Inner-Outer Inverse Iteration for Eigenvector Computations

Wilfried N. Gansterer[*†‡]
Christoph W. Ueberhuber[*§]

March 6, 1998

Abstract

Inverse iteration is a standard technique for finding selected eigenvectors associated with eigenvalues which are known only approximately. At each iteration step the solution of a linear system of equations is required. In standard implementations these linear systems are solved using LU decomposition.

Instead of direct factorization *iterative* methods can be applied for solving the linear systems. This results in an *inner-outer iterative scheme* with several attractive properties: (i) Only a very low accuracy in the inner iterations is needed to ensure convergence of the outer iteration in a few steps. This is especially true in two cases: Either when the eigenvectors are to be computed with low or medium accuracy or when good starting vectors are available. Using an iterative method is, in this case, much more effective than using a direct method in the inner loop because the number of operations is reduced significantly. (ii) Variable shifts to improve the eigenvalue approximation can be handled very efficiently. Thus, rapidly converging Rayleigh quotient iterations can be applied advantageously. (iii) If several eigenvectors have to be computed, all corresponding inner iterative steps can be performed *simultaneously*. This property is essential for efficient implementations on modern computer architectures.

In this paper a detailed description of the inner-outer iterative approach is given. Convergence issues based on experiments on prototype implementations are dealt with. Results have been very promising: Using the stabilized BiCG method or the QMR method in the inner iteration instead of LU decomposition results in significant reductions in the computational effort. On average, the complexity is reduced by a factor of two, and in some cases even by a factor of four.

[*]Institute for Applied and Numerical Mathematics, Technical University of Vienna
[†]Supported by the Austrian Science Foundation (FWF).
[‡]e-mail: `ganst@aurora.tuwien.ac.at`
[§]e-mail: `christof@uranus.tuwien.ac.at`

1 Introduction

If approximations to the corresponding eigenvalues are available, the standard method for computing selected eigenvectors of a matrix $B \in \mathbb{R}^{n\times n}$ is inverse iteration (Golub, Van Loan [3]). This method requires the solution of a linear system in every iteration step. In the context of traditional eigensolvers, inverse iteration is performed on *reduced* matrices, i. e. on Hessenberg matrices (if B is unsymmetric) or on tridiagonal matrices (if B is symmetric). Since direct factorization is comparatively cheap for reduced matrices (with a complexity $O(n^2)$ for Hessenberg matrices or $O(n)$ for tridiagonal matrices), standard implementations of inverse iteration use LU factorization and repeated backsubstitutions. This procedure will be referred to as *standard inverse iteration* or as *direct inverse iteration*.

Blocked algorithms recently developed to solve symmetric eigenproblems (Gansterer, Kvasnicka [1], [2]) apply inverse iteration not to maximally reduced systems, but to banded systems with semibandwidth $b > 1$. This way the factorization involves $O(nb^2)$ flops. Increasing the value of b results, on the one hand, in better floating-point performance in the reduction step, but, on the other hand, makes factorization increasingly expensive. Eventually, inverse iteration becomes a bottleneck in the solution of the eigenproblem. This observation is what lead us to consider solving the linear systems *iteratively*, i. e. replacing the LU factorization and the backsubstitutions with a series of Level 2 (matrix-vector) operations.

The resulting two-level iteration, which will be referred to as *inner-outer inverse iteration*, is investigated in this paper. It is plain to see that using inner-outer inverse iteration instead of standard inverse iteration reduces the workload only if the total number of inner iterations required is kept within reasonable bounds. In the following, circumstances under which the new method is preferable to standard inverse iteration are investigated.

Problem Setting. Given a normal matrix $B \in \mathbb{R}^{n\times n}$ and an approximation $\hat{\lambda}$ to one of its eigenvalues, the corresponding eigenvector x must be computed.

Standard Inverse Iteration. One step of standard inverse iteration, which will be referred to in this paper as an *outer step*, basically consists of solving a linear system of the form

$$(B - \hat{\lambda}I)z^{(i)} = x^{(i-1)}. \tag{1}$$

By factorizing the system matrix $(B - \hat{\lambda}I)$ before of the main loop is executed this can be done by solving two triangular systems per step.

Synopsis. The rest of this paper is organized as follows: In Section 2 some background information is presented. In Section 3 convergence properties of standard and inner-outer inverse iteration are compared. Section 4 contains a discussion of how to implement inner-outer inverse iterations efficiently. In Section 5 the advantages of the new method over standard inverse iteration are summarized. In Section 6 some experimental results are given.

