USF UNIVERSITY OF SOUTH FLORIDA

Department of Mechanical Engineering

COLLEGE OF ENGINEERING

EML3041: Computational Methods

Semester and Year
Spring 2020

Due Dates
Mini project 1 - Thursday, March 26 at 3:30PM
Mini project 2 - Thursday, April 16 at 3:30PM

Title
Cooling the Aluminum Cylinder Experiment to Illustrate Use of Numerical Methods

Points
100 points

Background
A solid aluminum cylinder treated as a lumped-mass[1] system is immersed in a bath of iced water. Let us develop the mathematical model for the problem to find how the temperature of the cylinder would behave as a function of time.

When the cylinder is placed in the iced-water bath, the cylinder loses heat to its surroundings by convection.

Rate of heat loss due to convection $= hA(\theta(t) - \theta_a)$.

(1)

where

$\theta(t) =$ the temperature of cylinder as a function of time t, ºC

h = the average convective cooling coefficient, W/(m^2-ºC)

[1] It implies that the internal conduction in the trunnion is large enough that the temperature throughout the trunnion is uniform. This allows us to assume that the temperature is only a function of time and not of the location in the trunnion. This means that if a differential equation governs this physical problem, it would be an ordinary differential equation for a lumped system and a partial differential equation for a non-lumped system. In your heat transfer course, you will learn when a system can be considered lumped or non-lumped. In simplistic terms, this distinction is based on the material, geometry, and heat exchange factors of the ball with its surroundings.

A =surface area, m^2

θ_a =the ambient temperature of iced water, ºC

The energy stored in the mass is given by

Energy stored by mass at a particular time = $mC\theta(t)$ (2)

where

m = mass of the cylinder, kg

C = specific heat of the cylinder, J/(kg- ºC)

From an energy balance,

The rate at which heat is gained − Rate at which heat is lost = Rate at which heat is stored

gives

$$0 - hA(\theta(t) - \theta_a) = mC\frac{d\theta(t)}{dt}$$

$$-hA(\theta(t) - \theta_a) = mC\frac{d\theta(t)}{dt} \quad (3)$$

The ordinary differential equation is subjected to

$$\theta(0) = \theta_0$$

where

θ_0 =the initial temperature of cylinder, ºC

Assuming the convective cooling coefficient, h to be a constant function of temperature, the exact solution to the differential equation (3) is

$$\theta(t) = \theta_a + (\theta_0 - \theta_a)e^{-\frac{hAt}{mC}} \quad (4a)$$

It can now also be written in a normalized form as

$$\frac{\theta(t)-\theta_a}{\theta_0-\theta_a} = e^{-\frac{hAt}{mC}} \quad (4b)$$

Grading

This project is part of the Special Assignment/Project grade. Your solution will be graded on the following categories:

- The merit of the conceptual portion
- The merit of programming portion
- The format of submission

Academic Dishonesty

For this project, you may not receive ANY help from anyone outside of the instructor or the TA. Refer to academic dishonesty policy of the University of South Florida at http://ugs.usf.edu/policy/AcademicIntegrityOfStudents.pdf. Also, visit the new ethics resources at USF http://www.usf.edu/ethics/index.aspx for even more information.

Learning Objectives

- Identify and implement the correct procedure for a given problem
- Improve existing programming skills
- Reinforce prerequisite knowledge
- Solve real-world problems

Formatting

- Follow the sample project format including cell formatting, published html format, commenting, etc. http://www.eng.usf.edu/~kaw/class/EML3041/homework/sample_experimental.html
- Use MATLAB to solve all the problems, unless mentioned otherwise.
- Use comments, display commands and fprintf statements, sensible variable names and units to explain your work. **Use SI system of units throughout.**

What we do in the laboratory

1. Fill the ice-cooler with half-water and half-ice. It is better to use the water from the water-cooler, as it is cooler than the tap water. Keep stirring the ice, so that ice cubes are not stuck to each other.
2. Take the thermocouple wires and connect them properly (+ to +, - to -) to the temperature indicator. Two thermocouples are attached to illustrate the concept of a lumped system.
3. Turn the temperature indicator on and wait for a few seconds to record the initial temperature of the cylinder.
4. Record the temperature of the iced-water using a temperature indicator.
5. Immerse the aluminum cylinder in a bath of iced water and start the stopwatch simultaneously. Every five to ten seconds, record the temperature of the cylinder as a function of time.

