

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
DEPARTMENT OF MECHANICAL ENGINEERING
CAMBRIDGE, MASSACHUSETTS 02139
2.29 NUMERICAL FLUID MECHANICS — SPRING 2015

Problem Set 1

Issued: Wednesday, February 11, 2015

Due: Monday, February 23, 2015

Grading Note: Please provide your solutions either as hand-written/hard-copy solutions or by submitting via course website. MATLAB[®] codes should be submitted via course website. The bulk of the grades will be given to detailed explanations and to algorithms and numerical schemes that capture the essence of the numerical problems. We know that successful coding of numerical schemes can be time consuming and prone to small errors. Such small errors or omissions in a code will not be heavily penalized.

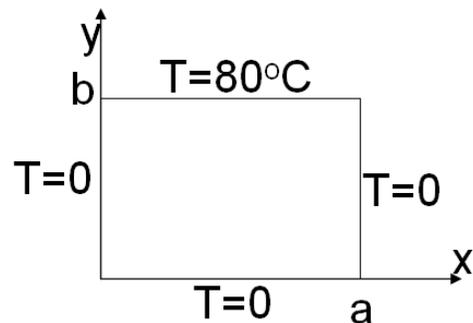
Problem 1

Do problems 3.7 and 3.8 of Chapra and Canale, pages 79-80.

Problem 2

Consider the steady heat transfer within a thin rectangular flat plate whose sides are kept at fixed temperatures. Three sides of the plate are at 0°C and the last one at $T_1 = 80^\circ\text{C}$. The dimensions of the rectangular plate are $a = 5$ m and $b = 4$ m. For the given boundary conditions $T(x,y)$ can be expressed analytically by Fourier series (Erwin Kreyszig, “Advanced Engineering Mathematics”, John Wiley and Sons, 1993):

$$T(x,y) = \frac{4T_1}{\pi} \sum_{n=1}^{\infty} \frac{\sin[(2n-1)\frac{\pi x}{a}] \sinh[(2n-1)\frac{\pi y}{a}]}{(2n-1) \sinh[(2n-1)\frac{\pi b}{a}]}$$



For the following numerical computations, divide the length ‘a’ into 40 segments and the width ‘b’ into 32 segments. To add term in the series, please use the limited precision addition function radd.m (was presented in lecture and is available on course website).

- a) Determine and plot (e.g. 3-D surface plot) the temperature distribution $T(x,y)$ within the plate using 2 significant digits in the 'radd' function in the following two ways:
- Carrying out the summation from $n = 1$ to $n = 20$.
 - Carrying out the summation in the reverse order, i.e. from $n = 20$ to $n = 1$.
- b) Determine the difference in the results of these two summation procedures and generate a surface plot for this difference. Which method gives more accurate results? Why?
- MATLAB help: Look at meshgrid and surf commands in MATLAB and in the MATLAB review and recitation provided on Wednesday, 4 February, 2015.

Problem 3

Do problem 4.13 of Chapra and Canale, page 108.

Problem 4: Truncation Error, Round-off Error, Order of Accuracy and Noisy Data.

This problem studies the errors occurring during the evaluation of the first derivative $f'(x)$ of a given function $f(x)$ using the forward and central difference schemes (of course, in real fluid problems, f would be unknown). For the example here, $f(x) = \sin(x)$.

First, we evaluate the effect of the discretization step size h on the estimate of $f'(x)$.

- Set $h = 10^i$, with $i = [-20: 0.5: 0]$ and $x = \pi - 0.1$. Write a short program that evaluates the total errors of both the forward and central difference estimates of $f'(x)$, for each of these values of h . Plot the results, i.e. plot two total error curves as a function of h in a single figure panel. For each case, plot also the leading term of the truncation (discretization) error.
- Briefly discuss and explain the results, in terms of h .
- Briefly explain the behavior of the total error at the largest values of h ? (Hint: which of the forward or central difference is most accurate at these values of h ?).

Bonus (do this only if you are interested, it is not needed for full credit). We now evaluate the effects differentiating noisy data, i.e. the effects that small random uncorrelated noise in $f(x)$ have on the values of derivative estimates. Assume that $f(x)$ is perturbed with noise that is randomly distributed zero-mean Gaussian, with amplitude of 1%: in other words, the perturbed $f(x) = (1 + 0.01\omega) \sin(x)$ with $\omega \in N(0,1)$.

- Set $h = 0.01$. For the values of $x = [0: h: 2\pi]$, evaluate the perturbed $f(x)$ and the corresponding forward and central difference estimates of $f'(x)$.
- Plot the results as a function of x : e.g. three plots, $f(x)$ and then the two $f'(x)$ each overlaid on the unperturbed derivative $\cos(x)$. Discuss your results.
- Do the general propagation error formula and/or the standard error estimate adequately explain your results in e)?

