APMA 4302 – Homework 2 Solutions

Problem 1: Condition Number and Error Bounds (10 pts)

The condition number of a matrix A is $\kappa(A) = \|A\|\|A^{-1}\|$, where $\|Ax\| \leq \|A\|\|x\|$.

(a) Assume A is invertible and $Au = b$. Show that the relative error in the solution satisfies

$$\frac{\|u - \hat{u}\|}{\|u\|} \leq \kappa(A) \frac{\|b - \hat{b}\|}{\|b\|},$$

where $\hat{u}$ solves $A\hat{u} = \hat{b}$ (perturbed right-hand side).

(b) Suppose $Au = b$ and $Ae = r$. Show that

$$\frac{1}{\kappa(A)} \frac{\|r\|}{\|b\|} \leq \frac{\|e\|}{\|u\|} \leq \kappa(A) \frac{\|r\|}{\|b\|},$$

where $e = u - \hat{u}$ and $r = b - A\hat{u}$ (residual). Give an interpretation.

Solution

Part (a). We have $u = A^{-1}b$ and $\hat{u} = A^{-1}\hat{b}$, so $u - \hat{u} = A^{-1}(b - \hat{b})$. By the matrix norm property,

$$\|u - \hat{u}\| = \|A^{-1}(b - \hat{b})\| \leq \|A^{-1}\|\|b - \hat{b}\|.$$

Also $b = Au$ gives $\|b\| \leq \|A\|\|u\|$, so $\|u\| \geq \|b\|/\|A\|$. Hence

$$\frac{\|u - \hat{u}\|}{\|u\|} \leq \frac{\|A^{-1}\|\|b - \hat{b}\|}{\|u\|} \leq \|A^{-1}\|\|b - \hat{b}\| \cdot \frac{\|A\|}{\|b\|} = \kappa(A) \frac{\|b - \hat{b}\|}{\|b\|}.$$

Part (b). By definition, $r = b - A\hat{u} = A(u - \hat{u}) = Ae$, so $Ae = r$.

Upper bound: $e = A^{-1}r$ implies $\|e\| \leq \|A^{-1}\|\|r\|$. With $\|b\| \leq \|A\|\|u\|$ we get $\|u\| \geq \|b\|/\|A\|$. Thus

$$\frac{\|e\|}{\|u\|} \leq \kappa(A) \frac{\|r\|}{\|b\|}.$$

Lower bound: $r = Ae$ gives $\|r\| \leq \|A\|\|e\|$, so $\|e\| \geq \|r\|/\|A\|$. Also $\|u\| = \|A^{-1}b\| \leq \|A^{-1}\|\|b\|$. Therefore

$$\frac{\|e\|}{\|u\|} \geq \frac{1}{\kappa(A)} \frac{\|r\|}{\|b\|}.$$

Interpretation. The relative error $\|e\|/\|u\|$ is bracketed by the relative residual $\|r\|/\|b\|$ scaled by $1/\kappa(A)$ and by $\kappa(A)$. A small residual can still correspond to a large relative error when $\kappa(A)$ is large (ill-conditioning). For well-conditioned A, error and residual are of the same order.

Problem 2: Discrete Laplacian Matrix (10 pts)

Let

$$A = \frac{1}{h^2}\,\mathrm{tridiag}(-1, 2, -1)$$

be the symmetric positive definite matrix from the centered finite-difference discretization of $-u''(x)$ on $x \in [0, 1]$ with homogeneous Dirichlet boundary conditions. Here $h = 1/m$ and A is $(m-1) \times (m-1)$.

(a) Show that the eigenvectors of A are $v_j(i) = \sin(j\pi x_i)$ with $x_i = ih$, $i = 1, \ldots, m-1$.

(b) Find the corresponding eigenvalues λ_j.

(c) Show that $\kappa(A) \sim O(m^2)$ as $m \to \infty$.

Solution

Part (a): Eigenvectors. Apply the discrete Laplacian to v_j at interior index i. The stencil $(1/h^2)(-1, 2, -1)$ gives

$$(Av_j)(i) = \frac{1}{h^2}\big[-v_j(i-1) + 2v_j(i) - v_j(i+1)\big].$$

Using $x_i = ih$ and $v_j(i) = \sin(j\pi x_i) = \sin(j\pi ih)$:

$$-\sin(j\pi(i-1)h) + 2\sin(j\pi ih) - \sin(j\pi(i+1)h) = 2\sin(j\pi ih) - \big[\sin(j\pi ih - j\pi h) + \sin(j\pi ih + j\pi h)\big].$$

By $\sin(\alpha - \theta) + \sin(\alpha + \theta) = 2\sin(\alpha)\cos(\theta)$ with $\alpha = j\pi ih$, $\theta = j\pi h$:

$$= 2\sin(j\pi ih) - 2\sin(j\pi ih)\cos(j\pi h) = 2\sin(j\pi ih)\big[1 - \cos(j\pi h)\big] = 4\sin(j\pi ih)\,\sin^2(j\pi h/2).$$

Thus $(Av_j)(i) = (4/h^2)\sin^2(j\pi h/2)\,v_j(i)$, so v_j is an eigenvector with eigenvalue $\lambda_j = (4/h^2)\sin^2(j\pi h/2)$.

