

# 1      **RESTARTING THE NON-SYMMETRIC LANCZOS ALGORITHM** 2      **VIA THE IMPLICITLY SHIFTED LR ALGORITHM\***

3                      P. S. NEGI<sup>†</sup> AND C. ARRATIA<sup>‡</sup>

4      **Abstract.** The shifted QR iteration is used as a restart procedure for the Arnoldi method for  
5      the calculation of a few dominant eigenvalues of a large matrix. We show that the underlying idea  
6      can be utilized in much the same manner via the shifted LR iteration to create a restart procedure  
7      for the non-symmetric Lanczos algorithm for eigenvalue calculations. Additionally, we show that the  
8      (shifted) LR iteration can be performed implicitly in a manner similar to the Francis' algorithm,  
9      resulting in a bulge-chase type procedure which does not require the explicit construction of the full  
10     lower and upper triangular matrices.

11     **Key words.** LR algorithm, unsymmetric Lanczos, implicit restart

12     **MSC codes.** 68Q25, 68R10, 68U05

13     **1. Introduction.** The Arnoldi iteration [1] is a popular Krylov space method  
14     for calculating a few eigenvalues of a large matrix. The method relies on the gen-  
15     eration of a sequence of Krylov vectors which determine the subspace within which  
16     approximations of the eigenvalue-eigenvector pairs are obtained. Depending on the  
17     accuracy and number of eigenpair approximations needed, the Krylov space size can  
18     become exceedingly large so that the quality of the results may be limited by the  
19     available memory. Sorensen [9] introduced an elegant procedure for restarting the  
20     Arnoldi factorization based on polynomial filters, which are applied through the im-  
21     plicitly shifted QR iterations on the reduced Hessenberg matrix obtained through the  
22     Arnoldi method. In particular, the use of exact shifts was shown to be successful in the  
23     convergence process of the eigenspace[9] of the specified eigenvalues. The method has  
24     subsequently found widespread application through the ARPACK library [4]. The use  
25     of QR iterations ensures that the reduced matrix preserves its Hessenberg structure  
26     through the transforms that make up the restart process. If the underlying matrix is  
27     symmetric, the Arnoldi iteration reduces to the Lanczos algorithm and, the Hessen-  
28     berg matrix reduces to a symmetric tridiagonal matrix. The QR iteration preserves  
29     the symmetric tridiagonal structure as well and, as pointed out by Sorensen [9], the  
30     implicit restart process applies equally well for the Lanczos method for symmetric  
31     matrices.

32     One would like to extend this procedure to the case of the non-symmetric Lanczos  
33     method. However, the reduced matrix that one obtains is a non-symmetric tridiagonal  
34     matrix, with the tridiagonal structure being the result of the recurrence relations of the  
35     Lanczos algorithm [8]. Since the QR iterations do not preserve the banded structure  
36     of non-symmetric matrices, a straightforward application of the restart procedure  
37     propounded by Sorensen will lead to a loss of this tridiagonal structure of the reduced  
38     matrix (the Hessenberg structure will still be preserved). This loss of structure can  
39     be circumvented if one looks to the predecessor of the QR algorithm namely, the  
40     LR algorithm proposed by Rutishauser [6, 7, 5], which has the attractive property of  
41     preserving the band structure of a matrix. This property was already pointed out  
42     by Rutishauser in [6] where the banded matrices were referred to as striped matrices.

---

\*Submitted to the editors DATE.

**Funding:** This work was funded by the Fog Research Institute under contract no. FRI-454.

<sup>†</sup>Imagination Corp., Chicago, IL (prabal.negi@su.se, <http://www.imag.com/~ddoe/>).

<sup>‡</sup>Department of Applied Mathematics, Fictional University, Boise, ID (cristobal.arratia@su.se).

As we will show in the next section, the shifted LR transform is the appropriate generalization of the restart procedure to the case of non-symmetric Lanczos iteration. The process would necessarily require refining both the right as well as the left Krylov spaces simultaneously.

