QR-method lecture 2

SF2524 - Matrix Computations for Large-scale Systems

Outline:

1. Decompositions
   - Jordan form
   - Schur decomposition
   - QR-factorization
2. Basic QR-method
3. **Improvement 1: Two-phase approach**
   - Hessenberg reduction
   - Hessenberg QR-method
4. Improvement 2: Acceleration with shifts
5. Convergence theory

Improvement 1: Two-phase approach

We will separate the computation into two phases:

$$\begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix} \underset{\text{Phase 1}}{\rightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix} \underset{\text{Phase 2}}{\rightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & & \times & \times \\ & & & & \times \end{bmatrix}$$

Phases:

- Phase 1: Reduce the matrix to a Hessenberg with similarity transformations (Section 2.2.1 in lecture notes)
- Phase 2: Specialize the QR-method to Hessenberg matrices (Section 2.2.2 in lecture notes)

Phase 1: Hessenberg reduction

Idea for Hessenberg reduction

Compute unitary P and Hessenberg matrix H such that

$$A = PHP^*$$

Unlike the Schur factorization, this can be computed with a finite number of operations.

Key method: Householder reflectors

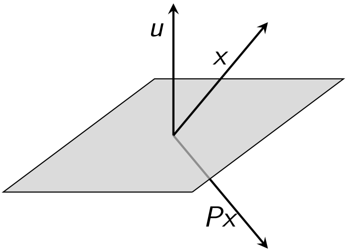
Phase 1: Hessenberg reduction

Definition

A matrix $P \in \mathbb{C}^{m\times m}$ of the form

$$P = I - 2uu^* \quad \text{where } u \in \mathbb{C}^m \text{ and } \|u\| = 1$$

is called a *Householder reflector*.



Properties

- $P^* = P^{-1} = P$
- $Pz = z - 2u(u^*z)$ can be computed with $O(m)$ operations.
- $\cdots$ (show on white board)

Householder reflectors satisfying $Px = \alpha e_1$

Problem

Given a vector x compute a Householder reflector such that

$$Px = \alpha e_1.$$

Solution (Lemma 2.2.3)

Let $\rho = \text{sign}(x_1)$,

$$z := x - \rho\|x\|e_1 = \begin{bmatrix} x_1 - \rho\|x\| \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

and

$$u = z/\|z\|.$$

Then, $P = I - 2uu^*$ is a Householder reflector that satisfies $Px = \alpha e_1$.

* Matlab demo showing Householder reflectors *

We will be able to construct $m-2$ householder reflectors that bring the matrix to Hessenberg form.

Elimination for first column

$$P_1 := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} 1 & 0^T \\ 0 & I - 2u_1u_1^T \end{bmatrix}.$$

Use Lemma 2.2.1 with $x^T = [a_{21}, \ldots, a_{n1}]$ to select u_1 such that

$$P_1 A = \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & \times & \times & \times & \times \end{bmatrix}$$

In order to have a similarity transformation mult from right:

$$P_1 A P_1^{-1} = P_1 A P_1 = \text{same structure as } P_1 A.$$

Elimination for second column

Repeat the process with:

$$P_2 = \begin{bmatrix} 1 & 0 & 0^T \\ 0 & 1 & 0^T \\ 0 & 0 & I - 2u_2u_2^T \end{bmatrix}$$

where u_2 is constructed from the $n-2$ last elements of the second column of $P_1AP_1^*$.

$$P_1AP_1 = \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & \times & \times & \times & \times \end{bmatrix} \underset{\substack{\text{mult. from} \\ \text{left with } P_2}}{\longrightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & \times & \times & \times \end{bmatrix}$$

$$\underset{\substack{\text{mult. from} \\ \text{right with } P_2}}{\longrightarrow} \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ & \times & \times & \times & \times \\ & & \times & \times & \times \\ & & \times & \times & \times \end{bmatrix} = P_2P_1AP_1P_2$$

* Matlab demo of the first two steps of the Hessenberg reduction *

The iteration can be implemented without explicit use of the P matrices.

Algorithm 2 Reduction to Hessenberg form

Input: A matrix $A \in \mathbb{C}^{n\times n}$

Output: A Hessenberg matrix H such that $H = U^* A U$.

for $k = 1, \dots, n-2$ **do**

Compute u_k using (2.4) where $x^T = [a_{k+1,k}, \cdots, a_{n,k}]$

Compute $P_k A$: $A_{k+1:n,k:n} := A_{k+1:n,k:n} - 2u_k(u_k^* A_{k+1:n,k:n})$

Compute $P_k A P_k^*$: $A_{1:n,k+1:n} := A_{1:n,k+1:n} - 2(A_{1:n,k+1:n} u_k) u_k^*$

end for

Let H be the Hessenberg part of A.

* show it in matlab *