Final Exam Solutions
Administered: Tuesday, December 12, 2023
10:30 AM – 12:45 PM
32 points

Problem 1. (6 points)

Diffusion in solids is often an activated process, in which the diffusivity can be approximated by an Arrhenius-type temperature dependence.

$$D = D_o \, exp\left(-\frac{E_a}{RT}\right)$$

where D is the diffusivity in m^2/s, T is the temperature in K, R is the gas constant (8.314 J/mol/K), E_a is the activation energy in J/mol and D_o is the exponential prefactor in m^2/s.

(a) Linearize this equation so that it is linear in the unknown parameters, E_a and D_o.
(b) Using the table of data providing D and T in the file "xm4p01_f23.txt" on the exam portion of the course website, perform a linear regression to determine the mean values of E_a and D_o for the diffusion of Cu in solid Cu.
(c) Also report the standard deviations of E_a and D_o.

Solution:

(a) Linearize this equation so that it is linear in the unknown parameters, E_a and D_o.

Convert the data into a linear form necessary for a linear regression.

$$D = D_o \, exp\left(-\frac{E_a}{RT}\right)$$

$$ln(D) = ln(D_o) - \frac{E_a}{RT}$$

This equation is in linear form, $y = mx + b$, where the independent variable, $x = -\frac{1}{RT}$, the dependent variable, $y = ln[D]$, the slope, $m = E_a$, and the intercept, $b = ln(kD_o)$.

(b) Using the table of data providing D and T in the file "xm4p01_f18.txt" on the exam portion of the course website, perform a linear regression to determine the mean values of E_a and D_o for this data.

I used the code linreg1.m for linear regression with one independent variable.

I wrote the small script xm4p01_f23.m

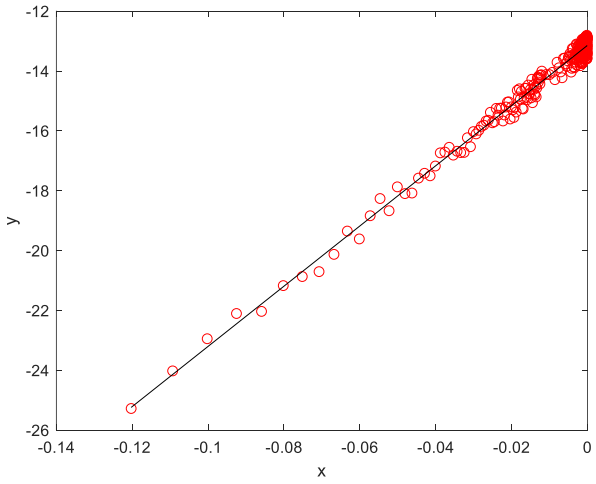
```
clear all;
format long;

M = [100     5.9143E-115
110    6.0003E-105
120    1.16695E-96
... interior data omitted for brevity ...
980    4.28653E-16
990    6.23922E-16
1000 5.57175E-16];

n = max(size(M));
R = 8.314;
for i = 1:1:n
    x(i) = -1.0/(R*M(i,1));
    y(i) = log(M(i,2));
end
[b,bsd,MOF] = linreg1(x, y)
Ea_mean = b(2)
Do_mean = exp(b(1))

Ea_sd = bsd(2)
Do_sd = abs(bsd(1)/b(1))*Do_mean
```

At the command line prompt, I executed the script

```
>> xm4p01_f23
```

This generated the following output for the means, standard deviations and Measure of Fit.

```
>> xm4p01_f23

b =   1.0e+05 *
  -0.000094690506610
   2.110271227399635

bsd =   1.0e+02 *
   0.000407818761763
   1.048196164192228

MOF = 0.999978042188756

Ea_mean = 2.110271227399635e+05
Do_mean = 7.720466610141368e-05

Ea_sd = 1.048196164192228e+02
Do_sd = 3.325096935155578e-07
```



The mean value of E_a is 211,000 J/mol.
The mean value of D_o is 7.72×10^{-5} cm^2/s

The code also generated a plot, shown here.

(c) Also report the standard deviations of E_a and D_o.

Since the slope was E_a, the standard deviation of E_a is simply the standard deviation of the slope. The standard deviation of E_a is 105 J/mol.

