INVERSE MULTIGRID CORRECTION FOR GENERALIZED EIGENVALUE COMPUTATIONS

ULRICH RÜDE* AND WERNER SCHMID†

Abstract. The inverse iteration is a classical method for the numerical computation of eigenvalues, which requires the solution of a linear system in each step. For large, sparse systems, a direct solution may be too expensive and iterative solvers must be used. However, the convergence of the outer eigenvalue iteration poses increasing demands on the accuracy of the inner iteration. Thus, high accuracy can be obtained only when the number of inner iteration steps is increased. We propose a variant of inverse iteration that is based on a correction approach whose accuracy requirements do not increase. This inverse correction method can then be used with only one cycle of a multigrid method for each outer iteration step. If combined with appropriate shifts, this method leads to a rapidly converging overall algorithm. Numerical examples for generalized eigenvalue problems indicate that inverse multigrid correction is an interesting alternative for large sparse eigenvalue problems.

Key words. Inverse iteration, Eigenvalue Problems, Iterative Methods, Multigrid.

AMS(MOS) subject classifications. 65N55, 65F15, 65N06, 65N25, 82D75

1. Introduction. In this note we study the problem of calculating an eigenvalue-eigenvector pair \((\rho, x)\) for a generalized eigenvalue problem of the form

\[(L - \rho F)x = 0 \quad \text{with} \quad x \neq 0.\]

We start our study with the classical inverse iteration (cf. [8]). This algorithm requires the inversion of \((L - \rho F)\) in each step of the iteration, where \(\hat{\rho}\) is an approximation for the eigenvalue \(\rho\). Since we are interested in the case when \(L\) and \(F\) arise from the discretization of partial differential equations, e.g. of the form (11) below, direct solvers are generally too expensive. The large and sparse matrices are most economically inverted by iterative methods. If employed within an outer inverse iteration process, such an iterative solver becomes an inner iteration. In our examples in Section 3, we will use the multigrid method as an approximate solver. Multigrid methods for elliptic systems are asymptotically optimal, that is, their rate of convergence does not deteriorate when the systems become larger. Thus they are particularly attractive for the solution of very large systems.

Combining inverse iteration with iterative inner solvers, two main difficulties arise:

- When used with shifts, the system to be inverted becomes (almost) singular.
- The accuracy which can be obtained in the inverse iteration is directly limited by the accuracy of the inversion process.

Thus the inverse iteration requires an increasing number of inner iterations, each of which tends to become more difficult to perform.

Therefore we suggest an alternative iteration process where the inverse iteration is reformulated in a correction style. Here the inner iteration is not applied to compute a new approximation of the eigenvector itself, but just a correction to the previous approximation. Thus, generally, if the previous approximation is already very accurate, the iterative solver has to provide just a small correction, and therefore it can be started with \(0\) as an initial guess. This new iteration will therefore need only a fixed accuracy in the iterative solver, and we will demonstrate that it is sufficient with this new iteration to use just a single cycle of the multigrid method in each step of the outer iteration.

2. The inverse correction method.

2.1. Inverse Iteration. The classical inverse iteration is given by

\[
\begin{align*}
  x_{i+1} &= (L - \hat{\rho}_i F)^{-1}F x_i; \\
  \hat{\rho}_{i+1} &= \frac{x_{i+1}^T}{\|x_{i+1}\|}; \\
  \rho_{i+1} &= x_{i+1}^T L x_{i+1} \geq x_{i+1}^T F x_{i+1}. 
\end{align*}
\]

Algorithm 1: Inverse Iteration

The key part of this algorithm is the step

\[x_i = (L - \hat{\rho}_i F)^{-1}F x_{i-1},\]

where a system with the shifted matrix \((L - \hat{\rho}_i F)\) must be solved. Here, the shift \(\hat{\rho}_i\) must be chosen as an approximation of the eigenvalue \(\rho\). One common strategy is to set \(\hat{\rho}_i = \rho_i\), the most recent approximation of eigenvalue.

