MA5233 Homework 1

(Due date: 10:00pm, September 26, 2016 (Monday))

1. Explore truncation error and rounding error, and resolution and accuracy. We seek to solve the following integral equation:

$$f(x) = \int_0^1 \left[t + \sin(x\, t^2) \right] \, dt.$$

(a) Write a program (in Matlab or other languages) to estimate the integral using a composite integration method (such as composite midpoint rule, composite trapezoidal rule and composite Simpson's rule), with the number of intervals $n = \frac{1}{\Delta t}$ as input. The program should be well documented so that other person could easily understand it and easily substitute a different quadrature by changing a few lines of the code. (You can use standard subroutine such as Numerical Recipes, Netlib://http.netlib.org; Matlab: type help funfun).

(b) Verify the correctness of the program by checking that it gives the right answer for small x. We can estimate the integral for small x using a few terms of its Taylor series. This series can be computed by integrating the Taylor series of $\sin(x\, t^2)$ term by term. Turn in a table showing your Taylor series approximation and the value returned by your code for a few small values of x, say, $x = 0.1, 0.2, 0.3$.

(c) With $x = 1$, do a convergence study to verify the second order accuracy of the composite trapezoidal rule (trapzd.f routine of Numerical Recipes, Section 4.2 or your own code made in question (a) in other programming languages) and the fourth order accuracy of the composite Simpson's rule (qsimp.f routine of Numerical Recipes, Section 4.2 or your own code in other programming languages). Turn in a table showing the results for different step size Δt, or a log-log plot of the error, and explain how your results demonstrate the correct order of accuracy. Try different ways if checking accuracy such as comparing with "correct" answer computed numerically by a very fine grid; or using the relation of

$$\frac{A(4\Delta t) - A(2\Delta t)}{A(2\Delta t) - A(\Delta t)}.$$

(d) As in (c), what happens when n is very large? Find the value n_0 such that the accuracy check in (c) fails when $n \geq n_0$. What is the connection with the round-off error? You will also find that accuracy check in (c) fails when n is very small. This is related to the resolution and will be explained in (e).

(e) For large values of x, the integrand will undergo many oscillations within the limits of the integration. Therefore, in order to achieve accuracy, the value of Δt must decrease in order to resolve the features of the integrand. Perform a convergence study for large x, say x = 100, 1000, 10000, and find out how many grid points per-wave structure in the integrand is needed in order to obtain a clean accuracy check. You shall use relative error in the convergence study.

(f) Write a routine that uses the composite trapezoidal rule and Richardson extrapolation to compute the integral to within a specified (absolute) error tolerance ε, e.g. $\varepsilon = 10^{-12}$. The desired error bound should be input. The output should be the estimated value of the integral and the number of function evaluations used. You can use or modify the routine qtrap.f from Numerical Recipes, Section 4.2. This routine should be robust enough to quit and report failure if it is unable to achieve the requested accuracy. Turn in a few examples with your input and output from the routine. Try to find an example where your code reports failure.

(g). For large x, you may find an asymptotic approximation for the intrgation using $\int_0^1 \sin(x\,t^2)\,dt = \int_0^\infty \sin(x\,t^2)\,dt - \int_1^\infty \sin(x\,t^2)\,dt$ and doing integration by parts for the second term (noting $\int_0^\infty \sin(z^2)dz = \sqrt{\pi/8}$). Make a few plots showing f and its approximations using one, two and all three terms on the right side of the above approximate formula for f with x in the range $1 \le x \le 1000$. In all cases we want to evaluate f so accurately that the error in our f value is much less than the error of the approximate formula. Note that even for a fixed level of accuracy, more points are needed for large x as explained in (e).

2. Consider an $n \times n$ tridiagonal matrix of the form

$$T_\alpha = \begin{pmatrix} \alpha & -1 & 0 & \dots & 0 \\ -1 & \alpha & -1 & \dots & 0 \\ 0 & -1 & \alpha & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \alpha \end{pmatrix},$$

where α is a real parameter.

