The next step in the never-ending process of generalizing Francis’s implicitly-shifted $QR$ algorithm

David S. Watkins
watkins@math.wsu.edu

Department of Mathematics
Washington State University
This is joint work ...
This is joint work . . .

- . . . with Raf Vandebril.
This is joint work . . .

- ... with Raf Vandebril.
- ... mostly Raf’s work!
Francis’s Algorithm
Francis’s Algorithm

- requires Hessenberg matrix
Francis’s Algorithm

- requires Hessenberg matrix
- we know how
Francis’s Algorithm

- requires Hessenberg matrix
- we know how

\[
\begin{pmatrix}
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\end{pmatrix}
\]
Reduce to Triangular Form
Reduce to Triangular Form

\[
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\Box & \times & \times & \times & \times & \times \\
\Box & \times & \times & \times & \times & \times \\
\Box & \times & \times & \times & \times & \times \\
\Box & \times & \times & \times & \times & \times \\
\end{bmatrix}
= \begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\end{bmatrix}
\]
Reduce to Triangular Form

\[
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \\
\times & \times & \times & \times & \times & \\
\times & \times & \times & \times & \\
\times & \times & & & & \\
\end{bmatrix}
= 
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \\
\times & \times & \times & \times & \times & \\
\times & \times & \times & \times & \times & \\
\times & \times & \times & \times & \times & \\
\end{bmatrix}
\]

This yields a QR decomposition.
\[ QR \text{ Decomposed Hessenberg matrix} \]

\[
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\end{bmatrix}
\Rightarrow
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\end{bmatrix}
\]
Decomposed Hessenberg matrix

\[
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\end{bmatrix}
\rightarrow
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\end{bmatrix}
\]

... a way to represent the matrix.
Hessenberg matrix (from now on)
Inverse of a Hessenberg matrix
Inverse of a Hessenberg matrix

\[
\begin{bmatrix}
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times & \\
\times & \times & \times & \times & \\
\times & \times & \times & \\
\times & \\
\end{bmatrix}
\]

…an attainable form!
Another Possibility

\[
\begin{bmatrix}
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\end{bmatrix}
\]
Another Possibility

<table>
<thead>
<tr>
<th>CMV form</th>
</tr>
</thead>
</table>
Another Possibility

- CMV form
- Some rotations commute.
CMV Form

\[
\begin{array}{cccccc}
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\times & \times & \times & \times & \times \\
\end{array}
\]
CMV Form

also attainable
CMV Form

- also attainable
- rotators can appear in any order
also attainable

rotators can appear in any order

There are variants of Francis’s algorithm for all of these forms.
Allowed Operations
Allowed Operations

- fusion

\[ \Downarrow \quad \Downarrow \quad \Rightarrow \quad \Downarrow \]
Allowed Operations

- fusion

- shift through
Allowed Operations, continued
Allowed Operations, continued

- shift through triangular matrix

\[
\begin{bmatrix}
\times & \times & \times \\
\times & \times & \\
& \times & 
\end{bmatrix}
\overset{\Longleftrightarrow}{\longrightarrow}
\begin{bmatrix}
\times & \times & \times \\
& \times & \times \\
& & \times 
\end{bmatrix}
\]

- structure commutes
Francis iteration on Hessenberg form
Francis iteration on Hessenberg form

- single shift for simplicity
Francis iteration on Hessenberg form

- single shift for simplicity  (can do any number)
Francis iteration on Hessenberg form

- single shift for simplicity (can do any number)
- create a bulge
Francis iteration on Hessenberg form

- single shift for simplicity (can do any number)
- create a bulge and chase it
Francis iteration on Hessenberg form

- single shift for simplicity (can do any number)
- create a bulge and chase it
Suppress the triangular matrix.
Think of the unitary case.
Eliminate rotator in rows 2 and 3.
- Eliminate rotator in rows 2 and 3.
- Don’t touch first row.
Eliminate rotator in rows 3 and 4.
Eliminate rotator in rows 4 and 5.
Done!
Francis iteration on inverse Hessenberg
Francis iteration on inverse Hessenberg

(triangular matrix suppressed)
Now eliminate the rotator on the right.
Done!
Francis iteration on an “arbitrary” pattern

(triangular matrix suppressed)
Now go the other way.

and so on ...
Comparing start with finish

Pattern moves upward by one.
Two ways to finish

Bottom rotator can be on left or right.
Does it work?
Does it work?

- Raf tried it out.
Does it work?

- Raf tried it out.
- It works great!
Does it work?

- Raf tried it out.
- It works great!
- Can we establish some convergence theory?
Does it work?

- Raf tried it out.
- It works great!
- Can we establish some convergence theory?
- Yes, we can!
Does it work?

- Raf tried it out.
- It works great!
- Can we establish some convergence theory?
- Yes, we can!
- Multishift iterations of any degree
What is Francis’s algorithm?
What is Francis’s algorithm?

- It’s nested subspace iteration ...
What is Francis’s algorithm?

- It’s nested subspace iteration . . . with changes of coordinate system.
What is Francis’s algorithm?

- It’s nested subspace iteration . . . with changes of coordinate system.
- No reliance on implicit-\(Q\) theorem.
What is Francis’s algorithm?