For large problems and sparse matrices, we must perform the inversion in equation (2) by an iterative method. Unfortunately, the outer inverse eigenvector iteration depends critically on how accurately (2) is performed. In particular, the accuracy requirements increase, the better the approximation to the eigenvalue and eigenvector become. Additionally, the solution of (2) becomes more difficult, when the shift becomes more accurate and the shifted system gets closer to being singular.

2.2. Inverse Correction. We therefore suggest an alternative iteration based on a correction approach. Thus we first compute the residual \(d_i = (L - \rho_i F)x_i\) and then solve for the correction

\[v_i = (L - \hat{\rho}_i F)^{-1}d_i.\]

Finally, we update the new eigenvector approximation \(x_{i+1} = x_i - v_i\).

With an exact system solver for (3), this iteration is equivalent to the inverse iteration:

\[x_{i+1} = x_i - (L - \hat{\rho}_i F)^{-1}(L - \rho_i F)x_i = (\rho_i - \hat{\rho}_i)(L - \hat{\rho}_i F)^{-1}F x_i.\]

With an additional normalization step, as in the inverse iteration algorithm, (4) is equivalent to equation (2), provided \(\rho_i \neq \hat{\rho}_i\).
For a somewhat more detailed analysis, we now assume that \( e \) is an eigenvector for the
eigenvalue \( \rho^* \) of the generalized eigenvalue problem satisfying \( L = \rho^*F \). To \( e \),

\[
(L - \rho e) e = (1 - \rho_1 / \rho^*) L e \quad \text{and} \quad (L - \bar{\rho}_1 e)^{-1} L e = \frac{\rho^* e}{\rho^* - \bar{\rho}_1}
\]

so that, combined, iteration (4) applied to \( x_t = e \) leads to

\[
x_{t+1} = \left(1 - \frac{(\rho^* - \bar{\rho}_1)}{(\rho^* - \hat{\rho}_1)}\right) e = \frac{(\rho_1 - \hat{\rho}_1)}{(\rho^* - \hat{\rho}_1)} e.
\]

Clearly, the amplification factor for the eigenvector is \( \sigma = (\rho_1 - \hat{\rho}_1) / (\rho^* - \bar{\rho}_1) \), if \( \bar{\rho}_1 \neq \rho^* \). However, if we assume that both \( \bar{\rho}_1 \) and \( \rho_1 \) are approximations of \( \rho^* \) of the form

\[
\rho_1 = \rho^*(1 + \epsilon_t), \quad \bar{\rho}_1 = \rho^*(1 + \hat{\epsilon}_t),
\]

then the amplification factor becomes

\[
\sigma = \frac{\rho^*(\hat{\epsilon}_t - \epsilon_t)}{\rho^*(\hat{\epsilon}_t)} = \frac{\hat{\epsilon}_t - \epsilon_t}{\hat{\epsilon}_t}.
\]

We now observe that \( \lim_{t \to \infty} \epsilon_t = 0 \), independent of \( \hat{\epsilon}_t \neq 0 \). However,

\[
\lim_{t \to \infty} \sigma = 0.
\]

For the case of interest, that is, both \( \epsilon_t \) and \( \hat{\epsilon}_t \) small, the behavior of \( \sigma \) is nontrivial. If we consider, e.g., approaches of both \( \epsilon_t, \hat{\epsilon}_t \to 0 \) along lines \( \epsilon_t = \gamma \epsilon_t \) for constant \( \gamma \), then

\[
\lim_{t \to \infty} \sigma = \lim_{t \to \infty} \frac{\epsilon_t(\gamma - 1)}{\epsilon_t} = \frac{\gamma - 1}{\gamma}.
\]

Because we want \( \sigma = O(1) \), we must make \( \gamma \) as large as possible.

If we assume that both \( L \) and \( F \) are symmetric and positive definite, any vector \( z \)

be accomplished, if \( \rho_1 \) is chosen as the best available approximation of the eigenvalue
\( \rho^* = \eta_0 \), while \( \hat{\rho}_1 \) is slightly perturbed.

