
AN ENHANCED SHIFTED QR ALGORITHM FOR EFFICIENT EIGENVALUE COMPUTATION OF SQUARE NON-HERMITIAN MATRICES

 **Chahat Ahuja***
chahat22138@iiitd.ac.in

 **Partha Chowdhury†**
parthac@iiitd.ac.in

 **Subhashree Mohapatra‡**
subhashree@iiitd.ac.in

ABSTRACT

This work presents a novel approach to compute the eigenvalues of non-Hermitian matrices using an enhanced shifted QR algorithm. The existing QR algorithms fail to converge early in the case of non-hermitian matrices, and our approach shows significant improvement in convergence rate while maintaining accuracy for all test cases. In this work, though our prior focus will be to address the results for a class mid- large sized non-Hermitian matrices, our algorithm has also produced significant improvements in the case of comparatively larger matrices such as 50×50 non-Hermitian matrices.

Keywords Eigenvalue computation · non-Hermitian matrices · Shifted QR algorithm · Convergence acceleration · Matrix deflation.

1 Introduction

The eigenvalue problem involves finding a scalar λ for a corresponding vector $\mathbf{v}$ such that $A\mathbf{v} = \lambda\mathbf{v}$ holds good. But when matrices become extremely large or ill-conditioned then the eigen value problem faces difficulties to reach its desired results. QR factorization being one of the methodologies to solve an eigen value problem, involves finding a orthogonal matrix Q and an upper triangular matrix R for a given matrix $A([1],[2])$. For Hermitian matrices where $A = A^H$ (A^H is the Conjugate transpose of A), efficient QR algorithms exist because of their well-structured spectral properties, including real eigenvalues and orthogonal eigenvectors [3]. However, for non-Hermitian matrices, the problem becomes more challenging because of its complex eigenvalues and non-orthogonal eigenvectors. In this paper we have proposed an algorithm to address the drawbacks of QR algorithm in case of non-Hermitian matrices. Our algorithm performs significantly well in comparison with existing QR algorithms. While existing algorithms fail to achieve satisfactory results due to their slower convergence and less accuracy, the proposed algorithm addresses these challenges, making the convergence faster and maintaining the accuracy of eigenvalues for randomly generated non-Hermitian matrices.

2 Problem Statement

Given a $n \times n$ non-Hermitian matrix $A \in \mathbb{C}^{n \times n}$, the goal is to compute the eigenvalues $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \dots \in \mathbb{C}$ such that,

$$Av_i = \lambda_i v_i \quad \forall i = 1, 2, 3, 4, 5, \dots, n$$

where $v_i \in \mathbb{C}^n$ are the corresponding eigenvectors. Our goal is to develop a fast and efficient algorithm for computing the eigenvalues of non-Hermitian matrices while ensuring numerical stability. The proposed method aims to reduce computational complexity by accelerating convergence using advanced shift strategies and early deflation techniques.

*B.Tech CSAM, Department of Mathematics, IIIT Delhi

†Ph.D. Student, IIIT Delhi

‡Assistant Professor, Department of Mathematics, IIIT Delhi

By efficiently handling large and complex matrices, our approach minimizes iteration count and improves robustness against numerical errors. These enhancements make the algorithm well-suited for practical applications requiring high-speed and accurate eigenvalue computations.

3 Brief Literature Survey

The QR algorithm remains one of the most fundamental techniques for computing eigenvalues of matrices, and various refinements have been proposed over the years. The classical QR method for determining all eigenvalues of real square matrices was extensively studied by Francis and Kublanovskaya, whose work laid the foundation for modern implementations [4]. The convergence properties of the QR method, particularly in the context of shifted iterations for varied dimension matrices, have been analysed using structured matrices. Batterson[5] examined the behaviour of the shifted QR algorithm for 3×3 standard matrices, highlighting cases where convergence issues arise. The restarted QR algorithm proposed by Calvetti [6] demonstrated significant efficiency improvements for structured matrices by periodically restarting iterations to enhance numerical stability. Meanwhile, Braman et.al. [7] introduced aggressive early deflation techniques for multi-shift QR algorithms, significantly reducing computational cost by identifying and deflating converged eigenvalues early. The convergence characteristics of multi-shift QR algorithms have also been examined for symmetric tridiagonal matrices, revealing optimal shift selection strategies to improve performance [8]. Additionally, Ahu and Tisseur proposed a novel deflation criterion to enhance the robustness of multi-shift QR iterations[9]. Parlett [10] derived necessary and sufficient conditions for converging the QR algorithm on Hessenberg matrices, contributing to a deeper theoretical understanding of QR iterations. These studies form the basis for developing our Enhanced shifted QR algorithm, which aims to achieve faster convergence for non-Hermitian matrices while addressing the limitations of existing literature.

