The Multishift QZ Algorithm with Aggressive Early Deflation*

Björn Adlerborn, Bo Kågström, and Daniel Kressner

Department of Computing Science, Umeå University, S-901 87 Umeå, Sweden.
{adler,bokg,kressner}cs.umu.se

Abstract. Recent improvements to the QZ algorithm for solving generalized eigenvalue problems are summarized. Among the major modifications are novel multishift QZ iterations based on chasing chains of tiny bulges and an extension of the so called aggressive early deflation strategy. The former modification aims to improve the execution time of the QZ algorithm on modern computing systems without changing the number of floating point operations (flops) significantly. In contrast, the new deflation strategy results in a considerable convergence acceleration and consequently in a reduction of both, flops and computing time.

1 Introduction

The QZ algorithm is the most widely used method for computing all n eigenvalues λ of a regular matrix pair (A, B) with $A, B \in \mathbb{R}^{n\times n}$, which satisfy

$$\det(A - \lambda B) = 0.$$

The QZ algorithm was developed by Moler and Stewart in [14] and relies on computing orthogonal matrices Q and Z such that $(S, T) = (Q^T AZ, Q^T BZ)$ is in real generalized Schur form, i.e., S is quasi-upper triangular with 1×1 and 2×2 blocks on the diagonal, while T is upper triangular. This equivalence transformation preserves the eigenvalues of (A, B), which then can be easily extracted from the block diagonals of S and T. The LAPACK [2] implementation of the QZ algorithm is mainly based on [14], with some improvements proposed in [10, 16, 18]. It consists of the following subroutines:

`DGGBAL` performs an optional preliminary balancing step [17] aiming to improve the accuracy of subsequently computed eigenvalues.

`DGGHRD` reduces a general matrix pair (A, B) to Hessenberg-triangular form, i.e., it computes in a finite number of steps orthogonal matrices Q_1 and Z_1 such that $H = Q_1^T AZ_1$ is upper Hessenberg while $T = Q_1^T BZ_1$ is upper triangular.

* This work was supported by the DFG Emmy Noether fellowship KR 2950/1-1 and by the *Swedish Research Council* under grant VR 621-2001-3284 and by the *Swedish Foundation for Strategic Research* under grant A3 02:128. This research was conducted using the resources of the High Performance Computing Center North (HPC2N).

`DHGEQZ` reduces (H, T) further, by applying single- and double-shift QZ iterations combined with deflations, to real generalized Schur form.

`DTGSEN` and `DTGEVC` post-process the output of `DHGEQZ` to compute selected eigenvectors and deflating subspaces [9] of (A, B).

Additionally, there are a number of support and driver routines for solving generalized eigenvalue problems. The focus of the improvements summarized in the following are the QZ iterations and deflations implemented in `DHGEQZ`, see also [1, 8]. Improvements to `DGGBAL`, `DGGHRD` and `DTGSEN`, which are also considered for inclusion in the next LAPACK release, can be found in [6, 11–13, 15].

2 Multishift QZ Iterations

Given a matrix pair (H, T) in Hessenberg-triangular form, with T nonsingular, an implicit double-shift QZ iteration starts with computing the vector

$$v = (HT^{-1} - \sigma_1 I)(HT^{-1} - \sigma_2 I), \tag{1}$$

where I denotes the $n \times n$ identity matrix and $\sigma_1, \sigma_2 \in \mathbb{C}$ are suitably chosen shifts. Next an orthogonal matrix Q (e.g., a Householder reflector [7]) is computed such that $Q^T v$ is mapped to a scalar multiple of the first unit vector e_1. This transformation is applied from the left to H and T:

