Treating breakdowns and near breakdowns in JHESS-algorithm for a reducing a matrix to upper $J$-Hessenberg form

Ahmed Salam$^1$ Haithem Ben Kahla $^1$

$^1$Université Lille Nord de France, ULCO, LMPA, Calais, France

NASCA18, July 2-6, 2018.
Kalamata, Greece
Motivation
SRDECO algorithm
Existence of SR decomposition - link with SRDECO
JHESS algorithm
Curing breakdowns or treating near-breakdowns in JHESS
Numerical experiments
The aim of this talk is to treat breakdown and near breakdown in JHESS algorithm.

JHESS (and recent variants) reduces a matrix to a condensed form: the upper J-Hessenberg form, via elementary symplectic transformations.

The $J$-Hessenberg reduction is a key step in the $SR$-algorithm for computing eigensystems of a class of structured matrices as Hamiltonian, skew Hamiltonian, symplectic matrices.

The $J$-Hessenberg form is crucial for structure-preserving model reduction methods for a class of large and sparse structured matrices.
We recall that a matrix \( H = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \in \mathbb{R}^{2n \times 2n} \), is upper J-Hessenberg when \( H_{11}, H_{21}, H_{22} \) are upper triangular and \( H_{12} \) is upper Hessenberg.

For example, \( n = 4 \), we have

\[
H = \begin{bmatrix}
  x & x & x & x & x & x & x & x \\
  0 & x & x & x & x & x & x & x \\
  0 & 0 & x & x & x & x & x & x \\
  0 & 0 & 0 & x & x & x & x & x \\
  x & x & x & x & x & x & x & x \\
  0 & x & x & x & x & x & x & x \\
  0 & 0 & x & x & x & x & x & x \\
  0 & 0 & 0 & x & x & x & x & x \\
  0 & 0 & 0 & 0 & x & x & x & x \\
  0 & 0 & 0 & 0 & x & x & x & x \\
\end{bmatrix}
\]
Van Loan’s transformations

Let $J_{2n}$ be $J_{2n} = J = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix}$. A matrix $S$ is symplectic iff $S^TJS = J$.

Orthogonal and symplectic transformations

- Van Loan’s Householder transformation: $H(k, w) = \begin{pmatrix} \text{diag}(I_{k-1}, P) & 0 \\ 0 & \text{diag}(I_{k-1}, P) \end{pmatrix}$, where $P = I - 2ww^T/w^Tw$, $w \in \mathbb{R}^{n-k+1}$.

- Van Loan’s Givens transformation: $J(k, c, s) = \begin{pmatrix} C & S \\ -S & C \end{pmatrix}$, where $C = \text{diag}(I_{k-1}, c, I_{n-k})$ and $S = \text{diag}(0_{k-1}, s, 0_{n-k})$, with $c^2 + s^2 = 1$. 
Symplectic Gauss transformations

The $SR$ factorization (and reduction to J-Hessenberg) can not be performed for a general matrix by using the sole $H$ and $J$ transformations. A third type $G$, introduced in Bunse-Gestner 1986, is given by

$$G(k, \nu) = \begin{pmatrix} D & F \\ 0 & D^{-1} \end{pmatrix},$$  \hspace{1cm} (1)

where $k \in \{2, \ldots, n\}$, $\nu \in \mathbb{R}$ and $D$, $F$ are the $n \times n$ matrices

$$D = I_n + \left(\frac{1}{(1 + \nu^2)^{1/4}} - 1\right)(e_{k-1}e_{k-1}^T + e_k e_k^T),$$

$$F = \frac{\nu}{(1 + \nu^2)^{1/4}}(e_{k-1}e_k^T + e_k e_{k-1}^T).$$

The matrix $G(k, \nu)$ is symplectic and non-orthogonal.
SR-decomposition

It is the equivalent of QR decomposition.

- **SR decomposition**: \( A = SR \), with \( S \) symplectic and \( R \) \( J \)-triangular.

- \( R \) \( J \)-triangular means \( R = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \) with \( R_{11}, R_{12}, R_{22} \) are upper triangular and \( R_{21} \) is strictly upper triangular.
The SRDECO algorithm, as introduced in Bunse -Gestner et al 86: Given $A \in \mathbb{R}^{2n \times 2n}$, the algorithm determines an $SR$ decomposition of $A$, using functions

- $vlg$ (symplectic and orthogonal)
- $vlh$ (symplectic and orthogonal)
- $gau$ (symplectic but not orthogonal)
The function \texttt{vlg} uses Van Loan’s Givens transformation \( J(k, c, s) \) as follows: for a given integer \( 1 \leq k \leq n \) and a vector \( a \in \mathbb{R}^{2n} \), it determines coefficients \( c \) and \( s \) such that the \( n + k \)th component of \( J(k, c, s)a \) is zero. All components of \( J(k, c, s)a \) remain unchanged, except eventually the \( k \)th and the \( n + k \).