The rest of the paper is organized as follows. In the next section we start with the introduction of the non-symmetric Lanczos iteration and then develop the restart procedure. We also show that the LR algorithm can be implemented as a bulge-chase method, similar to the Francis' algorithm. In section 3 we apply the restart process to the Grcar matrix, and make some concluding remarks in section 4.

**2. Non-symmetric Lanczos.** Lanczos first introduced his algorithm in [3] as a method for tridiagonalizing a matrix, but also realized that the method could be used iteratively to find eigenvalues. For an arbitrary matrix  $A$ , the method generates a pair of Krylov subspaces  $\{v_1, \dots, v_j\}$  and  $\{w_1, \dots, w_j\}$ , through repeated action of  $A$  and  $A^H$  respectively. We refer to these as the right and left Krylov spaces respectively and they satisfy the biorthogonality relation  $w_i^H v_j = \delta_{ij}$ . The two subspaces are generated through a three term recurrence relation which, for a Krylov space of size  $m$ , can be written in matrix form as

$$\begin{aligned} (2.1a) \quad & AV_m - V_m T_m = v_{m+1} e_m^T \\ (2.1b) \quad & A^H W_m - W_m T_m^H = w_{m+1} e_m^T \\ (2.1c) \quad & W_m^H V_m = I_m \end{aligned}$$

where  $I_m$  represents the Identity matrix of size  $m$ ,  $T_m$  is a tri-diagonal matrix of size  $m$  and,  $T_m^H$  is the Hermitian conjugate of  $T_m$ . If either  $v_{m+1}$  or  $w_{m+1}$  vanishes it represents the convergence of the right or the left Krylov subspaces to an invariant subspace of dimension  $m$ . A more serious breakdown occurs if  $w_{m+1}^H v_{m+1} = 0$  with both  $v_{m+1} \neq 0$  and  $w_{m+1} \neq 0$ . We do not address that issue here since it is not specifically related to the restart procedure. We refer the reader to [2] for a comprehensive overview on Lanczos type solvers and the related issues of breakdown.

As Sorensen points out for the Arnoldi method [9], if one is interested in an invariant subspace of dimension  $m$ , the starting vector of Krylov subspace must not contain components of the generator of a cyclic subspace of dimension greater than  $m$ . This applies equally for the right and left Krylov subspaces generated through the Lanczos recurrence relations. Hence a non-vanishing  $v_{m+1}$  (respectively  $w_{m+1}$ ) implies that  $v_1$  (respectively  $w_1$ ) contains components of an invariant subspace of dimension greater than  $m$ . The idea behind restarts then is to discard the components of the starting vector  $v_1$  (and  $w_1$ ) along the unwanted dimensions, such that each restart process moves the Krylov space(s) closer to being invariant. For the Arnoldi method Sorensen [9] proposed to achieve this via polynomial filtering, *i.e.* replacing

$$\begin{aligned} (2.2a) \quad & v_1 \leftarrow \psi(A)v_1, \\ (2.2b) \quad & \psi(\lambda) = (1/\tau) \prod_{j=1}^p (\lambda - \mu_j). \end{aligned}$$

Obviously  $\psi(\lambda)$  is the filtering polynomial,  $\tau$  is a normalization constant and each  $\mu_j$  specifies a node of the polynomial. The polynomial acts on  $v_1$  to filter out the part of the spectrum of  $A$  that is close to each  $\mu_j$ . If a particular  $\mu_j$  corresponds to an exact eigenvalue of  $A$ , then components of the corresponding eigenvector are completely filtered out from  $v_1$ . The node  $\mu_j$  is referred to as a shift since the application of the polynomial filtering relies on the shifted QR algorithm, where  $\mu_j$  is used as the shift. As shown below for the case of a single shift, an analogous procedure can be

