Solving large scale eigenvalue problems

Lecture 5, March 23, 2016: The QR algorithm II

http://people.inf.ethz.ch/arbenz/ewp/

Peter Arbenz
Computer Science Department, ETH Zürich
E-mail: arbenz@inf.ethz.ch

Survey of today's lecture

The QR algorithm is the most important algorithm to compute the Schur form of a dense matrix.

- Basic QR algorithm
- Hessenberg QR algorithm
- QR algorithm with shifts
- Double step QR algorithm for real matrices
- The symmetric QR algorithm
- The QZ algorithm for solving $A\mathbf{x} = \lambda B\mathbf{x}$.

Spectral decomposition

Theorem

Let $A \in \mathbb{C}^{n \times n}$ be hermitian, $A^* = A$. Then there is a unitary matrix $U \in \mathbb{C}^{n \times n}$ such that

$$U^* A U = \Lambda = diag(\lambda_1, \ldots, \lambda_n) \tag{1}$$

is diagonal. The diagonal elements λ_i of Λ are the eigenvalues of A.

Let $U = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n]$. Then

$$A\mathbf{u}_i = \lambda_i \mathbf{u}_i, \qquad 1 \leq i \leq n.$$

$\mathbf{u}_i$ is the eigenvector associated with the eigenvalue λ_i.

The symmetric QR algorithm

- The QR algorithm can be applied straight to Hermitian or symmetric matrices.
- The QR algorithm generates a sequence $\{A_k\}$ of symmetric matrices.
- Taking into account the symmetry, the performance of the algorithm can be improved considerably.
- Hermitian matrices have a real spectrum. Therefore, we can restrict ourselves to single shifts.

Reduction to tridiagonal form

Apply a sequence of Householder transformations to arrive at tridiagonal (= symmetric Hessenberg) form.
First step: Let

$$P_1 = \begin{bmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & I_{n-1} - 2\mathbf{u}_1\mathbf{u}_1^* \end{bmatrix}, \qquad \mathbf{u}_1 \in \mathbb{C}^n, \quad \|\mathbf{u}_1\| = 1.$$

Then,

$$\begin{aligned} A_1 := P_1^* A P_1 &= (I - 2\mathbf{u}_1\mathbf{u}_1^*)A(I - 2\mathbf{u}_1\mathbf{u}_1^*) \\ &= A - \mathbf{u}_1\underbrace{(2\mathbf{u}_1^* A - 2(\mathbf{u}_1^* A \mathbf{u}_1)\mathbf{u}_1^*)}_{\mathbf{v}_1^*} - \underbrace{(2A\mathbf{u}_1 - 2\mathbf{u}_1(\mathbf{u}_1^* A \mathbf{u}_1))}_{\mathbf{v}_1}\mathbf{u}_1^* \\ &= A - \mathbf{u}_1\mathbf{v}_1^* - \mathbf{v}_1\mathbf{u}_1^*. \end{aligned}$$

Reduction to tridiagonal form (cont.)

In the k-th step of the reduction we similarly have

$$A_k = P_k^* A_{k-1} P_k = A_{k-1} - \mathbf{u}_{k-1}\mathbf{v}_{k-1}^* - \mathbf{v}_{k-1}\mathbf{u}_{k-1}^*,$$

where the last $n-k$ elements of $\mathbf{u}_{k-1}$ and $\mathbf{v}_{k-1}$ are nonzero.

Essential computation in the kth step:

$$\mathbf{v}_{k-1} = 2A_{k-1}\mathbf{u}_{k-1} - 2\mathbf{u}_{k-1}(\mathbf{u}_{k-1}^* A_{k-1}\mathbf{u}_{k-1})$$

which costs $2(n-k)^2 + \mathcal{O}(n-k)$ flops.

Altogether, the reduction to tridiagonal form costs

$$\sum_{k=1}^{n-1} \left(4(n-k)^2 + \mathcal{O}(n-k)\right) = \frac{4}{3}n^3 + \mathcal{O}(n^2) \text{ flops.}$$

The explicit tridiagonal QR algorithm

In the explicit form, a QR step is essentially

1: Choose a shift μ
2: Compute the QR factorization $A - \mu I = QR$
3: Update A by $A = RQ + \mu I$.

Of course, this is done by means of plane rotations and by respecting the symmetric tridiagonal structure of A.

