Computing the eigenvalues of a companion matrix

Structured Linear Algebra Problems:
Analysis, Algorithms, and Applications
Cortona, Italy
September 15-19, 2008

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren
Dept. Computer Science, K.U.Leuven, Belgium
Dept. of Math. Eng., Catholic University of Louvain, Belgium

Contents

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

The problem
Companion matrix
Working with Givens transformations
Representation
Fusion and shift-through operations
Unitary plus rank one matrices
Structure under a *QR*-step
A representation for the unitary matrix
Representation of the unitary plus rank one matrix
Implicit *QR*-algorithm with single shift
Initialization
The chasing
The last Givens transformation
Numerical Experiments
Scaling
Comparison

Outline

The problem
- Companion matrix

Working with Givens transformations
- Representation
- Fusion and shift-through operations

Unitary plus rank one matrices
- Structure under a *QR*-step
- A representation for the unitary matrix
- Representation of the unitary plus rank one matrix

Implicit *QR*-algorithm with single shift
- Initialization
- The chasing
- The last Givens transformation

Numerical Experiments
- Scaling
- Comparison

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

The problem
Companion matrix
Working with Givens transformations
Representation
Fusion and shift-through operations
Unitary plus rank one matrices
Structure under a *QR*-step
A representation for the unitary matrix
Representation of the unitary plus rank one matrix
Implicit *QR*-algorithm with single shift
Initialization
The chasing
The last Givens transformation
Numerical Experiments
Scaling
Comparison

Outline

The problem
Companion matrix

Working with Givens transformations
Representation
Fusion and shift-through operations

Unitary plus rank one matrices
Structure under a *QR*-step
A representation for the unitary matrix
Representation of the unitary plus rank one matrix

Implicit *QR*-algorithm with single shift
Initialization
The chasing
The last Givens transformation

Numerical Experiments
Scaling
Comparison

The problem
Companion matrix
Working with Givens transformations
Representation
Fusion and shift-through operations
Unitary plus rank one matrices
Structure under a *QR*-step
A representation for the unitary matrix
Representation of the unitary plus rank one matrix
Implicit *QR*-algorithm with single shift
Initialization
The chasing
The last Givens transformation
Numerical Experiments
Scaling
Comparison

Companion matrix

Definition

Given a monic polynomial

$$p(z) = p_0 + p_1 z + p_2 z^2 + \ldots + p_{n-1} z^{n-1} + z^n$$

with the coefficients $p_i \in \mathbb{R}$ or $\mathbb{C}$
the associated companion matrix C_p is defined as:

$$C_p = \begin{bmatrix} 0 & & & -p_0 \\ 1 & \ddots & & -p_1 \\ & \ddots & & \vdots \\ & & 1 & -p_{n-1} \end{bmatrix}.$$

The eigenvalues of the companion matrix coincide with the zeros of the associated polynomial $p(z)$, because

$$p(z) = \det(zI - C_p).$$

Unitary plus rank one matrix

$$\begin{aligned} C_p &= H && \text{upper Hessenberg} \\ &= U + \mathbf{u}\mathbf{v}^H && \text{unitary plus rank one} \end{aligned}$$

with

$$\begin{aligned} U &= \begin{bmatrix} 0 & 0 & \cdots & 0 & \pm 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \\ \mathbf{v}^T &= (0, 0, \ldots, 0, 1) \\ \mathbf{u}^T &= (-p_0 \mp 1, -p_1, \ldots, -p_{n-1}) \end{aligned}$$

Hence, computing the zeros of the polynomial $p(z)$ is equivalent to computing the eigenvalues of the upper Hessenberg, unitary plus rank one matrix

$$C_p = H = U + \mathbf{u}\mathbf{v}^H.$$

Outline

The problem
- Companion matrix

Working with Givens transformations
- Representation
- Fusion and shift-through operations

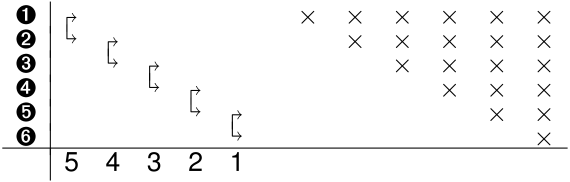
Unitary plus rank one matrices
- Structure under a *QR*-step
- A representation for the unitary matrix
- Representation of the unitary plus rank one matrix

Implicit *QR*-algorithm with single shift
- Initialization
- The chasing
- The last Givens transformation

Numerical Experiments
- Scaling
- Comparison

Representation

- Givens transformations are a powerful tool for working with structured matrices.
- representation of sequences of Givens transformations by a graphical scheme.
- **example:** the QR-factorization of a 6×6 Hessenberg matrix



The figure corresponds to $G_5 G_4 \ldots G_1 R$.
Rewriting the formula we have $G_1^H \ldots G_5^H H = R$.
Hence G_5^H annihilates the first subdiagonal element, G_4^H the second and so forth.

Outline

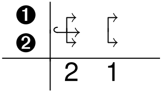
Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

Fusion operation

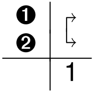
Lemma

Suppose two Givens transformations G_1 and G_2 are given. Then we have that $G_1 G_2 = G_3$ is again a Givens transformation. We will call this the fusion of Givens transformations.

The proof is trivial. In our graphical schemes, we will depict this as follows:



resulting in



.

Shift-through operation

Often Givens transformations of higher dimensions are considered, i.e., the corresponding 2×2 Givens transformation is embedded in the identity matrix of dimension n.

