Homework Assignment for Week 16

1. Section 7.4: Problems 7(c), 13.

2. Read Theorem 7.11 for evaluation of $\|A\|_\infty$.

3. Let A be the $(m+1)\times(m+1)$ matrix from Homework 13, Problem 4 (with $K(x,t) = -\frac{1}{2}e^{-|x-t|}$). Show that $\|T_j\|_\infty \approx 1 - \frac{1}{\sqrt{e}}$ for large m and therefore Jacobi iteration converges. Then estimate the total number of operations (multiplication/division, to leading order) for Jacobi iteration to reach $\|e^{(k)}\|_\infty = h^2$ (assuming $\|e^{(0)}\|_\infty = 1$) where $h = (b-a)/m = \frac{1}{m}$ and $e^{(k)} = u^{(k)} - u_e$. Compare the operation count with that of the Gaussian Elimination/LU decomposition approach.

4. Let A be the matrix from Problem 4, Homework 15.

   (a) Give the operation count to leading order (on multiplication/division) for one iteration of Jacobi, Gauss Seidel and SOR on this A, respectively.

   (b) It is a fact (the proof is beyond this course) that **for this** A, $\rho(T_j) \approx 1 - \frac{\pi^2}{2}h^2$, $\rho(T_g) \approx 1 - \pi^2 h^2 \approx \rho(T_j)^2$. Note that both numbers are very close to 1. Take this fact for granted, estimate the number of iterations Jacobi and Gauss-Siedel take to reach $\|e^{(k)}\| = h^2$, respectively (assuming $\|e^{(0)}\| = 1$) where $e^{(k)} = u^{(k)} - u_e$. Then estimate the total number of multiplications/divisions needed for Jacobi and Gauss-Siedel, respectively. You will need that $\log(1+x) \approx x$ for $|x| << 1$.

   (c) It is another fact (the proof is also beyond this course) that with the optimal $\omega = \omega^* \approx 2 - 2\pi h$, we will have $\rho(T_{\omega^*}) \approx 1 - 2\pi h$ for SOR (for this A). Take this fact for granted again, estimate the number of iterations and total multiplications/divisions the optimal SOR takes to reach $\|e^{(k)}\| = h^2$ with $\|e^{(0)}\| = 1$.

   (d) The facts that $\rho(T_j) \approx 1 - C_1 h^2$, $\rho(T_g) \approx 1 - C_2 h^2$ and $\rho(T_{\omega^*}) \approx 1 - C_3 h$ remain valid in the 3D case, with different constants C_i. Repeat the above problems for the 3D case.

   Compare these results (operation count) with that of the Gaussian Elimination/LU decomposition approach.

5. Derive the Jacobi version of SOR. Express $T_{\omega,j}$ in terms of T_j.

6. Section 7.5: Problems 1(a,c), 3(a,c), 11, 12(a).

7. Section 7.5: The Hilbert matrices are well known ill-conditioned matrices.

   In addition to the example provided in problem 9, you can use the matlab built-in command 'hilb' and 'cond' to find the condition numbers of the Hilbert matrices $H^{(n)}$ in problem 9 directly (i.e. without finding $(H^{(n)})^{-1}$ first). Do this for $n = 6, \cdots, 12$ and observe how fast the condition number grows with n.

8. Section 8.1: Problems 2, 14.

9. Derive the continuous version of least square problem:

   Give n and $f(x) : [0,1] \mapsto R$, find $a_0, \cdots a_n$ to minimize the quantity

   $$\int_0^1 (f(x) - (a_0 + a_1 x + \cdots a_n x^n))^2 \; dx$$

   Derive the normal equation for the coefficient vector $(a_0, \cdots a_n)$.

   Remark: The matrix corresponding to this linear system is ill-conditioned for large n (why?). The discrete counter part, problem 14, is similarly ill-conditioned for large n. The proper way of solving these problems numerically for large n, say $n > 5$, can be found in section 8.2.