BIT Numerical Mathematics
2003, Vol. 43, No. 1, pp. 001–018

0006-3835/03/4301-0001 $16.00

CONVERGENCE RATES FOR INEXACT INVERSE ITERATION WITH APPLICATION TO PRECONDITIONED ITERATIVE SOLVES *

M. A. FREITAG[1] and A. SPENCE[1] †

February 8, 2005

[1] *Department of Mathematical Sciences, University of Bath, Claverton Down, BA2 7AY, United Kingdom. email:* {*m.freitag, a.spence*}*@maths.bath.ac.uk*

Abstract.

In this paper we study inexact inverse iteration for solving the generalised eigenvalue problem $\mathbf{Ax} = \lambda\mathbf{Mx}$. We show that inverse iteration is a Newton method and inexact inverse iteration is a modified Newton method. Using the convergence theory of modified Newton's method we obtain convergence rates for various versions of inexact inverse iteration for the calculation of an algebraically simple eigenvalue. In particular, if the inexact solves are carried out with a tolerance chosen proportional to the eigenvalue residual then quadratic convergence is achieved. We also analyse preconditioned iterative solves using an ILU preconditioner. We show how modifying the right hand side in inverse iteration still provides a convergent method, but the rate of convergence may only be linear. Finally we introduce a tuned ILU preconditioner which has advantages both for the standard form of inverse iteration and for the version with a modified right hand side. Numerical examples are given to illustrate the theoretical results.

AMS subject classification (2000): Primary 65F15, Secondary 65F10.

Key words: Inverse iteration, Newton's method, preconditioned iterative methods.

1 Introduction.

This paper is about inexact inverse iteration applied to the generalised nonsymmetric eigenvalue problem $\mathbf{Ax} = \lambda\mathbf{Mx}$, where $\mathbf{A}$ and $\mathbf{M}$ are large, sparse (possibly complex) matrices.

Many methods for the iterative calculation of eigenvalues of $\mathbf{Ax} = \lambda\mathbf{Mx}$ involve the repeated solution of the shifted linear system

$$(\mathbf{A} - \sigma\mathbf{M})\mathbf{y} = \mathbf{Mx}, \tag{1.1}$$

for some $\sigma \in \mathbb{C}^n$, with the simplest iterative method being inverse iteration. We assume that it is unfeasible to factorise the matrices, due to excessive memory requirements, and so to solve the large, sparse, linear shifted systems iterative

*

†

methods are used. These require only matrix-vector products and can therefore exploit the sparse structure of the matrices.

Previous studies on inexact inverse iteration applied to the standard symmetric eigenvalue problem include the very early paper [12] and more recent [14], [13] and [1]. In these papers Rayleigh quotient shifts were used in the inverse iteration algorithm and various aspects are discussed, for example, the convergence theory and the implementation and performance of the iterative solver. For preconditioned iterative solves, [13] introduced the idea of modifying the right hand side to improve the performance of the iterative solver. For inexact inverse iteration applied to the nonsymmetric eigenvalue problem we refer to [10] and [6] for a fixed shift, and [3] for a variable shift strategy. In many of these papers the analysis is based on eigenvector expansions and convergence is determined by looking at a (generalised) tangent of the error in the desired eigendirection. Other related papers on the topic of inexact eigenvalue solvers include [9] and [11].

In this paper a completely different and novel approach to the analysis for variable shifts is used which provides a much simpler analysis, and also suggests a way of analysing preconditioned iterative solves when the right hand sided is modified as in [13]. We show that inexact inverse iteration is a modified Newton's method and hence obtain a convergence analysis for inexact inverse iteration applied to the calculation of an algebraically simple eigenvalue. In addition, the approach here suggests a "tuning" strategy for the preconditioner that works well in the example considered here.

The plan of the paper is as follows. Section 2 contains a review of some known results about inverse iteration. The main theory of the paper is in Section 3 where the convergence results for inexact inverse iteration are obtained. In Section 4 we discuss the convergence theory for a version of inexact inverse iteration where the right-hand side is modified to improve the performance of a preconditioned iterative solver. This analysis is illustrated for an ILU preconditioner. In addition, we introduce a tuned ILU preconditioner, which turns out to have a significantly improved performance over the usual ILU preconditioner. Section 5 provides a summary of the main results in the paper.