Lemma (Shift through lemma)

Suppose three 3×3 Givens transformations $\check{G}_1, \check{G}_2$ and $\check{G}_3$ are given, such that the Givens transformations $\check{G}_1$ and $\check{G}_3$ act on the first two rows of a matrix, and $\check{G}_2$ acts on the second and third row (when applied on the left to a matrix).
Then there exist three Givens transformations $\hat{G}_1, \hat{G}_2$ and $\hat{G}_3$ such that

$$\check{G}_1 \check{G}_2 \check{G}_3 = \hat{G}_1 \hat{G}_2 \hat{G}_3,$$

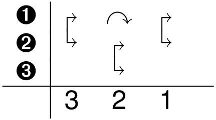
where $\hat{G}_1$ and $\hat{G}_3$ work on the second and third row and $\hat{G}_2$, works on the first two rows.

Proof: a 3×3 unitary matrix can be factorized in different ways. Gives the possibility

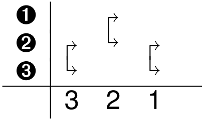
- to interchange the order of Givens transformations and
- to obtain different patterns.

Shift-through operation

Graphically we will depict this rearrangement as follows.



resulting in



.

Outline

The problem
Companion matrix

Working with Givens transformations
Representation
Fusion and shift-through operations

Unitary plus rank one matrices
Structure under a *QR*-step
A representation for the unitary matrix
Representation of the unitary plus rank one matrix

Implicit *QR*-algorithm with single shift
Initialization
The chasing
The last Givens transformation

Numerical Experiments
Scaling
Comparison

Outline

Structure under a *QR*-step

The companion matrix C_p:

$$C_p = H = U + \mathbf{u}\mathbf{v}^H, \qquad (1)$$

with H Hessenberg, U unitary and $\mathbf{u}$ and $\mathbf{v}$ two vectors.
Given a shift μ, perform a step of the *QR*-iteration onto the matrix H:

$$\begin{aligned} H - \mu I &= QR \\ \hat{H} &= RQ + \mu I = Q^H H Q. \end{aligned}$$

Applying the similarity transformation onto the terms of (1):

$$\begin{aligned} \hat{H} &= Q^H H Q = Q^H U Q + Q^H \mathbf{u}\mathbf{v}^H Q \\ &= \hat{U} + \hat{\mathbf{u}}\hat{\mathbf{v}}^H, \end{aligned}$$

with $\hat{H}$ Hessenberg, $\hat{U}$ unitary and $\hat{\mathbf{u}} = Q^H\mathbf{u}$ and $\hat{\mathbf{v}} = Q^H\mathbf{v}$ two vectors.
Hence the unitary plus rank one structure of the Hessenberg matrix is preserved under a step of the *QR*-iteration.
Exploiting the structure leads to an efficient implicit *QR*-method.

Outline

A representation for the unitary matrix

Consider

$$U = H - \mathbf{u}\mathbf{v}^H.$$

H is Hessenberg

$\Rightarrow$ H has zeros below the subdiagonal

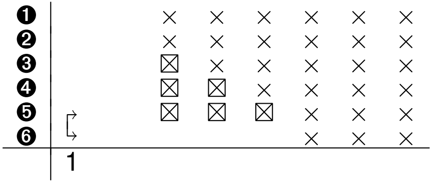
$\Rightarrow$ the matrix U needs to be of rank 1 below the subdiagonal

$$U = \begin{bmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \boxtimes & \times & \times \end{bmatrix}$$

A representation for the unitary matrix

$$U = \begin{bmatrix} \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \boxtimes & \times & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \times & \times & \times & \times \\ \boxtimes & \boxtimes & \boxtimes & \times & \times & \times \\ \otimes & \otimes & \otimes & \boxtimes & \times & \times \end{bmatrix}$$

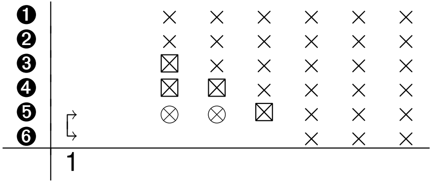
By a single Givens transformation acting on row five and six, three elements are annihilated.
The elements to be annihilated are marked with $\otimes$.



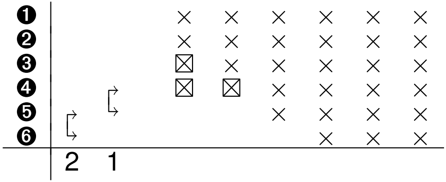
Mathematically, the figure depicts

$$U = G_1 U_1 .$$

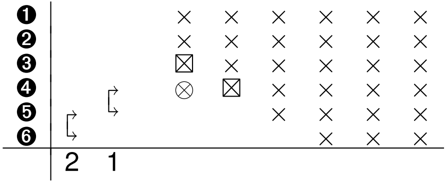
A representation for the unitary matrix



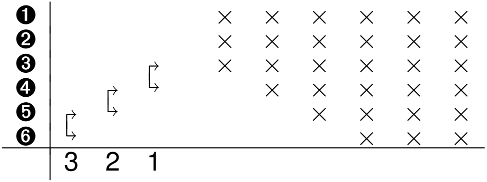
Applying a second transformation gives us $U = G_1 G_2 U_2$.



A representation for the unitary matrix

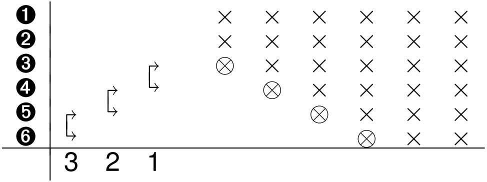


Apply a third Givens transformation on rows three and four.

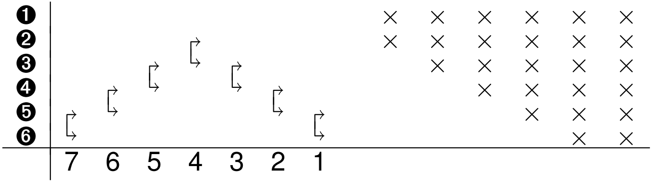


$\Rightarrow$ the low rank part is completely removed
$\Rightarrow$ The matrix remaining on the right side is now a generalized Hessenberg matrix having two subdiagonals.

