

Homework Assignment Number Four Solutions

Problem (1)

Consider the following boundary value problem

$$\frac{d^2 y}{dx^2} = c_1 \frac{dy}{dx} + c_2 y + c_3 \sin(x) + c_4 \exp\left(\frac{dy}{dx}\right)$$

with the boundary conditions

$$y(x=0) = y_o = 1.0$$

$$y(x=10) = y_f = 1.0$$

- (a) Convert this single second-order ODE, to a system of two first-order ODEs.
 (b) Let $c = [1, -2, 2, 0]$ Determine the behavior of $y(x)$ and $y'(x)$ from $0 \leq x \leq 10$. Show the behavior in a plot. Clearly identify which curve corresponds to which function. State what value of the initial condition for $y'(x=0)$ led to the final solution.
 (c) Let $c = [0, 1, 2, -2]$ Determine the behavior of $y(x)$ and $y'(x)$ from $0 \leq x \leq 10$. Show the behavior in a plot. Clearly identify which curve corresponds to which function. State what value of the initial condition for $y'(x=0)$ led to the final solution.

Solution:

- (a) Convert this single second-order ODE, to a system of two first-order ODEs.

Follow the three step process. First, define new variables.

$$y_1 = y \qquad y_2 = \frac{dy}{dx}$$

Second write the ODEs in the new variables.

$$\frac{dy_1}{dx} = y_2$$

$$\frac{dy_2}{dx} = c_1 y_2 + c_2 y_1 + c_3 \sin(x) + c_4 \exp(y_1)$$

Third, write the conditions in terms of the new variables.

$$y_1(x=0) = y_o = 1.0$$

$$y_1(x=10) = y_f = 1.0$$

(b) Let $c = [1.0, -2.0, 2.0, 0.0]$ Determine the behavior of $y(x)$ and $y'(x)$ from $0 \leq x \leq 10$. Show the behavior in a plot. Clearly identify which curve corresponds to which function. State what value of the initial condition for $y'(x=0)$ led to the final solution.

To solve this BVP, I will use both the code for Newton-Raphson method with Numerical Derivatives for 1 equation (nrnd1.m) and the classical 4th-order Runge-Kutta method for N equations (rk4n.m).

I modified the input function in nrnd1.m as follows:

```
function f = funkeval(x)
xo = 0;
yo_1 = 1;
yo_2 = x;
xf = 10.0;
yf = 1.0;
n = 1000;
[x,y]=rk4n(n,xo,xf,[yo_1,yo_2]);
yf_calc = y(n+1,1);
f = yf_calc-yf;
```

I entered the ODEs in the input file for rk4n.m as follows

```
function dydx = funkeval(x,y);
dydx(1) = y(2);
dydx(2) = y(1) - 2.0*y(2) + 2.0*sin(x);
```

At the command line prompt, I typed

```
>> [x0,err] = nrnd1(0.5)
```

where 0.5 was my initial guess for the initial slope. This command generated the following output.

```
icount = 1 xold = 5.000000e-01 f = -1.095578e+00 df = -2.160443e-02 xnew = -5.021082e+01 err = 1.000000e+02
icount = 2 xold = -5.021082e+01 f = -6.264989e-13 df = -2.160443e-02 xnew = -5.021082e+01 err = 5.775100e-13

x0 = -50.2108

err = 5.7751e-13
```

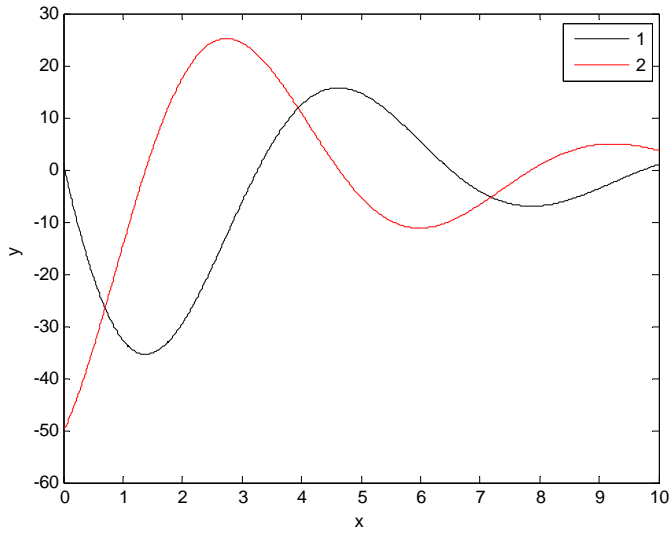
The initial slope is -50.2108.