2 Inner-Outer Inverse Iteration

The method proposed in this paper is based on using *iterative* methods as an alternative to performing direct matrix factorizations for solving the linear systems (1) arising in the inverse iteration process. Within the (outer) iteration process an *inner iteration* for solving the linear system arising in every step of the outer iteration is performed.

Choice of the Inner Iterative Method. Some work has already been done in a similar direction: Ruhe and Wiberg [6] used the conjugate gradient method to solve the linear systems (1). However, they only dealt with the eigenvectors associated with the *largest* or with the *smallest* eigenvalue of B.

Ruhe and Wiberg showed that for good starting vectors convergence can be achieved after a small number of inner steps. They used the Rayleigh quotient to improve the eigenvalue approximation.

In this paper Ruhe and Wiberg's approach is developed further and generalized in several ways: When computing *arbitrary* eigenvectors, the system matrices involved are generally *indefinite*. Hence the standard CG method is not applicable. Other iterative methods have to be used in the inner iteration. In numerical experiments the best results were achieved using stabilized BiCG and QMR (Saad [7]).

3 Rate of Convergence

The key to success of inner-outer inverse iteration is to keep the number of inner iterations as low as possible without prohibitively slowing down the convergence rate of the outer iteration. Since an "exact solution" (except for rounding errors) obtained from a direct factorization is replaced by an iterative approximation (involving truncation error), one might suspect that the inner iteration cannot be terminated before very high accuracy is achieved.

Moreover, the systems to be solved are near-singular and hence badly *ill-conditioned*. This ill-conditioning slows down the convergence of iterative solvers, causing doubts about the benefits of the new approach.

However, surprisingly enough, it turns out that very high accuracy as achieved with direct factorizations is *not* required. In fact, the inner iteration can be terminated after reaching *very low* accuracy approximations. This early termination makes it possible for the overall flop count to be lower than in the standard method. In summary, the arithmetic complexity can be reduced despite the slow convergence of the inner iteration, which is a consequence of the ill-conditioning.

High Accuracy Requirements. When highly accurate approximations to the eigenvectors are sought, the potential for work reduction decreases: In this case highly accurate eigenvalue approximations are required since the error in the eigenvector approximations is at least as large as the error in the associated eigenvalue approximation (Ipsen [5]). In addition, the system matrices of (1) become more and more ill-conditioned the better the eigenvalue approximation is. This ill-conditioning severely affects the convergence rate of the inner iteration. Even for less stringent inner tolerances the number of inner iteration steps required is possibly so large that the overall complexity is higher than in standard inverse iteration. Sometimes inner-outer inverse iteration even fails to converge.

The computational effort can be reduced significantly when good starting vectors are available. If the starting vector $x^{(0)}$ is chosen from a subspace spanned by only a few of the eigenvectors of B, then inner-outer inverse iteration is competitive even for high accuracy requirements.

4 Implementation

On modern computer architectures, Level 3 (matrix-matrix) operations achieve much better floating-point performance than Level 2 (matrix-vector) operations. This is an implication of the fact that the locality of reference put into practice in Level 3 routines is better. Deviating from direct matrix factorizations, which benefit from Level 3 operations, and proceeding to iterative methods, which are based on Level 2 and Level 1 (vector-vector) operations, seems therefore not to be very effective.

However, if $k > 1$ eigenvectors are to be computed, it is possible to take advantage of the special structure of the system matrices of (1). All corresponding inner iteration steps can be combined into one single step which makes use of Level 3 operations.

5 Performance Assessment

Operation Count. The main benefit of inner-outer inverse iteration is the complexity reduction resulting from the replacement of a matrix factorization by several matrix-vector products. As long as the total number of inner iterations is not exceedingly large, the arithmetic complexity of inner-outer iteration is smaller than that of standard inverse iteration.

When comparing the performance of inner-outer inverse iteration with standard inverse iteration, it is necessary to distinguish three types of matrices:

1. B *sparse*: Ordering operations and fill-in make direct factorization much more expensive than matrix-vector products. Accordingly, inner-outer inverse iteration is much more efficient than standard inverse iteration.
2. B *banded* (semibandwidth b): For very small values of b (e. g. for a tridiagonal matrix B, $b = 1$), the new approach does not bring noticeable improvements since the effort needed to solve the linear systems is very small anyway.
3. B *full* ($b = n-1$): The potential complexity reduction is highest among all banded matrices. This makes inner-outer inverse iteration attractive in cases where an eigenvalue approximation is available, but a reduced form of B is not.