Computationally, the inverse correction iteration (4) combined with an iterative
inner solver has a significant advantage compared to inverse iteration. Note that in
the inverse correction algorithm, the iterative process is applied to compute a correction
to a previous approximation of the eigenvector. This correction converges to zero,
when the outer iteration converges, and when we succeed in making the corresponding
amplification factor \( \sigma = 1 \). This simplifies the task of the iterative solver and usually
requires less inner iterations. Only a relatively crude approximation of the correction
must be computed to improve the current iterate. Furthermore, the inverse correction
algorithm does not need a frequent rescaling of the eigenvector approximation.

\[
\text{Inverse Multigrid Correction:}
\]

\[
\text{Given: Initial vector } x_0, \text{ estimates } \hat{\rho}_1, \text{ initial eigenvalue approximation } \rho_0
\]

\[
\text{For } i = 0, 1, 2, \ldots \quad d_i = (L - \rho_i F)x_i; \\
\epsilon_i = MG[(L - \hat{\rho}_i F), d_i, 0]; \\
\eta_i = MG[(L - \rho_i F), d_i, 0]; \\
x_{i+1} = x_i - \epsilon_i;
\]

\[
\rho_{i+1} = < \eta_{i+1}, Lx_{i+1} > / \eta_{i+1}, Fx_{i+1} >;
\]

Algorithm 2: Inverse Multigrid Correction

On the other hand, it is no disadvantage for the inverse correction that the outer
iteration converges only linearly, because the speed of convergence is naturally limited
by the precision of the inner iteration.

Using only one multigrid cycle \( MG[(L - \hat{\rho}_i F), d_i, 0] \) with initial guess \( \theta \) to approximately compute \((L - \hat{\rho}_i F)^{-1} d_i \), we arrive at the inverse multigrid correction
algorithm (Algorithm 2). The estimate \( \hat{\rho}_i \) has to be chosen carefully to guarantee the convergence
of multigrid. For \( \hat{\rho}_i \), too close to \( \rho_i \), \( \sigma \) tends to zero, as in (8). Algorithm 2 seems to provide an efficient numerical technique to solve the generalized eigenvalue problem.

**2.3. Related Work.** The multigrid method applied to inverse iteration has been analyzed by Bank [3] in the case of finite element discretizations. The analysis in this paper assumes a fixed shift smaller than the smallest eigenvalue in order to guarantee convergence for the multigrid method. An alternative multigrid approach for the eigenvalue problem has been suggested by Hackbusch [9, Chapter 12]. Our algorithm is similar, however, the algorithm there applies an additional projection step on each level of the multigrid cycle to restrict the corrections to a subspace which is approximately orthogonal to the eigenvector.

The linearized method analyzed in Cai, Mandel, McCormick [5] for generalized
eigenvalue problems with symmetric positive definite operators is also similar to our
inverse correction multigrid method, but it also employs a projection step on the coarsest
level. The main focus of this paper is on a more general algorithm based on a multi-
level minimization of the Rayleigh quotient, where the eigenvalue estimate is iteratively
improved within the multigrid cycle.
In the paper [4] by Brand and Petrowa, the idea of preconditioning the power method is discussed. Though the emphasis of this paper is on preconditioners based on incomplete factorization, it provides an interesting framework, where our algorithm can be interpreted as using a multigrid approximation to the inverse of \((L - \beta F)\) as a preconditioner within the power method. In this paper, the relation of this method to the so-called generalized (preconditioned) Davidson method is discussed, see also Morgan and Scott [11].

Finally, in a recent paper by Zaslavsky [15], an algorithm identical to our inverse correction multigrid algorithm is used. Since the focus of this paper is on the adaptation of the algebraic multigrid method to the special requirements of the eigenvector iteration, it does not provide any analysis of the inverse correction method. As in our paper, the application in [15] are the two-group neutron diffusion equations. Here, we extend the results there to mixed finite element discretizations in three space dimensions and provide a systematic comparison of the inverse correction to inverse iteration.