- It’s nested subspace iteration . . . with changes of coordinate system.
- No reliance on implicit-$Q$ theorem.
- DSW, A M Monthly (May 2011)
What is Francis’s algorithm?

- It’s nested subspace iteration . . . with changes of coordinate system.
- No reliance on implicit-$Q$ theorem.
- DSW, A M Monthly (May 2011)
What is Francis’s algorithm?

- It’s nested subspace iteration . . . with changes of coordinate system.
- No reliance on implicit-$Q$ theorem.
- DSW, A M Monthly (May 2011)

- Check this out!
What is Francis’s algorithm?
What is Francis’s algorithm?

- It’s nested subspace iteration ...
What is Francis’s algorithm?

- It’s nested subspace iteration . . . on Krylov subspaces. (from Hessenberg form)
What is Francis’s algorithm?

- It’s nested subspace iteration . . .
  on Krylov subspaces. (from Hessenberg form)

\[ \text{span}\{e_1\} \]
\[ \text{span}\{e_1, Ae_1\} \]
\[ \text{span}\{e_1, Ae_1, A^2e_1\} \]
\[ \text{span}\{e_1, Ae_1, A^2e_1, A^3e_1\} \]
What is Francis’s algorithm?

- It’s nested subspace iteration . . .
  on Krylov subspaces. (from Hessenberg form)

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2e_1\} \\
\text{span}\{e_1, Ae_1, A^2e_1, A^3e_1\}
\end{align*}
\]

- For other forms, adjust the Krylov subspaces
Example: inverse Hessenberg form
Example: inverse Hessenberg form

$$\text{span}\{e_1\}$$
$$\text{span}\{e_1, A^{-1}e_1\}$$
$$\text{span}\{e_1, A^{-1}e_1, A^{-2}e_1\}$$
$$\text{span}\{e_1, A^{-1}e_1, A^{-2}e_1, A^{-3}e_1\}$$
Example: inverse Hessenberg form

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, A^{-1}e_1\} \\
\text{span}\{e_1, A^{-1}e_1, A^{-2}e_1\} \\
\text{span}\{e_1, A^{-1}e_1, A^{-2}e_1, A^{-3}e_1\}
\end{align*}
\]

and in general ...
An “arbitrary” pattern
An “arbitrary” pattern
An “arbitrary” pattern

\[ \text{span}\{e_1\} \]
An “arbitrary” pattern

\[ \text{span}\{e_1\} \]
\[ \text{span}\{e_1, Ae_1\} \]
An “arbitrary” pattern

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2e_1\}
\end{align*}
\]
An “arbitrary” pattern

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2 e_1\} \\
\text{span}\{A^{-1} e_1, e_1, Ae_1, A^2 e_1\}
\end{align*}
\]
An “arbitrary” pattern

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2e_1\} \\
\text{span}\{A^{-1}e_1, e_1, Ae_1, A^2e_1\} \\
\text{span}\{A^{-2}e_1, \ldots, A^2e_1\}
\end{align*}
\]
An “arbitrary” pattern

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2 e_1\} \\
\text{span}\{A^{-1} e_1, e_1, Ae_1, A^2 e_1\} \\
\text{span}\{A^{-2} e_1, \ldots, A^2 e_1\} \\
\text{span}\{A^{-3} e_1, \ldots, A^2 e_1\}
\end{align*}
\]
An “arbitrary” pattern

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2e_1\} \\
\text{span}\{A^{-1}e_1, e_1, Ae_1, A^2e_1\} \\
\text{span}\{A^{-2}e_1, \ldots, A^2e_1\} \\
\text{span}\{A^{-3}e_1, \ldots, A^2e_1\} \\
\text{span}\{A^{-3}e_1, \ldots, A^3e_1\}
\end{align*}
\]
An “arbitrary” pattern

\[
\begin{align*}
\text{span}\{e_1\} \\
\text{span}\{e_1, Ae_1\} \\
\text{span}\{e_1, Ae_1, A^2e_1\} \\
\text{span}\{A^{-1}e_1, e_1, Ae_1, A^2e_1\} \\
\text{span}\{A^{-2}e_1, \ldots, A^2e_1\} \\
\text{span}\{A^{-3}e_1, \ldots, A^2e_1\} \\
\text{span}\{A^{-3}e_1, \ldots, A^3e_1\} \\
\text{span}\{A^{-3}e_1, \ldots, A^4e_1\}
\end{align*}
\]
Final Remarks
Final Remarks

- With the new spaces, the convergence theory carries through as before.
Final Remarks

- With the new spaces, the convergence theory carries through as before.
- Position of final rotator affects convergence rate.
Final Remarks

- With the new spaces, the convergence theory carries through as before.
- Position of final rotator affects convergence rate.
- Subspace iteration:

\[ A - \rho I \quad \text{or} \quad A^{-1} - \rho^{-1} I \]
Final Remarks

- With the new spaces, the convergence theory carries through as before.
- Position of final rotator affects convergence rate.
- Subspace iteration:
  \[
  A - \rho I \quad \text{or} \quad A^{-1} - \rho^{-1} I
  \]
- I must be about out of time.
Final Remarks

- With the new spaces, the convergence theory carries through as before.
- Position of final rotator affects convergence rate.
- Subspace iteration:

\[ A - \rho I \quad \text{or} \quad A^{-1} - \rho^{-1} I \]

- I must be about out of time.
- Thank you for your attention.