Part (b): Eigenvalues.

$$\lambda_j = \frac{4}{h^2}\sin^2\Big(\frac{j\pi h}{2}\Big) = \frac{4}{h^2}\sin^2\Big(\frac{j\pi}{2m}\Big), \qquad j = 1, \ldots, m-1.$$

Equivalently, $\lambda_j = 4m^2\sin^2(j\pi/(2m))$.

Part (c): Condition number. For SPD A, $\kappa(A) = \lambda_{\max}/\lambda_{\min}$. The largest eigenvalue is $\lambda_{m-1} = (4/h^2)\sin^2((m-1)\pi/(2m)) \to 4/h^2 = 4m^2$ as $m \to \infty$. The smallest is $\lambda_1 = (4/h^2)\sin^2(\pi/(2m))$; for large m, $\sin(\pi/(2m)) \approx \pi/(2m)$, so $\lambda_1 \approx \pi^2$. Hence $\kappa(A) \sim (4m^2)/\pi^2 = O(m^2)$ as $m \to \infty$.

Problem 3: Boundary Value Problem with PETSc (20 pts)

BVP: $-u''(x) + \gamma u(x) = f(x)$ on $x \in [0, 1]$ with Dirichlet boundary conditions.

(a) Find $f(x)$ for the manufactured solution

$$u(x) = \sin(k\pi x) + c\Big(1 - \frac{1}{2}\Big)^3,$$

where k is a positive integer and c is a real constant.

(b) Modify `tri.c` (p4pdes Ch. 2) to solve this BVP with PETSc: run with `mpiexec -np P ./bvp -options_file options_file`; assemble matrix and RHS in parallel; use `MatZeroRowsColumns` to enforce Dirichlet BCs; compute and print relative error; output solution, exact solution, and RHS to HDF5; use `plot_bvp.py` to visualize.

(c) Modify `plot_bvp.py` to plot error vs. h for $\gamma = 0$, $k = 1, 5, 10$, and $m = 40, 80, 160, \ldots, 1280$. Report observed order of convergence.

Solution

Part (a): Manufactured solution and $f(x)$. With the given manufactured solution we have

$$u(x) = \sin(k\pi x) + c\Big(1 - \frac{1}{2}\Big)^3 = \sin(k\pi x) + \frac{c}{8}.$$

Differentiating: $u'(x) = k\pi \cos(k\pi x)$ and $u''(x) = -k^2\pi^2 \sin(k\pi x)$. Substituting into $-u'' + \gamma u = f(x)$ gives

$$-u'' + \gamma u = k^2\pi^2 \sin(k\pi x) + \gamma \sin(k\pi x) + \frac{\gamma c}{8},$$

hence

$$\boxed{f(x) = (k^2\pi^2 + \gamma)\sin(k\pi x) + \frac{\gamma c}{8}.}$$

Note. If the intended solution was $u(x) = \sin(k\pi x) + c(1 - x)^3$, then f would include extra terms from the $(1 - x)^3$ part.

3(b) PETSc solution and verification plot

I modified `tri.c` to solve

$$-u''(x) + \gamma u(x) = f(x), \qquad x \in [0, 1],$$

with Dirichlet boundary conditions, using PETSc options handling and parallel assembly. Dirichlet conditions were enforced in a symmetry-preserving way using `MatZeroRowsColumns` after assembling the operator and vectors. The code computes and prints the relative error in the solution and writes `u`, `uexact`, and `f` to an HDF5 file for plotting.

Figure 1 compares the numerical and exact solutions and shows the pointwise error. The numerical and exact curves are visually indistinguishable at the plot scale, and the error remains small (on the order of 10^{-5} in this run).