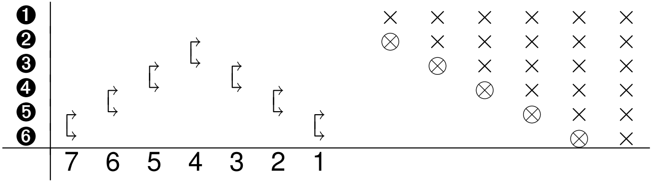
A representation for the unitary matrix



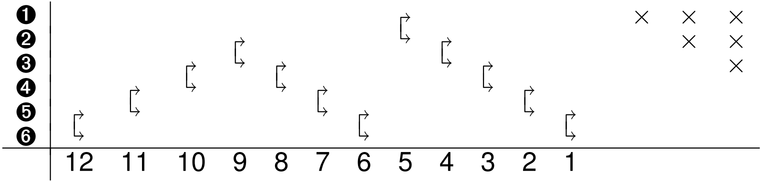
Peeling of the second subdiagonal, removing successively all elements marked with $\otimes$, will give us an extra descending sequence of Givens transformations.
These Givens transformations can be found in positions 1 up to 4.



A representation for the unitary matrix

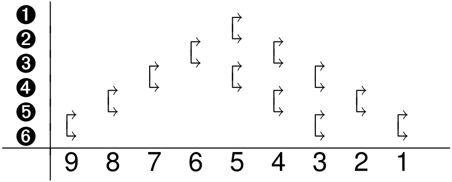


In the final step the remaining subdiagonal from the matrix on the right is removed.



In fact we have computed now a *QR*-factorization of the unitary matrix $U = QR$ consisting of three sequences of Givens transformations.

A representation for the unitary matrix



(2)

We have factored the unitary matrix U as the product of three unitary matrices

$$U = VWX$$

with

- V denoting the sequence from 9 to 7,
- W the lower sequence from 6 to 3 and
- X the upper sequence from 5 to 1.

This representation of the unitary matrix will be used in the sum

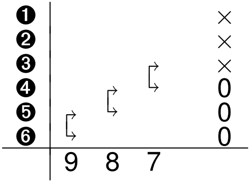
$$C_p = H = U + \mathbf{u}\mathbf{v}^H.$$

Outline

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

A representation for the unitary plus rank one matrix

relation between the low rank part in U and $\mathbf{u}\mathbf{v}^H$
$\Rightarrow$ we can decompose the vector $\mathbf{u}$:



Givens transformations in position 9, 8 and 7 are equal to V. Hence,

$$
\begin{aligned}
H &= U + \mathbf{u}\mathbf{v}^H \\
&= VWX + \mathbf{u}\mathbf{v}^H \\
&= V\left(WX + V^H\mathbf{u}\mathbf{v}^H\right) \\
&= V\left(WX + \hat{\mathbf{u}}\mathbf{v}^H\right),
\end{aligned}
$$

where the vector $\hat{\mathbf{u}}$ has only the first three elements different from zero.

Outline

Outline

Determining the first Givens transformation

The implicit *QR*-step performed on a Hessenberg matrix is determined by the first Givens transformation G_1 such that

$$G_1^H (H - \mu I)\mathbf{e}_1 = \pm \|(H - \mu I)\mathbf{e}_1\| \mathbf{e}_1 .$$

We will now perform the similarity transformation $G_1^H H G_1$, exploiting the factorization of the matrix H.

$$H \quad = V\left(WX + \hat{\mathbf{u}}\mathbf{v}^H\right). \tag{3}$$

$\Rightarrow$ a bulge is created. After a complete *QR*-step is finished:

- bulge is chased away
- we obtain again a Hessenberg matrix
- represented as in (3).

Hence, throughout the entire procedure we want the representation to remain as closely as possible to the original representation of H.

Performing the first similarity transformation

Let us perform the first similarity transformation onto the matrix $H = H_0 = V_0\left(W_0 X_0 + \hat{\mathbf{u}}_0 \mathbf{v}_0^H\right)$. We obtain

$$
\begin{aligned}
H_1 &= G_1^H V_0 \left(W_0 X_0 + \hat{\mathbf{u}}_0 \mathbf{v}_0^H \right) G_1 \\
&= G_1^H V_0 \left(W_0 X_0 G_1 + \hat{\mathbf{u}}_0 \mathbf{v}_0^H G_1 \right) \\
&= G_1^H V_0 \left(W_0 X_0 G_1 + \hat{\mathbf{u}}_0 \mathbf{v}_1^H \right).
\end{aligned}
$$

Since the Givens transformation G_1^H acts on the first two rows, and V_0 acts on the rows 3 up to 6, they commute and we obtain the following:

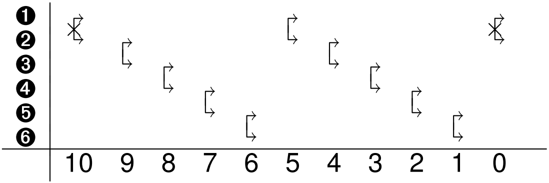
$$
\begin{aligned}
H_1 &= V_0 \left(G_1^H W_0 X_0 G_1 + G_1^H \hat{\mathbf{u}}_0 \mathbf{v}_1^H \right) \\
&= V_0 \left(G_1^H W_0 X_0 G_1 + \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right),
\end{aligned}
$$

with $\tilde{\mathbf{u}}_1 = G_1^H \mathbf{u}_0$, having only the first three elements different from zero.