Shifting strategies: Rayleigh quotient shifts, Wilkinson shifts

The implicit tridiagonal QR algorithm

In the more elegant implicit form of the algorithm we first compute the first Givens rotation $G_0 = G(1, 2, \vartheta)$ of the QR factorization that zeros the $(2, 1)$ element of $A - \mu I$,

$$\begin{bmatrix} c & s \\ -s & c \end{bmatrix} \begin{bmatrix} a_{11} - \mu \\ a_{21} \end{bmatrix} = \begin{bmatrix} * \\ 0 \end{bmatrix}, \qquad c = \cos(\vartheta_0), \quad s = \sin(\vartheta_0). \tag{2}$$

Performing a similarity transformation with G_0 we have $(n = 5)$

$$G_0^* A G_0 = A' = \begin{bmatrix} \times & \times & + & & \\ \times & \times & \times & & \\ + & \times & \times & \times & \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix}$$

The implicit tridiagonal QR algorithm (cont.)

Similarly as with the double step Hessenberg QR algorithm we chase the bulge down the diagonal.

$$
A \xrightarrow[= G(1,2,\vartheta_0)]{G_0} \begin{bmatrix} \times & \times & + & & \\ \times & \times & \times & & \\ + & \times & \times & \times & \\ & & \times & \times & \times \\ & & & \times & \times \end{bmatrix} \xrightarrow[= G(2,3,\vartheta_1)]{G_1} \begin{bmatrix} \times & \times & 0 & & \\ \times & \times & \times & + & \\ 0 & \times & \times & \times & \\ & + & \times & \times & \times \\ & & & \times & \times \end{bmatrix}
$$

$$
\xrightarrow[= G(3,4,\vartheta_2)]{G_2} \begin{bmatrix} \times & \times & 0 & & \\ \times & \times & \times & & \\ & \times & \times & \times & + \\ & 0 & \times & \times & \times \\ & & + & \times & \times \end{bmatrix} \xrightarrow[= G(4,5,\vartheta_3)]{G_3} \begin{bmatrix} \times & \times & & & \\ \times & \times & \times & & \\ & \times & \times & \times & 0 \\ & & \times & \times & \times \\ & & 0 & \times & \times \end{bmatrix}
$$

The implicit tridiagonal QR algorithm (cont.)

The full step is given by

$$\overline{A} = Q^* A Q, \qquad Q = G_0\, G_1\, \cdots\, G_{n-2}.$$

Because $G_k \mathbf{e}_1 = \mathbf{e}_1$ for $k > 0$ we have

$$Q\,\mathbf{e}_1 = G_0\, G_1 \cdots G_{n-2}\,\mathbf{e}_1 = G_0\,\mathbf{e}_1.$$

Both explicit and implicit QR step form the same first plane rotation G_0. By referring to the Implicit Q Theorem we see that explicit and implicit QR step compute *essentially* the same $\overline{A}$.

Symm. tridiag. QR algo with Wilkinson shifts

```
Let T ∈ ℝ^{n×n} be a symmetric tridiagonal matrix with diagonal
entries a_1, ..., a_n and off-diagonal entries b_2, ..., b_n.
This algorithm computes the eigenvalues λ_1, ..., λ_n of T and
corresponding eigenvectors q_1, ..., q_n. The eigenvalues are
stored in a_1, ..., a_n. The eigenvectors are stored in the matrix
Q, such that TQ = Q diag(a_1, ..., a_n).

m = n {Actual problem dimension. m is reduced in the convergence check.}
while m > 1 do
    d := (a_{m-1} - a_m)/2; {Compute Wilkinson's shift}
    if d = 0 then
        s := a_m - |b_m|;
```

Symm. tridiag. QR algo with Wilkinson shifts (cont.)

else
  $s := a_m - b_m^2/(d + \text{sign}(d)\sqrt{d^2 + b_m^2})$;
end if
$x := a(1) - s$; {Implicit QR step begins here}
$y := b(2)$;
for $k = 1$ to $m - 1$ **do**
  if $m > 2$ **then**
    $[c, s] :=$ **givens**(x, y);
  else
    Determine $[c, s]$ such that $\begin{bmatrix} c & -s \\ s & c \end{bmatrix} \begin{bmatrix} a_1 & b_2 \\ b_2 & a_2 \end{bmatrix} \begin{bmatrix} c & s \\ -s & c \end{bmatrix}$ is diagonal
  end if
  $w := cx - sy$;

Symm. tridiag. QR algo with Wilkinson shifts (cont.)

    $d := a_k - a_{k+1}; \quad z := (2cb_{k+1} + ds)s;$
    $a_k := a_k - z; \quad a_{k+1} := a_{k+1} + z;$
    $b_{k+1} := dcs + (c^2 - s^2)b_{k+1};$
    $x := b_{k+1};$
    if $k > 1$ **then**
        $b_k := w;$
    end if
    if $k < m - 1$ **then**
        $y := -sb_{k+2}; \quad b_{k+2} := cb_{k+2};$
    end if
    $Q_{1:n;k:k+1} := Q_{1:n;k:k+1} \begin{bmatrix} c & s \\ -s & c \end{bmatrix};$
end for{Implicit QR step ends here}
if $|b_m| < \varepsilon(|a_{m-1}| + |a_m|)$ **then** {Check for convergence}

Symm. tridiag. QR algo with Wilkinson shifts (cont.)