4 Background

The classical QR algorithm is an iterative method for computing the eigenvalues. It involves factorizing A into a product of an orthogonal matrix Q and an upper triangular matrix R . Then, the matrix A is updated as:

$$A_k = QR$$

$$A_{k+1} = RQ$$

The algorithm iterates this process until the matrix A_k converges to a diagonal matrix whose diagonal elements are the eigenvalues of A . QR Decomposition can be performed using three methods: Householder QR Factorization (uses Householder reflections for numerical stability), Gram-Schmidt QR Factorization (constructs an orthonormal basis, available in classical and modified forms), and Givens Rotation QR Factorization (uses Givens rotations, suitable for sparse matrices).

4.1 Shifted QR Decomposition

The shifted QR algorithm [11] enhances the classic QR method. It accelerates the algorithm's convergence by applying a shift to the matrix A_k at each iteration. The idea is to use a shift σ_k that improves the convergence of the eigenvalues.

$$A_k = Q_k R_k$$

$$A_{k+1} = R_k Q_k + \sigma_k I$$

where σ_k is a scalar shift and I is the identity matrix. Basically, shifts accelerate convergence to the dominant eigenvalues reducing the off-diagonal entries faster.

Now we will define the shifting strategy i.e the wilkinson shift that have been used in the algorithm.

Wilkinson Shift[11]: Let B be the lower-right 2×2 submatrix of $A^{(k)}$. The Wilkinson shift is the eigenvalue of B closer to a_m , defined as:

$$\mu = a_m - \text{sign}(\delta) b_{m-1}^2 \left(|\delta| + \sqrt{\delta^2 + b_{m-1}^2} \right),$$

where

$$\delta = \frac{a_{m-1} - a_m}{2}$$

and $\text{sign}(\delta)$ can be chosen arbitrarily if $\delta = 0$. This shift accelerates the QR algorithm to cubic convergence.

4.1.1 Deflation

Deflation [11] is a technique used to reduce the size of a matrix, thereby decreasing computational complexity and improving efficiency. It eliminates already converged eigenvalues, progressively reducing the matrix size m . Here, we impose deflation on the off-diagonal elements so they can be deflated correctly in a certain tolerance threshold, making the algorithm computationally efficient.

Suppose, after several iterations, a matrix becomes the following form:

$$\begin{bmatrix} \lambda_1 & * & * \\ 0 & \lambda_2 & * \\ 0 & 0 & \lambda_3 \end{bmatrix}$$

where the sub-diagonal elements are effectively zero ($\approx 10^{-12}$) [Ex: $A_{32} = 1.2 * 10^{-12}$]

At this point, the subdiagonal norm would be ≈ 0 which is less than 10^{-10} . The matrix is now nearly upper triangular, so we can deflate extracting λ_3 as an approximate eigenvalue. We also reduced the matrix size by removing the last row and column.

5 Methodology

The Enhanced Shifted QR Algorithm¹ computes the eigenvalues of a matrix by iteratively applying the QR decomposition with an implicit Wilkinson shift for accelerated convergence. Additionally, after every iteration, it incorporates deflation by eliminating converged eigenvalues, progressively reducing the matrix size.

