

Final Project

In this project, you will develop a numerical method to solve system of first order ordinary differential equations. As an application of developed numerical method, you will solve the one-dimensional partial differential equation (PDE). The task also includes optimization problem to be solved using two different approaches.

1 Problem set 1 (20 marks)

Let $u = u(t) = [u_1(t), u_2(t), \dots, u_n(t)]^T$ be a $n \times 1$ vector of functions of time t , where $0 \leq t \leq T$. Let u satisfies following:

$$\frac{d}{dt}u(t) = Au(t) + f(t), \quad (1)$$

where $A = [a_{ij}]$ is a $n \times n$ matrix independent of time t and u , and $f = f(t) = [f_1(t), f_2(t), \dots, f_n(t)]^T$ a $n \times 1$ vector of functions. Above equation satisfies the following initial condition

$$u(0) = u^0, \quad (2)$$

where u^0 is a $n \times 1$ vector of given numbers.

Let Δt is the step size and $t_1 = 0, t_2 = \Delta t, \dots, t_{N_t+1} = N_t \Delta t$ are the discrete times, N_t being the number of steps.

- (i) **Implement** forward Euler method to solve (1) for $u(t_1), u(t_2), \dots, u(t_{N_t+1})$.
- (ii) **Implement** backward Euler method to solve (1) for $u(t_1), u(t_2), \dots, u(t_{N_t+1})$.
- (iii) **Test** your implementation by comparing your results for the following specific problem with $n = 4, T = 1, \Delta t = 10^{-5}, f(t) = [0, 0, 0, 0]^T, u^0 = [1, 0, 0, 0]^T$, and the matrix A given by

$$A = \begin{bmatrix} -2 & 1 & 0 & 0 \\ 1 & -2 & 1 & 0 \\ 0 & 1 & -2 & 1 \\ 0 & 1 & -1 & 0 \end{bmatrix}. \quad (3)$$

Compare your results from both the forward Euler and backward Euler with solution $u(T)$ at final time T given by

$$u(T) = [0.2165, 0.1931, 0.1138, 0.1138]^T. \quad (4)$$

- (iv) **Verify** that for large T , say $T = 1000$, and $\Delta t = T/10^6$, $u(T)$ is very small (of the order of 10^{-87}). This is due to the fact that the problem is dissipative.
- (v) Take $T = 10$ and different $\Delta t = T/2^k, k = 1, 2, 3, \dots, 10$. **Provide** table of $\max(u(T))$, i.e., maximum of four-element vector $u(T)$, for different k (or equivalently different Δt).

Remark 1. Note that forward Euler is unstable for large Δt . Specifically, it is unstable when Δt is such that $\rho(I + \Delta t A) > 1$, where I is the identity matrix, A is the matrix in ODE, Δt is the size of time step, and $\rho(A)$ is the spectral radius of matrix A . Spectral radius can be computed by first computing the eigenvalues of A , then taking the absolute of the eigenvalues, and finally picking the maximum of absolute of eigenvalues. In Matlab you can do this as follows:

$$rho = \max(abs(eig(I + \Delta t A))). \quad (5)$$

For a simple ODE $dv/dt = -\lambda v$ with a positive number λ , forward Euler is unstable if $|1 - \Delta t \lambda| > 1$ and stable if Δt is such that $|1 - \Delta t \lambda| \leq 1$. Thus, forward Euler is stable as long as $\Delta t \leq 2/\lambda$. As you can see, when λ is large, one needs to have small Δt to keep the forward Euler stable. On the other hand, backward Euler is stable as long as $\frac{1}{|1 + \Delta t \lambda|} \leq 1$ which is satisfied for any positive Δt , i.e., for any Δt , backward Euler is stable. In certain situations backward Euler can compute solutions faster by taking large Δt .

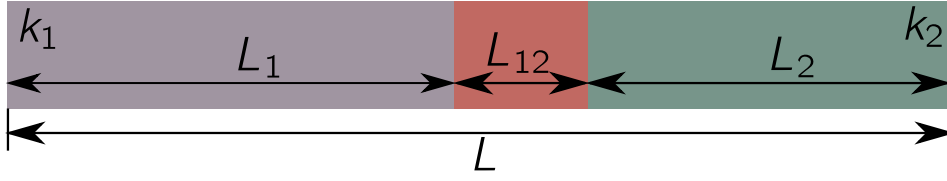


Figure 1: Composite bar with different material properties. In orange region, transition of conductivity k_1 to k_2 takes place linearly.

2 Problem set 2 (35 marks)

Consider a composite bar in Figure 1 of length L . We are interested in modeling temperature in this bar as a function of position on the bar and time. Let $h = h(t, x)$ be the temperature in bar at point x , $0 \leq x \leq L$, and time t , $0 \leq t \leq T$. The left end is at $x = 0$ and the right end is at $x = L = L_1 + L_{12} + L_2$.