        $m := m - 1$;
    end if
end while

Remark on deflation

$$T = \begin{bmatrix} a_1 & b_2 & & & & \\ b_2 & a_2 & b_3 & & & \\ & b_3 & a_3 & 0 & & \\ & & 0 & a_4 & b_5 & \\ & & & b_5 & a_5 & b_6 \\ & & & & b_6 & a_6 \end{bmatrix}$$

- Shift for next step is determined from second block.
- First plane rotation is determined from shift and first block!
- The implicit shift algorithm then chases the bulge down the diagonal. Procedure finishes already in row 4 because $b_4 = 0$.
- This shift does not improve convergence.
- Explicit QR algorithm converges rapidly, but first block is not treated properly

Complexity of QR algorithm

	nonsymmetric case		symmetric case	
	without	with	without	with
	Schurvectors		eigenvectors	
transformation to Hessenberg/tridiagonal form	$\frac{10}{3}\, n^3$	$\frac{14}{3}\, n^3$	$\frac{4}{3}\, n^3$	$\frac{8}{3}\, n^3$
real double step Hessenberg/ tridiagonal QR algorithm (2 steps per eigenvalues)	$\frac{20}{3}\, n^3$	$\frac{50}{3}\, n^3$	$24\, n^2$	$6\, n^3$
total	$10\, n^3$	$25\, n^3$	$\frac{4}{3}\, n^3$	$9\, n^3$

The QZ algorithm for $A\mathbf{x} = \lambda B\mathbf{x}$

Theorem (Generalized Schur decomposition)

Let $A, B \in \mathbb{C}^{n \times n}$. Then there are unitary matrices Q and Z such that $Q^*AZ = T$ and $Q^*BZ = S$ are both upper triangular matrices. If for some k, $t_{kk} = s_{kk} = 0$ then $\sigma(A; B) = \mathbb{C}$. Otherwise,

$$\sigma(A; B) = \left\{ \frac{t_{ii}}{s_{ii}} \mid s_{ii} \neq 0 \right\}.$$

Remark: (1) It is possible that $\infty \in \sigma(A; B)$. This is equivalent with $0 \in \sigma(B; A)$.
(2) There is a real version of this theorem. There T is quasi-upper triangular and S is upper triangular.

Proof.

Let B_k a sequence of nonsingular matrices converging to B. Let $Q_k^*(AB_k^{-1})Q_k = R_k$, $k \geq 0$, the Schur decomposition of AB_k^{-1}. Let Z_k be unitary and $Z_k^*(B_k^{-1}Q_k) = S_k^{-1}$ upper triangular. Then

$$Q_k^* A Z_k Z_k^* B_k^{-1} Q_k = R_k \implies Q_k^* A Z_k = R_k S_k$$

is upper triangular.
Bolzano–Weierstrass: the sequence $\{(Q_k, Z_k)\}$ has convergent subsequence, $\lim(Q_{k_i}, Z_{k_i}) = (Q, Z)$.
Q, Z are unitary and Q^*AZ, Q^*BZ are upper triangular.
Statement on eigenvalues follows from

$$\det(A - \lambda B) = \det(QZ^*) \prod_{i=1}^{n} (t_{ii} - \lambda s_{ii}).$$

The QZ algorithm: step 1

Step 1: Reduction to Hessenberg-triangular form.
Transform B into upper triangular form (QR factorization of B)

$$A \leftarrow U^* A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow U^* B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

The QZ algorithm: step 1 (cont.)

Transform A into Hessenberg form by a sequence of Givens rotations w/o destroying the zero pattern of B

$$A \leftarrow Q_{45}^* A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \color{red}{0} & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow Q_{45}^* B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \color{red}{+} & \times \end{pmatrix}$$

$$A \leftarrow AZ_{45} = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \color{red}{0} & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow BZ_{45} = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \color{red}{0} & \times \end{pmatrix}$$

The QZ algorithm: step 1 (cont.)

$$A \leftarrow Q_{34}^* A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow Q_{34}^* B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & + & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

$$A \leftarrow A Z_{34} = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow B Z_{34} = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

The QZ algorithm: step 1 (cont.)

$$A \leftarrow Q_{23}^* A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow Q_{23}^* B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & + & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

$$A \leftarrow A Z_{23} = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \end{pmatrix}, \quad B \leftarrow B Z_{23} = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

Now the first column of A has the desired structure.

