Numerical Solutions of Differential Equations - Project #1

due 2020 NOV 03, 2:55 p.m.

1 The Assignments

I. Write a `Matlab` or `C++` program to plot the damping coefficients of two-grid correction with weighted Jacobi. Output six plots for $n = 64$ and the cases of $(\nu_1, \nu_2) = (0, 0), (0, 2), (1, 1), (2, 0), (2, 2), (4, 0)$.

II. Write a `C++` package to implement the multigrid solver discussed in the notes for the one-dimensional Poisson equation

$$u''(x) = f(x) \tag{1}$$

on $\Omega = [0, 1]$ with the Dirichlet boundary condition $u(0) = u_0$ and $u(1) = u_1$. Your package must give the user the following options:

(a) boundary conditions: homogeneous and nonhomogeneous;

(b) restriction operators: full-weighting and injection ($v_j^{2h} = v_{2j}^h$);

(c) interpolation operators: linear and quadratic;

(d) cycles: V-cycle and full multigrid cycle;

(e) stopping criteria: the number of maximum iterations and the relative accuracy ϵ of the solution;

(f) the initial guess.

As for the bottom solver, you can either implement a Gaussian elimination in your package or use the one in the BLAS or LAPACK package.

III. For the function

$$u(x) = \exp(\sin(x)), \tag{2}$$

derive the corresponding $f(x)$ and the boundary conditions. For $\epsilon = 10^{-8}$ and the zero-vector initial guess, test your multigrid solver for all combinations of (b,c,d) in II on grids with $n = 128, 256, 512, 1024$, report the residual and the reduction rate of the residuals for each V-cycle. Report the maximum norm of the error vector and the corresponding convergence rates on the four grids.

IV. Gradually reduce ϵ towards 2.2×10^{-16}, under which critical value of ϵ does your program fail to achieve the preset accuracy? Why?

V. Design a test with homogeneous boundary conditions and repeat III.

VI. (Optional) Augment your `C++` package to two dimensions on $\Omega = [0, 1]^2$ and repeat II-V. For III, use the exact solution

$$u(x, y) = \exp(\sin(x) \sin(y)) \tag{3}$$

to derive the Dirichlet conditions.

2 How to submit

Your submission must contain

(a) the LaTeX source code and its Makefile so that the command "`make answers`" generates a document that contains detailed answers to problems in Section 1,

(b) the LaTeX source code of your math document so that the command "`make math`" generates the math document of your own story on geometric multigrid.

(c) the LaTeX source code of your design document so that the command "`make design`" generates the design document that contains the class diagrams and other UML diagrams describing the interaction of classes.

(d) a `C++` package so that the command "`make run`" would trigger the compilation of your source code, the production of the executable, the running of your tests, and the display of test results.

You should archive your source code in a single gzipped tar ball (**format**: `YourName_project1.tar.gz`) and send it to the TA's email. A number of tips are given as follows.

(i) You can use either `GNU Make` or `cmake` or a mixture of them.

(ii) You can use Chinese or English for the math document and the design document.

(iii) Your gzipped tar ball should neither contain anything that can be generated from your Makefile, nor contain anything irrelevant to this homework. In other words, your answers to this homework should be both *sufficient* and *necessary.*

(iv) You are encouraged to use the unit test framework `CppUnit`, which is required for my graduate students.

Problem I weighs 10 points and problems II-V weigh 40 points. Each of the math document and the design document weighs 20 points. The quality of your `C++` code weighs 10 points. Problem VI weighs 20 points and is optional, but is required for my graduate students.