Numerical Analysis - Institutionen för Matematik   SF2524, SF3580 - 2016-03-16, kl 08-12
KTH - Royal Institute of Technology

SF2524 Matrix Computations for Large-scale Systems
Exam - Solutions

Aids: None   Time: Four hours

Grades: E: 16 points, D: 19 points, C: 22 points, B: 25 points, A: 28 points (out of the possible 35 points, including bonus points from homeworks).

Problem 1 (4p)

(a) How is the Rayleigh quotient iteration for the matrix $A \in \mathbb{R}^{n\times n}$ defined? Answer with formulas and/or algorithm.

(b) Suppose $Q \in \mathbb{R}^{n\times m}$ is an orthogonal matrix ($Q^TQ = I$) with $n > m$. Describe the Gram-Schmidt (GS) procedure.

(c) Suppose Q is as in (b). Describe the double Gram-Schmidt (DGS) procedure.

(d) What are the advantages / disadvantages of GS vs DGS?

Solution:

(a) $r(x) := x^TAx/x^Tx$ and $z_{k+1} = (A - r(x_k)I)^{-1}x_k$ and $x_{k+1} = z_{k+1}/\|z_{k+1}\|$.

(b-c) GS: $h = Q^Tx$, $z = x - Qh$ and $y = z/\|z\|$
DGS: $h = Q^Tx$, $z = x - Qh$, $g = Q^Tz$, $z = z - Qg$ and $y = z/\|z\|$ set $h = h + g$.

(d) GS requires less operations than DGS but is more sensitive to round-off errors.

Problem 2 (4p)

(a) Let A be a non-singular matrix with eigenvalues in all four quadrants of complex plane. Give a definition of the matrix sign function for this matrix?

(b) Let $A \in \mathbb{R}^{n\times n}$. Suppose $X_0 = 2A$ and let X_k be defined by

$$X_{k+1} = \frac{1}{2}X_k + X_k^{-1}A, \quad k = 1, \ldots .$$

If the sequence $X_0, X_1, \ldots$ converges, what does it converge to?

(c) Suppose we have very reliable algorithms to compute the matrix functions f and g. Derive a formula that produces the first k derivatives at $x = 0$ of $h(x) = f(g(x))$ for scalar-valued x using matrix functions f and g.

Solution:

(a) It can be (easily) defined as $\operatorname{sign}(A) = A^{-1}\sqrt{A^2}$ or $\operatorname{sign}(A) = \sqrt{A^2}A^{-1}$. It can also be defined with the Jordan definition or the Cauchy integral formula (with $f(s) = s^{-1}\sqrt{s^2}$). It cannot be defined with a Taylor definition since the sign function is not analytic in the origin and we cannot find a disc including all the eigenvalues but not the origin.

(b) Newton's method for the matrix square root $\sqrt{B}$ is $X_{k+1} = \frac{1}{2}(X_k + X_k^{-1}B)$ and initialized with $X_0 = B$. We can directly identify the given algorithms as Newton's method with the choice $B = 2A$. Since Newton's method for the square root is globally convergent (if matrix square root is well-defined). Therefore, $X_k \to \sqrt{B} = \sqrt{2A} = \sqrt{2}\sqrt{A}$.
Alternative proof: Since this is a fixed point iteration, convergence can only occur to a fixed point. At the fixed point we have $X_* = \frac{1}{2}X_* + X_*^{-1}A$. The solution to this equation is $X_* = \sqrt{2}\sqrt{A}$.

(c) We can directly use the Jordan form definition as a procedure to compute the derivative: We set $J \in \mathbb{R}^{(k+1)\times(k+1)}$ and evaluate $h(J) = f(g(J))$. The first row will contain

$$\begin{bmatrix} h(0) & \frac{h'(0)}{1!} & \cdots & \frac{h^{(k)}(0)}{k!} \end{bmatrix}$$

from which we can directly read-off the derivatives.