$$H \leftarrow Q^T H, \qquad T \leftarrow Q^T T.$$

The Hessenberg-triangular structure of the updated matrix pair is destroyed in the first three rows and the rest of the implicit QZ iteration consists of reducing it back to Hessenberg-triangular form without touching the first row of H or T. Due to the special structure of (H, T), this process requires $O(n^2)$ flops and can be seen as chasing a pair of 3×3 bulges along the subdiagonals of H and T down to the bottom right corner, see [14, 18]. If the shifts are chosen to be the eigenvalues of the 2×2 lower bottom submatrix pair of (H, T) then typically the $(n-1, n-2)$ subdiagonal entry of H converges to zero. Such a subdiagonal entry is explicitly set to zero if it satisfies

$$|h_{j+1,j}| \leq \mathbf{u}(|h_{jj}| + |h_{j+1,j+1}|), \tag{2}$$

where $\mathbf{u}$ denotes the unit roundoff. This criterion not only ensures numerical backward stability but may also yield high relative accuracy in the eigenvalues for graded matrix pairs, see [8] for more details. Afterwards, the QZ iterations are continued on the deflated lower-dimensional generalized eigenvalue problems.

The described QZ iteration performs $O(n^2)$ flops while accessing $O(n^2)$ memory. This poor computation/communication ratio limits the effectiveness of the QZ algorithm for larger matrices. An idea which increases the ratio without affecting the convergence of QZ iterations, has been extrapolated in [8] from existing techniques for the QR algorithm, see, e.g., [4]. Instead of only one bulge pair corresponding to one double shift, a tightly coupled chain of bulge pairs corresponding to several double shifts is introduced and simultaneously chased. This allows the use of level 3 BLAS without a significant increase of flops in the overall QZ algorithm.

3 Aggressive Early Deflation

Another ingredient, which may drastically lower the number of iterations needed by the QZ algorithm, is aggressive early deflation introduced in [5] and extended in [1, 8]. Additionally to the classic deflation criterion (2), the following strategy is implemented. First, H and T are partitioned

$$(H, T) = \left(\begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ 0 & H_{32} & H_{33} \end{bmatrix}, \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ 0 & T_{22} & T_{23} \\ 0 & 0 & T_{33} \end{bmatrix} \right),$$

such that $H_{32} \in \mathbb{R}^{m \times 1}$ and $H_{33}, T_{33} \in \mathbb{R}^{m \times m}$ (typical choices of m are between 40 and 240). Then (H_{33}, T_{33}) is reduced to real generalized Schur form. By applying the corresponding left orthogonal transformation to H_{32}, a spike is introduced in H. If the trailing $d \leq m$ spike elements can be safely set to zero (see [5] for various criteria) then the bottom right $d \times d$ submatrix pair can be deflated. Otherwise, the Schur form of (H_{33}, T_{33}) is reordered to move other, untested eigenvalues to its bottom right corner, see [8] for more implementation details.

4 Further Improvements

Also described in [8] are more efficient algorithms for deflating infinite eigenvalues to guarantee the nonsingularity of T assumed in the beginning of Section 2. For a careful implementation, we will carefully reinvestigate the use of ad hoc shifts to avoid convergence failures in the QZ algorithm as well as the optimal use of the pipelined QZ iterations described in [6] to address medium-sized subproblems.

5 Selected Numerical Experiments

To give an indication of the performance of the new QZ algorithm, we summarize a few of the numerical experiments reported in [8], which were performed on a dual AMD Opteron 248 (2.2GHz) with 64 kB instruction and 64 kB data L1 Cache (2-way associative) and a 1024 kB unified L2 Cache (16-way associative). We used the Portland F90 6.0 compiler under Debian GNU/Linux 3.1 with Goto BLAS 0.94.

First, we tested a random 2000×2000 matrix pair reduced to Hessenberg-triangular form. LAPACK's `DHGEQZ` requires 270 seconds, while the multishift QZ algorithm described in Section 2 with 60 simultaneous shifts requires 180 seconds (on machines with smaller L2 cache this reduction was even more significant). Applying aggressive early deflation with $m = 200$ reduced the execution time further, to remarkable 28 seconds. This significant reduction of execution time carries over to other, practically more relevant examples. For instance, the multishift QZ algorithm with aggressive early deflation needed 12 seconds (instead of 147 seconds needed by `DHGEQZ`) when applied to the 2003×2003 matrix pair BCSSTK13/BCSSTM13 from the Matrix Market collection [3].