The temperature field h satisfies the following one-dimensional heat (or diffusion) partial differential equation, for all $0 \leq t \leq T$ and $0 \leq x \leq L$,

$$\frac{\partial}{\partial t} h(t, x) = k(x) \frac{\partial^2}{\partial x^2} h(t, x) + q_{ext}(t, x), \quad (6)$$

where $\frac{\partial}{\partial t} h(t, x)$ is the partial derivative of h with respect to t , similarly $\frac{\partial}{\partial x} h(t, x)$ is the partial derivative with respect to x , $k = k(x)$ is the thermal conductivity coefficient as a function of x , $q_{ext} = q_{ext}(t, x)$ external source.

Because the bar is composite, we consider thermal conductivity k as a function of x . We consider k as a following function (notice how this function is related to bar in Figure 1)

$$k(x) = \begin{cases} k_1, & \text{if } x \leq L_1, \\ k_2, & \text{if } x \geq L_1 + L_{12}, \\ k_1 + (k_2 - k_1) \frac{(x - L_1)}{L_{12}}, & \text{otherwise.} \end{cases} \quad (7)$$

In the above, k goes to k_1 at $x \leq L_1$ to k_2 at $x \geq L_1 + L_{12}$ linearly over the length L_{12} . We fix $L_{12} = 0.1$ and keep L_1 and L_2 as flexible. Note that since $L_1 + L_2 + L_{12} = L$, if L and L_{12} are fixed, only one of the two, L_1 or L_2 , will be independent.

We also consider the following external source

$$q_{ext}(t, x) = \alpha \exp(-\beta x) \sin(\theta x), \quad (8)$$

where $\alpha = 200, \beta = 2, \theta = 20$.

Since we are dealing with the partial differential equation that includes the first order time derivative and the second order space derivative, we need one initial condition and two boundary conditions. We consider the following initial condition

$$h(0, x) = 0, \quad 0 \leq x \leq L \quad (9)$$

and boundary conditions

$$h(t, 0) = 100, \quad \frac{\partial h(t, L)}{\partial x} = 0, \quad 0 < t \leq T. \quad (10)$$

We basically fixed the temperature at the left-end of the bar to 100 and the heat flux at the right-end to zero.

Parameters. Let $T = 1, L = 1, N_t = 1000, \Delta t = T/N_t, N_x = 1000, \Delta x = L/N_x, k_1 = 10, k_2 = 0.1, L_{12} = 0.1, L_1 = 0.7, L_2 = 0.2, \alpha = 200, \beta = 2, \theta = 20$. Functions k and q_{ext} are given in (7) and (8), respectively.

2.1 Numerical method to solve the heat equation

Let Δx be the mesh size such that $x_1 = 0, x_2 = \Delta x, x_3 = 2\Delta x, \dots, x_{N_x+1} = N_x \Delta x$ are the discrete points on the bar. Let $H_i = H_i(t) = h(t, x_i)$ is the temperature at point x_i on the bar as a function of time t .

- (i) **Obtain** the system of the first order ODE from (6) (through approximation of $\partial^2 h(t, x_i)/\partial x^2$) of the form:

$$\frac{d}{dt}H(t) = AH(t) + Q_{ext}(t), \quad (11)$$

where $H(t) = [H_2(t), H_3(t), \dots, H_{N_x+1}(t)]^T$ is the $N_x \times 1$ vector of function of temperature at points $x_2, x_3, \dots, x_{N_x+1}$, A is the $N_x \times N_x$ matrix, $Q_{ext} = Q_{ext}(t)$ is the vector of function. **Provide** the structure of matrix A and Q_{ext} for general Δx . Note that in above I have used $dH(t)/dt$ because vector function H is just a function of t .

Note that matrix A will be familiar to you. However, in this problem because of the boundary condition $\partial h(t, L)/\partial x = 0$, the matrix A will be slightly different. Follow the hints.

- (ii) Consider discrete times: $t_1 = 0, t_2 = \Delta t, t_3 = 2\Delta t, \dots, t_{N_t+1} = N_t \Delta t$. Using the parameters specified in the previous section and the method developed in **Problem set 1**, numerically **solve** for $H(t_k)$ for $k = 2, 3, \dots, N_t + 1$. Note that from the initial condition (9), we know $H(t_1)$. **Use backward Euler** method.

Plot (on the same plot using 'hold on') $H(t_k)$, $k = 201, 401, 601, 801, 1001$, as a function of x . For a given t , $H(t)$ is a vector with each element of the vector corresponding to different points x on the bar. So you can plot the points x_i on bar in the x-axis and corresponding temperature $H_i(t)$ in the y-axis.