89 followed using a shifted LR algorithm which achieves the same effect of applying a  
 90 polynomial filter to the starting vector  $v_1$ . Starting with the Lanczos relation for the  
 91 right subspace 2.1a, and adding and subtracting  $\mu V_m$  we obtain

$$\begin{aligned} 92 \quad (2.3a) \quad & (A - \mu I)V_m - V_m(T_m - \mu I) = v_{m+1}e_m^T \\ 93 \quad (2.3b) \quad & (A - \mu I)V_m - V_m(L_1 R_1) = v_{m+1}e_m^T \\ 94 \quad (2.3c) \quad & (A - \mu I)V_m L_1 - V_m(L_1 R_1)L_1 = v_{m+1}e_m^T L_1 \\ 95 \quad (2.3d) \quad & A(V_m L_1) - (V_m L_1)(R_1 L_1 + \mu I) = v_{m+1}e_m^T L_1 \\ 96 \quad (2.3e) \quad & AV'_m - V'_m T'_m = v_{m+1}e_m^T L_1. \end{aligned}$$

97 Here we have set  $V'_m = V_m L_1$  and  $T'_m = (R_1 L_1 + \mu I)$ . The matrices  $L_1, R_1$  are the  
 98 lower and upper triangular matrices obtained from the LU decomposition of  $(T_m - \mu I)$ .  
 99 The matrix  $L_1$  can be required to be unit triangular (all entries on the main diagonal  
 100 are ones), in which case the LU decomposition is unique. Furthermore,  $L_1$  for a  
 101 tridiagonal matrix only consists of one sub-diagonal (in addition to the main diagonal).  
 102 One can easily recognize that the new reduced matrix  $T'_m$  is a result of one step of the  
 103 shifted LR iteration. Hence  $T'_m$  retains the tridiagonal structure of the  $T_m$  [6, 5]. The  
 104 relationship between generating vectors of the two spaces  $V_m$  and  $V'_m$  can be obtained  
 105 by multiplying (2.3b) by  $e_1$ , *i.e.*

$$\begin{aligned} 106 \quad & (A - \mu I)V_m e_1 - (V'_m)R_1 e_1 = v_{m+1}e_m^T e_1 \\ 107 \quad & \implies (A - \mu I)v_1 = v'_1 \rho_{11}, \end{aligned}$$

108 where  $\rho_{11} = e_1^T R_1 e_1$ . This clearly shows the filtering operation done on the original  
 109 vector  $v_1$  to generate the new vector  $v'_1$ .

110 Since the Lanczos method creates a biorthogonal basis, one must simultaneously  
 111 transform the left basis  $W_m$  to maintain the biorthogonality property. It is easy to see  
 112 that the necessary transform to maintain biorthogonality is  $W'_m = W_m L_1^{-H}$ , since,

$$113 \quad W'^H V'_m = (W_m L_1^{-H})^H (V_m L_1) = L_1^{-1} (W_m^H V_m) L_1 = I$$

114 We can obtain the modified Lanczos relation for the left Krylov space as

$$\begin{aligned} 115 \quad (2.4) \quad & A^H W_m - W_m (R_1^H L_1^H + \bar{\mu} I) = w_{m+1}e_m^T \\ 116 \quad (2.5) \quad & A^H (W_m L_1^{-H}) - W_m (L_1^{-H} L_1) (R_1^H L_1^H + \bar{\mu} I) L_1^{-H} = w_{m+1}e_m^T L_1^{-H} \\ 117 \quad (2.6) \quad & A^H (W_m L_1^{-H}) - (W_m L_1^{-H}) (L_1^H R_1^H + \bar{\mu} I) = w_{m+1}e_m^T L_1^{-H} \\ 118 \quad (2.7) \quad & A^H W'_m - W'_m (T'_m)^H = w_{m+1}e_m^T L_1^{-H}. \end{aligned}$$

119 Conveniently the structure of the Lanczos iteration for the left Krylov space is also  
 120 preserved. Noting that  $L_1^{-H}$  is upper triangular, one can again expose the relationship  
 121 between the generating vectors of the two left Krylov spaces as

$$\begin{aligned} 122 \quad & W'_1 e_1 = (W_m L_1^{-H}) e_1 \\ 123 \quad & \implies w'_1 = w_1 (e_1^T L_1^{-H} e_1). \end{aligned}$$

124 Clearly  $w'_1$  is simply a scalar multiple of the old vector  $w_1$  and no filtering of the  
 125 generating vector has occurred. In order to ensure that we filter the left Krylov space  
 126 as well, we perform one step of the shifted LR iteration with the conjugated shift  $\bar{\mu}$   
 127 on the reduced matrix  $(T'_m)^H$  obtained in equation 2.7. The right Krylov space is  
 128 modified again to maintain orthogonality.