References

1. B. Adlerborn, K. Dackland, and B. Kågström. Parallel and blocked algorithms for reduction of a regular matrix pair to Hessenberg-triangular and generalized Schur forms. In J. Fagerholm et al., editor, *PARA 2002*, LNCS 2367, pp 319–328. Springer, 2002.
2. E. Anderson, Z. Bai, C. H. Bischof, S. Blackford, J. W. Demmel, J. J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. C. Sorensen. *LAPACK Users' Guide*. SIAM, Philadelphia, PA, third edition, 1999.
3. Z. Bai, D. Day, J. W. Demmel, and J. J. Dongarra. A test matrix collection for non-Hermitian eigenvalue problems (release 1.0). Tech. Report CS-97-355, Dept. of Computer Science, Univ. of Tennessee, Knoxville, TN, USA, 1997. Also available online from http://math.nist.gov/MatrixMarket.
4. K. Braman, R. Byers, and R. Mathias. The multishift QR algorithm. I. Maintaining well-focused shifts and level 3 performance. *SIAM J. Matrix Anal. Appl.*, 23(4):929–947, 2002.
5. K. Braman, R. Byers, and R. Mathias. The multishift QR algorithm. II. Aggressive early deflation. *SIAM J. Matrix Anal. Appl.*, 23(4):948–973, 2002.
6. K. Dackland and B. Kågström. Blocked algorithms and software for reduction of a regular matrix pair to generalized Schur form. *ACM Trans. Math. Software*, 25(4):425–454, 1999.
7. G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, MD, third edition, 1996.
8. B. Kågström and D. Kressner. Multishift variants of the QZ algorithm with aggressive early deflation. Report UMINF-05.11, Dept. of Computing Science, Umeå University, Ume, Sweden, 2005.
9. B. Kågström and P. Poromaa. Computing eigenspaces with specified eigenvalues of a regular matrix pair (A, B) and condition estimation: theory, algorithms and software. *Numer. Algorithms*, 12(3-4):369–407, 1996.
10. L. Kaufman. Some thoughts on the QZ algorithm for solving the generalized eigenvalue problem. *ACM Trans. Math. Software*, 3(1):65–75, 1977.
11. D. Kressner. *Numerical Methods and Software for General and Structured Eigenvalue Problems*. PhD thesis, TU Berlin, Inst. für Mathematik, Berlin, Germany, 2004.
12. D. Kressner. Block algorithms for reordering standard and generalized Schur forms, 2005. To appear in *ACM Trans. Math. Software.*
13. D. Lemonnier and P. Van Dooren. Balancing regular matrix pencils, 2004. To appear in *SIAM J. Matrix Anal. Appl.*
14. C. B. Moler and G. W. Stewart. An algorithm for generalized matrix eigenvalue problems. *SIAM J. Numer. Anal.*, 10:241–256, 1973.
15. G. Quintana-Ortí and E. S. Quintana-Ortí. An efficient algorithm for computing the Hessenberg-triangular form. Technical report ICC 2006-05-01, Universidad Jaime I, Castellón, Spain, 2006.
16. R. C. Ward. The combination shift QZ algorithm. *SIAM J. Numer. Anal.*, 12(6):835–853, 1975.
17. R. C. Ward. Balancing the generalized eigenvalue problem. *SIAM J. Sci. Statist. Comput.*, 2(2):141–152, 1981.
18. D. S. Watkins and L. Elsner. Theory of decomposition and bulge-chasing algorithms for the generalized eigenvalue problem. *SIAM J. Matrix Anal. Appl.*, 15:943–967, 1994.