Assignment 05 -Numerical Differentiation and Numerical Integration

ESO208 - Computational methods in engineering — 2024-25 I
Instructor: Dr.Balaji Devaraju [dbalaji] — **Issue: 24.10.2024**
Deadline: 07.11.2024 (20:00 hrs)

Manual Problems

1. Give proper mathematical explanation for the following questions:

   i. Given the values of $f(x)$ at 3 unequally spaced points $x - h_1$, x, and $x + h_2$, where h_1 and h_2 are not equal, determine the finite difference approximation for $f''(x)$. What is the order of the truncation error? **(10 marks)**

   ii. Demonstrate from first principles that the Trapezoidal rule does not offer any improvement in the order of local truncation error over the rectangular rule in a single interval. Using this conclusion, demonstrate that Simpson's $\frac{1}{3}^{rd}$ rule achieves a local truncation error of 5^{th} order accuracy. **(12 marks)**

   iii. **BONUS:** Show that $R_{k,2}$ in Romberg integration is identical to the composite Simpson's $\frac{1}{3}^{rd}$ Rule with 2_{k-1} panels. **(10 marks)**

2. For the function e^{-x} at x = 1:

   i. Compute the second derivative using second order central difference formula using a 6-digit and 8-digit precision computer. Take step size ranging from 0.64 to 0.00125, reducing it by half each time. (Tabulate the results properly) **(12 marks)**
   Assuming $f''(x) = e^{-1} = 0.36787944$ to be the true value.

   ii. Identify the optimum step size (h) for 6-digit precision and 8-digit precision computer. To what significant digits is the result accurate in both the computers and how many significant digits are lost due to truncation and round off errors in both the computers. Does the extra precision computer help in the computations? **(5 marks)**

   iii. In this problem we have decreased the step size by half as we progress. Does decreasing step size always improve your results? Explain. What different measures would you suggest overcoming these errors? **(5 marks)**

   iv. If the error can be expressed in the form ch^p, can you suggest a numerical technique that would improve the results for a 6-digit precision computer as close to 8-digit precision computer? Using this technique find the second derivative of $f(x)$ using any one suitable h value.
   (5 marks)

   v. In general, if we know the functional form of error of any numerical algorithm, it can be used for error control and extrapolation. Suppose the numerical algorithm given as: $f_{exact} = f(h) + Ah^n + Bh^{n+m} + Ch^{n+2m} +$ ... approximates the exact solution with an error that depends on step size h. Derive the leading error in terms of error $E(h)$ and $E(h/\mathbb{R})$ where $\mathbb{R}$ is any real number. **(6 marks)**

3. Iron man is battling Ivan Vanko to protect Happy Hogan and Pepper Potts, who are seated in a car of mass, $m = 2000\,\text{kg}$. A table below shows the power (P) delivered to the car's wheels at different speed (v). Determine the time (t) that Iron man needs to hold off Ivan Vanko to allow Happy to accelerate the car to reach a velocity of $6\,\text{m/s}$ and escape. Choose appropriate method and give reason for your choice. **(10 marks)**

Table: Problem 3

Velocity v (m/s)	Power P (kW)
0.0	0.0
1.0	4.7
1.8	12.2
2.4	19.0
3.5	31.8
4.4	40.1
5.1	43.8
6.0	43.2

Hint: You may use Newton's s law to formulate the equation.

4. **BONUS:** Ethan Hunt managed to hold on to the aircraft carrying the plutonium package. Benji had managed to hack into two nearby radar stations (say A and B) used to track any aircraft taking off from this airport. The radar stations are separated by 500 m apart, and they record the angles α and β at one-second intervals. Benji managed to gather 3 successive radar readings tabulated below: **(10 marks)**