Figure 1. Cooling the Aluminum Cylinder

What to submit

There are two submission on CANVAS. One is the whole miniproject and another is the mfile.

For the whole project submission

1. A signed typed affidavit sheet (Your printed name can be considered to be the signature) (http://www.eng.usf.edu/~kaw/class/EML3041/homework/affidavit_sheet_individual_projects.pdf) (http://www.eng.usf.edu/~kaw/class/EML3041/homework/affidavit_sheet_individual_projects.doc)
2. Published mfile format. Use any format that allows you to submit it successfully as a pdf file. See https://autarkaw.org/2020/03/25/how-to-publish-in-matlab/
3. Any typed pages when asked for. Each answer needs to start on a fresh page.
4. Attach completed checklist given at end of this assignment. Check-mark the boxes you have accommodated in your assignment. Do not do this blindly.
5. Upload the whole project as a SINGLE pdf file. See https://autarkaw.org/2020/03/26/how-to-make-a-pdf-file/ for help.

For the mfile submission on CANVAS

Submit the mfile in .m format under Computer Project.
Name it as lastname_firstinitial_conv_spring20_x.m, where x is one or two depending on the mini-project number. For example, if your name is Abraham Lincoln, the name of your file would be Lincoln_A_conv_spring20_one.m for mini-project 1 and Lincoln_A_conv_spring20_two.m for mini-project 2.

Why do I ask for a separate mfile submission?

Your mfile is put through a plagiarism checker along with mfiles from previous semesters. Some overlap is expected because of the nature of the program. Each program that is flagged for plagiarism is looked at manually as well.

Project Exercises

Mini project 1 (100 points) - Thursday, March 26, at class start time

1. On plain white paper or engineering graph paper, handwrite the data of temperature vs. time we collected in class and the following data.

   Diameter of cylinder = 49 mm
   Length of cylinder = 100 mm
   Density of aluminum = 2700 kg/m^3
   Specific heat of aluminum = 902 J/(kg-^{o}C)
   Thermal conductivity of aluminum = 240 W/(m-^{o}C)

   Table 1. Coefficient of thermal expansion vs. temperature for aluminum (*http://www.llnl.gov/tid/lof/documents/pdf/322526.pdf*)

Temperature (^{o}C)	Coefficient of thermal expansion (μm/m/^{o}C)
-10	5.8
77	9.3
127	13.9
177	25.5
227	32.6
277	34.1
327	36.1
377	38.9
427	39.8

2. Assign all the required input data (experimental data and other data that is needed for Mini project 1) to variables as MATLAB statements at the beginning of the mfile as one section. Do not change the units of the inputs – enter them as given. Any changes in the input data should not require one to change any part of the rest of the program, and that is what is called "avoiding hardcoding". Of course, fprintf/sprintf/disp the input data using the variables.
3. Change the units of input variables, if needed, to the SI system in a new section. There is no need to fprintf/sprintf/disp in this section.
4. Find the convective cooling coefficient h by a **crude** method as follows. Use the value of the temperature of the aluminum cylinder at the 3rd data point in your readings of temperature vs. time, and solve the nonlinear equation (4a) (do not simplify by hand) to calculate the value of h. *Hint: Use the solve command*.
5. **Only** using the experimental temperature vs time data, and no other data, estimate the rate at which temperature is changing with respect to time at the time corresponding to the 3rd data point.

6. Use the rate at which temperature changed with respect to time from #5 and the right-hand-side of equation (3) to estimate the rate of change of heat stored in the cylinder at the time corresponding to the 3rd data point.
7. Use the value of convective cooling coefficient *h* you found in problem#4 and the left-hand-side of equation (3) to estimate the rate of heat loss due to convection at the time corresponding to the 3rd data point.
8. In 100 words or more, answer the following quetsion.
    a. You obtained values in problems#6 and #7; what did you expect and why; if it did not turn out as expected, what are the possible reasons?