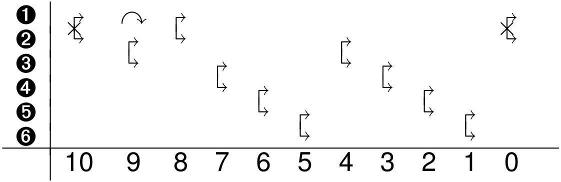
It seems that the matrix H_1 is already in the good form, except for the unitary matrix $G_1^H W_0 X_0 G_1$. Let us see how to change this matrix.

Performing the first similarity transformation

$$H_1 \quad = \quad V_0 \left(G_1^H W_0 X_0 G_1 + \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right)$$

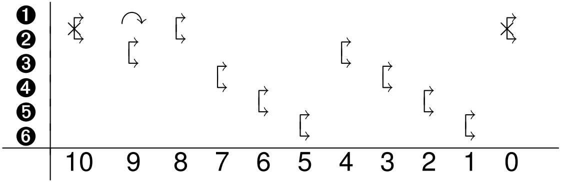


We will start by removing the Givens transformation in position 10. Reordering (moving the Givens from position 5 to position 8) permits the application of the shift through operation.

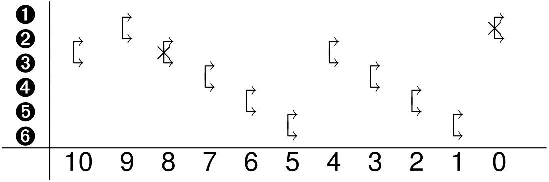


Performing the first similarity transformation

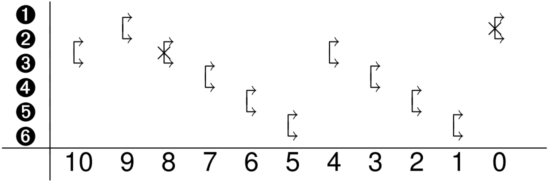
$$H_1 = V_0 \left(G_1^H W_0 X_0 G_1 + \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right)$$



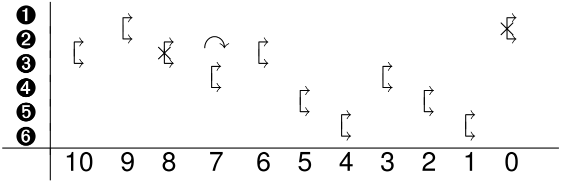
Apply the shift through operation.

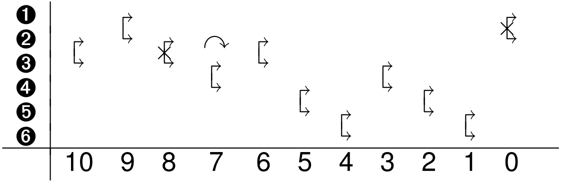


Performing the first similarity transformation

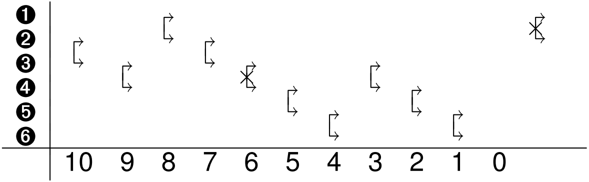


Rewriting.



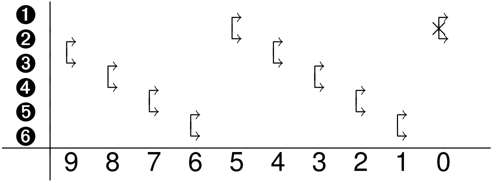


Apply the shift through operation.



Clearly the undesired Givens transformation moves down, creating a new sequence, starting in positions 10 and 9 and removing the sequence in positions 3 up to 1.

Performing the first similarity transformation



This corresponds to the following relations:

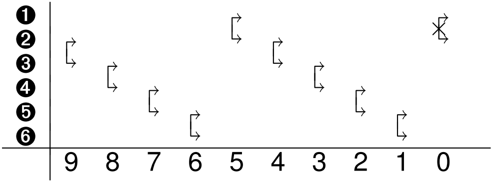
$$H_1 = V_0 \left(\tilde{W}_1 \tilde{X}_1 G_1 + \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right),$$

the unitary matrices $\tilde{W}_1$ and $\tilde{X}_1$ are two descending sequences of transformations.

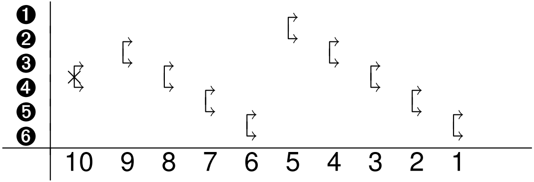
Let us continue, by trying to remove also the second undesired Givens transformation, the one in position 0.

Performing the first similarity transformation

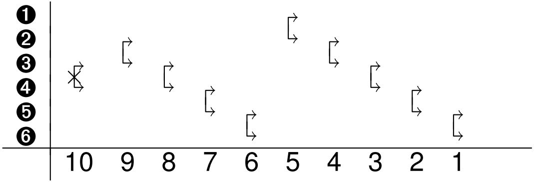
$$H_1 = V_0 \left(\tilde{W}_1 \tilde{X}_1 G_1 + \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right),$$



One can apply two times the shift through operation to obtain the following scheme.