Throughout this paper we use $\|\mathbf{z}\|=\|\mathbf{z}\|_\infty$. All computations are carried out with MATLAB, version 6.1 (R 12.1).

2 Inverse Iteration.

In this section we formulate inverse iteration as a Newton method (see, for example, [15]) and use the standard Newton theory to obtain a convergence result for inverse iteration.

Let $\mathbf{A}$ and $\mathbf{M}$ be real or complex $n \times n$ matrices, and consider the generalised eigenvalue problem

$$\mathbf{A}\mathbf{x} = \lambda \mathbf{M}\mathbf{x}, \quad \lambda \in \mathbb{C},\, \mathbf{x} \in \mathbb{C}^n. \tag{2.1}$$

Assume that $(\mathbf{x}_1, \lambda_1)$ is an algebraically simple eigenpair of (2.1) with $\mathbf{u}_1^H$ the corresponding left eigenvector, so that,

$$\mathbf{u}_1^H \mathbf{M}\mathbf{x}_1 \neq 0. \tag{2.2}$$

Also, for some non-zero constant vector $\mathbf{c} \in \mathbb{C}^n$ assume the normalisation condition

$$\mathbf{c}^H \mathbf{x}_1 = 1. \tag{2.3}$$

One version of inverse iteration is given by Algorithm 2.1.

ALGORITHM 2.1 (INVERSE ITERATION). *Given* $\lambda^{(0)}$ *and* $\mathbf{x}^{(0)}$ *with* $\mathbf{c}^H\mathbf{x}^{(0)} = 1$. *For* $i = 0, 1, 2, \dots$

(1) Solve $(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{M}\,\mathbf{x}^{(i)}$,

(2) Set $\Delta\lambda^{(i)} = \dfrac{1}{\mathbf{c}^H\mathbf{y}^{(i)}}$; $\quad \lambda^{(i+1)} = \lambda^{(i)} + \Delta\lambda^{(i)}$,

(3) Update $\mathbf{x}^{(i+1)} = \Delta\lambda^{(i)}\mathbf{y}^{(i)}$,

(4) Evaluate $\mathbf{r}^{(i+1)} = (\mathbf{A} - \lambda^{(i+1)}\mathbf{M})\mathbf{x}^{(i+1)}$,

(5) Test for convergence

Note that from steps (2) and (3) of Algorithm 2.1, $\mathbf{c}^H\mathbf{x}^{(i+1)} = 1$ and hence $\mathbf{c}^H\Delta\mathbf{x}^{(i)} = 0$, where $\Delta\mathbf{x}^{(i)} = \mathbf{x}^{(i+1)} - \mathbf{x}^{(i)}$.

Now, let us introduce the nonlinear system $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ where, for $\mathbf{z} := (\mathbf{x}, \lambda)^T$,

$$\mathbf{F}(\mathbf{z}) = \begin{bmatrix} (\mathbf{A} - \lambda\mathbf{M})\mathbf{x} \\ \mathbf{c}^H\mathbf{x} - 1 \end{bmatrix}. \tag{2.4}$$

Then the steps in Algorithm 2.1 may be rewritten in the following block matrix form

$$\begin{bmatrix} (\mathbf{A} - \lambda^{(i)}\mathbf{M}) & -\mathbf{M}\mathbf{x}^{(i)} \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}^{(i)} \\ \Delta\lambda^{(i)} \end{bmatrix} = \begin{bmatrix} -(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{x}^{(i)} \\ 0 \end{bmatrix}, \tag{2.5}$$

with

$$\begin{bmatrix} \mathbf{x}^{(i+1)} \\ \lambda^{(i+1)} \end{bmatrix} = \begin{bmatrix} \mathbf{x}^{(i)} + \Delta\mathbf{x}^{(i)} \\ \lambda^{(i)} + \Delta\lambda^{(i)} \end{bmatrix}. \tag{2.6}$$

Equations (2.5) and (2.6) are merely Newton's method applied to (2.4), namely,

$$\mathbf{J}(\mathbf{z}^{(i)})\Delta\mathbf{z}^{(i)} = -\mathbf{F}(\mathbf{z}^{(i)}), \quad \mathbf{z}^{(i+1)} = \mathbf{z}^{(i)} + \Delta\mathbf{z}^{(i)}, \tag{2.7}$$

where $\mathbf{J}(\mathbf{z}^{(i)})$ denotes the Jacobian

$$\mathbf{J}(\mathbf{z}^{(i)}) = \begin{bmatrix} (\mathbf{A} - \lambda^{(i)}\mathbf{M}) & -\mathbf{M}\mathbf{x}^{(i)} \\ \mathbf{c}^H & 0 \end{bmatrix}. \tag{2.8}$$