3. Numerical Examples. Our examples are motivated by reactor physics simulations, where the primary problem is the solution of the multi-group diffusion equations

\[
-\nabla \cdot (D \nabla \phi) + (\Sigma_{se} + \sum_{p=1}^{G} \Sigma_{cp}) \phi = 0
\]

Diffusion

\[
\sum_{p=1}^{G} \Sigma_{cp} \phi_p + \frac{1}{\lambda} \sum_{p=1}^{G} ((1 - \beta) \lambda_{cp} + \beta \lambda_{fp} \Sigma_{fp}) \phi_p.
\]

Absorption

Scattering

1/s. Fission

This can be interpreted as a differential eigenvalue problem that can be written in operator form (cf. Wachpress [14]):

\[
\begin{bmatrix}
D + A - \frac{1}{\lambda} F
\end{bmatrix} \phi = 0, \quad \text{i.e.,} \quad L\phi = \frac{1}{\lambda} F\phi.
\]

The following examples refer to this type of model problem with the number of groups \(G = 2\), so that (11) is a system of two elliptic partial differential equations. The system is discretized by either finite differences (for a simple one-dimensional model problem) or by mixed finite elements in a realistic three-dimensional model situation.

Unfortunately, problem (11) does not satisfy all the formal requirements of the analysis in Section 2, since we have to deal with operators which are not symmetric positive definite. However, since only the lower order terms are non-symmetric and the main part of the differential operators is symmetric, the numerical experience reported below is in good conformance with is in good conformance with the above analysis.

3.1. Model problem in one space dimension. In our first example we consider a one-dimensional model problem of the type (11) with constant coefficients. Though the solution of this one-dimensional system does not yet require the use of iterative methods, the experiments provide useful insight in the behavior of the algorithm.

The differential equation is discretized by central second order differences on a grid with 129 knots. For the sake of simplicity, only Dirichlet boundary conditions are used. The dominating eigenvalues of the discrete system are approximately \(0.9001, 0.9287, 0.9500, \ldots\). These values have been computed by a direct method. Note that the eigenvalues are poorly separated, so that a simple (inverse) power method would converge only slowly. We therefore use shifted methods in two variants. Both are based on a multigrid solver using V-cycles with 6 levels and two pre- and one post-smoothing steps on each level. The smoother is a simple Block-Jacobi method relaxing the two unknowns of the same node simultaneously. The plots in Fig. 1 show the number of multigrid cycles versus the accuracy of the eigenvalue approximation. Clearly, a fixed number of cycles with the shifted inverse iteration produces only limited accuracy. Only by increasing the number of inner iterations highly accurate results can be obtained.

The first algorithm tested is inverse iteration with shift where the above multigrid algorithm is used as a solver. In Fig. 1, we present results when 1, 2, and 3 cycles are used for the approximate inversion in each loop of the outer iteration. Here, clearly, the improvement in the accuracy of the inverse iteration is limited by the number of multigrid iterations in the inner loop. From Fig. 1 it is obvious that lower accuracy is required for the first few iterations. The first two iterations yield essentially the same accuracy for the eigenvalue, independent of the number of multigrid cycles employed.

Since it is obviously necessary to increase the accuracy of the iterative solver in later iterations, we present another experiment with a variable number of inner iterations. Note that a high number of fixed iterations would neither be economic, since then the iterative solver would provide too much accuracy in early iterations. The algorithm evaluated in Fig. 1 therefore uses 1 cycles of multigrid in the ith outer loop. Note that n outer iterations now cost \(O(n^3)\) multigrid cycles. In practice, this algorithm could be
further improved by adaptive stopping criteria for the inner iteration loop. This has in fact been used in our multidimensional experiments.

The inverse iteration based algorithms above are finally compared with the new inverse multigrid correction algorithm. This algorithm employs a single multigrid cycle in each inner iteration. In all algorithms, the shift \( \tilde{\lambda} \) is not the current eigenvalue approximation, since using exact shifts degrades multigrid convergence. Therefore, we have chosen to use \( \tilde{\lambda} = 0.99\tilde{\lambda}_\text{init} \) as the shift in the asymptotic phase of the algorithm, where \( \tilde{\lambda}_\text{init} \) is the best current eigenvalue approximation. In the first four iterations of the inverse correction algorithm, the shift \( \tilde{\lambda} = 0.95\tilde{\lambda}_\text{init} \) is used to avoid oscillations in the convergence behavior. This is reflected as a slower convergence in the starting phase for the algorithm. Despite this slower startup, the inverse correction algorithm is clearly superior to the inverse iteration in terms of accuracy and speed of convergence. After the fourth iteration, it converges to the eigenvalue with a rate consistently around 0.04 all the way down to roundoff error tolerance.