For the standard deviation of D_o, we use the relation for the propagation of error that the relative error of D_o is equal to the relative error of the intercept.

$$\frac{s_{Do}}{\overline{x}_{Do}} = \left|\frac{s_b}{\overline{x}_b}\right| \text{ so } s_{Do} = \left|\frac{s_b}{\overline{x}_b}\right| \overline{x}_{Do}$$

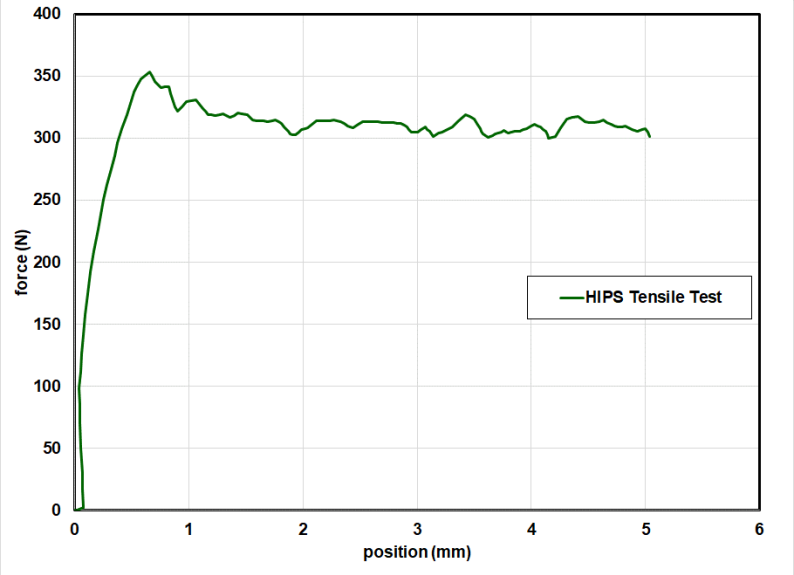
The standard deviation of D_o is 3.33×10^{-7} cm^2/s.

Problem 2. (6 points)

Work, W, is defined as the integral of the force, F, over distance, x,

$$W = \int_{x_o}^{x_f} F dx$$

The data for a tensile test experiment of High-Impact Polystyrene (HIPS) is reported in the file "xm4p02_f23.txt" on the exam portion of the course website. The first column of the data is position in mm and the second column of the data is force in Newtons. A plot of the data is shown below.



(a) What method can you use to numerically integrate this data to obtain the work done during the tensile test?
(b) Report the numerical integral of this data.
(c) Convert this result to SI units, i.e. Joules.

Solution:

(a) What method can you use to numerically integrate this data to obtain the work done during the tensile test?

The Trapezoidal rule can be used to numerically integrate this data.

(b) Report the numerical integral of this data.

To integrate this data, I pasted the data from the input file into the Matlab script, trapezoidal_data.m, presented in class.

```
clear all;
close all;
format long;

% paste date here
% two columns
% first column is x
% second column is integrand, f(x)
% data does not have to be equally spaced
datamat= [
0      0
0.002974705  -0.35
0.00594941   -0.7
... interior data omitted for brevity ...
5.86   0
5.88   -0.25
5.9    -0.5
];
% number of data points
ndata = length(datamat);
% initialize integral to zero.
integral = 0.0;
% compute area of each trapezoid and add them up
for i = 2:1:ndata
    base = datamat(i,1) - datamat(i-1,1);
    average_height = 0.5*(datamat(i,2) + datamat(i-1,2));
    trapezoid_area = base*average_height;
    integral = integral + trapezoid_area;
    %fprintf(1,'interval = %i; trap_area = %e; cumulative integral = %e
\n',i,trapezoid_area,integral);
end
fprintf(1,'data points = %i integral = %e \n',ndata,integral);
```

I executed this script, which I had renamed xm4p02_f23.m, at the Matlab prompt.

```
>> xm4p02_f23
```

The code returned one line.

```
data points = 197 integral = 1.627335e+03
```

So, the integral is 1,627 N-mm.

(c) Convert this result to SI units, e.g. Joules.

A Joule is a N-m. There are 1,000 mm/m, so we divide our result from (b) by 1,000.

The work done during the HIPS tensile test is 1.627 J.