5.1 Algorithm

Firstly, the algorithm incorporates the Wilkinson shift, a well-known shift strategy focusing on the eigenvalues of 2×2 sub-matrix to ensure faster convergence towards dominant eigenvalues. We also employed early deflation techniques to accelerate computation, reducing the adequate size of the matrix by removing converged eigenvalues from the matrix. We have also incorporated a balancing step into the algorithm to ensure the algorithm's numerical stability, minimizing sensitivity to rounding errors during iterations. These enhancements make our algorithm novel compared to the existing ones, overcoming the traditional limitations. The algorithm first aims to converge in fewer iterations than existing algorithms to its dominant eigenvalues for matrices with complex spectra. Secondly, the algorithm reduces computational complexity by efficiently deflating converged eigenvalues. Lastly, The algorithm maintains numerical stability throughout the iterative process (relative errors). Here we are providing the pseudocode and the details algorithm is given in the appendix 1.

5.2 Pseudocode of the Algorithm

Input:

- Matrix $A \in \mathbb{R}^{n \times n}$
- Maximum iterations $k_{\max}$
- Convergence tolerance ϵ
- Deflation tolerance δ

Output:

- Eigenvalues of A
- Iterations required for convergence
- Off-diagonal norms
- Deflation count

¹Python code of the algorithm available in <https://github.com/chahat6606/Enhanced-Shifted-QR-Algorithm>

5.3 Time and Space Complexity

In this Subsection we will discuss about the time and space complexity of the Enhanced Shifted QR algorithm in details.

The enhanced shifted QR algorithm is an iterative method for computing the eigenvalues of a given matrix A . In each iteration, the most computationally intensive operations involve orthogonal decomposition, which, when implemented using standard Gram-Schmidt or Householder reflections, requires $O(n^3)$ time for an $n \times n$ matrix. Since convergence usually requires $O(n)$ iterations in practice, the worst-case time complexity of the full algorithm is $O(n^4)$. However, empirical observations suggest that convergence often occurs in less than the worst-case bound, achieving at least $O(n^3)$ complexity in practice.

The algorithm operates in place on the matrix A , iteratively modifying it. The additional space required is for the matrices Q and R , both of which are $O(n^2)$ in size. Thus, the overall space complexity is $O(n^2)$. Since no additional extensive data is stored, the space complexity remains manageable, making the algorithm feasible for moderately large matrices.

5.4 Performance Analysis on matrices of arbitrary dimension

We carried out extensive experiments using our algorithm with square non-hermitian matrices of various dimensions and summarized the findings as follows. The performance criteria were assessed with regard to the convergence speed, accuracy, and computational efficiency, tested across diverse matrix sizes.

5.4.1 Case 1: For 3×3 Matrices

Firstly, we tested the Enhanced Shifted QR algorithm on various randomly generated 3×3 non-Hermitian matrices, comparing its performance with all other existing algorithms. While all test cases followed a general trend, we present a representative example in Figure 1, showcasing a randomly generated matrix and its results.