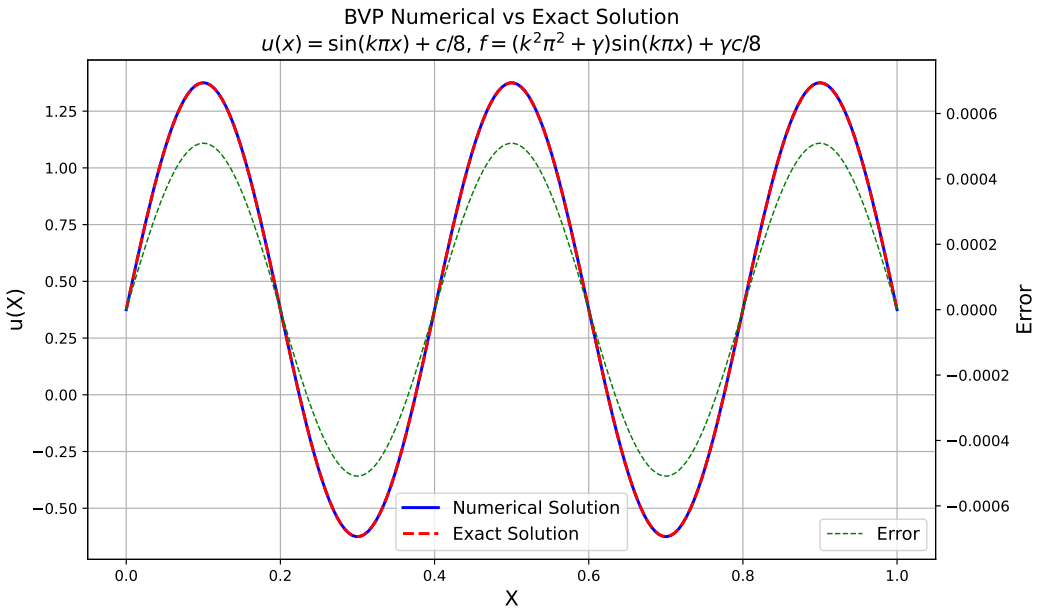

Figure 1: BVP numerical vs exact solution and the pointwise error for the PETSc solve in Q3(b).

3(c) Convergence study and observed order

To study convergence, I ran the solver with $\gamma = 0$ and

$$m = 40, 80, 160, \ldots, 1280, \qquad h = \frac{1}{m},$$

for $k \in \{1, 5, 10\}$, and computed the relative error in the 2-norm for each grid. Figure 2 plots relative error versus h on log-log axes. The lines are approximately straight with slope ≈ 2, indicating

$$\|e\|_2 = O(h^2),$$

which matches the expected second-order accuracy of the centered finite difference discretization.

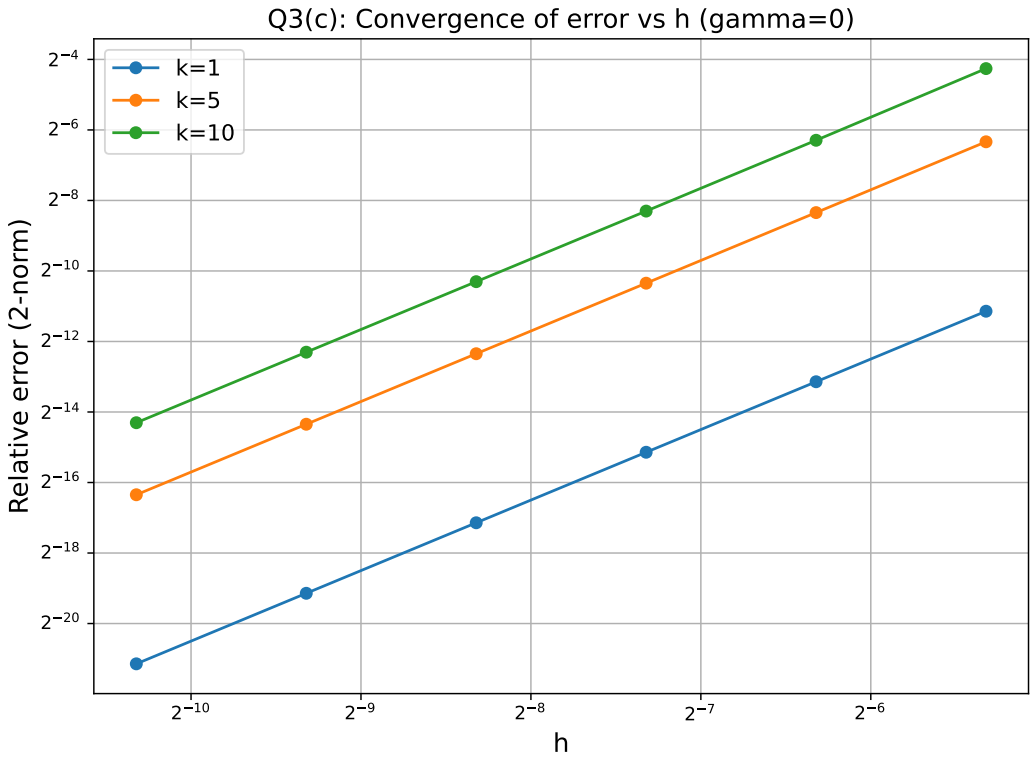

Figure 2: Q3(c): Convergence of relative error (2-norm) versus h for $\gamma = 0$ and $k = 1, 5, 10$. The observed slope is approximately 2, indicating second-order convergence.