Not surprisingly this converged in one iteration (with a second iteration required for confirmation) because the ODEs are linear in the unknown, y .

The converged solution can be viewed by typing the following command,

```
>> [x,y]=rk4n(1000,0,10,[1,-50.2108]);
```

The resulting figure is provided below.



We can confirm that this is the solution to the boundary condition by checking the value of y at the final value of x .

```
>> y(1001,1)
ans =    1.0000
```

(c) Let $c = [0,1,2,-2]$ Determine the behavior of $y(x)$ and $y'(x)$ from $0 \leq x \leq 10$. Show the behavior in a plot. Clearly identify which curve corresponds to which function. State what value of the initial condition for $y'(x=0)$ led to the final solution.

To solve this BVP, I will again use both the code for Newton-Raphson method with Numerical Derivatives for 1 equation (`nrnd1.m`) and the classical 4th-order Runge-Kutta method for N equations (`rk4n.m`).

I used the same input function in `nrnd1.m` as was used in part(b) above.

I entered the ODEs in the input file for `rk4n.m` as follows

```
function dydx = funkeval(x,y);
dydx(1) = y(2);
dydx(2) = y(1) - 2.0*exp(y(2)) + 2.0*sin(x);
```

At the command line prompt, I typed

```
>> [x0,err] = nrnd1(0.5)
```

where 0.5 was my initial guess for the initial slope. This command generated the following output.

```
>> [x0,err] = nrnd1(0.5)
icount = 1 xold = 5.000000e-01 f = -1.018489e+02 df = 2.346654e+02 xnew = 9.340174e-01 err = 1.000000e+02
icount = 2 xold = 9.340174e-01 f = -2.609997e+01 df = 1.171209e+02 xnew = 1.156864e+00 err = 1.926297e-01
icount = 3 xold = 1.156864e+00 f = -7.326415e+00 df = 5.035093e+01 xnew = 1.302371e+00 err = 1.117247e-01
icount = 4 xold = 1.302371e+00 f = -2.382988e+00 df = 2.191844e+01 xnew = 1.411092e+00 err = 7.704722e-02
icount = 5 xold = 1.411092e+00 f = -5.168776e-01 df = 1.353624e+01 xnew = 1.449276e+00 err = 2.634745e-02
icount = 6 xold = 1.449276e+00 f = -3.448829e-02 df = 1.180901e+01 xnew = 1.452197e+00 err = 2.011095e-03
icount = 7 xold = 1.452197e+00 f = -1.814707e-04 df = 1.169306e+01 xnew = 1.452212e+00 err = 1.068682e-05
icount = 8 xold = 1.452212e+00 f = -6.589710e-08 df = 1.169245e+01 xnew = 1.452212e+00 err = 3.880885e-09

x0 = 1.4522

err = 3.8809e-09
```

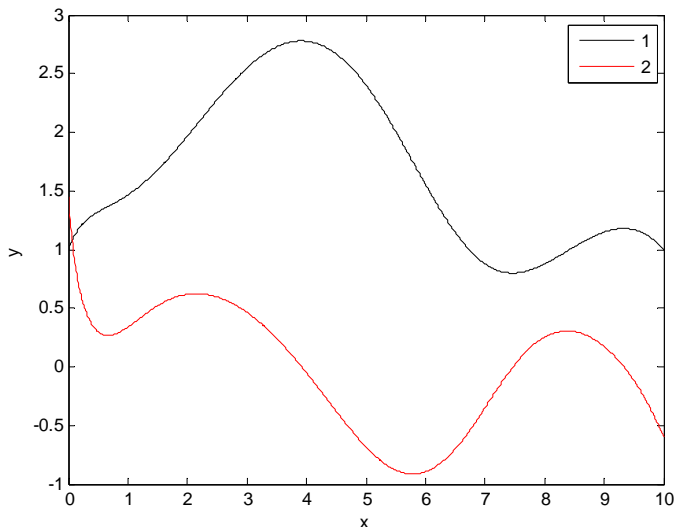
The initial slope is 1.4522.

This problem took more iterations to converge because it is nonlinear and our initial guess wasn't particularly good.

The converged solution can be viewed by typing the following command,

```
>> [x,y]=rk4n(1000,0,10,[1,1.4522]);
```

The resulting figure is provided below.