\texttt{vlg}

\begin{verbatim}
function [c, s] = vlg(k, a)
    1. twon = length(a); n = twon/2;
    2. r = sqrt(a(k)^2 + a(n+k)^2);
    3. if r = 0 then c = 1; s = 0;
    4. else 5. c = a(k)/r; s = a(n+k)/r;
    end
\end{verbatim}
The function \textit{vlh} uses Van Loan's Householder transformation \(H(k, w)\) as follows: for a given integer \(k \leq n\) and a vector \(a \in \mathbb{R}^{2n}\), a vector \(w = (w_1, \ldots, w_{n-k+1})^T\) is determined such that the components \(k+1, \ldots, n\) of \(H(k, w)a\) are zeros. All components \(1, \ldots, k-1\) and \(n+1, \ldots, n+k-1\) remain unchanged.

\textbf{vlh}

\begin{verbatim}
function[\beta,w]=vlh(k,a)
1. twon = length(a); n = twon/2;
  \% w = (w_1, \ldots, w_{n-k+1})^T;
2. r1 = \sum_{i=2}^{n-k+1} a(i + k - 1)^2;
3. r = \sqrt{a(k)^2 + r1};
4. w_1 = a(k) + sign(a(k))r;
5. w_i = a(i + k - 1) for i = 2, \ldots, n - k + 1;
6. r = w_1^2 + r1; \beta = \frac{2}{r};
  \%P = I - \beta w w^T; (H(k, w)a)_i = 0 for i = k + 1, \ldots, n.
\end{verbatim}
The function \( gau \) uses the transformation \( G(k, \nu) \) as follows: for a given integer \( k \leq n \) and a vector \( a \in \mathbb{R}^{2n} \), satisfying the condition \( a_{n+k} = 0 \) only if \( a_{k+1} = 0 \), it determines \( \nu \) such that the \( k+1 \)th component of \( G(k, \nu)a \) is zero.

**gau**

function\([\nu] = gau(k, a)\)
1. \( \text{twon} = \text{length}(a); \ n = \text{twon}/2; \)
2. if \( a_{k+1} = 0 \)
3. \( \nu = 0; \)
4. else
5. \( \nu = -\frac{a_{k+1}}{a_{n+k}}; \)
6. end
7. end
The matrix \( A \) is overwritten by the \( J \)-upper triangular matrix.

**SRDECO**

function \([S,A]=SRDECO(A)\)

0. \( S = I \),

1. For \( j = 1, \ldots, n \)

2. For \( k = n, \ldots, j \)

3. Zero entry \((n+k,j)\) of \( A \): \([c,s] = vlg(k,A(:,j)), J_{k,j} = J(k,c,s)\).

4. Update \( A = J_{k,j}A \) and \( S = SJ_{k,j}^T \)

5. End for.

6. Zero entries \((j+1,\ldots,n)\) of \( A(:,j) \): \([\beta,w] = vlh(j,A(:,j))\),

7. \( H_j = H(j,w) \). Update \( A = H_j A \) and \( S = SH_j^T \).

8. If \( j \leq n - 1 \)

9. For \( k = n, \ldots, j + 1 \)

10. Zero the entry \((n+k,n+j)\) of \( A \) by running the function \([c,s] = vlg(k,A(:,n+j))\) and computing \( J_{k,n+j} = J(k,c,s) \).
11. Update \( A = J_{k,n+j}A \) and \( S = SJ_{k,n+j}^T \).
12. End for.
13. Zero the entries \((j + 2, n + j), \ldots, (n, n + j)\) of \( A \) by running the function \([β, w] = vlh(j + 1, A(:, j))\) and computing \( H_{n+j} = H(j + 1, w) \).
14. Update \( A = H_{n+j}A \) and \( S = SH_{n+j}^T \).
15. If the entry \((j + 1, n + j)\) of \( A \) is nonzero and the entry \((n + j, n + j)\) is zero then stop the algorithm, %breakdown
16. else
17. Zero the entry \((j + 1, n + j)\) of \( A \) by running the function \([ν] = gau(j + 1, A(:, n + j))\) and computing \( G_{j+1} = G(j + 1, ν) \).
18. Update \( A = G_{j+1}A \) and \( S = SG_{j+1}^{-1} \). % \( G_{j+1}^{-1} = J^T G_{j+1}^T J \).
19. End if
20. End if
Existence of SR decomposition