    The work needs to be professional, typed on a separate sheet(s) of paper, with the above question, all the variables defined, and with appropriate equation editors of your word processor used.
9. Estimate the change in the diameter of the aluminum cylinder if it was placed in iced water for several hours, and the iced water temperature is assumed to stay constant.

Mini project 2 (100 *points*) – *Thursday April 16, at class start time*

1. Type in a word processor the data of temperature vs. time we collected in class and the following data even if it is not used.

    Diameter of cylinder = 49 mm
    Length of cylinder = 100 mm
    Density of aluminum = 2700 kg/m^3
    Specific heat of aluminum = 902 J/(kg-°C)
    Thermal conductivity of aluminum = 240 W/(m-°C)

    Table 1. Coefficient of thermal expansion vs. temperature for aluminum (http://www.llnl.gov/tid/lof/documents/pdf/322526.pdf)

Temperature (°C)	Coefficient of thermal expansion (μm/m/°C)
-10	5.8
77	9.3
127	13.9
177	25.5
227	32.6
277	34.1
327	36.1
377	38.9
427	39.8

2. Assign all the required input data (experimental data and any other data that is needed for Mini project 2 to variables as MATLAB statements at the beginning of

the mfile as one section. Do not change the units of the inputs – enter them as given. Any changes in the input data should not require one to change any part of the rest of the program, and that is what is called "avoiding hardcoding". Of course, fprintf/sprintf/disp the input data using the variables.

3. Change the units of input variables, if needed, to the SI system in a new section. There is no need to fprintf/sprintf/disp in this section.
4. In a MATLAB section, find the convective cooling coefficient *h* by a **crude** method as follows. Use the value of the temperature of the aluminum cylinder at the 3rd data point in your readings of temperature vs. time, and solve the nonlinear equation (4a) (do not simplify by hand) to calculate the value of h. *Hint: Use the solve command.*
5. Regress the temperature vs time data to the model

   $$\theta(t) = \theta_a + (\theta_0 - \theta_a)e^{-\frac{hAt}{mC}}$$

   to find the convective cooling coefficient *h* of the regression model.

   You are **not** allowed to transform the data. You should not do any part of this problem by hand.
6. In a MATLAB section, regress the temperature vs. time data to the model

   $$\theta(t) = \theta_a + (\theta_0 - \theta_a)e^{-\frac{hAt}{mC}}.$$

   to find the convective cooling coefficient *h* of the regression model by transforming the data for this exercise.

   Hint: Rewrite the regression model given above in the form of equation (4b), and then take the natural log of both sides. You will end up using the linear regression formulas of the $y = a_1x$ straight-line model.

   The derivation of the model for the transformed data needs to be done manually The derivation needs to be professional, typed on a separate sheet(s) of paper, and should be complete including defining all the variables, using appropriate equation editors of your word processor, writing appropriate sentences to explain steps, etc. See any derivation given in Chapter 06.04 as an example of the expected format.

   Use the final derived formula directly then in a MATLAB section of the program to find the convective cooling coefficient ***h*.**
7. In 100 words or more, do the following. The work needs to be professional, typed on a fresh page, with all the variables defined, and with appropriate equation editors of your word processor used.
   a. Type the values of convective cooling coefficient *h* obtained from problems #4, #5, and #6 with an appropriate description.
   b. Type this question and type the answer: Before running the program, what did you expect in the comparison of the three convection coefficient values and why?

c. Type this question and type the answer: If these values turned out or did not turn out as expected, what are the possible reasons?

8. Plot in one figure the temperature vs. time data that shows individual data points and the temperature vs. time curves using the values of convective cooling coefficient *h* from the problems #4, #5 and #6. Use axes labels with names, symbols and units, figure title, and legends.
9. Find the 2nd order polynomial regression model of thermal expansion coefficient vs temperature by setting up equations in matrix form and then solving the three simultaneous linear equations. Show the answer as a polynomial of temperature. *Hint*: Given n data pairs $(x_1, y_1), (x_2, y_2), \ldots\ldots, (x_n, y_n)$, 2nd order polynomial $y = a_0 + a_1 x + a_2 x^2, n > 2$, regression constants are given by solving these equations.

