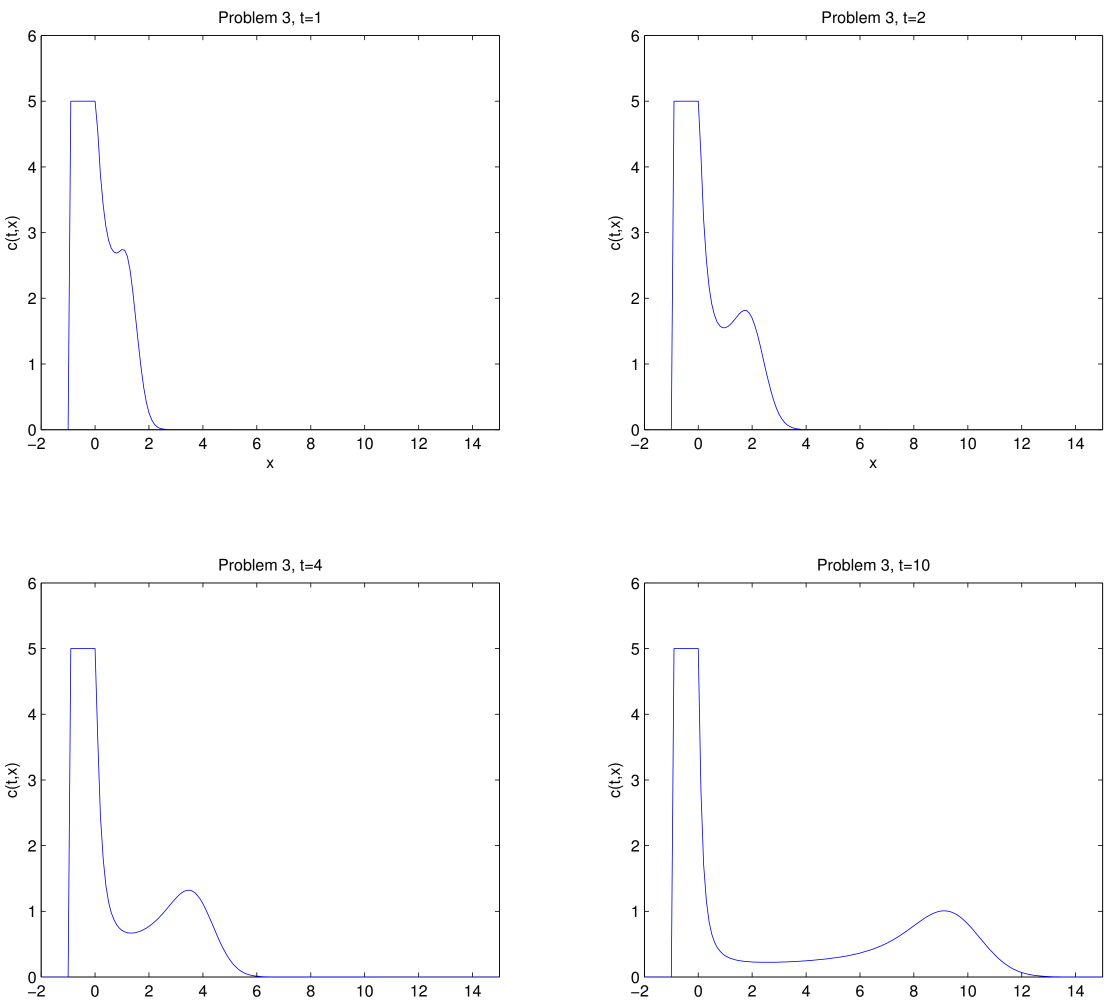

Figure 3: Graphs for Problem 3

MATLAB Code

```
clear; clc; figure;
dx = 1e-1;
dt = 1e-3;
a = -2;
b = 15;
T = 10;

x = a:dx:b;
n = T/dt;
c(1,:) = zeros(1,length(x));
I = find(abs(x) < 1);
c(1,I) = 5;

U = zeros(1, length(x));

for j = 1:length(x)
    val = x(j);
    if  val > 0
        U(j) = val^2/(1 + val^2);
    end
end

pos=0;

for i = 1:n
    for j = 2:length(x)
        c(i+1, j) = c(i,j) - U(j)*(dt/dx)*(c(i,j) - c(i, j-1))...
            - (U(j)-U(j-1))*(dt/dx)*c(i,j);
    end

    if sum(((i+1)*dt) == [1,2,4,10]) > 0
        pos = pos + 1;
        subplot(2,2,pos), plot(x, c(i+1, :)),
        title(['Problem 3, t=', num2str((i+1)*dt)]),
        axis([-2,15,0,6]),xlabel('x'), ylabel('c(t,x)')
    end
end
```

4. Friedman & Littman, p.35, Problem **2.5.2**
You should produce a plot that looks similar to those in the text (p. 36), so define $c_0(x)$ as on p. 35. We will change a few details, though. Use $dx = dy = 0.1$, $dt = 0.01$ and fix the x-y plane to $[-20, 20] \times [-20, 20]$. The MATLAB commands

```
z = squeeze(c(n,:,:));
surf(x,y,z', 'EdgeColor', 'none');
```

may be helpful to plot a 3D graph. Finally, compute the maximum concentration at time $t = 3$.