$$\begin{aligned}
(2.8a) \quad & (A - \mu I)^H W'_m - W'_m (T'_m - \mu I)^H = w_{m+1} e_m^T L_1^{-H} \\
(2.8b) \quad & (A - \mu I)^H W'_m - W'_m (L_2 R_2) = w_{m+1} e_m^T L_1^{-H} \\
(2.8c) \quad & (A - \mu I)^H W'_m L_2 - W'_m (L_2 R_2) L_2 = v_{m+1} e_m^T L_1^{-H} L_2 \\
(2.8d) \quad & A(W'_m L_2) - (W'_m L_2)(R_2 L_2 + \bar{\mu} I) = v_{m+1} e_m^T L_1^{-H} L_2 \\
(2.8e) \quad & A W''_m - W''_m (T''_m)^H = w_{m+1} e_m^T L_1^{-H} L_2.
\end{aligned}$$

One may again obtain the relation between the starting vectors as

$$(2.9) \quad (A^H - \bar{\mu} I) w'_1 = w''_1 (e_1^T R_2 e_1).$$

Obviously the appropriate transform of the right subspace to maintain orthogonality is  $V''_m = V'_m L_2^{-H} = V_m L_1 L_2^{-H}$ . Again, note that the upper triangular  $L_2^{-H}$  implies that the new  $v'_1$  is simply the scalar multiple of  $v'_1$  and no additional filtering occurs for  $v_1$  in this step. One can write the modified Lanczos relation as

$$(2.10) \quad A V''_m - V''_m T''_m = v_{m+1} e_m^T L_1 L_2^{-H}.$$

The above process can be repeated for  $p$  unwanted shifts. We Denote by  $L_1^p = L_{11} L_{12} \dots L_{1p}$  as the product of the lower triangular matrices generated due to  $p$  shifted-LR steps for the right Krylov space  $V_m$ , and by  $L_2^p = L_{21} L_{22} \dots L_{2p}$  as the product of the lower triangular matrices due to the  $p$  shifted-LR iterations for the left Krylov space. Then for a Krylov space size of  $m = k + p$  we have two modified Lanczos relations

$$(2.11a) \quad A V''_{k+p} - V''_{k+p} T''_{k+p} = v_{k+p+1} e_{k+p}^T L_1^p (L_2^p)^{-H}$$

$$(2.11b) \quad A^H W''_{k+p} - W''_{k+p} (T''_{k+p})^H = w_{k+p+1} e_{k+p}^T (L_1^p)^{-H} L_2^p.$$

We may take a closer look at the structure of the residual matrices on the right hand side of equation (2.11a).  $L_1^p$  is a product of  $p$  matrices that are lower triangular with just one subdiagonal.  $L_1^p$  then is lower triangular with  $p$  non-zero subdiagonals.  $(L_2^p)^{-H}$  is upper triangular and the product  $L_1^p (L_2^p)^{-H}$  therefore has  $p$  non-zero subdiagonals. Left multiplication by  $e_{k+p}^T$  therefore has the form

$$(2.11a) \quad e_{k+p}^T L_1^p (L_2^p)^{-H} = (\underbrace{0, 0, \dots, \theta_{k+p}}_k, \underbrace{b_1^T}_p)$$

where,  $\theta_{k+p} = e_{k+p}^T (L_1^p (L_2^p)^{-H}) e_k$ . Therefore matrix on the right hand side of (2.11b) has zeros in the first  $k - 1$  columns and the  $k^{th}$  column is simply  $\theta_{k+p} v_{k+p+1}$ . The remaining columns are non-zero in general. A very similar structure is obtained for the residual matrix in the right hand side of equation (2.11b) with zeros in the first  $k - 1$  columns and the  $k^{th}$  column being equal to  $\phi_{k+p} w_{k+p+1}$ , with  $\phi_{k+p} = e_{k+p}^T ((L_1^p)^{-H} L_2^p) e_k$