$$\begin{bmatrix} n & \left(\sum_{i=1}^{n} T_i\right) & \left(\sum_{i=1}^{n} T_i^2\right) \\ \left(\sum_{i=1}^{n} T_i\right) & \left(\sum_{i=1}^{n} T_i^2\right) & \left(\sum_{i=1}^{n} T_i^3\right) \\ \left(\sum_{i=1}^{n} T_i^2\right) & \left(\sum_{i=1}^{n} T_i^3\right) & \left(\sum_{i=1}^{n} T_i^4\right) \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{n} \alpha_i \\ \sum_{i=1}^{n} T_i \alpha_i \\ \sum_{i=1}^{n} T_i^2 \alpha_i \end{bmatrix}$$

10. Compare your answer from problem#9 with the MATLAB *polyfit* command by showing the thermal expansion coefficient as a polynomial function of temperature.

Help

If you need assistance, some resources are listed below.

- Instructor office hours
- TA office hours
- Make an appointment outside of office hours with the instructor or TA.
- How do I do that in MATLAB: http://mathforcollege.com/nm/blog_entries.html#How_do_do_that_in_MATLAB (go to the end of the webpage).

How to approach solving problems on paper

This following is meant to help students approach engineering problems effectively and efficiently. Without the proper approach, engineering problems can be very confusing. The following guidelines are written with common correct and incorrect approaches in mind. Remembering and implementing these approaches can not only help you find a solution faster, but it can increase your understanding of the problem and its conceptual basis. Most of these guidelines are not relegated to this class; you can use them in any engineering class!

- Start with what you know. If you do not know where to start, start with what you know. It's a little bit like connecting the dots. You cannot connect the dots until you have put some down.
    - Look at the information you're given.
    - Look at the applicable equations.
        - What are the restrictions on these equations?

  - Be methodical in your approach.
    - Often students will say, "I don't know anything about this!" Typically, this is because they don't know what they know and what they don't know. Start with what you know!
- Use dimensional analysis as a hint.
  - If you can't find a mistake in your work, check the unit consistency in the problem.
  - If you don't know how to solve a problem, determine the units of the solution and then look to see what units you're missing in the solution.
- Don't cut corners! This WILL hurt you sooner or later.

How to approach programming

- Start with what you know.
  - If you're having trouble programming a problem, start by working through the problem on paper.
  - Don't try to think up the whole program in your head and then type it out!
- When translating the problem solution into a program, display each part of the code. Fix one piece at a time.
- Avoid using ";" at the end of statements while debugging the program. You can add the ";" later when the program is finalized.
- Look at the 'How do I do that in MATLAB series'.
- Use the MATLAB help site (http://www.mathworks.com/help/matlab/) to look up error codes, syntax, etc.
  - If you're looking for syntax examples, click the "example" links on the right side of MathWorks sections for a sample program.

Common mistakes in programming

- Hard coding
- Incorrect format
- Misunderstanding the conceptual (paper) solution
- Inefficient program debugging
- The published mfile is cut off
- Unit errors/no units/SI System not used
- Professional presentation lacking;
- Missing comments
- Unsuppressed lines
- PUBLISHED mfile not in the soft copy but a mfile- how will I grade it;
- Vector data not in table form
- Questions not read properly, and hence answered incorrectly.
- Section numbers not matching problem numbers.

- Spelling mistakes

Look at the checklist on the next page that needs to be attached to the hard copy of your submission.

Checklist for submission

- ☐ I followed the general format as given in the sample project.
- ☐ I uploaded the mfile as a seprate submission.
- ☐ I attached the affidavit sheet.
- ☐ I wrote the code only by myself.
- ☐ I did not show my code to anyone else.
- ☐ I attached any handwritten or typed pages if asked for.
- ☐ I followed the section format as given in the sample project.
- ☐ I published the mfile in published format - HTML format.
- ☐ I wrote proper and reasonable comments.
- ☐ I put the comments on their own lines as seen in the sample project mfile (not at the end of a code line).
- ☐ I CLEARLY identified my methods for each problem.
- ☐ I suppressed all statements.
- ☐ I showed input and output variables using fprintf and disp statements for all exercises unless specified otherwise.
- ☐ I checked for cut off errors in the published file.
- ☐ I avoided all hard-coding (i.e., the program should still work if ANY of the input data is changed).