LEMMA 2.1. *If* $\mathbf{z}_1 = (\mathbf{x}_1, \lambda_1)^T$ *is an algebraically simple eigenpair of (2.1) then under (2.3),* $\mathbf{J}(\mathbf{z}_1)$ *is nonsingular.*

PROOF. Lemma 2.8 in [8] (see also Lemma 2 in [7]) shows that if $\mathrm{Rank}(\mathbf{A} - \lambda_1\mathbf{M}) = n - 1$, and if (2.2), (2.3) hold then $\mathbf{J}(\mathbf{z}_1)$ is nonsingular. □

Standard convergence theory for Newton's method (see, for example, [5]) applied to (2.7) provides the following well-known convergence result.

COROLLARY 2.2. *If* $\mathbf{z}_1 = (\mathbf{x}_1, \lambda_1)^T$ *is an algebraically simple eigenpair of (2.1) and if (2.3) holds, then Algorithm 2.1 converges quadratically for a close enough starting guess.*
This quadratic rate of convergence is observed in practice (see, for example, the numerical results given by the solid line in Figure 3.1).

We end this section with some comments about the eigenvalue residual term. For $(\mathbf{x}^{(i)}, \lambda^{(i)})^T$ the eigenvalue residual

$$\mathbf{r}^{(i)} = (\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{x}^{(i)}, \tag{2.9}$$

is calculated in step (4) of Algorithm 2.1. Since $\mathbf{c}^H\mathbf{x}^{(i)} = 1$, $\forall i$, we have $\|\mathbf{r}^{(i)}\| = \|\mathbf{F}(\mathbf{z}^{(i)})\|$. Now with $\mathbf{z}_1 = (\mathbf{x}_1, \lambda_1)$ denoting the root of $\mathbf{F}(\mathbf{z}) = \mathbf{0}$,

$$\|\mathbf{F}(\mathbf{z}^{(i)})\| = \|\mathbf{F}(\mathbf{z}^{(i)}) - \mathbf{F}(\mathbf{z}_1)\| \leq C_1 \ \|\mathbf{z}^{(i)} - \mathbf{z}_1\|,$$

holds, where C_1 is a bound on the norm of $\mathbf{J}(\mathbf{z})$ in some ball centered on $\mathbf{z}_1$. Hence

$$\|\mathbf{r}^{(i)}\| \leq C_1 \ \|\mathbf{z}^{(i)} - \mathbf{z}_1\| \ . \tag{2.10}$$

In this section we have formulated inverse iteration as a Newton method. In the next section we show how inexact inverse iteration can be regarded as a modified Newton's method and hence derive corresponding convergence results.

3 Inexact Inverse Iteration & Modified Newton's Method.

Let us now consider a version of inexact inverse iteration that introduces two changes from Algorithm 2.1. First, as the name implies, we solve the linear systems iteratively to a given residual tolerance (and hence the linear systems are solved 'inexactly'). Second, instead of (1.1) we consider the linear system

$$(\mathbf{A} - \sigma\mathbf{M})\mathbf{y} = \mathbf{Z}(\lambda)\mathbf{x}, \tag{3.1}$$

where $\mathbf{Z}(\lambda)$ is a complex $n \times n$ matrix depending on λ. If $\mathbf{Z}(\lambda) = \mathbf{M}$, then (3.1) reduces to (1.1). However, in Section 4 we consider the system $(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{P}\mathbf{x}^{(i)}$, where $\mathbf{P}$ is a preconditioner for $\mathbf{A}$, and so we consider the convergence theory for the more general form given by (3.1). Thus we discuss the following algorithm.