We can confirm that this is the solution to the boundary condition by checking the value of y at the final value of x .

```
>> y(1001,1)
ans = 0.9999
```

Problem (2)

Find an application from your own experience or a classical problem in your own field of research that results in a ODE boundary value problem.

- (a) Describe the physical problem from which the equation arises. Describe it in sufficient detail that an engineer from a different discipline could understand it.
- (b) Write the ODE(s). Write a complete set of reasonable boundary conditions.
- (c) If known, provide the analytical solution.
- (d) Numerically solve and plot the solution.
- (e) Explain the physical significance of the solution(s) and its behavior.

Problem (3)

Consider the following one-dimensional linear parabolic P.D.E., commonly known as the heat equation:

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} \right) + \frac{f(t, x, y, z)}{\rho \hat{C}_p}$$

where , α , the thermal diffusivity is defined as

$$\alpha = \frac{k}{\rho \hat{C}_p}$$

- (a) Consider an aluminum cylindrical rod 1.0 meter long connecting two heat reservoirs. One of the reservoirs is maintained at T=300K, the other reservoir at T=400 K. Initially, the cylinder is at 300 K. There is no heat loss from the rod. Consider the system to be one-dimensional.
 - (i) Write the IC and BC's.
 - (ii) What does the initial profile look like?
 - (iii) What does the steady state profile look like? Explain.
 - (iv) What is the temperature 0.5 meters into the rod at steady state?
 - (v) What is the temperature 0.5 meters into the rod after 1000 seconds?
 - (vi) Approximately how long does it take for the midpoint of the rod to get within 1% of the steady state value?

Solution:

- (i) Write the PDE, IC, and BC's.

PDE: $\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} \right)$

Initial Condition	$T(x, t = 0) = 300$
Boundary Condition	$T(x = 0, t) = 300$
Boundary Condition	$T(x = L, t) = 400$

(ii) What does the initial profile look like?

$T(x, t = 0) = 300$ A horizontal line at $T = 300$ K on a T vs x plot.

(iii) What does the steady state profile look like?

A linear profile from one boundary condition to the other.

$T(x, t \rightarrow \infty) = 300 + x \cdot (400 - 300)$

(vi) What is the temperature 0.5 meters into the rod at steady state?

$T(x = 0.5, t \rightarrow \infty) = 300 + 0.5 \cdot (400 - 300) = 350$ K

(v) What is the temperature 0.5 meters into the rod after 1000 seconds?

To solve this single, one-dimensional linear parabolic PDE, I will use the code `linparapde_crank_anyBC.m`.

I modified the input functions in `linparapde_crank_anyBC.m` as follows.

I set both boundary conditions to Dirichlet.

```
BC(1) = 'D';
BC(2) = 'D';
```

I set the final time to 4000 seconds and chose dt to be 40 seconds, so I had 100 temporal intervals.

```
% discretize time
to = 0;
tf = 4.0e+3;
dt = 4.0e+1;
```

The rod spans from 0 to 1 meter. I set dx to be 0.1 m, so I had 10 spatial intervals.

```
% discretize space
xo = 0;
xf = 1.0;
dx = 1.0e-1;
```

I defined the PDE in the following functions. Only c and d are non-zero. I obtained the physical properties of Al from the internet.

```
%
% functions defining PDE
%
```

```

function a = afunk(x,t);
a = 0;

function b = bfunk(x,t);
b = 0;

function c = cfunk(x,t);
% rho = density [kg/m^3]
rho = 2700.0;
% Cp = heat capacity [J/kg/K]
Cp_molar = 24.2; % J/mol/K
MW_gpm = 26.9815385; % g/mol
MW_kgpm = MW_gpm/1000.0; % kg/mol
Cp = 24.2/MW_kgpm;
% k = thermal conductivity [W/m/K]
k = 237.0;
% alpha = thermal diffusivity
alpha = k/rho/Cp;
c = alpha;

function dcdx = dcdxfunk(x,t);
dcdx = 0;

function d = dfunk(x,t);
d = 1;

function f = ffunk(x,t);
f = 0;

```

I defined the IC and BCs in the functions below.

```

%
% function defining initial condition
%

function ic = icfunk(x);
ic = 300;

%
% functions defining LHS boundary condition
%

function f = aBCo(t);
f = 1;

function f = bBCo(t);
f = 0;

function f = cBCo(t);
f = -300;

%
% functions defining RHS boundary condition
%