t (s)	9	10	11
α (degrees)	54.8°	54.06°	53.34°
β (degrees)	65.59°	64.59°	63.62°

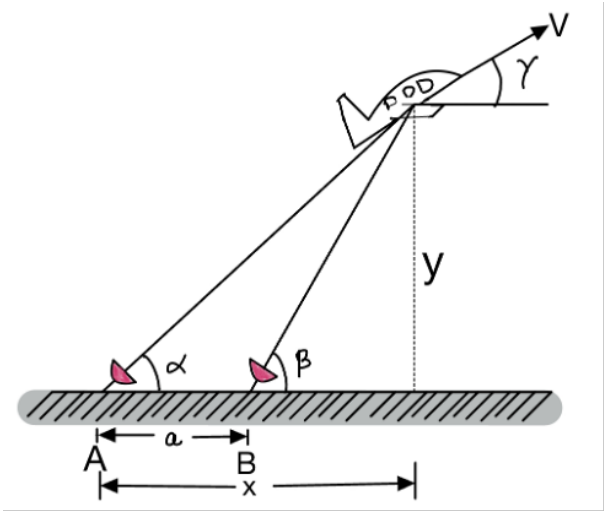

Figure 1: Aircraft with radar stations

Calculate the speed v (accurate to 4 significant digits) of the plane and climb angle γ at $t = 10\,\text{s}$. The coordinates of the aircraft are obtained from geometry as:

$$x = a\frac{\tan\beta}{\tan\beta - \tan\alpha}$$

and

$$y = a\frac{\tan\alpha\tan\beta}{\tan\beta - \tan\alpha}$$

Programming Problems

1. The Debye model, proposed by Peter Debye in 1912, is a technique in thermodynamics and solid-state physics for estimating the heat capacity. The Debye model correctly predicts the low-temperature dependence of the heat capacity of solids. The Debye's formula for the heat capacity C_v of a solid is given as:

$$C_v = 9Nkg(u)$$

where:

$$\frac{C_v}{9Nk} = g\left(\frac{T}{\Theta_D}\right) = g(u) = \left(\frac{T}{\Theta_D}\right)^3 \int_0^{\frac{\Theta_D}{T}} \frac{x^4 e^x}{(e^x - 1)^2}\, dx$$

The terms of the equation are: - N: Number of particles in the solid - k: Boltzmann Constant - $u = \frac{T}{\Theta_D}$ - T: Absolute temperature - Θ_D: Debye temperature

A material scientist would like to explore the specific heat variation of an unknown solid. Without performing an experiment, the scientist would like to numerically evaluate the trend expected here. Write a Python program to compute $g(u)$ from $u = 0$ to 1.0 at intervals of 0.05 using Romberg Integration and tabulate the results. Do not use built-in functions. Plot the results of $g(u)$ at different values of u. **(15 marks)**

2. The figure (Figure 1) below shows a component of a robotic arm. The geometric analysis of the robotic arm is given in the table below: **(10 marks)**

$\theta\,^\circ$	$\beta\,^\circ$
0	59.96
30	56.42
60	44.10
90	25.72
120	-0.27
150	-34.29

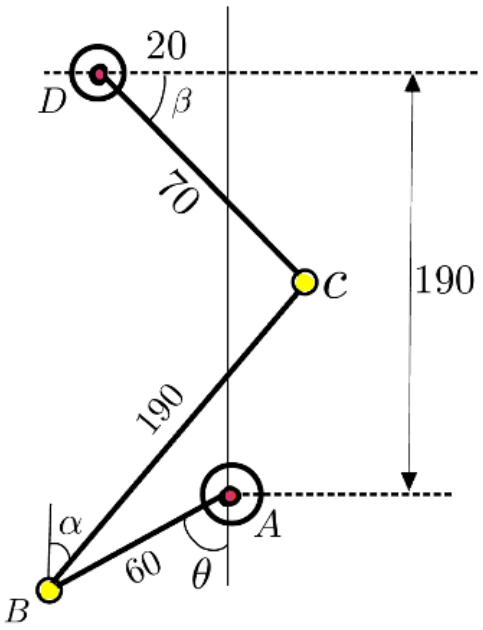

Figure 2: Robotic Arm (All dimensions in mm)

Assuming that the component AB of the robotic arm rotates with the constant angular velocity $\frac{d\theta}{dt} = 1\,\text{rad/s}$, compute $\frac{d\beta}{dt}$ in rad/s at the tabulated values of θ. Use cubic spline interpolation. Tabulate the results.

3. Write a general Python function to find the first-order derivatives of a series of data presented numerically in a matrix whose columns represent vectors of dependent variable i.e. each column in the matrix represents a vector of dependent variables (e.g., $y(x)$), where the rows represent data points corresponding to the values of the independent variable x. The user should be able to choose between backward, forward, or central differentiation as well as the order of the truncation error [Both these inputs must be given using a drop-down box/Button clicks, Also the program must be able to give multiple results without running the entire program repeatedly]. The code should automatically determine the appropriate truncation error order if the specified order cannot be evaluated with the given data.