ALGORITHM 3.1 (INEXACT INVERSE ITERATION). *Given* $\lambda^{(0)}$ *and* $\mathbf{x}^{(0)}$ *with* $\mathbf{c}^H\mathbf{x}^{(0)} = 1$. *For* $i = 0, 1, 2, \ldots$

(1) Choose $\tau^{(i)}$,

(2) Solve $(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}$ *inexactly, that is,*

$$\| (\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} - \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} \| \leq \tau^{(i)} \|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\|,$$

(3) Set $\Delta\lambda^{(i)} = \dfrac{1}{\mathbf{c}^H\mathbf{y}^{(i)}}; \quad \lambda^{(i+1)} = \lambda^{(i)} + \Delta\lambda^{(i)}$,

(4) Update $\mathbf{x}^{(i+1)} = \Delta\lambda^{(i)}\mathbf{y}^{(i)}$,

(5) Evaluate $\mathbf{r}^{(i+1)} = (\mathbf{A} - \lambda^{(i+1)}\mathbf{M})\mathbf{x}^{(i+1)}$,

(6) Test for convergence.

To analyse this algorithm let us introduce the linear system residual

$$\mathbf{res}^{(i)} := (\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} - \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} \tag{3.2}$$

which should not be confused with the eigenvalue residual $\mathbf{r}^{(i)}$ defined by (2.9). From step (2) in Algorithm 3.1 we know

$$\|\mathbf{res}^{(i)}\| \leq \tau^{(i)} \|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\| \ . \tag{3.3}$$

Now, using $\mathbf{x}^{(i+1)} = \Delta\lambda^{(i)}\mathbf{y}^{(i)}$ from step (4) of Algorithm 3.1, we may write (3.2) as

$$(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{x}^{(i+1)} = \Delta\lambda^{(i)}(\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} + \mathbf{res}^{(i)}),$$

or, equivalently,

$$(\mathbf{A} - \lambda^{(i)}\mathbf{M})\Delta\mathbf{x}^{(i)} - \Delta\lambda^{(i)}(\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} + \mathbf{res}^{(i)}) = -(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{x}^{(i)}.$$

This equation along with $\mathbf{c}^H\Delta\mathbf{x}^{(i)} = 0$ gives

$$\begin{bmatrix} \mathbf{A} - \lambda^{(i)}\mathbf{M} & -(\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} + \mathbf{res}^{(i)}) \\ \mathbf{c}^H & 0 \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}^{(i)} \\ \Delta\lambda^{(i)} \end{bmatrix} = \begin{bmatrix} -(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{x}^{(i)} \\ 0 \end{bmatrix}, \tag{3.4}$$

which with (2.6) we can write as

$$\widetilde{\mathbf{J}}(\mathbf{z}^{(i)})\Delta\mathbf{z}^{(i)} = -\mathbf{F}(\mathbf{z}^{(i)}), \quad \mathbf{z}^{(i+1)} = \mathbf{z}^{(i)} + \Delta\mathbf{z}^{(i)}, \tag{3.5}$$

where $\mathbf{z}^{(i)} = (\mathbf{x}^{(i)}, \lambda^{(i)})^T$ and $\widetilde{\mathbf{J}}$ is defined by

$$\widetilde{\mathbf{J}}(\mathbf{z}^{(i)}) = \begin{bmatrix} \mathbf{A} - \lambda^{(i)}\mathbf{M} & -(\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} + \mathbf{res}^{(i)}) \\ \mathbf{c}^H & 0 \end{bmatrix}. \tag{3.6}$$

Clearly (3.5) is a modified Newton method for $\mathbf{F}(\mathbf{z}) = \mathbf{0}$ with $\widetilde{\mathbf{J}}(\mathbf{z}^{(i)})$ being an approximation to the exact Jacobian $\mathbf{J}(\mathbf{z}^{(i)})$ given by (2.8). Hence the convergence of the inexact inverse iteration method given by Algorithm 3.1 can be proved using the convergence theory of modified Newton's method.

THEOREM 3.1 (CONVERGENCE OF INEXACT INVERSE ITERATION). *Let* $\mathbf{z}_1 = (\mathbf{x}_1, \lambda_1)^T$ *be an algebraically simple eigenpair of (2.1) satisfying (2.3). Since* $\mathbf{J}(\mathbf{z}_1)$ *defined by (2.8) is nonsingular we assume* $\|\mathbf{J}(\mathbf{z}_1)^{-1}\| \leq \beta$ *(see Lemma 2.1). For some* $\tau_{max}, r > 0$, *consider the use of Algorithm 3.1 with* $\tau^{(i)} \leq \tau_{max}, \quad \forall i$, *with starting value* $\mathbf{z}^{(0)} = (\mathbf{x}^{(0)}, \lambda^{(0)})^T \in \mathcal{B} = \mathcal{B}(\mathbf{z}_1, r)$. *If* r, τ_{max} *and* $\mathbf{Z}(\lambda)$ *are such that*