Elsner L. 79:

**Theorem**

Let $A \in \mathbb{R}^{2n \times 2n}$ be nonsingular and $P$ the permutation matrix $P = [e_1, e_{n+1}, e_2, e_{n+2}, \ldots, e_n, e_{2n}]$, where $e_i$ denotes the $i$th canonical vector of $\mathbb{R}^{2n}$. There exists $S \in \mathbb{R}^{2n \times 2n}$ symplectic and $R \in \mathbb{R}^{2n \times 2n}$ upper $J$-triangular, such that $A = SR$ if and only if all even leading minors of $P^T A^T J A P$ are nonzero.
We established

**Theorem**

Let $A \in \mathbb{R}^{2n \times 2n}$ be a matrix (not necessarily nonsingular), and let $R$ be the matrix that one obtains at stage $1 \leq j \leq n - 1$ of the algorithm SRDECO, by executing instructions 1. to 14. (corresponding to the current updated matrix $A$ in the process, at stage $j$ and until instruction 14.). Then the leading $2j$-by-$2j$ minor of $P^T A^T JAP$ satisfies

$$\det((P^T A^T JAP)_{[2j,2j]}) = \begin{bmatrix} r_{1,1} & r_{n+1,n+1} & \cdots & r_{i,i} & r_{n+i,n+i} & \cdots & r_{j,j} & r_{n+j,n+j} \end{bmatrix}^2. \quad (2)$$
**Corollary**

Let $A \in \mathbb{R}^{2n \times 2n}$ be a nonsingular matrix, and let $R$ be the matrix that one obtains at stage $1 \leq j \leq n - 1$ of the algorithm SRDECO or SRMSH, by executing instructions 1. to 14. (corresponding to the current updated matrix $A$ in the process, at stage $j$ and until instruction 14.). Then $A$ admits an SR decomposition if and only if $r_{n+j,n+j} \neq 0$, $\forall j \in \{1, \ldots, n\}$.

**Theorem**

Let $A \in \mathbb{R}^{2n \times 2n}$ be any matrix, and let $R$ be the matrix that one obtains at stage $1 \leq j \leq n - 1$ of the algorithm SRDECO (or SRMSH), by executing instructions 1. to 14. (corresponding to the current updated matrix $A$ in the process, at stage $j$ and until instruction 14.) Then $A$ admits an SR decomposition if and only if $(r_{n+j,n+j} \neq 0$ or $r_{j+1,n+j} = 0), \forall j \in \{1, \ldots, n\}$. 
JHESS (Bunse-Gestner 86) reduces (if possible) $A \in \mathbb{R}^{2n \times 2n}$ to upper $J$-Hessenberg form $H = \Pi^{-1} A \Pi$, with a symplectic matrix $\Pi$ whose first column is a multiple of $e_1$. $A$ is overwritten by the $J$-Hessenberg matrix $H$ and $S$ is overwritten by the transforming matrix $\Pi$. If this reduction of $A$ does not exist, the algorithm stops.”
JHESS algorithm

function [S,A]=JHESS(A)
1. For \( j = 1, \ldots, n - 1 \)
2. For \( k = n, \ldots, j + 1 \)
3. Zero the entry \((n + k, j)\) of \(A\) by running the function \([c, s] = vlg(k, A(:, j))\) and computing \(J_{k,j} = J(k, c, s)\).
4. Update \(A = J_{k,j}A J_{k,j}^T\) and \(S = SJ_{k,j}^T\).
5. End for.
6. Zero the entries \((j + 2, j), \ldots, (n, j)\) of \(A\) by running the function \([\beta, w] = vlh(j + 1, A(:, j))\) and computing \(H_j = H(j + 1, w)\).
7. Update \(A = H_j A H_j^T\) and \(S = S H_j^T\).
8. If the entry \((j + 1, j)\) of \(A\) is nonzero and the entry \((n + j, j)\) is zero then stop the algorithm % condition of breakdown
9. else
10. Zero the entry \((j + 1, j)\) of \(A\) by running the \([\nu] = gau(A(:, j))\)
JHESS algorithm