Problem 4: Solver Performance (10 pts)

Using the default options file, determine the number of iterations to converge for each solver and explain your results.

(a) Jacobi-preconditioned Richardson: `-ksp_type richardson -pc_type jacobi`
(b) Unpreconditioned CG, 1 proc: `-ksp_type cg -pc_type none`
(c) Unpreconditioned CG with $c = 0$.
(d) ICC-preconditioned CG, 1 proc: `-ksp_type cg -pc_type icc`
(e) Block Jacobi + ICC, 4 procs: `-ksp_type cg -pc_type bjacobi -pc_sub_type icc`
(f) MUMPS direct, 1 and 4 procs: `-ksp_type preonly -pc_type lu -pc_factor_solver_type mumps`

Problem 4: Iteration counts and explanation (default `options_file`)

I ran `./run_q4.sh` using the default PETSc options file and recorded the KSP convergence reason, iteration count, and final relative error (2-norm). The results are summarized in Table 1.

Part	Method / Preconditioner	Procs	KSP reason	Iters	Rel. error (ℓ_2)
(a)	Richardson + Jacobi	1	DIVERGED_ITS	10000 (max it)	1.159476×10^{-1}
(b)	CG, no PC	1	CONVERGED_RTOL	101	4.193903×10^{-4}
(c)	CG, no PC, $c = 0$	1	CONVERGED_RTOL	1	5.090952×10^{-4}
(d)	CG + ICC	1	CONVERGED_ATOL	1	4.193903×10^{-4}
(e)	CG + BJACOBI(ICC)	4	CONVERGED_RTOL	7	4.193903×10^{-4}
(f)	LU (MUMPS)	1	(direct)	–	4.193903×10^{-4}
(f)	LU (MUMPS)	4	(direct)	–	4.193903×10^{-4}

Table 1: Q4 iteration counts and relative errors from `./run_q4.sh` using the default options file.

(a) Richardson + Jacobi. Stationary iterations such as Richardson converge slowly for elliptic operators unless the relaxation parameter is well tuned and the spectrum is tightly clustered. For the 1D discrete Laplacian, the condition number grows like $\kappa(A) = O(m^2)$, so the iteration count required to reach a fixed tolerance can be very large. With only Jacobi scaling, the solver hit the maximum iteration cap (`DIVERGED_ITS` at 10000 iterations), and the resulting solution is noticeably less accurate.

(b) Unpreconditioned CG (1 proc). For SPD systems, CG convergence depends on the spectrum (equivalently the effective condition number). With no preconditioner, the iteration count grows roughly like $O(\sqrt{\kappa(A)})$, which is consistent with the observed 101 iterations to reach `CONVERGED_RTOL`. The final relative error is 4.19×10^{-4}.

(c) Unpreconditioned CG with $c = 0$. Setting $c = 0$ changes the manufactured solution and therefore the right-hand side, but does not change the matrix. CG can converge unusually fast if the initial error (or right-hand side in the zero-initial-guess case) lies in a very low-dimensional invariant subspace of A (e.g., dominated by a single eigenmode). In this run, CG converged in 1 iteration (to `RTOL`), but the final discretization error remains on the order of 10^{-4}, comparable to (b). The slightly different reported relative error (5.09×10^{-4}) reflects the changed exact solution and normalization, not a fundamentally different discretization accuracy.

(d) CG + ICC (1 proc). Incomplete Cholesky (ICC) is a strong preconditioner for this 1D SPD operator, substantially reducing the effective condition number. The solver converged in 1 iteration and reported `CONVERGED_ATOL`. The relative error matches (b), indicating the discretization error dominates once the linear solve is sufficiently accurate.

(e) CG + block Jacobi(ICC) (4 procs). Block Jacobi applies ICC within each subdomain block, which is typically weaker than a global ICC but is parallel-friendly. As expected, it requires more iterations than (d) but far fewer than (b): 7 iterations to `CONVERGED_RTOL`. The relative error again matches (b), consistent with discretization error domination.

(f) MUMPS direct solve (1 and 4 procs). With `-ksp_type preonly -pc_type lu` and MUMPS as the factorization backend, PETSc performs a direct solve (no Krylov iterations). The relative error matches the iterative methods when they converge tightly, again indicating the discretization error is the limiting factor for accuracy at this grid resolution.