Performing the first similarity transformation



Now there is an unwanted transformation in position 10. Denote this transformation with $\tilde{G}_1$. In formulas we obtain now:

$$H_1 = V_0 \left(\tilde{G}_1 \hat{W}_1 X_1 + \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right).$$

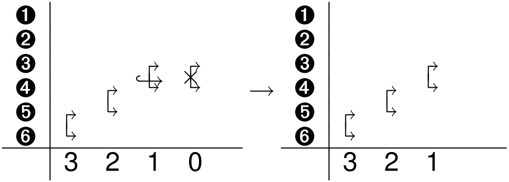
This gives us the following relations:

$$\begin{aligned} H_1 &= V_0 \tilde{G}_1 \left(\hat{W}_1 X_1 + \tilde{G}_1^H \tilde{\mathbf{u}}_1 \mathbf{v}_1^H \right), \\ &= V_0 \tilde{G}_1 \left(\hat{W}_1 X_1 + \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right), \end{aligned}$$

Two terms are of importance here: $V_0 \tilde{G}_1$ and $\tilde{G}_1^H \tilde{\mathbf{u}}_1$. Clearly the vector $\bar{\mathbf{u}}_1$, will loose one of his zeros. The vector will have now four nonzero elements.

Performing the first similarity transformation

$$H_1 \quad = \quad V_0 \tilde{G}_1 \left(\hat{W}_1 X_1 + \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right), \tag{4}$$



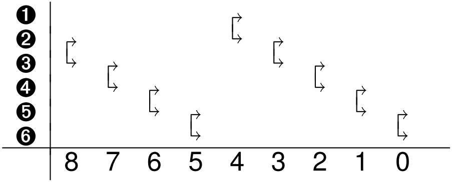
Hence,

$$H_1 \quad = \quad \tilde{V}_1 \left(\hat{W}_1 X_1 + \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right).$$

Performing the first similarity transformation

We are however not yet satisfied. We want to obtain a similar factorization as of the original matrix H. Hence, the four nonzero elements in the vector $\bar{\mathbf{u}}_1$ need to be transformed into three nonzero elements.
To do so, we need to rewrite

$$H_1 \quad = \quad \tilde{V}_1 \left(\hat{W}_1 X_1 + \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right).$$

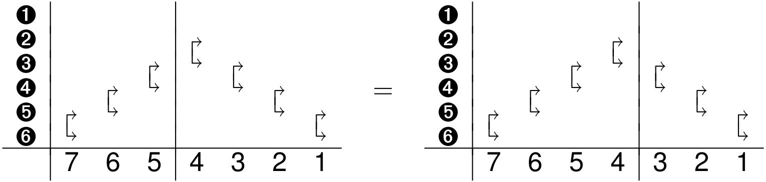


Denote the Givens transformation in position 8, with G^l, this means $G^l \hat{W}_1^l = \hat{W}_1$, we will drag this Givens transformation out of the brackets on the left. We denote all changed variables with $\cdot^l$ to clearly indicate that one extra Givens has moved to the left. Until we will move the Givens back inside the brackets we will indicate this on the affected elements.

Performing the first similarity transformation

$$
\begin{aligned}
H_1 &= \tilde{V}_1 \left(\hat{W}_1 X_1 + \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right) \\
&= \tilde{V}_1 \left(G^l \hat{W}_1^l X_1 + \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right) \\
&= \tilde{V}_1 G^l \left(\hat{W}_1^l X_1 + {G^l}^H \bar{\mathbf{u}}_1 \mathbf{v}_1^H \right) \\
&= \tilde{V}_1^l \left(\hat{W}_1^l X_1 + \bar{\mathbf{u}}_1^l \mathbf{v}_1^H \right)
\end{aligned}
$$

in which ${G^l}^H \bar{\mathbf{u}}_1 = \bar{\mathbf{u}}_1^l$ still has four elements different from zero, the matrix $\tilde{V}_1^l$ is now a sequence having one more Givens transformation than $\tilde{V}_1$. Graphically $\tilde{V}_1 \hat{W}_1 = \tilde{V}_1^l \hat{W}_1^l$ is depicted as follows.



The vertical line depicts the splitting of the product of the Givens transformations.

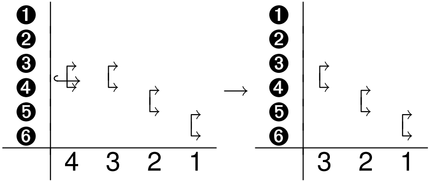
Performing the first similarity transformation

Construct now a Givens transformation acting on row 3 and 4 of the vector $\bar{\mathbf{u}}_1^l$ and annihilating the element in position 4.

Let us denote this Givens transformation by $\hat{G}_1$ such that $\hat{G}_1^H \bar{\mathbf{u}}_1^l = \hat{\mathbf{u}}_1^l$.

$$
\begin{aligned}
H_1 &= \tilde{V}_1^l \left(\hat{W}_1^l X_1 + \hat{G}_1 \hat{G}_1^H \bar{\mathbf{u}}_1^l \mathbf{v}_1^H \right) \\
&= \tilde{V}_1^l \left(\hat{G}_1 \hat{G}_1^H \hat{W}_1^l X_1 + \hat{G}_1 \hat{\mathbf{u}}_1^l \mathbf{v}_1^H \right) \\
&= \tilde{V}_1^l \hat{G}_1 \left(\hat{G}_1^H \hat{W}_1^l X_1 + \hat{\mathbf{u}}_1^l \mathbf{v}_1^H \right)
\end{aligned}
$$

Let us take a closer look now at $\hat{G}_1^H \hat{W}_1^l$. Applying one fusion, removes the Givens transformation in position 4.

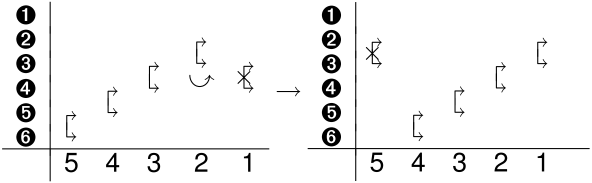


This results in a sequence of Givens transformations denoted as $\hat{G}_1^H \hat{W}_1 = W_1^l$.

Performing the first similarity transformation

$$H_1 = \tilde{V}_1^I \hat{G}_1 \left(W_1^I X_1 + \hat{\mathbf{u}}_1^I \mathbf{v}_1^H \right)$$

The matrix product $\tilde{V}_1^I \hat{G}_1$ is of the following form. Applying the shift through lemma gives us the following figure.