Rayleigh Quotient Iteration. For normal B and an approximate eigenvector x, the *Rayleigh quotient*

$$q(x) = \frac{x^\top Bx}{x^\top x} \tag{2}$$

approximates the corresponding eigenvalue (Golub, Van Loan [3]). Using (2) in the inverse iteration process leads to the *Rayleigh quotient iteration.*

In Rayleigh quotient iteration, the linear system to be solved *changes* in each outer iteration step. Performing a direct factorization per outer step requires a prohibitive amount of work. Therefore, standard inverse iteration normally cannot be generalized efficiently to Rayleigh quotient iterations. On the other hand, in inner-outer inverse iteration an update of the eigenvalue approximation incurs almost no overhead. It can be started with a rather rough eigenvalue approximation, which is improved in each outer iteration step. In practice, the eigenvalue approximation should only be updated at the end of steps in which the residual was reduced. As our experiments show, this reduction does not occur in every step.

Parallelism. When $k > 1$ eigenvectors have to be computed, standard inverse iteration can be implemented on parallel computers utilizing coarsely grained task parallelism: the k inverse iteration processes can be performed in parallel. More finely grained parallelism is not advisable since the backsubstitution process is inherently sequential and hard to parallelize efficiently.

In inner-outer inverse iteration, however, $k > 1$ leads to matrix-matrix products in the innermost loop, as illustrated in Section 4. Consequently, finely grained data parallelism can be utilized. Moreover, removal of the necessity to store k different LU factorizations reduces the storage requirements significantly.

6 Numerical Experiments

Experiments have been performed on MATLAB prototype implementations of inner-outer inverse iteration, and comparative experiments with standard inverse iteration have been performed using parameterized random test matrices.

The assessment of the two methods is based on flop counts, which gives some indication of the actual performance. Memory hierarchy effects have not been investigated yet.

In order to compare actual runtime behavior and floating-point performance, efficient Fortran implementations are required.

7 Conclusions

In this paper the concept of inner-outer inverse iteration for computing selected eigenvectors of normal matrices has been introduced. In contrast to standard implementations of inverse iteration the linear systems in each iteration step are solved iteratively.

In most cases only very low accuracy in the solution of the inner linear systems is needed to maintain fast convergence in the outer iteration process. Although convergence of the outer iteration is sometimes slower than in direct inverse iteration, the inner-outer approach leads to a considerable work reduction in many cases.

The desired accuracy for the eigenvector approximation is often achieved with less than half of the computational effort needed in standard inverse iteration. For low or medium accuracy requirements or when good starting vectors are available work can be reduced even more.

References

[1] W. Gansterer, D. Kvasnicka: *The Standard Eigenproblem in WIEN97 – Concepts.* Technical Report AURORA-5/3/97, Institute for Applied and Numerical Mathematics, Technical University Vienna, 1997.

[2] W. Gansterer, D. Kvasnicka: *The Standard Eigenproblem in WIEN97 – Experiments.* Technical Report AURORA-5/5/97, Institute for Applied and Numerical Mathematics, Technical University Vienna, 1997.

[3] G. H. Golub, C. F. Van Loan: *Matrix Computations*, 3rd ed. Johns Hopkins University Press, Baltimore London, 1996.

[4] E. Haunschmid, D. Kvasnicka: *Activities to Speed Up WIEN97 – Maximizing Cache Utilization*. Technical Report AURORA-5/2/97, Institute for Applied and Numerical Mathematics, Technical University of Vienna, 1997.

[5] I. C. F. Ipsen: *Computing an Eigenvector with Inverse Iteration*. SIAM Review 39 (1997), pp. 254 – 291.

[6] A. Ruhe, T. Wiberg: *The Method of Conjugate Gradients Used in Inverse Iteration*. BIT 12 (1972), pp. 543 – 554.

[7] Y. Saad: *Iterative Methods for Sparse Linear Systems*. PWS Publishing, Boston, 1996.

[8] C. W. Ueberhuber: *Numerical Computation*. Springer-Verlag, Berlin Heidelberg New York Tokyo, 1997.