Problem 3 (5p) Consider a matrix $A \in \mathbb{R}^{n\times n}$ with $\ell < n$ eigenvalues. Assume A is diagonalizable such that there exists invertible $X \in \mathbb{C}^{n\times n}$ and diagonal $\Lambda \in \mathbb{C}^{n\times n}$ when $A = X^{-1}\Lambda X$.

(a) How are the iterates of GMRES defined **and** computed? Answer with formulas and/or an algorithm.

(b) In this course we found that the iterates of GMRES satisfy $\|Ax_k - b\| = \min_{p\in P_k^0} \|p(A)b\|$. Use this to determine the error of GMRES after $k = \ell$ iterations, under the assumption that no premature breakdown occurs. Note that $\ell < n$ meaning that we have many multiple eigenvalues.

(c) Suppose $D \in \mathbb{R}^{n\times n}$ is a diagonal matrix with entries $d_1, \ldots, d_n$ and let $b = [b_1, \ldots, b_\ell, 0, \ldots, 0]^T \in \mathbb{R}^n$. What is the structure of the Krylov subspace $\mathcal{K}_k(D, b)$ when $k < \ell$? Use this to show that GMRES for $Dx = b$ is independent of $d_{\ell+1}, \ldots, d_n$.

Solution:

(a) The GMRES iterates are *defined* as minimizers of the residual over Krylov subspace:

$$\|Ax_k - b\| = \min_{x\in\mathcal{K}_k(A,b)} \|Ax - b\|$$

They are *computed* via the Arnoldi method, which generates an Arnoldi factorization $AQ_k = Q_{k+1}\underline{H}_k$. The minimization problem can be expressed as

$$\min_{x\in\mathcal{K}_k(A,b)} \|Ax - b\| = \min_{z\in\mathbb{R}^k} \|\underline{H}_k z - e_1\|b\|\|$$

which is a small overdetermined linear system of equations (solvable with for instance backslash). The approximate solution is constructed as $x_k = Q_k z_*$ where $\min_{z\in\mathbb{R}^k} \|\underline{H}_k z - e_1\|b\|\| = \|\underline{H}_k z_* - e_1\|b\|\|$.

(b) We first follow the same step as the derivation of the min-max bound in the lectures/course literature:

$$\|Ax_k - b\| \leq \min_{p\in P_k^0} \|p(A)b\| = \|X\|\|X^{-1}\| \min_{p\in P_k^0} \|p(\Lambda)\|\|b\| =$$

$$\|X\|\|X^{-1}\| \min_{p\in P_k^0} \max_{\lambda=\lambda_1,\ldots,\lambda_\ell} |p(\lambda_\ell)|\|\|b\|$$

We can explicitly construct a minimizers which gives 0 as follows for $k = \ell$. We set

$$p(z) = \frac{(\lambda_1 - z)\cdots(\lambda_\ell - z)}{\lambda_1 \cdots \lambda_\ell}$$

which satisfies $q \in P_k^0$ and $q(\lambda_1) = \cdots = q(\lambda_\ell) = 0$. Hence, $\|Ax_k - b\| = 0$.

(c) The result can be directly identified from the Krylov subspace:

$$D^j b = \begin{bmatrix} d_1^j b_1 \\ \vdots \\ d_\ell^j b_\ell \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \Rightarrow \quad \mathcal{K}_k(D,b) = \text{span}(\begin{bmatrix} b_1 \\ \vdots \\ b_\ell \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} d_1 b_1 \\ \vdots \\ d_\ell b_\ell \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \ldots \begin{bmatrix} d_1^{k-1} b_1 \\ \vdots \\ d_\ell^{k-1} b_\ell \\ 0 \\ \vdots \\ 0 \end{bmatrix})$$

GMRES is defined via $\|Dx_k - b\| = \min_{x\in\mathcal{K}_k(D,b)} \|Dx - b\|$. Since the right-hand side optimization set does not contain $d_{\ell+1}, \ldots, d_n$ and Dx does not contain $d_{\ell+1}, \ldots, d_n$ when $x \in \mathcal{K}_k(D,b)$, the right hand side cannot depend on $d_{\ell+1}, \ldots, d_n$. (The conclusion follows from uniqueness of the GMRES iteration.)