The maximum concentration at time $t = 3$ is approximately 94.1006. The associated simulation and code are given below.

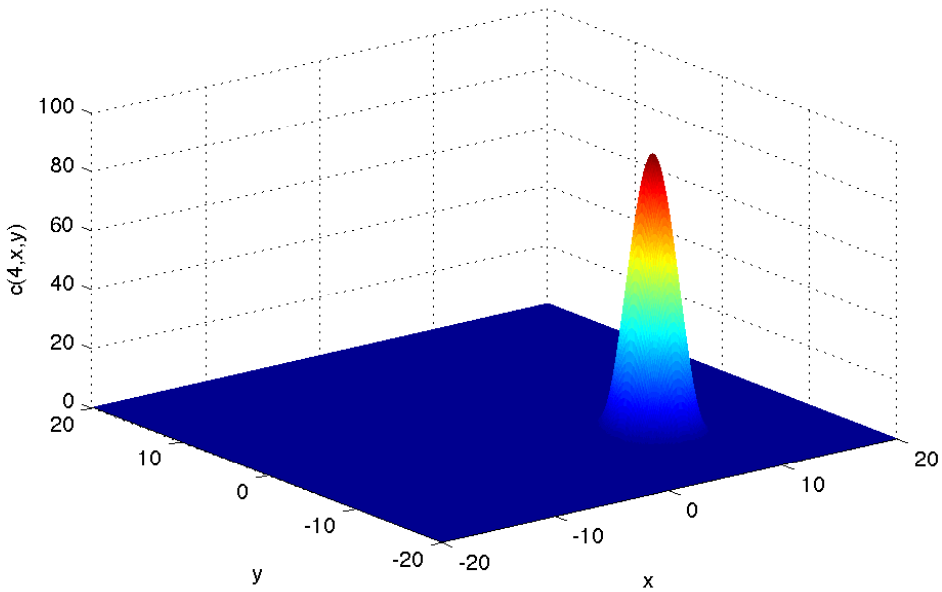

Figure 4: Graph for Problem 4

MATLAB Code

```
clear; clc; figure;
dx = 0.1;
dy = dx;
dt = 1e-2;
a = -20;
b = 20;
T = 3;
x = a:dx:b;
y = x;
n = T/dt;

c(1,:,:) = zeros(1, length(x), length(y));

%Initial Condition
for j = 1:length(x)
    for k = 1:length(y)
        R = sqrt((x(j)-5)^2 + (y(k)+10)^2);
```

```
        if (R < 4)
            c(1, j, k) = 50*(1+cos(pi*R/4));
        end
    end
end

for i = 1:n
    for j = 2:length(x)
        for k = 2:length(y)
            if x(j) == 0
                U = -sin(pi/2);
                V = cos(pi/2);
            else
                theta = atan(y(k)/x(j));
                U = -sin(theta);
                V = cos(theta);
            end

        c(i+1, j, k) = c(i,j,k) - U*(dt/dx)*(c(i,j,k) - c(i,j-1,k))...
            - V*(dt/dy)*(c(i,j,k) - c(i,j,k-1));
        end
    end
end

z = squeeze(c(n+1,:,:));
surf(x,y,z', 'EdgeColor', 'none'), xlabel('x'), ylabel('y'), zlabel('c(4,x,y)')


maxval = max(max(z))
```

5. Friedman & Littman, p.37, Problem **2.6.1**
In order to reduce the computational time in producing the graphs, use $dx = 0.5$ and $dt = 0.01$. Instead of generating 2D plots as in Problem 1, generate a 2×3 (with one entry empty) matrix of 3D plots on the coordinate system defined by $(x, t, c(t, x))$ as in Fig. 2.6. Using `surf` will help with this, and to get time in the same direction as the book use

```
set(gca, 'YDir', 'reverse');
```

Additionally, be sure to answer the questions stated within the problem.

The maximum concentrations at time $t = 4$ are approximately $0.8531, 0.7173, 0.6078, 0.5273$, and 0.6083, respectively, which are drastically less than the corresponding values in Problem 1. Thus, the addition of the diffusion term in the equation appears to decrease the maximum value of $c(t, x)$. As before, the simulations and code are given below.