Proceed similarly with columns 2 to $n-2$.

The QZ algorithm: step 2

Step 2: Deflation.
Let us assume

1. that A is an irreducible Hessenberg matrix and
2. that B is a nonsingular upper triangular matrix.

If 1. is not satisfied, e.g. $a_{k+1,k} = 0$, then

$$A - \lambda B = \begin{pmatrix} A_{11} - \lambda B_{11} & A_{12} - \lambda B_{12} \\ 0 & A_{22} - \lambda B_{22} \end{pmatrix}$$

and we can treat the smaller problems $A_{11} - \lambda B_{11}$ and $A_{22} - \lambda B_{22}$ individually.

The QZ algorithm: step 2 (cont.)

If 2. is not satisfied, then $b_{k,k} = 0$ for some k.

$$A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}, \quad B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

We chase the zero down the diagonal of B:

$$A \leftarrow Q_{34}^* A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & + & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}, \quad B \leftarrow Q_{34}^* B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

The QZ algorithm: step 2 (cont.)

$$A \leftarrow AZ_{23}^* = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}, \quad B \leftarrow BZ_{23}^* = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

$$A \leftarrow Q_{45}^* A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & + & \times & \times \end{pmatrix}, \quad B \leftarrow Q_{45}^* B = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

The QZ algorithm: step 2 (cont.)

$$A \leftarrow AZ_{34}^* = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & \color{red}{0} & \times & \times \end{pmatrix}, \quad B \leftarrow BZ_{34}^* = \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \color{red}{+} & \times & \times \\ 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$A \leftarrow AZ_{45} = \left[\begin{array}{cccc|c} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ \hline 0 & 0 & 0 & \color{red}{0} & \times \end{array}\right], \quad B \leftarrow B = BZ_{45} \left[\begin{array}{cccc|c} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & + & \times & \times \\ 0 & 0 & 0 & \color{red}{+} & \times \\ \hline 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

The QZ algorithm: step 2

Step 3: QZ step.
We now consider the pair (A, B) satisfing assumptions 1 and 2.
We execute an iteration that corresponds to a QR algorithm applied to AB^{-1}.
We look at a single step of the QR algorithm. We want to modify A and B,

$$\bar{A} - \lambda\bar{B} = \bar{Q}^*(A - \lambda B)\bar{Z}, \qquad \bar{Q}, \bar{Z} \text{ unitary.}$$

with $\bar{A}$ Hessenberg and $\bar{B}$ upper triangular. $\bar{A}\bar{B}^{-1}$ is the matrix that is obtained by one step of the QR algorithm applied to AB^{-1}.

The QZ algorithm: step 2 (cont.)

Set $M := AB^{-1}$. M is Hessenberg. Let

$$\mathbf{v} = (M - aI)(M - bI)\mathbf{e}_1$$

where a and b are the eigenvalues of the trailing 2×2 block of M.
($\mathbf{v}$ can be computed in $\mathcal{O}(1)$ flops.)
Let P_0 be the Householder reflector with

$$P_0\mathbf{v} = \pm\|\mathbf{v}\|\mathbf{e}_1.$$

Then,

$$A = P_0 A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ + & \times & \times & \times & \times \\ \hline 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{pmatrix}, \quad B = P_0 B \begin{pmatrix} \times & \times & \times & \times & \times \\ + & \times & \times & \times & \times \\ + & + & \times & \times & \times \\ \hline 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix}$$

The QZ algorithm: step 2 (cont.)

Now we restore the Hessenberg-triangular form:

$$A \leftarrow AZ_1Z_2 = \left[\begin{array}{ccc|cc} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ + & \times & \times & \times & \times \\ + & + & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \end{array}\right], \quad B \leftarrow BZ_1Z_2 = \left[\begin{array}{ccc|cc} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{array}\right]$$

$$A \leftarrow P_2P_1A = \left[\begin{array}{ccccc} \times & \times & \times & \times & \times \\ \hline \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ \hline 0 & 0 & 0 & \times & \times \end{array}\right], \quad B \leftarrow P_2P_1B = \left[\begin{array}{ccccc} \times & \times & \times & \times & \times \\ \hline 0 & \times & \times & \times & \times \\ 0 & + & \times & \times & \times \\ 0 & + & + & \times & \times \\ \hline 0 & 0 & 0 & 0 & \times \end{array}\right]$$

and so on, until the bulge drops out at the end of the matrix.

References

[1] G. H. Golub and C. F. van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, 4th ed., 2012.