- (iii) **Try** different Δt starting with a large to a very small and **report** the Δt such that it forward Euler method is stable for the above problem.

3 Problem set 3 (35 marks)

To be done independently. There will be a high penalty if we find any sign of collaboration on this problem set.

Notice that the composite bar consists of two materials. One of the material has a very high thermal conductivity whereas the other has a very low conductivity. The material with $k_2 = 0.1$ can be thought of as thermal insulator. Goal of this problem is to find the optimal length L_2 of the second material so that the temperature at the right-end of the bar at final time $T = 1$, i.e., $h(T, L)$, is about 50.

There are two ways to find the optimal L_2 such that $h(T, L) = 50$. But, before we perform this calculation, we change the partial differential equation slightly to also include the *convection* effect:

$$\frac{\partial}{\partial t}h(t, x) = k(x)\frac{\partial^2}{\partial x^2}h(t, x) - c\frac{\partial}{\partial x}h(t, x) + q_{ext}(t, x), \quad (12)$$

where the new term c is the convection velocity. We let $c = 0.001$.

- (i) As in the problem 2, discretize the bar with discrete points $x_1 = 0, x_2 = \Delta x, \dots, x_{N_x+1} = N_x \Delta x$. **Convert** (12) into the system of ordinary differential equation of type (through approximation of $\partial^2 h(t, x_i)/\partial x^2$ and $\partial h(t, x_i)/\partial x$):

$$\frac{d}{dt}H(t) = AH(t) + Q_{ext}(t). \quad (13)$$

In this case, **what** is the matrix A ? Use the *forward difference* to approximate $\partial h(t, x_i)/\partial x$.

- (ii) **Method 1: Regression method.** Consider $L_2 = 0.02, 0.02 + \Delta L_2, 0.02 + 2\Delta L_2, \dots, 0.02 + 15\Delta L_2$ with $\Delta L_2 = 0.05$. For each value of parameter L_2 , **solve** the (13) and obtain $h(T, L)$ (temperature at the final time and at the right-end of the bar, $h(t, x)$ with $t = T$ and $x = L$). Keep all parameters except L_2 same as described in Section 2. Also fix $c = 0.001$.

Step 1: Provide a table of values where the first column is L_2 and the second column is $h(T, L)$.

Step 2: Using the above table as data (L_2 is the independent variable and $h(T, L)$ is the dependent variable on L_2), **explain** what type of *curve fitting* (regression or interpolation) is appropriate.

Step 3: Perform curve fitting of the data in the table by choosing sufficiently high order polynomial function.

Step 4: From the curve-fitting, *find* the independent variable L_2 such that $h(T, L) = 50$.

Step 5: Use L_2 you obtained in step 4 and solve the problem (13). *How* much does the model solution $h(T, L)$ differ from the target value of 50 for the selected L_2 ?

- (iii) **Method 2: Optimization.** Write a code that takes L_2 as input and returns $h(T, L)$ by solving (13). Let $f = h(T, L)$, and since f depends on L_2 , let $f = f(L_2)$. So, your code should compute $f(L_2)$ for a given L_2 .

Define the *squared-error* cost function for the optimization by

$$J = J(L_2) = (f(L_2) - 50)^2. \quad (14)$$

With the initial guess of $L_2 = 0.07$, and the lower and upper bounds 0.05 and 0.2 on L_2 , respectively, *apply* 'fmincon' to find the optimal L_2 . *Report* the optimal L_2 and the corresponding value of $h(T, L)$.

4 Guidelines for preparing project report

Reports carry 10 marks. Your report will have four sections in the following order:

1. In this section, you will discuss the physical significance and application of the problem and general method, and what you have learned in doing this project.
2. Answers to all the problem sets 1, 2, and 3. You must prepare this report such that it is self-contained.
3. In this section, discuss the role of different members in completing this project. Provide some specific details of individual contributions just so we know everyone in the group worked sincerely.
4. Attach your codes as a supplement to this report.

Note. For problem set 3, collaboration is strictly not allowed. If we find any sign of collaboration on problem set 3, there will be heavy penalty.

The codes for problem set 2 must not be generalized to include the components of problem set 3. Basically, work on the problem set 3 separately from the other problem sets.

5 Grading for the final project

Because the problem set 3 really shows whether you have grasped the numerical method in problem set 1 and 2, we will assign following conditional marks:

Suppose you get marks a , where $0 \leq a \leq 35$, on problem set 3. We will subtract $\min\{35 - a, 10\}$ marks from the problem set 1 and $\min\{35 - a, 20\}$ from the problem set 2.

Example: if $a = 20$, then you will loose 10 and 15 marks from problem sets 1 and 2. If $a = 10$, then you will loose 10 and 20 from problem sets 1 and 2.

6 Some hints

This project builds on ideas you have already mastered in previous projects and assignments. There is just one part where you will require some help.