Problem 4 (3p) Felix the fluid mechanician has to compute $f(A)b$ where $f(z) = \sqrt{z}$ and $A \in \mathbb{R}^{n\times n}$ is a large and sparse matrix. One matrix vector product corresponding to A takes approximately $n^{1.5}/10$ time-units in a particular computing environment, and the orthogonalization of one vector against p (orthogonal) vectors takes $10pn$ time-units. Clearly state simplifying assumptions, and estimations when analyzing the following situations.

(a) Felix is an expert on convergence theory for Arnoldi's method for matrix functions and found that we can assume linear convergence for this iteration, and that the convergence factor can be estimated by $\rho = 0.2$. What is the computation time to reach machine precision (as a function n) with Arnoldi's method for matrix functions?

(b) Newton's method for the matrix square root has quadratic convergence. What is the computation time to reach machine precision for $f(A)$? Justify assumptions about computation time for matrix-matrix products.

Solution:

(a) First we determine the number of iterations required: Linear convergence gives an error estimated by ρ^k. Hence, from $10^{-16} \approx \rho^k$, with solution $k = 32$ for $\rho = 0.2$. In order to carry out 32 iterations, we have the following orthogonalization cost

$$\sum_{p=1}^{32} 10pn = 10n \sum_{p=1}^{32} p = 10n528 \approx 5000n.$$

In every step we need one matrix vector product so the matrix-vector product computation cost is

$$32n^{1.5}/10 \tag{*}$$

The matrix vector cost (*) will dominate when $n > n_* \approx 10^6$.

(b) Quadratic convergence implies that the error is squared in every step. If we **assume** the initial error is 0.1, we need 4 iterations to reach machine precision since $0.1^{2^4} = 10^{-16}$. Every step requires one matrix inversion and one matrix-matrix product, which both have essentially computation cost αn^3 for some α. The total a complexity $8\alpha n^3$. The above reasoning was done under the optimistic assumption that the initial error is 0.1. In practice it typically requires more iterations. (By comparing αn^3 with (*) the Arnoldi approach will be faster for large n.)

Problem 5 (4p)

(a) What is the (basic) QR-method? Answer with formulas and/or an algorithm.

(b) The QR-factorization of a matrix is not unique, unless the sign of the diagonal entries in the diagonal are fixed. Suppose $A = QR$ is a QR-factorization. Find a matrix P such that $\tilde{Q} = QP$ and $\tilde{R} = PR$ is a different QR-factorization

(c) Suppose A_1 is the result of one step of the shifted QR-method. Let $\tilde{A}_1$ be the result of one step of the shifted QR-method with the other QR-factorization in (b). Derive a formula for $\tilde{A}_1$, in terms of A_1? How does the non-uniqueness of the QR-factorization influence the QR-method?

Solution:

(a) $A_{k+1} = R_k Q_k$ where $A_k = Q_k R_k$ and $A_0 = A$.

(b) A general parameterization of all the QR-factorization is given by diagonal P that satisfy

$$P^2 = I.$$

This means that the diagonal elements of P are ±1. Any diagonal matrix different from identity with ±1 on the diagonal is a solution to the problem.

(c) The shifted QR-method is defined by

$$Q_k R_k = A_k - \mu I \tag{1}$$

$$A_{k+1} = R_k Q_k + \mu I = Q_k^T A_k Q_k - \mu Q_k^T Q_k + \mu I = Q_k^T A_k Q_k \tag{2}$$

If we instead use $A_k - \mu I = \tilde{Q}_k \tilde{R}_k$ for a different QR-factorization we have

$$\tilde{A}_{k+1} = \tilde{Q}_k^T A_k \tilde{Q}_k = P^T Q_k^T A_k Q_k P.$$

The influence of the non-uniqueness can be expressed as $\tilde{A}_{k+1} = P^T A_k P$. The magnitude of the off-diagonal elements are the same for A_k and $\tilde{A}_k$ and they are in that sense equally far from converged. The reasoning can be repeated (formalization with induction).