```

```
function f = aBCf(t);  
f = 1;
```

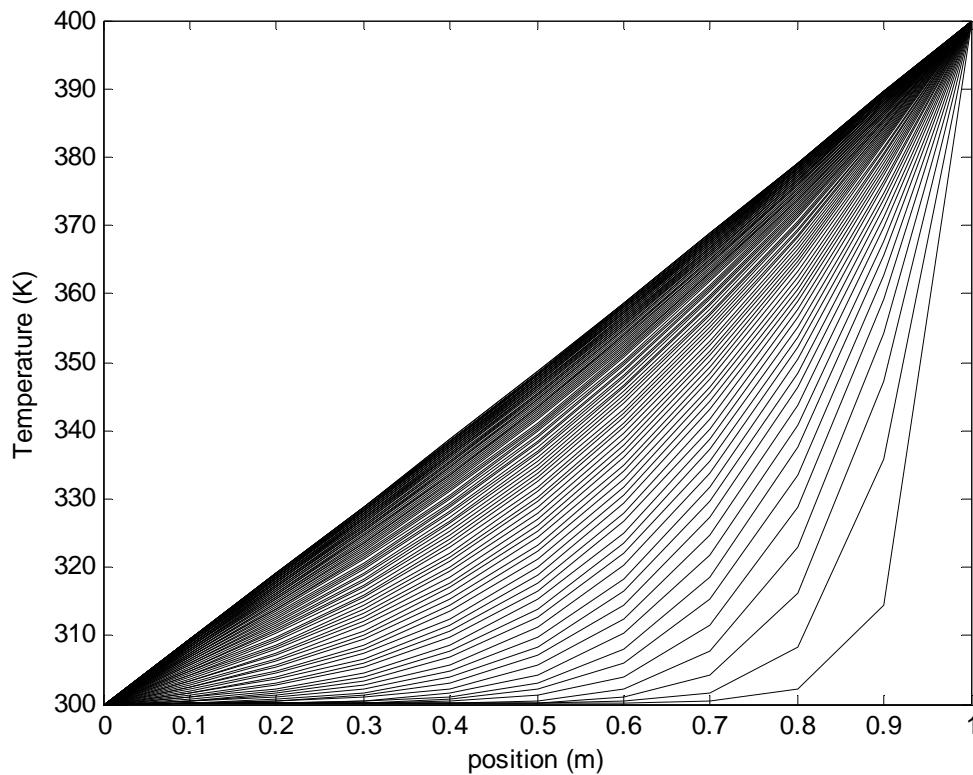
```
function f = bBCf(t);  
f = 0;
```

```
function f = cBCf(t);  
f = -400;
```

At the command line prompt, I typed

```
>> [xvec,tvec,Tmat] = linparapde_crank_anyBC;
```

This command generated the following plot.



To find the value at $t = 1000$ and $x = 0.5$, I first located the indices corresponding to those values.

```
>> tvec(26)  
ans = 1000
```

```
>> xvec(7)  
ans = 0.5000
```

```
>> Tmat(7,26)
```


ans = 325.3130

Therefore the temperature at the midpoint at 1000 seconds is 325.3 K.

(vi) Approximately how long does it take for the midpoint of the rod to get within 1% of the steady state value?

1% of steady state value is 346.5 K

I searched through the Tmat at the midpoint until I located the two time indices at the midpoint where T was below and above this threshold.

```
>> Tmat(7,76:77)
ans = 346.3660 346.5026
```

I then printed out the times corresponding to those indices.

```
>> tvec(76:77)
ans = 3000 3040
```

The times corresponding to those points is 3000 and 3040 seconds.

So, it takes between 3000 and 3040 seconds to reach this temperature.

Problem (4)

Completely rework Problem (3) with the following initial and boundary conditions.

Consider an aluminum cylindrical rod 1.0 meter long with one end connected to a heat reservoir at T=400 K. The other end is insulated. The entire rod is also insulated so that there is no heat loss to the surroundings. The initial temperature of the rod is 300 K.

Solution:

This problem could be done with the program, sypde1.m.

- (i) Write the PDE, IC, and BC's.

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} \right)$$

$$T(x, t = 0) = 300$$

$$T(x = 0, t) = 400$$

$$\frac{dT}{dx}(x = L, t) = 0$$

- (ii) What does the initial profile look like?

$T(x, t = 0) = 300$ A horizontal line at T= 300 K on a T vs x plot.