$$\beta\{|\lambda_1 - \lambda| \, \|\mathbf{M}\| + \|\mathbf{Z}(\lambda)\mathbf{x} - \mathbf{M}\mathbf{x}_1\| + \tau_{max} \, \|\mathbf{Z}(\lambda)\mathbf{x}\|\} =: \delta < 1 \tag{3.7}$$

for $\mathbf{z} = (\mathbf{x}, \lambda)^T \in \mathcal{B}$, *and if*

$$\left\{ \frac{\beta \, \|\mathbf{M}\| \, r}{1 - \delta} + \delta \right\} =: \alpha < 1, \tag{3.8}$$

then with $\mathbf{e}^{(i)} := \mathbf{z}^{(i)} - \mathbf{z}_1$,

a) Algorithm 3.1 converges linearly to $\mathbf{z}_1 = (\mathbf{x}_1, \lambda_1)^T$ *with*

$$\|\mathbf{z}^{(i+1)} - \mathbf{z}_1\| \leq \alpha \, \|\mathbf{z}^{(i)} - \mathbf{z}_1\|,$$

b) $\mathbf{e}^{(i+1)}$ *satisfies*

$$\|\mathbf{e}^{(i+1)}\| \leq \frac{\beta}{1 - \delta} K^{(i)} \, \|\mathbf{e}^{(i)}\|, \tag{3.9}$$

where $K^{(i)} = \|\mathbf{M}\| \|\mathbf{e}^{(i)}\| + \|\mathbf{M}\mathbf{x}^{(i)} - \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\| + \|\mathbf{res}^{(i)}\|$,

c) if, in addition, $\tau^{(i)}$ *in Algorithm 3.1 satisfies*

$$\tau^{(i)} \leq C_2 \, \|\mathbf{r}^{(i)}\|, \tag{3.10}$$

for some constant C_2 *independent of* i *with* $\mathbf{r}^{(i)}$ *given by (2.9), and*

$$\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)}\| \leq C_3 \, \|\mathbf{e}^{(i)}\|, \tag{3.11}$$

for some constant C_3 *independent of* i, *then Algorithm 3.1 converges quadratically.*

PROOF. The proof consists of verifying the conditions of Theorem A.1 on the convergence of modified Newton's method (given in the Appendix) for $\mathbf{F}$ defined by (2.4). First note that γ, the Lipschitz constant of $\mathbf{J}$, can be taken as $2\|\mathbf{M}\|$. Next by reducing r and τ_{max} and taking $\mathbf{Z}(\lambda)$ close enough to $\mathbf{M}$, conditions (3.7) and (3.8) can always be made to hold. Thus conditions (A.1) and (A.4) of Theorem A.1 hold and so the linear convergence of Algorithm 3.1 (part (a)) is proved. Part (b) of Theorem 3.1 follows immediately from part (b) in Theorem A.1. Under (3.10) and (3.11) (and recalling (2.10)),

$$\|\mathbf{J}(\mathbf{z}^{(i)}) - \widetilde{\mathbf{J}}(\mathbf{z}^{(i)})\| \leq \|\mathbf{M}\mathbf{x}^{(i)} - \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\| + \|\mathbf{res}^{(i)}\| \leq C_4 \, \|\mathbf{e}^{(i)}\|, \tag{3.12}$$

for some constant C_4 independent of i. The quadratic convergence follows from case (c) in Theorem A.1. □

We see from (3.9) that the possibility of achieving quadratic convergence in Algorithm 3.1 is determined by the size of $\|\mathbf{M}\mathbf{x}^{(i)} - \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\|$ and $\|\mathbf{res}^{(i)}\|$. We discuss the natural case $\mathbf{Z}(\lambda^{(i)}) = \mathbf{M}$ in the following subsection, but we end this theoretical section with the following corollary which will be useful in Section 4.

Corollary 3.2. *Assume that the conditions of Theorem 3.1 are satisfied. Let $\tau^{(i)}$ be chosen as in (3.10) and assume $\lambda^{(i)} \neq 0$. If $\mathbf{Z}(\lambda^{(i)})$ satisfies*

$$\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \frac{1}{\lambda^{(i)}}\mathbf{A}\mathbf{x}^{(i)}\| \leq C_5\|\mathbf{e}^{(i)}\|, \tag{3.13}$$

then Algorithm 3.1 exhibits quadratic convergence.