This scheme can be written as $G_2 V_1^I$, in which V_1^I incorporates the Givens from position 1 to 4 and G_2 is shown in the fifth position. This Givens transformation G_2 will determine the next chasing step. First we need however to bring the Givens transformation in position 1 again inside the brackets.

Plugging all of this into the equations above gives us:

$$H_1 \quad = \quad G_2 V_1^l \left(W_1^l X_1 + \hat{\mathbf{u}}_1^l \mathbf{v}_1^H \right).$$

Rewriting now the formula above by bringing the Givens transformation $\tilde{G}_1^l$ in position 1 of the matrix V_1^l inside the brackets does not change the formulas significantly. We obtain:

$$H_1 = G_2 V_1 \left(W_1 X_1 + \hat{\mathbf{u}}_1 \mathbf{v}_1^H \right),$$

with $V_1 = V_1^l \tilde{G}_1^H, \ W_1 = \tilde{G}_1 W_1^l, \ \hat{\mathbf{u}}_1 = \tilde{G}_1 \hat{\mathbf{u}}_1^l$.

The matrix H_1 is almost in Hessenberg form, except for a bulge in position $(2, 1)$.

The representation is also almost in the correct form, only the Givens transformation G_2 is undesired.

We will remove this transformation (removing thereby also the bulge in H_1), by applying another similarity transformation with G_2.

Outline

Chasing

Since we want our matrix H to become again of unitary plus low rank form obeing the designed representation, we want to remove the disturbance G_2. Performing the unitary similarity transformation with this Givens transformation removes the transformation in the left, but creates an extra transformation on the right. We obtain the following:

$$
\begin{aligned}
H_2 &= G_2^H H_1 G_2 \\
&= G_2^H G_2 V_1 \left(W_1 X_1 + \hat{\mathbf{u}}_1 \mathbf{v}_1^H \right) G_2 \\
&= V_1 \left(W_1 X_1 G_2 + \hat{\mathbf{u}}_1 \mathbf{v}_1^H G_2 \right) \\
&= V_1 \left(W_1 X_1 G_2 + \hat{\mathbf{u}}_1 \mathbf{v}_2^H \right),
\end{aligned}
$$

where $\mathbf{v}_2^H = \mathbf{v}_1^H G_2$.
Similarly as in the initial step we can drag G_2 through W_1 and X_1. We obtain $W_1 X_1 G_2 = \tilde{G}_2 W_2 X_2$.

This gives us

$$\begin{aligned} H_2 &= V_1\left(W_1 X_1 G_2 + \hat{\mathbf{u}}_1 \mathbf{v}_2^H\right), \\ &= V_1\left(\tilde{G}_2 W_2 X_2 + \hat{\mathbf{u}}_1 \mathbf{v}_2^H\right), \\ &= V_1 \tilde{G}_2\left(W_2 X_2 + \tilde{G}_2^H \hat{\mathbf{u}}_1 \mathbf{v}_2^H\right). \end{aligned}$$

Since the Givens transformation $\tilde{G}_2^H$ acts on row 4 and row 5, $\tilde{G}_2^H \hat{\mathbf{u}}_1 = \hat{\mathbf{u}}_1$ ($\hat{\mathbf{u}}_1$ has only the first three elements different from zero). Applying a final shift through operation for $V_1 \tilde{G}_2$ we obtain $G_3 V_2 = V_1 \tilde{G}_2$ giving us (with $\hat{\mathbf{u}}_2 = \hat{\mathbf{u}}_1$):

$$H_2 = G_3 V_2\left(W_2 X_2 + \hat{\mathbf{u}}_2 \mathbf{v}_2^H\right).$$

We have performed now a step of the chasing method since the Givens transformation G_3 has shifted down one position w.r.t. the Givens transformation G_2.
The chasing procedure can be continued in a straightforward manner.
Unfortunately this approach does not allow us to determine the last Givens transformation G_{n-1}.

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

Last Givens transformation

Suppose we have performed step $n-3$ and we have the following situation:

$$H_{n-3} \quad = \quad G_{n-2} V_{n-3} \left(W_{n-3} X_{n-3} + \hat{\mathbf{u}}_{n-3} \mathbf{v}_{n-3}^H \right).$$

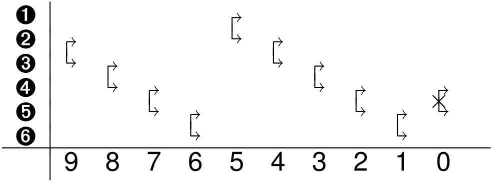
Performing the similarity transformation determined by G_{n-2} results in

$$\begin{aligned} H_{n-2} \quad &= \quad V_{n-3} \left(W_{n-3} X_{n-3} G_{n-2} + \hat{\mathbf{u}}_{n-3} \mathbf{v}_{n-3}^H G_{n-2} \right) \\ &= \quad V_{n-3} \left(W_{n-3} X_{n-3} G_{n-2} + \hat{\mathbf{u}}_{n-2} \mathbf{v}_{n-2}^H \right). \end{aligned}$$

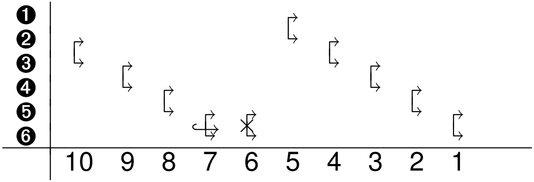
The Givens transformation G_{n-2} works on rows $n-2$ and $n-1$.