(iii) What does the steady state profile look like? Explain.

$$T(x, t \rightarrow \infty) = 400K$$

The steady state profile is constant at 400 K. The reason for this is that one of the boundaries continues to pump in heat until the temperature reaches 400 K. Once it reaches that, it stops pumping in heat since heat won't flow up hill.

(iv) What is the temperature 0.5 meters into the rod at steady state?

$$T(x = 0.5, t \rightarrow \infty) = 400K$$

(v) What is the temperature 0.5 meters into the rod after 1000 seconds?

To solve this single, one-dimensional linear parabolic PDE, I will use the code `linparapde_crank_anyBC.m`.

Starting with the code from Problem 3 above, I modified the input functions in `linparapde_crank_anyBC.m` as follows.

I set both boundary conditions to Dirichlet.

```
BC(1) = 'D';
BC(2) = 'N';
```

I didn't change the temporal discretization, the spatial discretization, the PDE, the IC. I changed the BCs as follows.

```
%
% functions defining LHS boundary condition
%
function f = aBCo(t);
f = 1;

function f = bBCo(t);
f = 0;

function f = cBCo(t);
f = -400;

%
% functions defining RHS boundary condition
%
function f = aBCf(t);
f = 0;

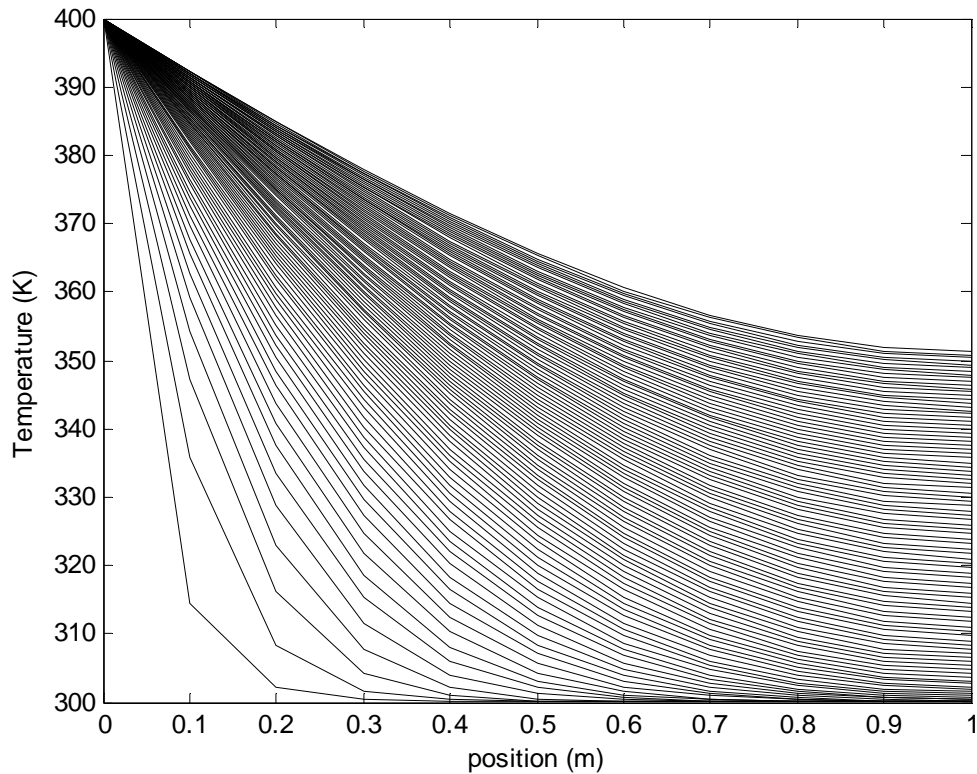
function f = bBCf(t);
f = 1;
```

```
function f = cBCf(t);
f = 0;
```

At the command line prompt, I typed

```
>> [xvec,tvec,Tmat] = linparapde_crank_anyBC;
```

This command generated the following plot.



To find the value at $t = 1000$ and $x = 0.5$, I first located the indices corresponding to those values.

```
>> tvec(26)
ans =      1000

>> xvec(7)
ans =      0.5000

>> Tmat(7,26)
ans =      325.4738
```

Therefore the temperature at the midpoint at 1000 seconds is 325.5 K.

(vi) Approximately how long does it take for the midpoint of the rod to get within 1% of the steady state value?

1% of steady state value of 400 K is 396 K.