Partitioning the matrices such that

$$(2.12a) \quad V''_{k+p} = (V''_k, V''_p), \quad T''_{k+p} = \begin{pmatrix} T''_k & \delta_{k+1} e_k e_1^T \\ \beta_{k+1} e_1 e_k^T & T''_p \end{pmatrix},$$

$$(2.12b) \quad W''_{k+p} = (W''_k, W''_p), \quad (T''_{k+p})^H = \begin{pmatrix} (T''_k)^H & \bar{\beta}_{k+1} e_k e_1^T \\ \bar{\delta}_{k+1} e_1 e_k^T & (T''_p)^H \end{pmatrix},$$

with the length of the  $e_i$  vectors understood to be such that the resulting matrices are consistent. We can write the modified Lanczos relations of (2.11b) and (2.11a) as

$$\begin{aligned} A(V_k'', V_p'') &= (V_k'', V_p'') \begin{pmatrix} T_k'' & \delta_{k+1} e_k e_1^T \\ \beta_{k+1} e_1 e_k^T & T_p'' \end{pmatrix} + (\theta_{k+p} v_{k+p+1} e_k^T, M_v), \\ A^H(W_k'', W_p'') &= (W_k'', W_p'') \begin{pmatrix} (T_k'')^H & \bar{\beta}_{k+1} e_k e_1^T \\ \bar{\delta}_{k+1} e_1 e_k^T & (T_p'')^H \end{pmatrix} + (\phi_{k+p} w_{k+p+1} e_k^T, M_w), \end{aligned}$$

Finally, equating the individual columns on both sides and discarding columns  $k + 1, \dots, k + p$  we are left with the new Krylov spaces of order  $k$  and the Lanczos relations

$$\begin{aligned} (2.13a) \quad & AV_k'' - V_k'' T_k'' = v_{k+1}'' e_k^T, \\ (2.13b) \quad & A^H W_k'' - W_k'' (T_k'')^H = w_{k+1}'' e_k^T, \\ (2.13c) \quad & (W_k'')^H V_k = I. \end{aligned}$$

The new residual vectors are defined as

$$\begin{aligned} (2.14a) \quad & v_{k+1}'' = \beta_{k+1} V_p'' e_1 + \theta_{k+p} v_{k+p+1} \\ (2.14b) \quad & w_{k+1}'' = \bar{\delta}_{k+1} W_p'' e_1 + \phi_{k+p} w_{k+p+1} \end{aligned}$$

which may be normalized appropriately such that their inner product is unity. The Lanczos process may now be carried out again to generate the next  $p$  vectors of the right and left Krylov spaces and the cycle may be repeated till an adequately converged subspace has been obtained. The entire process may be put together in the form of an algorithm as shown in algorithm 2.1.

