Computational PDEs M345N10, 2019-2020, Project 3

The project mark, will be weighted to comprise Total 30% of the overall Module.

Blackboard upload deadline : 6.00pm, 10 April

Please name your files in following way:

Proj3_partY_1of_X_yourCID.m (all your Matlab and/or Python scripts, with X the total number of scripts and Y referring to Part A or B etc.); Your accompanying technical report as: Proj3_yourCID.pdf.

A zipped folder will also be acceptable (ideally), in this case call your zipped folder: Proj3_yourCID.zip (etc.)

Project 3: Hyperbolic Systems

Part A: (20 Marks)

The one-dimensional wave equation for $u(x,t)$ is given by

$$u_{tt} = u_{xx}, \tag{1}$$

where x represents a spatial coordinate and t the time. At $t = 0$

$$u(x,0) = \cos(\frac{\pi x}{2\delta}) \; ; \quad \frac{\partial u}{\partial t}(x,0) = 0, \quad \text{for } (-\delta \leq x \leq \delta); \tag{2}$$

while elsewhere, i.e. $x > \pm\delta$ and $t = 0$, you may assume $u(x,0) = \partial u/\partial t = 0$. You are required to solve this equation numerically for the domain $x \leq \pm 2$, where δ is an arbitrary parameter with values $\delta = (0.01,\ 0.1,\ 0.5)$.

1. You are required to investigate how the solution to Eqn. 1 evolves for $t > 0$. Discretise the equation based on your lecture notes, such that it is second-order accurate in time and space using the so-called Leapfrog scheme:
$$\frac{U_j^{k+1} - 2U_j^k + U_j^{k-1}}{(\Delta t)^2} = \frac{U_{j-1}^k - 2U_j^k + U_{j+1}^k}{(\Delta x)^2}. \tag{3}$$
At the computational end points, namely $x = \pm 2$, investigate the treatment of boundary conditions which satisfy the following:
    (a) Minimal numerical reflections off the outer boundaries; i.e. the waves pass through the boundaries.
    (b) The solid wall condition.

Through appropriate numerical experiments investigate, discuss and demonstrate the accuracy of your numerical solution, and any dissipation and dispersive effects that you observe through appropriate plots, showing key features. You should include a discussion on the "modified PDE" that Eqn.(3) represents, and on the optimal choice of discretisation parameters giving you the most exact solution.

Undertake a discrete dissipation-dispersion analysis to explain your numerical observations.

Part B: (10 Marks)
The two-dimensional (2D) wave equation for $u(x, y, t)$ is given by

$$u_{tt} = u_{xx} + qu_{yy}, \tag{4}$$

where x, y represent spatial coordinates and q is a positive constant. At $t = 0$

$$u(x, y, 0) = \cos(\frac{\pi r}{2\delta}) \; ; \quad \frac{\partial u}{\partial t}(x, y, 0) = 0, \quad \text{for } (r \leq \pm\delta), \tag{5}$$

where $r = \sqrt{x^2 + y^2}$. Elsewhere, i.e. $r > \pm\delta$ and $t = 0$, you may assume $u(x, y, 0) = \partial u/\partial t = 0$. You are required to solve this equation with the 2D version of the method used in Part A, for the domain $(x, y) \leq \pm 2$, where $\delta = 0.2$ only.
You should only consider the case where the wave solutions pass through the computational boundaries with minimal reflection.

1. Setting $q = 1$ discretise the equation and compute solutions for large enough time such that the wave structure convects through the boundaries of your computational domain. Show with appropriate plots how your solution evolves with time along the $(x, y = 0)$ and $(x = 0, y)$ data planes. The plots should include sufficient detail and resolution of whether your treatment of the boundary-conditions is effective. Your report should describe the precise details of your finite-difference discretisation at the boundaries.

2. Investigate the effect on your solution accuracy for values of $q = (2., 0.5, 0.2)$ and discuss the results that you compute. You should report on any additional issues which arise as q varies which affects accuracy of your numerical result and steps you undertake to overcome these. These should be by appropriate plots (contour plots are instructive) and discussion.

For both Parts A and B, you should report how you ascertain that your results are grid independent (*subject to some computational tolerance you specify*).

Computational PDEs M345N10, 2019-2020, Project 4: Mastery

The project mark, will be weighted to comprise Total 20% of the overall Module.