Problem 3. (14 points)

A cylindrical zinc rod, of diameter, *d*, and length *L*, is horizontally suspended between two heat reservoirs, which maintain the temperature at one end (z=0) at 300 K and at the other end (z=l) at 500 K. Between them a fan blows on the rod to conduct heat away. The steady state heat equation describing this set up is given below as

$$0 = \frac{k_c}{\rho C_p}\frac{d^2T}{dz^2} - \frac{h}{\rho C_p}\frac{A}{V}(T - T_{surr})$$

where

- k_c is the thermal conductivity, $k_c = 116.0\,\frac{W}{m\cdot K}$
- ρ is the mass density, $\rho = 7140.0\ \frac{kg}{m^3}$
- C_p is the specific heat capacity, $C_p = 387.0\,\frac{J}{kg\cdot K}$
- d is the diameter of the rod, $d = 0.05\ m$
- l is the length of the rod, $l = 0.3\ m$
- A is the surface area of the rod, $A = \pi d l$
- V is the volume of the rod, $V = \frac{\pi}{4}d^2 l$
- A/V is the surface area to volume ratio of the rod, $A/V = \frac{4}{d}$
- T_{surr} is the surrounding temperature, $T_{surr} = 300\ K$
- h is an emprical heat transfer coefficient, $h = 40.0\,\frac{W}{m^2\cdot K}$

Answer the following questions and perform the following tasks.

(a) Is this ODE problem linear or nonlinear?
(b) Is this ODE problem an initial value problem or a boundary value problem?
(c) Convert this second order ODE into a system of two first order ODEs.
(d) Find the initial temperature gradient at *z* = 0.
(e) Sketch the temperature profile.
(f) Verify that your discretization resolution was sufficient.
(g) Report the temperature in the middle of the rod.

Solution

(a) Is this ODE problem linear or nonlinear?

The problem is linear.

(b) Is this ODE problem an initial value problem or a boundary value problem?

This problem is a boundary value problem because both conditions are not given at the same value of the independent variable, *z*.

(c) Convert this second order ODE into a system of two first order ODEs.

This conversion follows a three step process.

Step 1. Define new variables.

$$y_1 = T \qquad y_2 = \frac{dT}{dz}$$

Step 2. Write ODEs for the new variables.

In this transformation, the first equation is always

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

The second equation is substituting the variables in Step 1 into the original ODE.

$$0 = \frac{k_c}{\rho C_p}\frac{d^2T}{dz^2} - \frac{h}{\rho C_p}\frac{A}{V}(T - T_{surr})$$

or

$$\frac{k_c}{\rho C_p}\frac{d^2T}{dz^2} = \frac{h}{\rho C_p}\frac{A}{V}(T - T_{surr})$$

$$\frac{d^2T}{dz^2} = \frac{\rho C_p}{k_c}\frac{h}{\rho C_p}\frac{A}{V}(T - T_{surr}) = \frac{h}{k_c}\frac{A}{V}(T - T_{surr})$$

$$\frac{dy_2}{dz} = \frac{h}{k_c}\frac{A}{V}(y_1 - T_{surr})$$

Step 3. Write initial conditions for the new variables.

$$y_1(z=0) = T(z=0) = 400 \quad y_2(z=0) = \left.\frac{dT}{dz}\right|_0 \text{ (not given)}$$

(d) Find the initial temperature gradient at *z* = 0.
(e) Sketch the concentration profile.

This is a boundary value problem. I used as the starting points the two Matlab functions, rk4n.m and nrnd1.m, distributed on the course website in the odesolver_bvp folder.

I modified the input file for rk4n.m, which uses the classical fourth-order Runge-Kutta method to solve a system of n ODEs.

```
function dydx = funkeval(x,y);
h = 40.0; % W/m^2/K
kc = 116.0; % W/m/K
d = 0.05; % m
AoV = 4.0/d; % 1/m
Tsurr = 300.0; % K
con = h/kc*AoV; % 1/m^2
dydx(1) = y(2);
dydx(2) = con*(y(1)-Tsurr);
```

I also modified the input for nrnd1.m, which uses the Newton Raphson method with numerical derivatives,

```
function f = funkeval(x)
xo = 0.0;
yo_1 = 300.0;
yo_2 = x;
xf = 0.3;
yf = 500.0;
n = 1000;
[x,y]=rk4n_xm4f23(n,xo,xf,[yo_1,yo_2]);
yf_calc = y(n+1,1);
f = yf_calc-yf;
```