**Algorithm 2.1** Restarted Lanczos

---

**Input:**  $V_{k+p}, v_{k+p+1}, W_{k+p}, w_{k+p+1}, T_{k+p}$   
**Input:**  $\mu_1, \mu_2, \dots, \mu_p$  ▷ Unwanted shifts  
**Output:**  $V_k, v_{k+1}, W_k, w_{k+1}, T_k$   
**Require:**  $AV_{k+p} - V_{k+p}T_{k+p} = v_{k+p+1}e_{k+p}^T$  ▷ Right Lanczos relation  
**Require:**  $A^H W_{k+p} - W_{k+p}T_{k+p}^H = w_{k+p+1}e_{k+p}^T$  ▷ Left Lanczos relation  
**Require:**  $W_{k+p}^H V_{k+p} = I$ ;  $W_{k+p}^H v_{k+p+1} = 0$ ;  $V_{k+p}^H w_{k+p+1} = 0$ ; ▷ Bi-orthogonality  
1: **procedure** REFINERIGHTSUBSPACE  
2:    $L_r \leftarrow I_{k+p}$   
3:   **for**  $j \leftarrow 1$  to  $p$  **do**  
4:      $L, R \leftarrow \mathbf{LU}(T_{k+p} - \mu_j I)$  ▷ LU Decomposition  
5:      $T_{k+p} \leftarrow L^{-1}T_{k+p}L$  ▷ One step of LR with shift  $\mu_j$   
6:      $L_r \leftarrow LL_r$   
7:   **end for**  
8: **end procedure**  
9: **procedure** REFINELFTSUBSPACE  
10:    $L_l \leftarrow I_{k+p}$   
11:   **for**  $j \leftarrow 1$  to  $p$  **do**  
12:      $L, R \leftarrow \mathbf{LU}(T_{k+p}^H - \bar{\mu}_j I)$  ▷ LU Decomposition  
13:      $T_{k+p}^H \leftarrow L^{-1}T_{k+p}^H L$  ▷ One step of LR with shift  $\bar{\mu}_j$   
14:      $L_l \leftarrow LL_l$   
15:   **end for**  
16: **end procedure**  
17: **procedure** UPDATE  
18:    $\beta \leftarrow e_{k+1}^T T_{k+p} e_k$ ;    $\delta \leftarrow e_k^T T_{k+p} e_{k+1}$   
19:    $\theta = e_{k+p}^T (L_r (L_l)^{-H}) e_k$ ;    $\phi = e_{k+p}^T (L_r^{-H} L_l) e_k$   
20:    $v_{k+1} \leftarrow \beta V_{k+p} e_{k+1} + \theta v_{k+p+1}$  ▷ New right residual  
21:    $w_{k+1} \leftarrow \delta W_{k+p} e_{k+1} + \phi w_{k+p+1}$  ▷ New left residual  
22:    $V_k \leftarrow V_{k+p} (L_r L_l^{-H}) \begin{pmatrix} I_k \\ 0_p \end{pmatrix}$  ▷ New right Krylov space  
23:    $W_k \leftarrow W_{k+p} (L_r^{-H} L_l) \begin{pmatrix} I_k \\ 0_p \end{pmatrix}$  ▷ New left Krylov space  
24:    $T_k \leftarrow \begin{pmatrix} I_k & 0_p \end{pmatrix} T_{k+p} \begin{pmatrix} I_k \\ 0_p \end{pmatrix}$  ▷ New tridiagonal matrix  
25: **end procedure**  
26: **return**  $V_k, v_{k+1}, W_k, w_{k+1}, T_k$

---

182    **Acknowledgments.** We would like to acknowledge the assistance of volunteers  
183 in putting together this example manuscript and supplement.

## REFERENCES

- 185 [1] W. E. ARNOLDI, *The principle of minimized iterations in the solution of the matrix eigenvalue*  
186 *problem*, Quarterly of applied mathematics, 9 (1951), pp. 17–29.  
187 [2] M. H. GUTKNECHT, *Lanczos-type solvers for nonsymmetric linear systems of equations*, Acta  
188 numerica, 6 (1997), pp. 271–397.  
189 [3] C. LANCZOS, *An iteration method for the solution of the eigenvalue problem of linear differential*  
190 *and integral operators*, (1950).  
191 [4] R. B. LEHOUCQ, D. C. SORESENSEN, AND C. YANG, *ARPACK users' guide: solution of large-scale*

- 192 *eigenvalue problems with implicitly restarted Arnoldi methods*, SIAM, 1998.
- 193 [5] H. RUTISHAUSER, *Lectures on Numerical Mathematics*, Birkhauser, 1990.
- 194 [6] H. RUTISHAUSER, *Solution of eigenvalue problems with the lr-transformation*, National Bureau
- 195 of Standards, Applied Mathematics Series, 49 (1958), pp. 47–81.
- 196 [7] H. RUTISHAUSER AND H. SCHWARZ, *Their transformation method for symmetric matrices*, Nu-
- 197 merische Mathematik, 5 (1963), pp. 273–289.
- 198 [8] Y. SAAD, *The lanczos biorthogonalization algorithm and other oblique projection methods*
- 199 *for solving large unsymmetric systems*, SIAM Journal on Numerical Analysis, 19 (1982),
- 200 pp. 485–506.
- 201 [9] D. C. SORENSEN, *Implicit application of polynomial filters in ak-step arnoldi method*, Siam
- 202 journal on matrix analysis and applications, 13 (1992), pp. 357–385.