- g) Select a value of x and evaluate the total error in the forward and central derivative estimates as h is decreased. Compare your results to the theoretical truncation errors (without noise added).

Problem 5

Do problem 8.34 of Chapra and Canale.

Problem 6 (this is a short problem)

It is common in computations to determine random access memory (RAM) requirements.

- a) Calculate the RAM (in megabytes) necessary to store a multidimensional array of size $100 \times 100 \times 100 \times 2000$ (for example a discretized temperature field in 4d: 3d in space and 2000 time steps). This array is double precision, and each value requires a 64-bit word. Recall that a 64-bit word = 8 bytes and 1 megabyte = 10^6 bytes.
- b) If your laptop has 6 Gb of RAM, could you store the array in a) in your RAM?

For your information, when your memory requirements exceed the size of the available RAM, the CPU often uses “swap space”. Swap space is usually a portion of the disk space used for such temporary storage location, with the caveat that access to disk is much slower than RAM access.

Problem 7: (Round off Errors in Orthonormalization)

In this problem, we study the effect of round-off errors in a machine during an orthonormalization process. Given a matrix $V(m \times n)$, $m \geq n$ with linearly independent columns v_1, v_2, \dots, v_n , orthonormalization refers to the generation of orthonormal vectors q_1, q_2, \dots, q_n each of size $m \times 1$ such that they span the same n dimensional subspace of \mathbb{R}^m as vectors v_i , $i = 1, 2, \dots, n$. We define the inner product of two vectors q_i and q_j as $\langle q_i, q_j \rangle = q_i^T \cdot q_j$.

Vectors q_1, q_2, \dots, q_n are said to be orthonormal if they are all orthogonal to each other, i.e., $\langle q_i, q_j \rangle = 0$, for all $i \neq j$ and $1 < i, j < n$, and have unit magnitude i.e., $\|q_i\| = \sqrt{\langle q_i, q_i \rangle} = 1$, $i = 1, 2, \dots, n$.

The classical ‘**Gram-Schmidt**’ (CGS) process is a method for orthonormalizing a given set of vectors. It works as follows:

$$\begin{aligned} u_1 &= v_1, & q_1 &= \frac{u_1}{r_{1,1}}, \\ u_2 &= v_2 - r_{1,2}q_1, & q_2 &= \frac{u_2}{r_{2,2}}, \\ u_3 &= v_3 - r_{1,3}q_1 - r_{2,3}q_2, & q_3 &= \frac{u_3}{r_{3,3}}, \\ &\dots & & \\ u_n &= v_n - \sum_{i=1}^{n-1} r_{i,n}q_i, & q_n &= \frac{u_n}{r_{n,n}}, \end{aligned}$$

where $r_{i,j} = q_i^T \cdot v_j$, $i \neq j$ and $r_{i,i} = \|u_i\|$, $i = 1, 2, \dots, n$. The set of vectors $q_i, i = 1, 2, \dots, n$ are orthonormal and span the same subspace as $v_i, i = 1, 2, \dots, n$.

The ‘**Modified Gram-Schmidt**’ (MGS) process is another method for orthonormalization of vectors. It is very similar to the classic Gram-Schmidt process. In this method, the computation of q_1 is same as in the classical Gram-Schmidt above. For $k = 2, \dots, n$, q_k is computed as follows:

$$\begin{aligned}
 r_{1,k} &= \langle v_k, q_1 \rangle, & u_k^{(1)} &= v_k - r_{1,k}q_1, \\
 r_{2,k} &= \langle u_k^{(1)}, q_2 \rangle, & u_k^{(2)} &= u_k^{(1)} - r_{2,k}q_2, \\
 &\dots & & \\
 r_{k-2,k} &= \langle u_k^{(k-3)}, q_{k-2} \rangle, & u_k^{(k-2)} &= u_k^{(k-3)} - r_{k-2,k}q_{k-2}, \\
 r_{k-1,k} &= \langle u_k^{(k-2)}, q_{k-1} \rangle, & u_k &= u_k^{(k-1)} = u_k^{(k-2)} - r_{k-1,k}q_{k-1}, \\
 r_{k,k} &= \|u_k\|, & q_k &= \frac{u_k}{r_{k,k}}.
 \end{aligned}$$

Similar to the classical Gram-Schmidt, orthonormal q_i ’s are generated sequentially starting from $i = 1$ to $i = n$. As an example, to calculate q_3 , one would first calculate $u_3^{(1)}$ and $u_3^{(2)}$ using:

$$\begin{aligned}
 r_{1,3} &= \langle v_3, q_1 \rangle, & u_3^{(1)} &= v_3 - r_{1,3}q_1, \\
 r_{2,3} &= \langle u_3^{(1)}, q_2 \rangle, & u_3 &= u_3^{(2)} = u_3^{(1)} - r_{2,3}q_2, \\
 r_{3,3} &= \|u_3\|, & q_3 &= \frac{u_3}{r_{3,3}}.
 \end{aligned}$$