MATLAB Code

```
clear; clc;
```

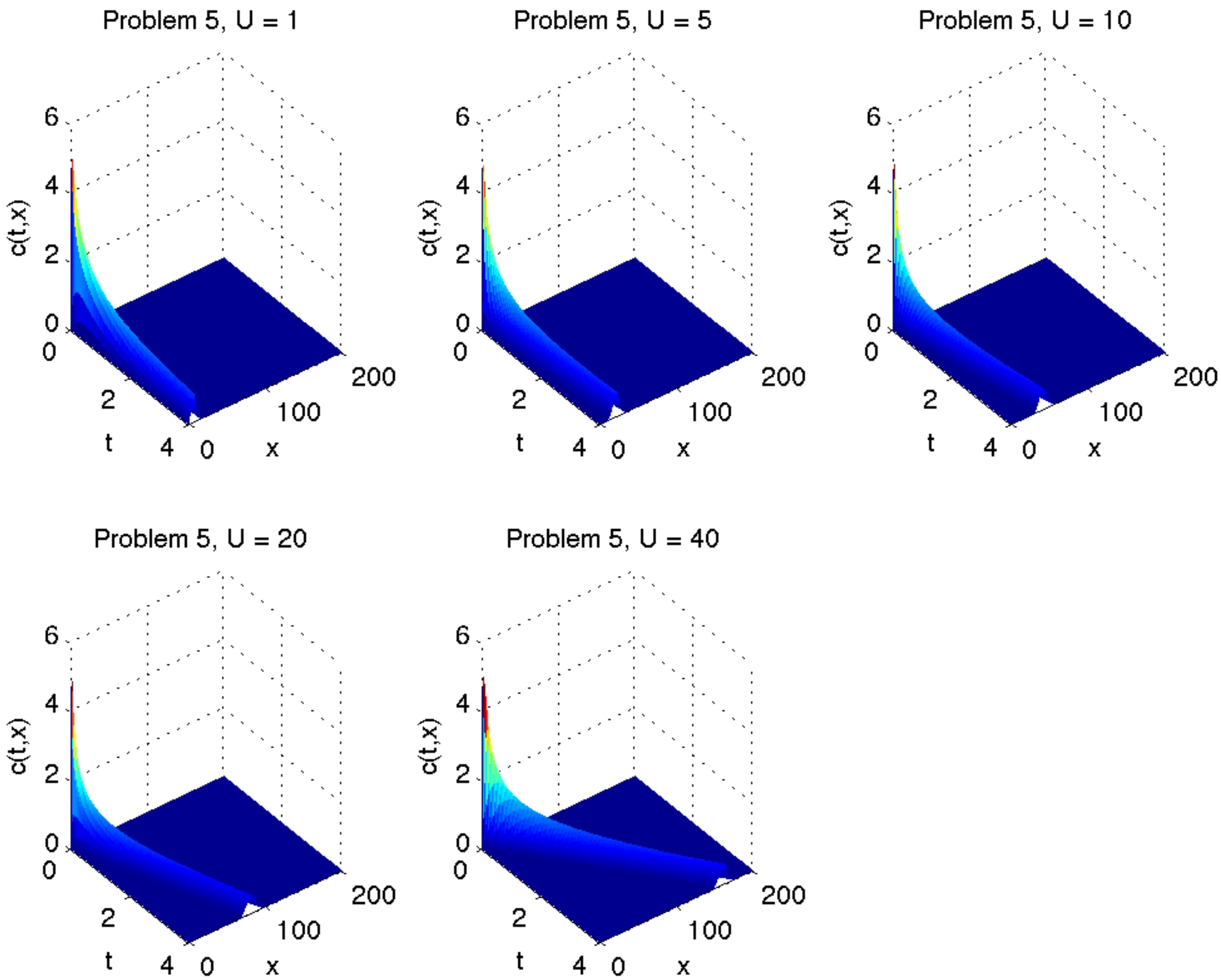

Figure 5: Graphs for Problem 5

```
U = [1, 5, 10, 20, 40];
dx = 5e-1;
dt = 1e-2; %1e-2;
a = -2;
b = 200;
T = 4;

for vel = 1:5
    x = a:dx:b;
    t = 0:dt:T;
    n = T/dt;
    I = find(abs(x) < 1);
    c(1,:) = zeros(1,length(x));
    c(1,I) = 5;
```

```
    for i = 1:n
        for j = 2:length(x)-1
            c(i+1, j) = c(i,j) - U(vel)*(dt/dx)*(c(i,j) - c(i, j-1)) +...
                (dt/dx^2)*(c(i,j+1) - 2*c(i,j) + c(i, j-1));
        end

    end
    subplot(2,3, vel);
    surf(x, t, c, 'EdgeColor', 'none'), title(['Problem 5, U = ', num2str(U(vel))]),
    axis([-3,200,0,4, 0, 6]),
    xlabel('x'), ylabel('t'), zlabel('c(t,x)')
    set(gca, 'YDir', 'reverse');

    %Maximum concentration at t=4
    M = max(c(end, :))
end
```

```
M =

    0.8531


M =

    0.7173


M =

    0.6078


M =

    0.5273


M =

    0.6083
```