Numerical Linear Algebra QS 6 *(MT 2019)*

Optional questions: Qns 1, 6

1. Show that the Chebyshev Polynomials $T_n(x) = \frac{1}{2^{n-1}} \cos n \left(\cos^{-1} x\right)$ satisfy

$$\int_{-1}^{1} \frac{T_n(x)\, T_m(x)}{\sqrt{1-x^2}}\, dx = 0$$

when $n \neq m$.

This shows that the Chebyshev polynomials are the orthogonal polynomials with respect to the inner product

$$< p, q >= \int_{-1}^{1} \frac{p(x)\, q(x)}{\sqrt{1-x^2}}\, dx$$

ie. $T_n \in \Pi_n, n = 0, 1, \ldots$ and $< T_n, T_m >$ is equal to zero except when $n = m$ (when it is clearly positive.)

2. If $A \in \mathbb{R}^{n\times n}$ is nonsingular, show that GMRES breaks down at the ℓ^{th} iteration (ie. $h_{\ell+1,\ell} = 0$) if and only if $x_\ell = x$ (ie. if and only if the solution of the linear system has been found).

3. Let $Q_k R_k$ be a QR factorization of $\widehat{H}_k$ where $Q_k = J_1, J_2, \ldots, J_k$ with J_j being a single $(j+1) \times (j+1)$ Givens rotation matrix for each j. If $\widehat{H}_{k+1}$ is computed from $\widehat{H}_k$ by appending the one further column computed by the next step of the Arnoldi algorithm, show that only one further Givens rotation J_{k+1} gives the QR factorization of $\widehat{H}_{k+1}$.

4. Continuing from the question above: If $s = \sin\theta$ in the Givens rotation in J_{k+1}, show that

$$\|r_k\|_2 = |s| \|r_{k-1}\|_2.$$

Hence for the sequence of succesive residuals $r_k, k = 0, 1, 2, \ldots$ computed by the GMRES method, $\{\|r_k\|_2, k = 0, 1, 2, \ldots\}$ must reduce monotonically. Are there any circumstances in which the convergence is not strictly monotonic?

5. Show how GMRES will converge on the linear system $Ax = b$ with $x_0 = 0$ when

$$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

Hand calculation (it is simple in this example to work out what the residual vectors must be!) is best here if you want to learn something!

6. If

$$A = \begin{bmatrix} I & B_2 & & & & \\ & I & B_3 & & & \\ & & I & \ddots & & \\ & & & I & B_{k-1} & \\ & & & & I & B_k \\ & & & & & I \end{bmatrix}$$

for arbitrary submatrices B_i of appropriate dimension, show that $(I - A)^k = 0$.

What is the maximum number of steps that unrestarted GMRES would take to solve a linear system with matrix A?

7. Use matlab (`[x,flag,relres,iter,resvec]=gmres(A,b,[],1.e-6,size(A,1))`) with suitably chosen matrices A and b as below to investigate the behaviour of GMRES.

Note in the form above matlab will use unrestarted GMRES, `flag=0` will indicate successful convergence (the relative residual norm - `relres` - less than 10^{-6} in less than dimension(A) `=size(A,1)` iterations), `iter` is the number of restarts (should be 1 with no restarting) and iterations taken and `resvec` is the vector of residual norms at each iteration (hence `semilogy(resvec)` will plot the convergence curve). See `help gmres` if you want to read more or change any of the defaults.

(i) `A=randn(n); b=ones(n,1);` for `n=7,47,...` as you choose (and have patience for! (note `ctrl C` will interrupt a computation). These are dense matrices!

(ii) `A=sprandn(100,100,0.1); b=ones(100,1);`. This is a sparse 100×100 matrix with approximately 10 non-zero entries per row (ie. `0.1` of the 10,000 entries non-zero).

(iii) `A=sprandn(100,100,0.1) +2*eye(100,100); b=ones(100,1);`

(iv) `A=sprandn(100,100,0.1) +4*eye(100,100); b=ones(100,1);`

(v) a diagonalisable matrix that has few distinct eigenvalues
eg. `X=randn(9,9); A=X*diag([1,1,-4,3,3,-4,-4,-4,3])/X`
(note `/X` is an more efficient way of computing `*inv(X)` and that it is possible that an X generated with random entries is singular, but is rarely so!)

(vi) any matrix

8. (entirely voluntary)

(I know this to be true (indeed for many matrices), but do not have as elementary a proof of it as I would like: hence I offer a pint of beer/gin&tonic/orange juice to the first person to show me a proof)

Prove that the Gauss-Seidel iteration matrix for the 5-point matrix is not diagonalisable.