### JHESS

11. Compute $G_{j+1} = G(j + 1, \nu)$.
12. Update $A = G_{j+1}A G_{j+1}^{-1}$ and $S = SG_{j+1}^{-1}$.
13. End if
14. For $k = n, \ldots, j + 1$
15. Zero the entry $(n + k, n + j)$ of $A$ by running the function $[c, s] = vlg(k, A(:, n + j))$ and compute $J_{k,n+j} = J(k, c, s)$.
16. Update $A = J_{k,n+j}A J_{k,n+j}^T$ and $S = SJ_{k,n+j}^T$.
17. End for.
18. If $j \leq n - 2$
19. Zero the entries $(j + 2, n + j), \ldots, (n, n + j)$ of $A$ by running the function $[\beta, w] = vlh(j + 1, A(:, n + j))$ and compute $H_{n+j} = H(j + 1, w)$.
20. Update $A = H_{n+j}A H_{n+j}^T$ and $S = SH_{n+j}^T$.
21. End if
22. End for.
Let $A_6$ be the 6-by-6 matrix

$$A_6 = \begin{pmatrix}
1 & 0 & 0 & 1 & 2 & 0 \\
2 & 1 & 0 & 2 & 1 & 0 \\
0 & 2 & 1 & 0 & 2 & 1 \\
0 & 2 & 0 & 1 & 0 & 0 \\
0 & 1 & 2 & 3 & 1 & 0 \\
0 & 0 & 1 & 0 & 3 & 1 \\
\end{pmatrix}. \quad (3)$$

- Breakdown ($1^{st}$ step): $A_6(2, 1) \neq 0 \& A_6(4, 1) = 0$.
- The SR-algorithm (QR-like algorithm) breaks also at the first step.
The $j$th iteration is stopped if breakdown or $\text{cond}(G_{j+1}) \geq \tau$ a certain tolerance.

A similarity $S_j^{-1}AS_j$ is computed, with $S_j = I - ww^T J$, where $w$ is a random vector with $\|w\|_2 = 1$.

The algorithm JHESS is then applied to $S_j^{-1}AS_j$.

If the number of encountered near-breakdowns/breakdowns exceeds a given bound, the whole process is definitively stopped.

Drawbacks:

- $\text{cond}(S_j^{-1}AS_j)$ will be worse than of $\text{cond}(A)$. ($S_j$ sympl., not orth.)
- The cost of forming the product $S_j^{-1}AS_j$ is $O(n^2)$, size($A$) = $2n$.
- The product $S_j^{-1}AS_j$ fills-up the matrix and destroys the previous partially created $J$-Hessenberg form of $A$. A cost of $O(n^3)$ is needed to restore the $J$-Hessenberg form.
Salam et al. 2017. Alternative: Compute a similarity transformation $S_j^{-1}A S_j$ for which:

- The proposed matrix $S_j$ is not only symplectic but also orthogonal. Thus, the condition number of $S_j^{-1}A S_j$ is preserved, and the process is numerically as accurate as possible.
- The cost for computing the product $S_j^{-1}A S_j$ is only $O(n)$. Thus, a gain of an order-of-magnitude is guaranteed.
- The product $S_j^{-1}A S_j$ does not fills-up the matrix and preserves all the created zeros in previous steps. Also, to restore the $J$-Hessenberg form of $A$, only a cheaper additional cost of $O(n^2)$ is needed.
The idea

Consider the example \( A_6 \). The idea: one seeks for a symplectic and orthogonal transforming matrix \( S \) so that the similar matrix \( SA_6S^{-1} \), may be reduced by JHESS. A judicious choice of \( S \) consists in taken \( S \) equal to Van Loan’s Householder matrix

\[
S = \begin{pmatrix}
H_2 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & H_2 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix},
\]

or Van Loan’s Givens matrix

\[
S = \begin{pmatrix}
G_2 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & G_2 & 0 \\
0 & 0 & 0 & 1
\end{pmatrix},
\]

\( H_2 \) (resp. \( G_2 \)) is a 2x2 Householder (resp. 2x2 Givens) matrix, such that

\[
H_2(1, 2)^T = \sqrt{5}(1, 0)^T \quad \text{resp.} \quad G_2(1, 2)^T = \sqrt{5}(1, 0)^T.
\]
The idea

With choice (5), we obtain

\[
SA_6 S^{-1} = \begin{pmatrix}
9/5 & -8/5 & 0 & 13/5 & -6/5 & 0 \\
2/5 & 1/5 & 0 & -6/5 & 3/5 & 0 \\
4/\sqrt{5} & 2/\sqrt{5} & 1 & 4/\sqrt{5} & 2/\sqrt{5} & 1 \\
8/5 & 4/5 & 4/\sqrt{5} & 11/5 & 12/5 & 0 \\
-6/5 & -3/5 & 2/\sqrt{5} & 3/5 & -1/5 & 0 \\
0 & 0 & 1 & 6/\sqrt{5} & 3/\sqrt{5} & 1 \\
\end{pmatrix}.
\]

(6)

We applied JHESS to the matrix \(SA_6 S^{-1}\) of (6). The algorithm run well and the reduction to the J-Hessenberg form is obtained. See Salam et al 2017, general strategy.