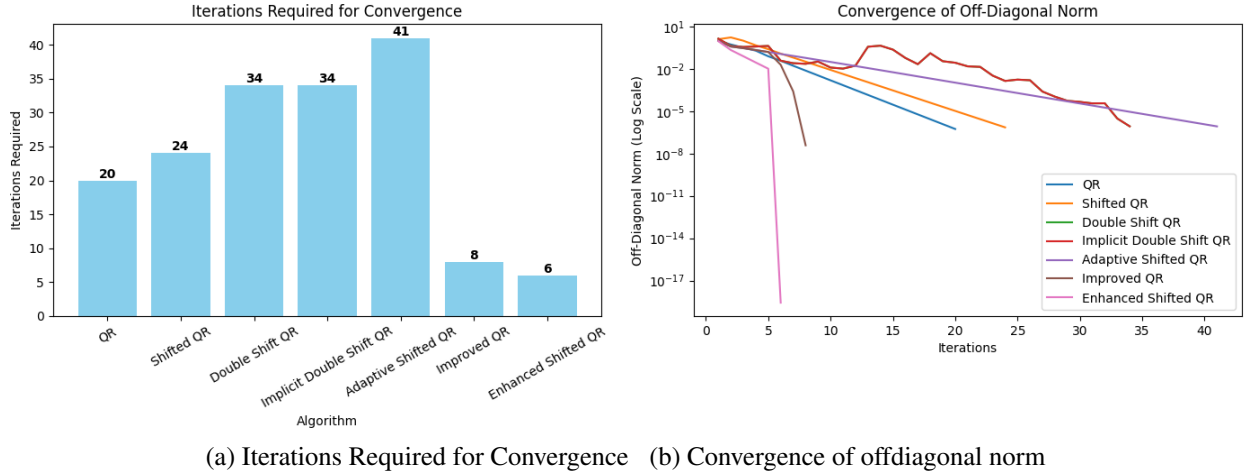


Figure 1: Iterations Required and Off-Diagonal Norm Convergence of a 3×3 randomly generated non-Hermitian matrix.

Figure 1(a) presents a comparative analysis of different QR-based methods in terms of convergence rate and off-diagonal norm reduction. It illustrates the number of iterations required for convergence, revealing that the Enhanced Shifted QR method exhibits the most rapid convergence, requiring only six iterations. In contrast to other algorithms which like Implicit Double Shift QR and Shifted QR methods that demand significantly higher iteration counts, 41 and 24, respectively, indicating slower convergence.

Figure 1(b) demonstrates the convergence behavior of the off-diagonal norm across various iterations on a logarithmic scale. The Enhanced Shifted QR method shows a steep decline, suggesting a more efficient reduction of the off-diagonal norm compared to other methods, meaning that the matrix converges to a diagonal form quickly.

Overall, our algorithm demonstrates the fastest performance for randomly generated 3×3 square non-Hermitian matrices. While other algorithms exist with nearly comparable efficiency in this low-dimensional setting, the difference

in computational speed becomes significantly more pronounced as the matrix dimension increases, making our approach particularly advantageous for high-dimensional eigenvalue computations.

5.4.2 Case 2: For 7×7 Matrices

We gradually increased the dimension of the matrices to test the performance of the algorithm. Further, we check the performance of our algorithm for randomly generated 7×7 matrices to observe emerging trends. The following is a representation of its performance in comparison with the existing ones.

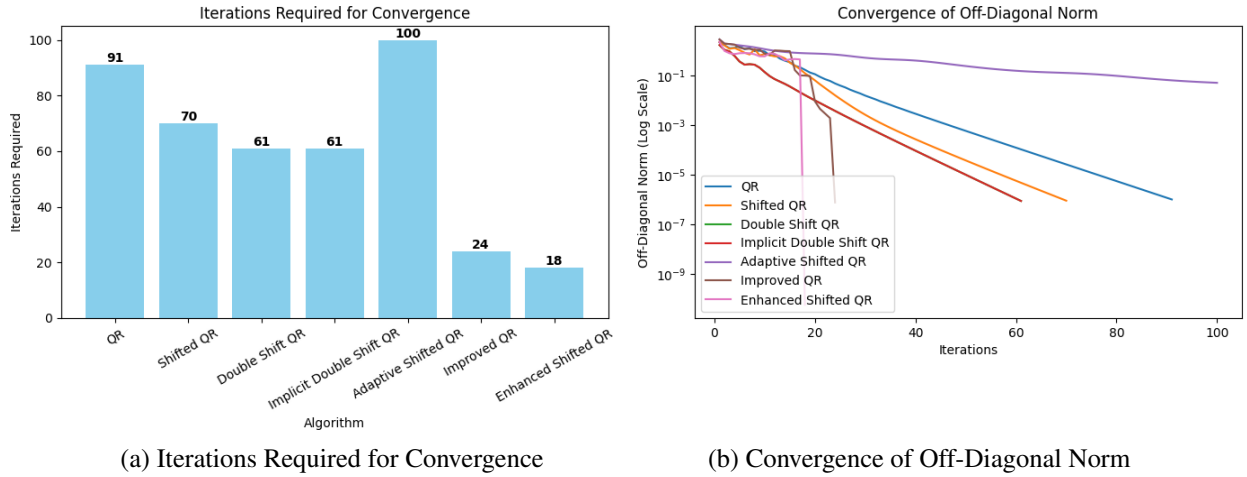


Figure 2: Iterations Required and Off-Diagonal Norm Convergence of a 7×7 randomly generated non-Hermitian matrix.