   A bench-scale gas-solid fluidized bed is a small-scale reactor that uses air or nitrogen to treat organic and inorganic solids in powder form. The axial pressure (kPa) profile of a bench-scale gas-solid fluidized bed is given in table below.

Axial Position z (m)	**Pressure (kPa)**
0.0	1.80
0.5	1.38
1.0	1.09
1.5	0.63
2.0	0.18

   Use the above Python function to:

   (a) Find the pressure gradient $\frac{dP}{dz}$ using the data given from the above table and tabulate the results. **(20 marks)** Tabulate the results for:
   - Forward difference [$O(h)$ to $O(h^4)$],
   - Backward difference [$O(h)$ to $O(h^4)$],
   - Central difference [$O(h^2)$ and $O(h^4)$].

   Sample Table

Finite Difference Method	**Order of Truncation Error**	**Calculated Derivative $\frac{dP}{dz}$**	**Actual Truncation Error Order**
Forward	1		
Forward	2		
Forward	3		
Forward	4		

   (b) Assuming fully developed solids flow in the riser and neglecting wall shear and solids stress, and using the momentum balance equation for the two-phase flow, we find that the pressure drop in the above-mentioned conditions is balanced by the weight of the bed, as described by the following equation: **(10 marks)**

$$-\frac{dP}{dz} = \left(\rho_g(1-\varepsilon_s) + \rho_s\varepsilon_s\right) g$$

   where:

   P: Pressure, z : Axial position, ρ_g and ρ_s : densities of gas and solids, respectively, ε_s: Volume fraction of the solids, g: Gravitational acceleration.

   Given the following values: $\rho_g = 1.2\,\text{kg/m}^3$, $\rho_s = 2650\,\text{kg/m}^3$, $g = 9.81\,\text{m/s}^2$.

   Calculate the solids volume fraction ε_s in the riser of a bench-scale gas-solid fluidized bed. Make a plot of the volume fraction of the solids ε_s versus height z. Analyze and discuss what can be inferred about the solid fraction from this plot.

4. Fermentation is a natural process that's been used by humans since ancient times to create many products, including food, medicines, and fuels. Fermentation process involves two important quantities: Rate of evolution of carbon dioxide and Rate of oxygen uptake. The analysis of inlet and outlet gases of a fermentor experiment along with flow rates, temperature and pressure of these gases would give the above two parameters required to study a fermentation process. The ratio of carbon dioxide evolution rate to oxygen uptake is a good barometer of metabolic activity of the microorganism. Penicillin is a group of antibiotics that treat bacterial infections by damaging the cell walls of bacteria. Below is a table given for the rate of evolution of carbon dioxide and rate of oxygen uptake during fermentation of Penicillium chlysogenum that produces the penicillin antibodies. **(10 marks)**

Time (h)	**Rate of CO_2 Evolution (g/h)**	**Rate of O_2 Uptake (g/h)**
140	15.72	15.49
141	15.53	16.16
142	15.19	15.35
143	16.56	15.13
144	16.21	14.20
145	17.39	14.23
146	17.36	14.29
147	17.42	12.74
148	17.60	14.74
149	17.75	13.68
150	18.95	14.51

i. Write a Python function to implement Simpson's $\frac{1}{3}^{rd}$ rule and calculate the total amount of carbon dioxide (CO_2) produced and oxygen (O_2) consumed over the 10-hour period of fermentation.

ii. **BONUS:** Compare these results with the trapezoidal rule. **(5 marks)**

Result Table

Quantity	**Simpson's 1/3 Rule (g)**	**Trapezoidal Rule (g)**
Total CO_2		
Total O_2		

Note:

i. No flowcharts are required.

ii. Please ensure that you arrange the answers in the same order as that of questions. You can use sections in colab to help you do the same for coding section.

iii. Each program should have the formulas used to solve the problem and supporting explanation attached to the code as a scanned image. You will be awarded zero marks if there is no supporting document.

iv. Ensure that you use the specified input data provided in the problem when generating results for assignment submission. Failure to adhere to the given input will result in a 50% deduction in marks.

v. The results must be presented in a format (Tables) similar to the sample provided in each problem. Marks will be deducted if the required format is not followed.