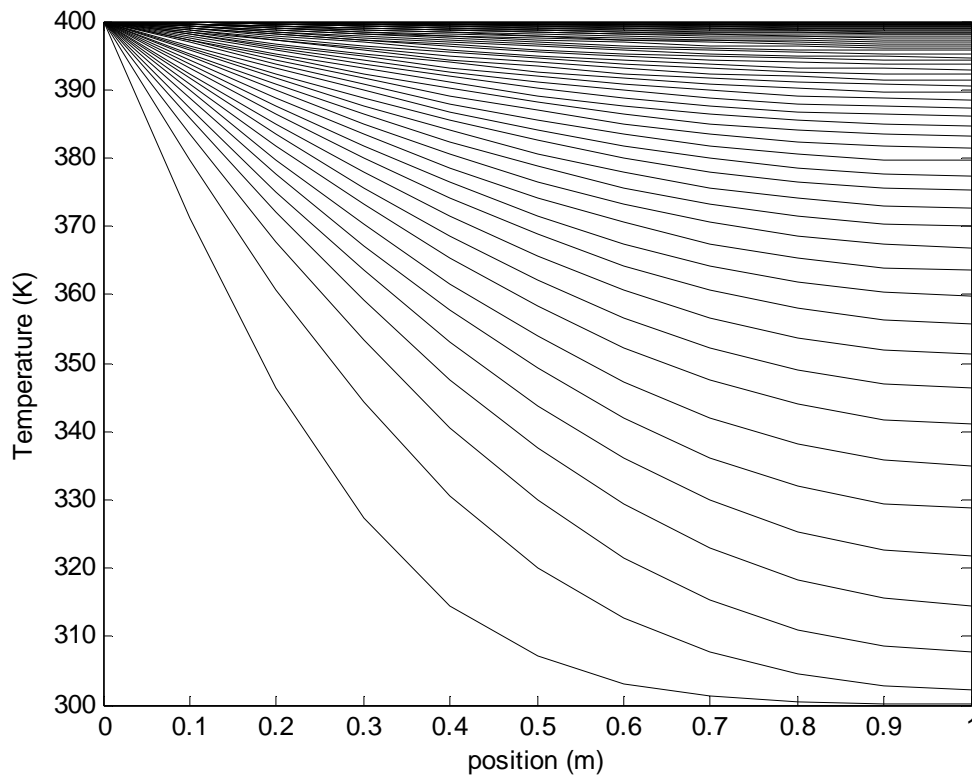
Clearly from the plot in part (v), we aren't close to that value at 4000. I reran the code running for a longer time with the same size time intervals. I modified the code to print out only every 10 profiles in the plot.

```
tf = 4.0e+4;  
dt = 4.0e+1;  
  
nskip = 10;
```

At the command line prompt, I typed

```
>> [xvec,tvec,Tmat] = linparapde_crank_anyBC;
```

This command generated the following plot.



I searched through the Tmat at the midpoint until I located the two time indices at the midpoint where T was below and above this threshold.

```
>> Tmat(7,324:325)  
ans = 395.9874 396.0259
```

I then printed out the times corresponding to those indices.

```
>> tvec(324:325)
ans =      12920      12960
```

The times corresponding to those points are 12,920 and 12,960 seconds.

So, it takes between 12,920 and 12,960 seconds to reach this temperature.

Problem (5)

Rework Problem (3) with the same initial conditions as given in Problem (3). However, in this case, the rod is not insulated so heat is lost from the rod, which has a radius of 10 cm. The surrounding temperature is 200 K. Use a heat transfer coefficient of 40.0 W/m²/K. Consider the system to be one-dimensional.

Solution:

- (i) Write the IC and BC's.

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial x^2} \right) + \frac{hA (T_{\text{surround}} - T(x, y, t))}{\rho \hat{C}_p V}$$

$$T(x, t = 0) = 300$$

$$T(x = 0, t) = 300$$

$$T(x = L, t) = 400$$

where for the case of a one-dimensional cylindrical rod, the area is the surface area of the rod, $A = \pi DL$. This diameter appears no where else in the P.D.E. but is needed for this type of generation term. The volume then is $V = \frac{\pi D^2 L}{4}$

- (ii) What does the initial profile look like?

$$T(x, t = 0) = 300 \quad \text{A horizontal line at } T = 300 \text{ K on a } T \text{ vs } x \text{ plot.}$$

- (iii) What does the steady state profile look like? Explain.