At the command line prompt, I needed an initial guess for the initial slope. I used the average slope as my initial guess.

```
>> dTdz_avg = (500-300)/0.3
dTdz_avg =     6.666666666666667e+02

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

This command generated the following output:

```
>> [x0,err] = nrnd1_xm4f23(dTdz_avg)
icount = 1 xold = 6.666667e+02 f = 9.366083e+01 df = 4.404912e-01 xnew = 4.540385e+02  err = 1.000000e+02
icount = 2 xold = 4.540385e+02 f = 4.495519e-09 df = 4.404912e-01 xnew = 4.540385e+02  err = 2.247754e-11

x0 =     4.540385424699608e+02
err =     2.247753542408721e-11
```

We know that the code converged because the error was less than the stated tolerance of 10^{-6}. The initial slope is

$$\left.\frac{dT}{dz}\right|_0 = 454.0385\ \frac{K}{m}$$

We can run the Runge-Kutta code with this converged initial condition to generate the result.

```
>> >> [x,y]=rk4n_xm4f23(1000,0,0.3,[300,454.0385]);
>> y(1001,1)
ans =     4.999999812923543e+02
```

When we use 1000 intervals in the Runge-Kutta code, we obtain a temperature at the far end of the rod of 500.0 K. That matches our specified boundary condition.

(f) Verify that your discretization resolution was sufficient.

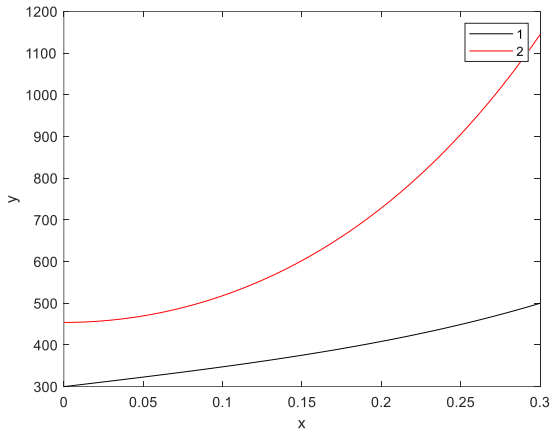
In order to verify that the spatial discretization was sufficiently fine, we also use 10,000 intervals in the Runge-Kutta code. For this finer resolution, we obtain a concentration at the far end of the

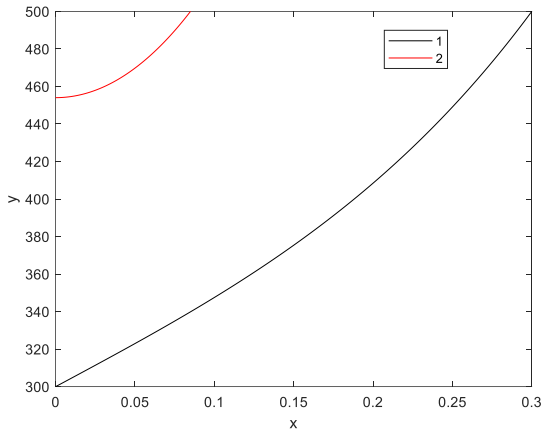
```
>> [x,y]=rk4n_xm4f23(10000,0,0.3,[300,454.0385]);
>> y(10001,1)

ans =     4.999999812923742e+02
```

The two results agree, so we had a good discretization resolution.

A plot of the profile is shown below. The black line marked "1" is the temperature. The red line marked "2" is the temperature gradient.

A second plot is also shown which zooms in on the temperature. It is clear that there is sufficient cooling that the temperature exhibits a nonlinear profile between the two reservoirs.





(g) Report the temperature in the middle of the rod.

```
>> [x,y]=rk4n_xm4f23(1000,0,0.3,[300,454.0385]);
>> x(501)
ans =   0.150000000000000
>> y(501,1)
ans =     3.753731104318554e+02
```

The temperature in the middle of the rod (x = 0.15 m) is 375.37 K.

Problem 4. (6 points)

Answer the following questions based on the course project that you completed.

(a) Which project did you complete, "Phase Diagrams from Regular Solution Theory" or "Data Analysis of Carbon Fiber Tensile Tests"?
(b) Describe in your own words how the numerical method works in general for either "parameter stepping" or "outlier analysis".
(c) Describe how either "parameter stepping" or "outlier analysis" was specifically applied in your project.

Solution

The solution is student-dependent.