Blackboard upload deadline : 6.00pm, 24 April 2020

Please name your files in following way:

Proj4M_1of_X_yourCID.m (all your Matlab and/or Python scripts, with X the total number of scripts); Your accompanying technical report as: Proj4M_yourCID.pdf. A zipped folder will also be acceptable (ideally), in this case call your zipped folder: Proj4M_yourCID.zip (etc.)

You are required to work with the code you developed in the assessed coursework Project 2 on the Elliptic equation.

The equation describing a compressible fluid flowing over a surface is given by

$$\frac{\partial(\hat{\rho}(1+u))}{\partial x} + \frac{\partial(\hat{\rho}v)}{\partial y} = 0, \tag{6}$$

where x, y represent spatial coordinates and (u, v) are the velocity perturbations given by the velocity potential ϕ

$$\frac{\partial \phi}{\partial x} = u; \qquad \frac{\partial \phi}{\partial y} = v. \tag{7}$$

The density $\hat{\rho} = 1 + \rho$ is given by

$$\hat{\rho} = 1 + \rho = \left(1 - \frac{\gamma - 1}{2} M^2 (2u + u^2 + v^2)\right)^{1/(\gamma-1)}, \tag{8}$$

where $\gamma = 1.4$ and M is the Mach number; thus ρ represents the density perturbation which tends to zero far away from the biconvex surface and when $M = 0$, $\rho = 0$ throughout the computational field.

The symmetrical biconvex aerofoil $y_b(x)$ is given by

$$y_b(x) = 2\tau x(1-x), \qquad 0 \leq x \leq 1, \tag{9}$$

where τ denotes the maximum thickness of y_b. The inviscid flow tangency condition on the surface $y_b(x)$ is transferred to $y = 0$, thus you are required to satisfy for $\tau << 1$,

$$\frac{\partial \phi}{\partial y} - (1 + \frac{\partial \phi}{\partial x})\frac{dy_b}{dx} = 0, \qquad 0 \leq x \leq 1, \quad \text{at} \quad y = 0. \tag{10}$$

On either side of the aerofoil, i.e. $y = 0$ a flow symmetry condition of

$$\frac{\partial \phi}{\partial y} = \frac{\partial \rho}{\partial y} = 0, \qquad \text{for} \quad x < 0 \quad \text{and} \quad x > 1, \tag{11}$$

is satisfied, while as $y \to \infty$ the normal velocity $v = 0$; also as $x \to \pm\infty$ the streamwise potential $\frac{\partial \phi}{\partial x} \to 0$. You may also assume $\rho \to 0$ at the boundaries.

1. Devise a numerical strategy which allows you to solve Eqn. (6). Your method should be based on guessing a starting solution for ρ, which then gets corrected during an iterative solution process. Only compute solutions for $\tau = 0.05$, and you may develop your code based on either the Gauss-Seidel or SOR approaches. (12 Marks)

2. Compute numerical solutions for $M = (0.01, 0.2, 0.4)$. Make plots of how the surface velocity perturbation component $U_{surf} = u = \frac{\partial \phi}{\partial x}$ on the aerofoil surface $0 \leq x \leq 1$ varies as M increases. Make plots of how U_{surf} varies with $x(0 : 1)$ and along the entire $y = 0, x(-q : s)$ plane. Likewise make plots of the surface $v-$ velocity and density ρ components along $y = 0$. A colour plot of your entire (u, v) and ρ perturbation field solutions over your complete (x, y) computational domain will also be instructive – use the *pcolor* Matlab command, or alternative. (6 Marks)

3. What is the largest value for M that your code converges? If there is a limit, why do you think this arises? Discuss if your code can be modified to correct this *(You are NOT expected to code for this! just describe, what you think may be occurring.)* (2 Marks)

For both Parts A, B and Mastery you should report how you ascertain that your results are grid independent (*subject to some computational tolerance you specify*).

Notes:

1. Marking will consider both the correctness of your code as well as the soundness of your analysis *and* clarity and legibility of the technical report.

2. In particular describe clearly your treatment of the discretised boundary conditions in your report.

3. All figures created by your code should be well-made and properly labelled.

4. In order to assign partial credit, if your code does not work, do describe the steps that you were attempting in your report, with some plots to indicate.

5. You are allowed to discuss general aspects of Matlab/Python with each other, however you are trusted not to discuss your code or analysis with other students.

Dr M. S. Mughal
25 March, 2020