In developing a numerical method for PDE

$$\frac{\partial}{\partial t} h(t, x) = k(x) \frac{\partial^2}{\partial x^2} h(t, x) + q_{ext}(t, x), \quad (15)$$

you will need to approximate the term $\frac{\partial^2}{\partial x^2} h(t, x_i)$ at some discrete point x_i for given t .

Definition of partial derivatives We note how partial derivatives are defined to construct an approximation. We have:

$$\begin{aligned}\frac{\partial}{\partial t}h(t, x) &= \lim_{a \rightarrow 0} \frac{h(t+a, x) - h(t, x)}{a} = \lim_{a \rightarrow 0} \frac{h(t, x) - h(t-a, x)}{a} = \lim_{a \rightarrow 0} \frac{h(t+a, x) - h(t-a, x)}{2a}, \\ \frac{\partial}{\partial x}h(t, x) &= \lim_{a \rightarrow 0} \frac{h(t, x+a) - h(t, x)}{a} = \lim_{a \rightarrow 0} \frac{h(t, x) - h(t, x-a)}{a} = \lim_{a \rightarrow 0} \frac{h(t, x+a) - h(t, x-a)}{2a}\end{aligned}\quad (16)$$

and

$$\begin{aligned}\frac{\partial^2}{\partial x^2}h(t, x) &= \lim_{a \rightarrow 0} \frac{\frac{\partial}{\partial x}h(t, x+a) - \frac{\partial}{\partial x}h(t, x)}{a} = \lim_{a \rightarrow 0} \frac{\frac{\partial}{\partial x}h(t, x) - \frac{\partial}{\partial x}h(t, x-a)}{a} \\ &= \lim_{a \rightarrow 0} \frac{\frac{\partial}{\partial x}h(t, x+a) - \frac{\partial}{\partial x}h(t, x-a)}{2a}.\end{aligned}\quad (17)$$

For the exterior points $x = 0$ and $x = L$ (and also $t = 0$ and $t = T$), one of these formula will not be valid if $x - a$ or $x + a$ is outside the domain of function $[0, L]$ (similarly, if $t - a$ or $t + a$ is outside $[0, T]$). The idea is that to get partial derivative with respect to one variable, we fix all other variables and change only one variable for which partial derivative is computed.

Approximation of partial derivatives For interior point x_i , i.e., $i = 2, 3, \dots, N_x$ (except $i = 1$ and $i = N_x + 1$), you can use the central-difference formula as follows:

$$\frac{\partial^2}{\partial x^2}h(t, x_i) \approx \frac{h(t, x_{i+1}) - 2h(t, x_i) + h(t, x_{i-1}))}{\Delta x^2}, \quad (18)$$

where Δx is the spacing between discrete points. For the exterior point $x_1 = 0$, you do not need to approximate this term, as $h(t, x_1)$ is given by the boundary condition.

For the remaining exterior point $x_{N_x+1} = L$, we need to be little careful as we are also given the boundary condition:

$$\frac{\partial}{\partial x}h(t, x_{N_x+1}) = \frac{\partial}{\partial x}h(t, L) = 0. \quad (19)$$

We consider the following approximation of $\frac{\partial^2}{\partial x^2}h(t, x_{N_x+1})$:

$$\begin{aligned}\frac{\partial^2}{\partial x^2}h(t, x_{N_x+1}) &\approx \frac{\frac{\partial}{\partial x}h(t, x_{N_x+1}) - \frac{\partial}{\partial x}h(t, x_{N_x})}{\Delta x} \\ &= \frac{-\frac{\partial}{\partial x}h(t, x_{N_x})}{\Delta x} \\ &\approx \frac{-1}{\Delta x} \left[\frac{h(t, x_{N_x+1}) - h(t, x_{N_x})}{\Delta x} \right],\end{aligned}\quad (20)$$

where we used the boundary condition $\frac{\partial}{\partial x}h(t, x_{N_x+1}) = 0$ in the second last step, and used the forward difference formula to approximation $\frac{\partial}{\partial x}h(t, x_{N_x})$. Above provides the approximation of $\frac{\partial^2}{\partial x^2}h(t, x_{N_x+1})$.

Similarly, for the problem set 3, you may approximate the $\frac{\partial}{\partial x}h(t, x_i)$ at any interior points using the forward difference approximation

$$\frac{\partial}{\partial x}h(t, x_i) \approx \frac{h(t, x_{i+1}) - h(t, x_i)}{\Delta x}. \quad (21)$$

Again, you do not need to worry about the exterior point $x_1 = 0$. For the other exterior point $x_{N_x+1} = L$, you do not need to do anything as the boundary condition sets $\frac{\partial}{\partial x}h(t, x_{N_x+1}) = 0$.