Figure 2(a) shows the number of iterations required for convergence, indicating that our proposed method (Enhanced Shifted QR) requires 18 iterations, which is significantly less in comparison to standard QR methods requiring around 50-100 iterations. The Improved-QR algorithm demonstrates comparable performance to our method; however, as the dimensionality increases, the performance gap becomes more pronounced.

Figure 2(b) further validates this observation by illustrating the convergence rate of the off-diagonal norm. Our approach demonstrates the steepest decline, confirming the fast rate of the matrix of transforming into a diagonal form.

Finally, Our algorithm shows substantial improvement over traditional approaches, and its efficiency gain scales favorably with increasing matrix size. This makes it a highly promising method for high-dimensional eigenvalue computations, where computational cost becomes a critical factor.

5.4.3 Case 1: For 50×50 Matrices

To obtain a more general understanding of its scalability and effectiveness, we directly test it on a significantly larger matrix of size 50×50 . This approach allows us to observe the overall trend rather than focusing on case-specific behavior. The results from this high-dimensional setting provide a clearer insight into the algorithm's performance improvements as matrix size increases.

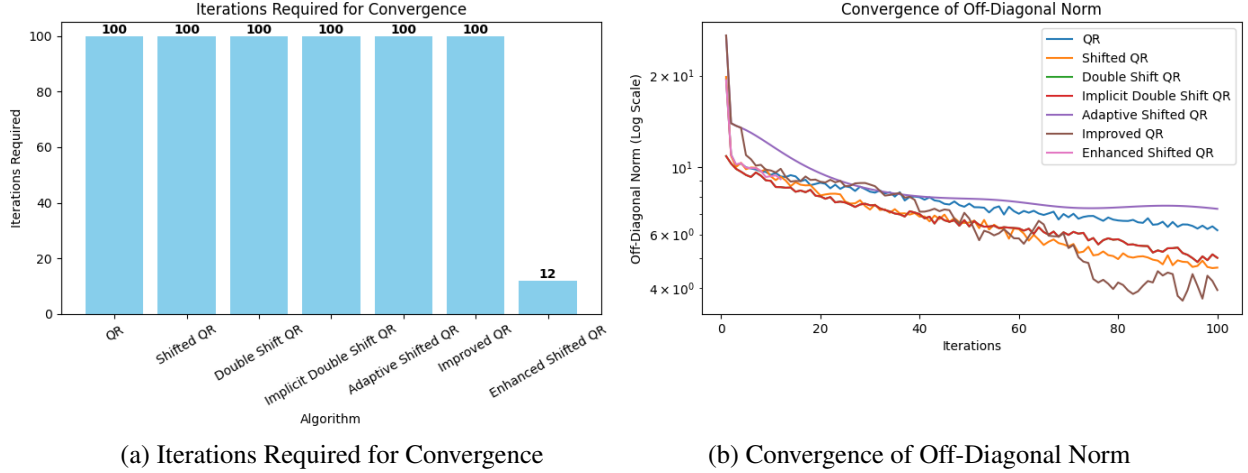


Figure 3: Iterations Required and Off-Diagonal Norm Convergence of a 50×50 randomly generated non-Hermitian matrix.

Figure 3(a) illustrates the performance of various QR-based algorithms for a larger matrix. The left plot clearly shows that while most traditional QR-based methods require the maximum iteration count to converge, our proposed Enhanced Shifted QR method converges in exceptionally fewer iterations, demonstrating its superior efficiency.

Figure 3(b) further reinforces this by depicting the rapid decay of the off-diagonal norm for our method, indicating faster convergence. This highlights that as matrix size increases, our approach not only outperforms others but also maintains a substantial efficiency advantage, making it a robust choice for high-dimensional eigenvalue computations.

Overall, the proposed algorithm consistently demonstrates a superior convergence rate, with the performance gap widening as the matrix dimension increases. Notably, it achieves the same level of accuracy as existing methods while significantly reducing the number of iterations required for convergence. This efficiency gain becomes increasingly pronounced in higher-dimensional settings, establishing our approach as a robust and scalable solution for eigenvalue computations.