Remark: breakdowns/near-breakdowns in JHESS may occur only when the function \(gau\) is called, and hence it concerns only columns from the first half of the current matrix.
The near-breakdown occurs when the coefficients $A_{n+j,j}$ and $A_{j+1,j}$ are both different from zero but are near to the situation of breakdown: the ratio $\frac{A_{j+1,j}}{A_{n+j,j}}$ is very large.

In this case, the non-orthogonal and symplectic transformations involved in JHESS become ill-conditioned.

To remedy: proceed as for a breakdown: the test $A_{n+j,j} = 0$ and $A_{j+1,j} \neq 0$ (corresponding to a breakdown) is replaced by the $\frac{A_{j+1,j}}{A_{n+j,j}} \geq \tau$ (corresponding to a near-breakdown), where $\tau$ is a certain tolerance.
Numerical experiments

To illustrate our purpose, we consider the following numerical example. Let $A$ be the 12-by-12 matrix

$$
A = \begin{pmatrix}
1 & 5 & 7 & 9 & 5 & 1 & 1 & 3 & 1 & 3 & 7 & 2 \\
0 & 1 & 4 & 6 & 1 & 2 & 2 & 1 & 5 & 4 & 3 & 5 \\
0 & 0 & 1 & 2 & 3 & 2 & 0 & 0 & 1 & 2 & 5 & 3 \\
0 & 0 & 2 & 1 & 9 & 8 & 0 & 0 & 2 & 1 & 2 & 4 \\
0 & 0 & 0 & 2 & 1 & 3 & 0 & 0 & 5 & 2 & 1 & 2 \\
0 & 0 & 0 & 4 & 2 & 1 & 0 & 0 & 4 & 3 & 2 & 1 \\
1 & 4 & 7 & 2 & 1 & 3 & 1 & 7 & 6 & 1 & 6 & 7 \\
0 & 1 & 9 & 3 & 5 & 1 & 0 & 1 & 4 & 5 & 8 & 3 \\
0 & 0 & 0 & 2 & 7 & 9 & 0 & 0 & 1 & 3 & 4 & 5 \\
0 & 0 & 0 & 1 & 2 & 8 & 0 & 0 & 3 & 1 & 7 & 3 \\
0 & 0 & 0 & 2 & 1 & 2 & 0 & 0 & 4 & 3 & 1 & 2 \\
0 & 0 & 0 & 9 & 3 & 1 & 0 & 0 & 1 & 2 & 3 & 1
\end{pmatrix}.
$$
Numerical experiments

JHESS (also its variants JHMSH and JHMSH2) breaks down at the step $j = 3$.

<table>
<thead>
<tr>
<th></th>
<th>Loss of $J$-Orthogonality $| I - S^J S |_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>JHESS</td>
<td>1.8553e − 15</td>
</tr>
<tr>
<td>MJHESS</td>
<td>5.0842e − 15</td>
</tr>
<tr>
<td>$JHM^2 SH$</td>
<td>6.6428e − 15</td>
</tr>
<tr>
<td>$JHM^2 SH2$</td>
<td></td>
</tr>
<tr>
<td>fails</td>
<td></td>
</tr>
</tbody>
</table>

- The $J$-orthogonality is numerically preserved up to the machine precision for MJHESS (respectively $JHM^2 SH$ and $JHM^2 SH2$).

<table>
<thead>
<tr>
<th></th>
<th>Reduction error $| A - SHS^{-1} |_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>JHESS</td>
<td>3.2709e − 14</td>
</tr>
<tr>
<td>MJHESS</td>
<td>3.8777e − 13</td>
</tr>
<tr>
<td>$JHM^2 SH$</td>
<td>2.7653e − 13</td>
</tr>
<tr>
<td>$JHM^2 SH2$</td>
<td></td>
</tr>
<tr>
<td>fails</td>
<td></td>
</tr>
</tbody>
</table>

- The error in the reduction to $J$-Hessenberg form is very satisfactory for MJHESS (respectively $JHM^2 SH$ and $JHM^2 SH2$).


Thank you.