Answer the following questions:

a. Write MATLAB functions `cgs.m` and `mgs.m` that perform the classical and the modified Gram-Schmidt orthonormalization respectively. In particular, your functions should take a single matrix $V(m \times n)$, $m \geq n$ as the input and return the matrices Q (matrix with columns q_i , $i = 1, 2, \dots, n$) and R (upper triangular matrix with elements $r_{i,j}$) as the two outputs.

b. Generate the matrix $V(m \times m)$ following these steps in MATLAB:

```

rand('state',100)
A = qr(rand(m));
B = qr(rand(m));
S = diag(2.^(1:m));
V = A*S*B';

```

c. For each of the following values of m : 5, 10, 50, 100, 200, 250, 500, 750, 1000, generate the V matrix using part (b), and compute the corresponding Q, R matrices as:

```

[Qcgs,Rcgs] = cgs(V);
[Qmgs,Rmgs] = mgs(V);

```

As error indicators, we will use the deviation of $Q'Q$ from the identity $I(m \times m)$.

For each of the above cases, compute the errors:

$$\text{ecgs} = \text{norm}(\text{Qcgs}' * \text{Qcgs} - \text{eye}(m));$$

$$\text{emgs} = \text{norm}(\text{Qmgs}' * \text{Qmgs} - \text{eye}(m));$$

Plot these errors as a function of m . What do you observe? Explain briefly. Is one method superior to the other?

d. For $m = 100$, plot the diagonal entries of Rcgs and Rmgs on a semilog scale, i.e. for example:

$$\text{semilogy}(1:m, \text{diag}(\text{Rcgs}), 'o');$$

What do you observe? **BONUS:** Try to explain the behavior you observe.

Problem 8: Root finding, Finite Difference, and Time integration.

A significant model of population growth was derived in 1838 by Belgian Pierre-Francois Verhulst. His model eliminated the unrealistic effects of unlimited exponential growth. He suggested that when a population increases, there is a tendency for individuals to compete with each other and so reduce growth. His model has come to be known as the logistic growth equation.

Consider a turtle's population modeled by such a logistic growth equation:

$$\frac{\partial x}{\partial t} = f(x, t) = rx \left(1 - \frac{x}{K} \right)$$

where: x is the number of turtles in the population
 $K = 700$ is the carrying capacity
 $r = 0.1$ is the growth rate

- Using Taylor series, apply the forward, backward, and central differences to the first order (time) derivative in the above equation and derive the corresponding discrete time-integration schemes. Re-arrange the terms such that the unknown values (at future time steps) are on the left-hand side and known values are on the right-hand side.
- The backwards difference formula requires the evaluation of the function $f(x, t)$ at a time in the future where the value of x is unknown. This results in a root-finding problem. Write a function **NewtonRaphson** that has a header of the type:

Function $x = \text{NewtonRaphson}(f, df, x0, TOL, maxit)$

Where: f is the "function handle" of the function whose root you require
 df is the "function handle" of the derivative of f
 $x0$ is an initial guess value and x the final value
 TOL is the error tolerance
 $Maxit$ is the maximum number of iterations

A skeleton of this function can be downloaded from the course website site.

- c) Compute the number of turtles from time 0 to time 120 years using the forward, central, and backwards difference schemes. Use 30, 60, 120, and 240 time intervals (that is $\Delta t = 120 / 30$, for example). For each of these four time step cases, plot the “Population vs. Time” values of the backward, central, and forward difference schemes and of the analytical solution (all four estimates on the same graph for each time step case). Discuss the results as the number of time intervals increase. Initially, we have two turtles and the exact analytical solution is given by:

$$x(t) = \frac{2 e^{rt}}{1 + 2 (e^{rt} - 1) / K}$$

We mainly provide you this so you can initialize the time integration with the central difference (you need two time-steps to start the scheme). You can also use the forward difference solution to initialize (which would be common practice), either way is fine.

- d) Plot the L^2 norm of the error versus the number of time steps used for the three schemes on the same graph. Use a loglog() plot.
For the L^2 norm use: $\text{norm}(x-x_{\text{exact}})/\text{sqrt}(\# \text{ intervals}+1)$. Here x is a vector containing the solution at each timestep. That is, x has a length= (#intervals+1). So, calculating the error for forward difference using 30 intervals you will get a single number.
- e) Determine the order of convergence for each scheme. Do they converge at the expected order?
- f) For which scheme and for which number of divisions do you obtain the fewest turtles at 100 years? (Do you think your colleagues would be happy if you used this numerical solution in your population growth research?)

MIT OpenCourseWare
<http://ocw.mit.edu>

2.29 Numerical Fluid Mechanics

Spring 2015

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.