(a) Verify that the eigenvalues of T_α are given by

$$\lambda_j = \alpha - 2\cos(j\theta), \qquad j = 1, 2, \dots, n,$$

where $\theta = \frac{\pi}{n+1}$ and that an eigenvector associated with each eigenvalue λ_j is

$$\mathbf{q}_j = [\sin(j\theta), \sin(2j\theta), \dots, \sin(nj\theta)]^T \in \mathbb{R}^n.$$

Under what condition on α does this matrix become positive definite?

(b) Now we take $\alpha = 2$. Will the Jacobi and Gauss-Seidel iterations converge for this matrix? For which values of ω will the SOR iteration converge?

3. The symmetric successive over-relaxation (SSOR) iteration for solving the linear system $A\mathbf{x} = \mathbf{b}$ with $A \in \mathbb{R}^{n\times n}$ and $\mathbf{b} \in \mathbb{R}^n$ is

```
for $i = 1, n$
    $$x_i^{(m+1/2)} = (1-\omega)x_i^{(m)} + \frac{\omega}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(m+1/2)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(m)}\right),$$
for $i = n, -1, 1$
    $$x_i^{(m+1)} = (1-\omega)x_i^{(m+1/2)} + \frac{\omega}{a_i i}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(m+1/2)} - \sum_{j=i+1}^{n} a_{ij}x_j^{(m+1)}\right).$$
```

Prove that the iteration matrix R_ω of the SSOR iteration, defined as

$$R_\omega = (D - \omega\tilde{U})^{-1}\left[\omega\tilde{L} + (1-\omega)D\right](D - \omega\tilde{L})^{-1}\left[\omega\tilde{U} + (1-\omega)D\right],$$

can be expressed as

$$R_\omega = I - \omega(2-\omega)(D-\omega\tilde{U})^{-1}D(D-\omega\tilde{L})^{-1}A.$$

4. What can you say about the convergence of the Jacobi iteration if A is symmetric positive definite? Prove the convergence or create a counter-example.

5. Explore the Gauss-Seidel, SOR, steepest decent and conjugate gradient methods. Approximate the following two dimensional (2D) Poisson equation with Dirichlet boundary condition:

$$\begin{aligned}
&-\Delta u(x,y) = -\partial_{xx}u(x,y) - \partial_{yy}u(x,y) = f(x,y), \qquad 0<x<1, \quad 0<y<1,\\
&u(0,y) = u(1,y) = 0, \qquad 0 \le y \le 1,\\
&u(x,0) = u(x,1) = 0, \qquad 0 \le x \le 1;
\end{aligned}$$

by the standard second-order central finite difference scheme:

$$\begin{aligned}
&-\frac{1}{h^2}\left[u_{i-1,j} + u_{i,j-1} - 4u_{i,j} + u_{i+1,j} + u_{i,j+1}\right] = f(x_i,y_j), \qquad i,j = 1,2,\ldots,N-1,\\
&u_{i,0} = u_{i,N} = 0, \qquad i = 0,1,2,\ldots,N,\\
&u_{0,j} = u_{N,j} = 0, \qquad j = 1,2,\ldots,N-1;
\end{aligned}$$

where $h = \frac{1}{N}$ is the mesh size, $x_i = i\,h$ $(i = 0,1,2,\ldots,N)$ and $y_j = jh$ $(j = 0,1,2,\ldots,N)$ are the computational grid points, and $u_{i,j}$ is an approximation of $u(x_i,y_j)$.

(a) Write the difference scheme as a linear system $A\,\mathbf{u} = \mathbf{b}$ (where the component of $\mathbf{u}$: $u_{i,j}$, $j = 1,2,\ldots,N-1$, $i = 1,2,\ldots,N-1$) and show that A is positive definite (use similar approach as in Problem 2).

(b). Sketch the steps and write a program for Gauss-Seidel, SOR with different relaxation constant $1 < \omega < 2$, steepest decent and conjugate methods solving the linear system $A\,\mathbf{u} = \mathbf{b}$. The program should contain two subroutines. One computes the inner product $\mathbf{u}^T\mathbf{v}$ and the other multiplies a matrix by a vector (so you are able to avoiding formulating the matrix A in your code).

(c) Choose $f(x,y) = \sin(5\pi x)\sin(7\pi y)$. Take small (e.g. $N = 10$ or 20) and large N (e.g. $N = 100$ or 500 or 1000) to summarize the rate of convergence of these three methods in the residual.

(d) What conclusion can you get from your computations?