Last Givens transformation

$$H_{n-2} \quad = \quad V_{n-3}\left(W_{n-3}X_{n-3}G_{n-2} + \hat{\mathbf{u}}_{n-2}\mathbf{v}_{n-2}^{H}\right).$$



Applying once the shift through operation.



Unfortunately we cannot shift the Givens transformation through anymore, a single fusion can be applied:

$$H_{n-2} \quad = \quad V_{n-2}\left(W_{n-2}X_{n-2} + \hat{\mathbf{u}}_{n-2}\mathbf{v}_{n-2}^{H}\right).$$

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

Explicit determination of last Givens

$$H_{n-2} \quad = \quad V_{n-2}\left(W_{n-2}X_{n-2} + \hat{\mathbf{u}}_{n-2}\mathbf{v}_{n-2}^{H}\right).$$

The representation is of the desired form, but it represents a Hessenberg matrix with an extra bulge in position $(n, n-1)$. This final Givens transformation can hence not be determined implicitly anymore. We determine this Givens transformation explicitly by computing the matrix vector product $H_{n-2}\mathbf{e}_{n-1}$, determine now G_{n-1} such that $\mathbf{e}_n^H G_{n-1}^H H_{n-2}\mathbf{e}_{n-1} = 0$.
The final similarity transformation results in:

$$H_{n-1} \quad = \quad G_{n-1}^{H} V_{n-2}\left(W_{n-2}X_{n-2}G_{n-1} + \hat{\mathbf{u}}_{n-2}\mathbf{v}_{n-2}^{H}G_{n-1}\right).$$

Applying a fusion for the product $G_{n-1}^H V_{n-2}$ and a fusion for the product $X_{n-2}G_{n-1}$ results in

$$H_{n-1} \quad = \quad V_{n-1}\left(W_{n-1}X_{n-1} + \hat{\mathbf{u}}_{n-1}\mathbf{v}_{n-1}^{H}\right),$$

which is a new Hessenberg matrix, i.e., a sum of a unitary and a rank one matrix, represented using the desired representation.

Outline

Outline

Scaling / balancing

- For a general square matrix A one can use a diagonal similarity transformation

$$\det(\lambda I - A) = \det(\lambda I - DAD^{-1})$$

with D a suitable diagonal matrix.
In general this will destroy the unitary plus rank one structure of the companion matrix C_p.
- The zeros of the monic polynomial $p(z)$ are α times the zeros of the monic polynomial $\tilde{p}(\tilde{z}) = \frac{p(\alpha\tilde{z})}{\alpha^n}$, i.e.,

$$\det(\lambda I - C_p) = \det(\lambda I - DC_pD^{-1}) = \det(\lambda I - \alpha C_{\tilde{p}})$$

with $D = \text{diag}(1, \alpha, \alpha^2, \ldots, \alpha^{n-1})$.
- The choice of a good value for the scaling parameter α is very important.

Marc Van Barel
Joint work with Raf Vandebril and Paul Van Dooren

Outline

Different methods

D. Bindel, S. Chandrasekaran, J. W. Demmel, D. Garmire, and M. Gu.
A fast and stable nonsymmetric eigensolver for certain structured matrices.
Technical report, Department of Computer Science, University of California, Berkeley, California, USA, May 2005.

D. A. Bini, F. Daddi, and L. Gemignani.
On the shifted *QR* iteration applied to companion matrices.
Electronic Transactions on Numerical Analysis, 18:137–152, 2004.

D. A. Bini, Y. Eidelman, L. Gemignani, and I. C. Gohberg.
Fast QR eigenvalue algorithms for Hessenberg matrices which are rank-one perturbations of unitary matrices.
SIAM Journal on Matrix Analysis and Applications, 29(2):566–585, 2007.

D. A. Bini, L. Gemignani, and V. Y. Pan.
Fast and stable *QR* eigenvalue algorithms for generalized companion matrices and secular equations.
Numerische Mathematik, 100(3):373–408, 2005.

S. Chandrasekaran, M. Gu, J. Xia, and J. Zhu.
A fast QR algorithm for companion matrices.
Operator Theory: Advances and Applications, 179:111–143, 2007.

S. Delvaux, K. Frederix, and M. Van Barel.
An algorithm for computing the eigenvalues of block companion matrices.
Technical report, Department of Computer Science, Katholieke Universiteit Leuven, 2008.
In preparation.

Methods considered

- method eig from Matlab
- method eig-nobalancing from Matlab
- method Bini et al.: compqr.m (website of Gemignani)
- method Delvaux et al.
- method talk but implemented with double shift (real arithmetic)

Criteria

- maximum relative backward error on the coefficients of the corresponding polynomial

$$\max_i \frac{|\tilde{p}_i - p_i|}{|p_i|}$$

where the coefficients are computed in high precision based on the zeros
- average number of iterations per eigenvalue
- check the $O(n^2)$ complexity

Results: computational complexity

polynomials with coefficients uniformly random from $[1,2]$
degrees: $25, 50, 100, 200, 400$

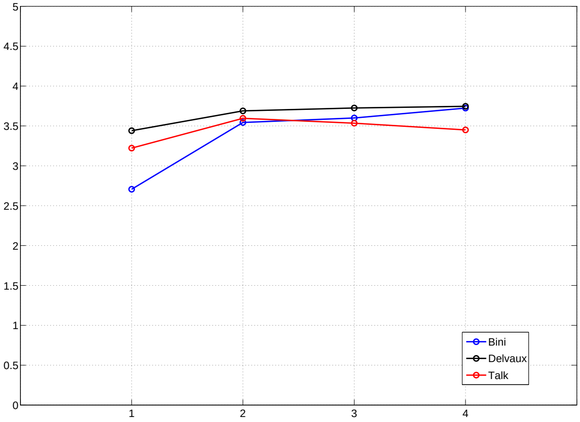


Figure: T_k / T_{k-1} with T_k the execution time for size $n = 25 * 2^k$ (5 samples)