See the plot in part (iv) below.

- (iv) What is the temperature 0.5 meters into the rod at steady state?

Starting with the code from Problem 3 (not Problem 4) above, I modified the input functions in `linparapde_crank_anyBC.m` as follows.

I changed the PDE. I didn't change the type of BCs, the temporal discretization, the spatial discretization, the IC or the BCs. I changed functions a and f .

```

function a = afunk(x,t);
% rho = density [kg/m^3]
rho = 2700.0;
% Cp = heat capacity [J/kg/K]
Cp_molar = 24.2; % J/mol/K
MW_gpm = 26.9815385; % g/mol
MW_kgpm = MW_gpm/1000.0; % kg/mol
Cp = 24.2/MW_kgpm;
% length of rod [m]
L = 1.0;
% heat transfer coefficient in [W/m^2/K]
h_tran_coeff = 40;
% diameter in [m]
radius = 0.1;
diameter = 2.0*radius;
% Area in [m^2]
Area = pi*diameter*L;
% Volume in [m^3]
Volume = pi/4*diameter^2*L;
a = h_tran_coeff*Area/(rho*Cp*Volume);

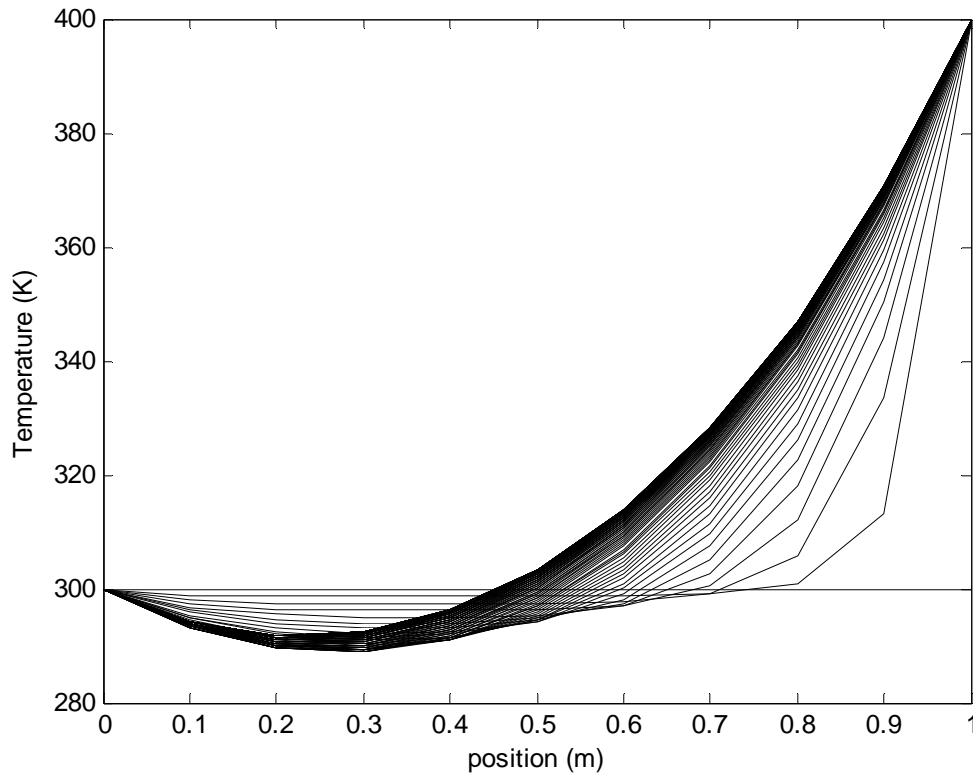
function f = ffunk(x,t);
% rho = density [kg/m^3]
rho = 2700.0;
% Cp = heat capacity [J/kg/K]
Cp_molar = 24.2; % J/mol/K
MW_gpm = 26.9815385; % g/mol
MW_kgpm = MW_gpm/1000.0; % kg/mol
Cp = 24.2/MW_kgpm;
% length of rod [m]
L = 1.0;
% heat transfer coefficient in [W/m^2/K]
h_tran_coeff = 40;
% diameter in [m]
radius = 0.1;
diameter = 2.0*radius;
% Area in [m^2]
Area = pi*diameter*L;
% Volume in [m^3]
Volume = pi/4*diameter^2*L;
a = h_tran_coeff*Area/(rho*Cp*Volume);
% Temperature of the surroundings [K]
Tsurround = 200.0;
f = a*Tsurround;

```

At the command line prompt, I typed

```
>> [xvec,tvec,Tmat] = linparapde_crank_anyBC;
```

This command generated the following plot.