6 Results and Findings

The results clearly highlight the efficiency of the Enhanced Shifted QR algorithm over traditional QR-based methods, particularly as the matrix size increases. For smaller matrices, such as 3×3 , our method achieves convergence in just **6 iterations**, while the Adaptive Shifted QR and Shifted QR methods require **41** and **24** iterations, respectively. This rapid convergence is also reflected in the off-diagonal norm reduction, where our approach consistently exhibits a steeper decline. As the matrix dimension increases to 7×7 , the efficiency gap widens further, with our method converging in **18 iterations**, compared to the **70-100** iterations observed for traditional QR methods. The Improved QR method performs relatively well, but our approach maintains a distinct advantage in both iteration count and norm reduction. When tested on large-scale 50×50 matrices, the Enhanced Shifted QR method continues to outperform others by requiring substantially fewer iterations, often well below the **100+** iterations observed in standard QR-based techniques. The rapid decay of the off-diagonal norm in the log-scale plots further validates the robustness of our approach, demonstrating that it efficiently drives the matrix towards a diagonal form with minimal computational effort. The superior performance of our algorithm becomes increasingly pronounced as matrix size grows, making it an ideal choice for high-dimensional eigenvalue problems. Overall, our method not only ensures faster convergence but also maintains numerical stability, offering a significant computational advantage over existing algorithms.

7 Conclusion

The Enhanced Shifted QR algorithm establishes itself as a superior alternative to traditional QR-based eigenvalue solvers by consistently demonstrating rapid convergence and robust numerical stability. The significant reduction in iteration count across different matrix dimensions confirms the efficiency of our method, particularly in high-dimensional cases where computational complexity is a major concern. Traditional QR and Shifted QR methods exhibit a slower rate of off-diagonal norm decay, often requiring a substantially higher number of iterations to achieve a comparable diagonal

form. The superior performance of our algorithm in both iteration reduction and norm minimization makes it an attractive choice for large-scale problems, where efficiency is paramount.

The future scope of this algorithm extends towards optimizing its implementation for parallel computing environments, allowing for even faster execution in high-performance computing systems. Additionally, incorporating adaptive heuristics for dynamic shift selection could further refine the convergence behavior, making the method even more robust for ill-conditioned matrices. Potential applications include quantum computing, spectral clustering, and numerical solutions to large-scale differential equations, where rapid eigenvalue decomposition is essential. Furthermore, extending the method to non-square and structured matrices could unlock new possibilities in data science and applied mathematics.

In conclusion, our Enhanced Shifted QR algorithm not only outperforms existing methods in terms of convergence speed and accuracy but also demonstrates scalability and robustness, making it a highly effective tool for modern eigenvalue computations. As computational demands continue to grow, our approach provides a promising direction for the development of even more efficient and powerful numerical techniques.

Appendices

A Appendix 1: Enhanced Shifted QR Algorithm for Eigenvalue Computation

```

1: function ENHANCEDSHIFTEDQR( $A, k_{\max}, \epsilon, \delta$ )
2:    $A_k \leftarrow A$ 
3:    $n \leftarrow \text{size}(A)$ 
4:   Initialize an empty list for eigenvalues
5:   for  $i = 1$  to  $k_{\max}$  do
6:     for  $j = n - 1$  down to 1 do
7:       if  $\|A_k[j, : j]\| < \delta$  then                                     ▷ Small sub-diagonal element
8:         Append  $A_k[j, j]$  to eigenvalues
9:         Remove the  $j$ th row and column from  $A_k$ 
10:        Update  $n \leftarrow n - 1$ 
11:        Break                                                         ▷ Restart iteration with reduced size
12:      end if
13:    end for
14:    if  $n == 1$  then
15:      Append the last remaining diagonal element  $A_k[0, 0]$  to eigenvalues
16:      Break
17:    end if
18:    Compute Wilkinson shift  $\mu$  from the last  $2 \times 2$  submatrix
19:    Compute QR decomposition:  $Q, R \leftarrow$  QR decomposition of  $(A_k - \mu I)$ 
20:    Update matrix:  $A_k \leftarrow RQ + \mu I$ 
21:    if  $\|A_k\|_{\text{off-diagonal}} < \epsilon$  then
22:      Append all diagonal elements of  $A_k$  to eigenvalues
23:      Break
24:    end if
25:  end for
26:  return Eigenvalues, iteration count, deflation count
27: end function