Problem 6 (4p) Suppose we have computed an Arnoldi factorization $AQ_k = Q_{k+1}\underline{H}_k$.

(a) How are the eigenvalue approximations for Arnoldi's method for eigenvalue problems computed from the Arnoldi factorization?

(b) Suppose $h_{k+1,k} = 0$. Show that an eigenvalue of H_k is an eigenvalue of A.

Solution:

(a) By definition: The eigenvalues of H_k are taken as approximations of A.

(b) In this case we have $AQ_k = Q_k H_k$. Let $Hx = \mu x$. Then, $AQ_k x = Q_k H_k x = \mu Q_k x$, so μ is an eigenvalue of A with eigenvector $Q_k x$.

Problem 7 (5p)

Let v_{k+1}, w_{k+1}, be generated by carrying out k steps of Algorithm X where $A \in \mathbb{R}^{n\times n}$.

(a) Prove that v_{k+1} and w_{k+1} are elements of certain Krylov subspaces? Which ones?

(b) Simplify the Algorithm X for the case A is symmetric. Under this symmetry assumption, Algorithm X is equivalent to an algorithm in this course. Which one?

(c) The iterates of the Algorithm X satisfy

$$\begin{aligned} AV_k &= V_{k+1}\underline{T}_k \\ A^T W_k &= W_{k+1}\underline{T}_k \end{aligned}$$

where $\underline{T}_k \in \mathbb{R}^{(k+1)\times k}$. Express V_k, W_k and $\underline{T}_k$ in terms of quantities in the algorithm.

Algorithm X:

```
1.  $\tilde{v}_1 = b - Ax_0,\ v_1 = w_1 = \tilde{v}_1/\|\tilde{v}_1\|$
    for $k = 1, \ldots$ until converged
2.      $\tilde{v}_{k+1} = Av_k$
3.      $\tilde{w}_{k+1} = A^T w_k$
4.      $\alpha_k = w_k^T \tilde{v}_{k+1}$
5.      $\tilde{v}_{k+1} = \tilde{v}_{k+1} - \alpha_k v_k$
6.      $\tilde{w}_{k+1} = \tilde{w}_{k+1} - \alpha_k w_k$
7.      if $k > 1$
8.          $\tilde{v}_{k+1} = \tilde{v}_{k+1} - \beta_{k-1} v_k$
9.          $\tilde{w}_{k+1} = \tilde{w}_{k+1} - \beta_{k-1} w_k$
10.     $\gamma_k = \|\tilde{v}_{k+1}\|,\ v_{k+1} = \tilde{v}_{k+1}/\gamma_k$
11.     $\beta_k = \|\tilde{w}_{k+1}\|,\ w_{k+1} = \tilde{w}_{k+1}/\beta_k$
    end
```

Solution:

(a) To carry out the proof for v_{k+1} we do induction and use steps 2,5 and 8 and 10 and the initialization in step 1: Initialization step we set $v_1 = (b - Ax_0)/\|b - Ax_0\|$. Use induction hypothesis, $v_k \in \mathcal{K}_k(A, v_1)$. From step 2 we find that $\tilde{v}_{k+1} \in \mathcal{K}_{k+1}(A, v_1)$. In the operations 5,8,10 we let v_{k+1} be linear combination of $\tilde{v}_{k+1}$, v_k such that $v_{k+1} \in \mathcal{K}_{k+1}(A, v_1)$. The proof is analogous but with a transpose $w_{k+1} \in \mathcal{K}_{k+1}(A^T, v_1)$.

(b) If A is symmetric $w_k = v_k$ for all k and the algorithm reduces to Lanczos.

(c) $V_k = [v_1, \ldots, v_k]$, $W_k = [w_1, \ldots, w_k]$. The matrix $\underline{T}_k$ is given by

$$\underline{T}_k = \begin{bmatrix} \alpha_1 & \beta_1 & & \\ \gamma_1 & \ddots & \ddots & \\ & \ddots & \ddots & \beta_{k-1} \\ & & \ddots & \alpha_k \\ & & & \gamma_k \end{bmatrix}$$