To find the value in the midpoint at steady state, I assumed my last step was at steady state. Since I have 100 intervals, I have 101 temporal nodes.

```
>> Tmat(7,101)
ans = 303.2782
```

To confirm that 4000 s was close to steady state, I ran the simulation to 40,000 s as well. Keeping the same time step, I now have 1000 intervals, so I have 1001 temporal nodes.

```
>> Tmat(7,1001)
ans = 303.3694
```

Close enough for me. The steady state temperature in the rod at the midpoint is 303.4 K.

(v) What is the temperature 0.5 meters into the rod after 1000 seconds?

To find the value at $t = 1000$ and $x = 0.5$, I first located the indices corresponding to those values.

```
>> tvec(26)
ans = 1000

>> xvec(7)
ans = 0.5000
```

```
>> Tmat(7,26)
ans = 299.0156
```

Therefore the temperature at the midpoint at 1000 seconds is 299.0 K.

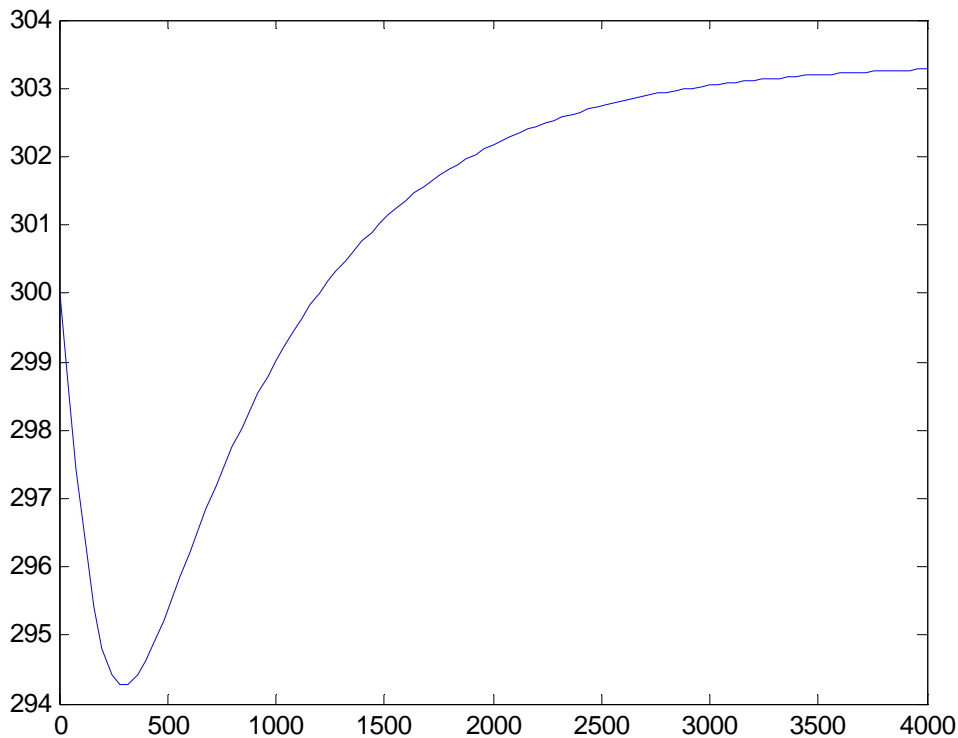
(vi) Approximately how long does it take for the midpoint of the rod to get within 1% of the steady state value?

The steady state temperature in the rod at the midpoint is 303.4 K from part (iv).

Within 1% of steady state value is between 300.37 K and 306.43.

I decided to generate a plot of the time behavior of the temperature at the center of the node.

```
>> figure(2)
>> plot(tvec,Tmat(7,:));
```



The temperature first decreases due to the cold surroundings. Then it increases due to the hot boundary condition.

I searched through the Tmat at the midpoint until I located the two time indices at the midpoint where T was below and above this threshold.

```
>> Tmat(7,33:34)
ans =
 300.3317 300.4842
```


I then printed out the times corresponding to those indices.

```
>> tvec(33:34)
ans =      1280      1320
```

The times corresponding to those points are 1280 and 1320 seconds.

So, it takes between 1280 and 1320 seconds to reach this temperature.