```

References

- [1] David S. Watkins. “QR-like algorithms for eigenvalue problems”. en. In: *Journal of Computational and Applied Mathematics* 123.1-2 (Nov. 2000), pp. 67–83. ISSN: 03770427. DOI: 10.1016/S0377-0427(00)00402-7. <https://linkinghub.elsevier.com/retrieve/pii/S0377042700004027>.
- [2] Dario A. Bini, Luca Gemignani, and Victor Y. Pan. “Fast and stable QR eigenvalue algorithms for generalized companion matrices and secular equations”. en. In: *Numerische Mathematik* 100.3 (May 2005), pp. 373–408. ISSN: 0029-599X, 0945-3245. DOI: 10.1007/s00211-005-0595-4. <http://link.springer.com/10.1007/s00211-005-0595-4>.

- [3] M. Nikpour, J.H. Manton, and I.M.Y. Mareels. “A novel approach to the shifted QR algorithm for the hermitian eigenvalue problem”. en. In: *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*. Vol. 3. Maui, Hawaii, USA: IEEE, 2003, pp. 2364–2367. ISBN: 978-0-7803-7924-4. DOI: 10.1109/CDC.2003.1272972. <http://ieeexplore.ieee.org/document/1272972/> .
- [4] Eyaya Fekadie Anley. “The QR Method for Determining All Eigenvalues of Real Square Matrices”. en. In: *Pure and Applied Mathematics Journal* 5.4 (2016), p. 113. ISSN: 2326-9790. DOI: 10.11648/j.pamj.20160504.15.
- [5] Steve Batterson. “Convergence of the shifted QR algorithm on 3 x 3 normal matrices”.
- [6] Daniela Calvetti, Sun-Mi Kim, and Lothar Reichel. “The restarted QR-algorithm for eigenvalue computation of structured matrices”. en. In: *Journal of Computational and Applied Mathematics* 149.2 (Dec. 2002), pp. 415–422. ISSN: 03770427. DOI: 10.1016/S0377-0427(02)00486-7. <https://linkinghub.elsevier.com/retrieve/pii/S0377042702004867>.
- [7] Karen Braman, Ralph Byers, and Roy Mathias. “The Multishift QR Algorithm. Part II: Aggressive Early Deflation”. en. In: *SIAM Journal on Matrix Analysis and Applications* 23.4 (Jan. 2002), pp. 948–973. ISSN: 0895-4798, 1095-7162. DOI: 10.1137/S0895479801384585. <http://epubs.siam.org/doi/10.1137/S0895479801384585> .
- [8] Qifang Su. “The convergence of multi-shift QR algorithm for symmetric matrices”. en. In: *Applied Mathematics and Computation* 222 (Oct. 2013), pp. 343–355. ISSN: 00963003. DOI: 10.1016/j.amc.2013.07.061. <https://linkinghub.elsevier.com/retrieve/pii/S0096300313008114> .
- [9] Mario Ahues and Francoise Tisseur. “A New Deflation Criterion for the QR Algorithm”. In: (Apr. 1997). <https://api.semanticscholar.org/CorpusID:61130>.
- [10] Beresford Parlett. “Necessary and sufficient conditions for convergence of the QR algorithm on Hessenberg matrix”. en. In: *Proceedings of the 1966 21st national conference on -*. Not Known: ACM Press, 1966, pp. 13–19. DOI: 10.1145/800256.810675. <http://portal.acm.org/citation.cfm?doid=800256.810675>.
- [11] Gorka Erana Robles. “Implementing the QR algorithm for efficiently computing matrix eigenvalues and eigenvectors”.