### 3.2. Problems in three space dimensions

The second and third example are simulations in three space dimensions using realistic data from standard reactor physics benchmark problems. The data for example 2 is taken from the thesis of Vaudoscal [13], example three is a simplified version of the three-dimensional IAEA benchmark.

For both examples 2 and 3, the discretization is based on mixed hybrid finite elements which are equivalent to so-called non-conforming nodal finite elements, a commonly used discretization in reactor criticality calculations. For more details see Hennart and Del Valle [10], Arbogast and Chen [2]), and Schmid [12]. The examples have 78176 and 87584 unknowns, respectively, on the finest grid. Multigrid is applied with three levels, using W-cycles and two block-ILU-pre- and post-smoothing steps. Again, we relax the two unknowns of the same node simultaneously.

![Convergence History for 3-D Problem](image)

**Fig. 2. Convergence History for 3-D Neutron Diffusion Example 2.**

The multigrid convergence for these examples is approximately 0.5 per cycle (independent of the number of levels) and thus significantly worse than for the one-dimensional model problem, in particular when one considers that a total of four ILU method iterations is a very strong smoother. However, in the literature, it has been observed that the multigrid rates for non-conforming elements are significantly worse than those usually observed for conforming discretizations, see e.g. Chen and Kwak [6]. The number of iterations required in industrial codes is often also significantly worse than ours as can be seen e.g. from Finneman et al. [7]. Compared to this, our results are already quite acceptable.

However, our method clearly leaves room for additional improvement. Our multigrid method is a straightforward implementation within the mixed hybrid finite element context. Since in coefficients in examples 2 and 3 have discontinuities, we expect that for optimal multigrid performance, operator dependent grid transfer operators and appropriate constructions for the coarse grid operator are required. These approaches are well known in the context of simpler discretizations, see e.g. Alcouffe, Brandt, Dendy, and Painter [1], or the algebraic multigrid approach of Zelovanský [15]. Unfortunately, the extension of these techniques to the mixed hybrid finite element discretization is not straightforward.

For examples 2 and 3, the exact eigenvalue is not available so that we plot the scaled Euclidean residual norm \( \|A\|_R \) versus the number of multigrid cycles in Figs. 2 and 3, respectively. As for the one-dimensional case, we present results for inverse iteration, when a fixed number of 1, 2, and 3 cycles are used for the approximate inversion in each loop of the outer iteration. Inverse iteration with variable cycle numbers was based on the condition that the residual norm of the inner iteration had to be less than a 1/100 of the outer residual norm.

This is compared to inverse correction with a single multigrid step in each outer iteration. Here, in all calculations a fixed shift of \( \tilde{\lambda} = 0.95 \) was used since a new shift value requires the calculation of a new ILU-decomposition for \((L - \tilde{\lambda}F)^{-1}\).

The results for both three-dimensional examples in Figs. 2 and 3 confirm the
general picture observed for the model case in Fig. 1. In all cases, the inverse correction algorithm seems to produce a more accurate result in fewer overall cycles.

4. Conclusion. In this note, we have proposed a correction variant of inverse iteration and showed, that — provided a proper shift is used — it has computational advantages over inverse iteration, when inner iterations must be used. Inverse correction can thus be an interesting alternative to classical inverse iteration. Further work will deal with a more thorough analysis of the convergence behavior of multigrid in this context, especially it will be interesting to reveal its relationship with other algorithms, as indicated in section 2.3. It will also be interesting to see which of the many variants can be used to further improve the robustness and speed of the present algorithm. For real life problems in three dimensions, we expect that the performance of multigrid itself can be improved significantly by tuning the multigrid algorithm.

REFERENCES