Results: iterations / eigenvalue

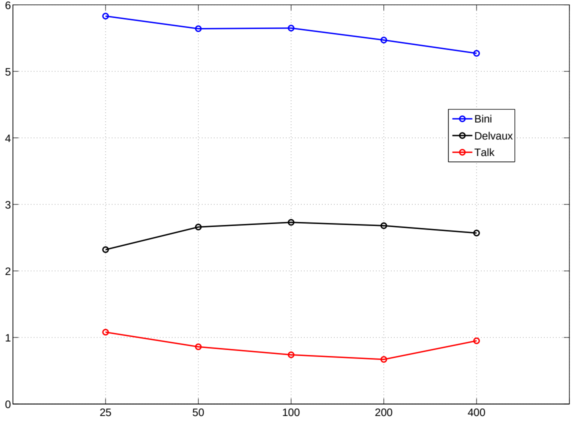


Figure: average number of iterations per eigenvalue (5 samples)

Results: relative backward error

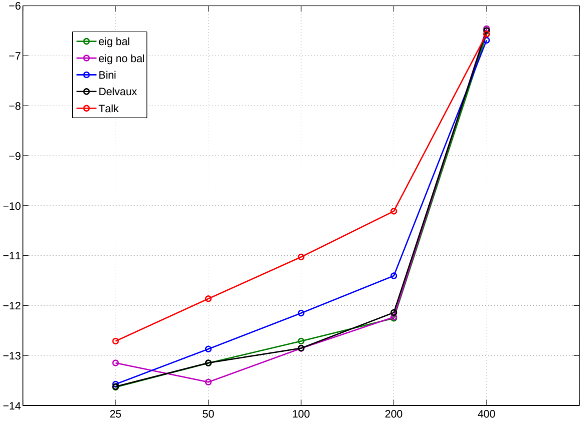


Figure: average number of maximum relative backward error on the coefficients (5 samples) coefficients are the same as before except their sign is randomly chosen

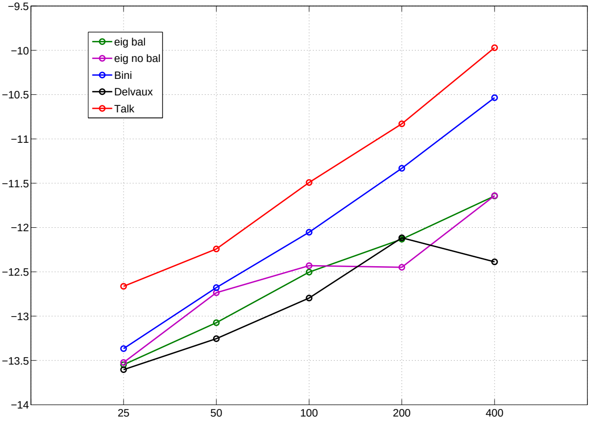


Figure: average number of maximum relative backward error on the coefficients (5 samples)

Results: influence of scaling

Polynomials of degree $n = 20$:

1. Wilkinson polynomial with roots: k with $k = 1, \ldots, n$.
2. Monic polynomial with roots: $[-2.1 : 0.2 : 1.7]$.
3. Monic polynomial with roots: 2^k with $k = -10, \ldots, 9$.
4. Scaled Wilkinson polynomial with roots: $\frac{k}{n}$ with $k = 1, \ldots, n$.
5. Reversed Wilkinson polynomial with roots: $\frac{1}{k}$ with $k = 1, \ldots, n$.
6. Polynomial with well separated roots: $\frac{1}{2^k}$ with $k = 1, \ldots, n$.
7. Polynomial $p(z) = (20!) \sum_{k=0}^{20} \frac{z^k}{k!}$.
8. Polynomial $p(z) = z^{20} + z^{19} + \cdots + z + 1$.

Results: relative backward error

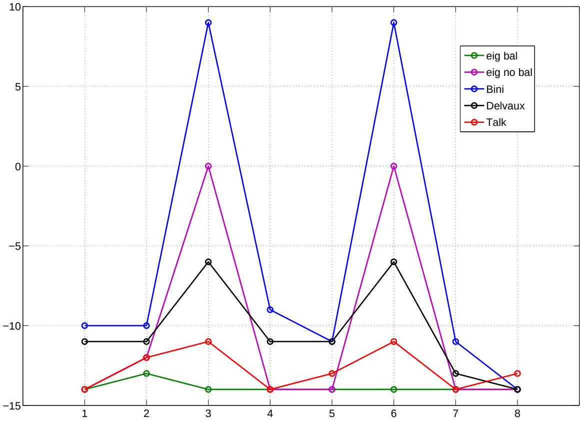


Figure: maximum relative backward error on the coefficients (for the optimal choice of the scaling parameter α); the scaling parameter is chosen as a power of 2

Results: relative backward error, influence of scaling

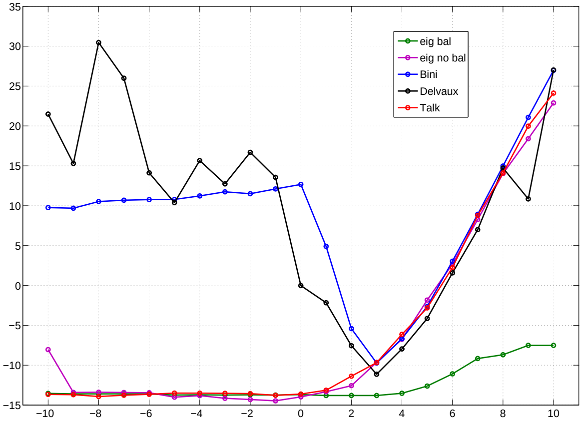


Figure: maximum relative backward error on the coefficients (for different choices of the scaling parameter α)