Proof. We have

$$\begin{aligned}\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)}\| \leq \|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} &- \frac{1}{\lambda^{(i)}}\mathbf{A}\mathbf{x}^{(i)}\| \\ &+ \|\frac{1}{\lambda^{(i)}}(\mathbf{A}\mathbf{x}^{(i)} - \lambda^{(i)}\mathbf{M}\mathbf{x}^{(i)})\| \, .\end{aligned} \tag{3.14}$$

Using (3.13) and the fact that $(\mathbf{x}^{(i)}, \lambda^{(i)})^T$ is an approximate eigenpair we can apply the properties of the eigenvalue residual (2.9) and (2.10) to get

$$\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)}\| \leq \; C_5 \; \|\mathbf{e}^{(i)}\| + \frac{1}{|\lambda^{(i)}|}C_1 \; \|\mathbf{e}^{(i)}\| \leq C_3 \; \|\mathbf{e}^{(i)}\|,$$

where $C_3 := C_5 + \frac{1}{|\lambda^{(i)}|}C_1$. Hence (3.10) and (3.11) in Theorem 3.1 hold, proving that the convergence is quadratic. □

Thus we see from (3.13) that if the right hand side of (3.1) can be made to approximate $\frac{1}{\lambda^{(i)}}\mathbf{A}\mathbf{x}^{(i)}$ then there is the possibility of superlinear or even quadratic convergence.

3.1 Standard Inverse Iteration

In this subsection we consider the standard form of inverse iteration by making the choice $\mathbf{Z}(\lambda^{(i)}) = \mathbf{M}$. In this context we discuss two choices of $\tau^{(i)}$ in Algorithm 3.1, namely, either $\tau^{(i)}$ is chosen to decrease or $\tau^{(i)}$ is held fixed (cases (a) and (b) respectively in the following corollary).

Corollary 3.3. *Assume $\mathbf{Z}(\lambda^{(i)}) = \mathbf{M}$ in Algorithm 3.1 and let the conditions of Theorem 3.1 hold. Then we obtain the following rates of convergence, depending on the tolerance $\tau^{(i)}$.*

a) Decreasing tolerance. *If $\tau^{(i)} \leq C_2 \; \|\mathbf{r}^{(i)}\|$ in step (1) of Algorithm 3.1 then, for a close enough starting guess, Algorithm 3.1 achieves quadratic convergence, which is equal to the rate achieved by Algorithm 2.1 for exact inverse iteration.*

b) Fixed tolerance. *If $\tau^{(i)} = \tau$ in step (1) of Algorithm 3.1, where τ is fixed but small enough, then, for a close enough starting guess, Algorithm 3.1 converges linearly.*

PROOF. For $\mathbf{Z}(\lambda^{(i)}) = \mathbf{M}$ condition (3.11) in Theorem 3.1 is obviously satisfied with $C_3 = 0$. In the case of a decreasing tolerance (a), condition (3.10) of Theorem 3.1 is assumed and therefore quadratic convergence follows immediately from Theorem 3.1. In the case of a fixed tolerance (b), the bound in (3.3) becomes

$$\|\mathbf{res}^{(i)}\| \leq \tau\, \|\mathbf{M}\mathbf{x}^{(i)}\|, \tag{3.15}$$

where τ is fixed. Then the $K^{(i)}$ term in (3.9) does not decrease with i and hence only linear convergence can be proved. ☐

3.2 Numerical Example

We now present some numerical results to illustrate the theory in Corollary 3.3, and also provide a comparison with Algorithm 2.1 (exact solves).

EXAMPLE 3.1. *Consider the standard eigenvalue problem $\mathbf{A}\mathbf{x} = \lambda\mathbf{x}$ where $\mathbf{A}$ is the finite difference discretisation (central differences) on a 32×32 grid of the following eigenvalue problem of the convection-diffusion operator*

$$-\Delta u + 5u_x + 5u_y = \lambda u \quad on \quad (0,1)^2,$$

with homogeneous Dirichlet boundary conditions. Take $\mathbf{Z}(\lambda^{(i)}) = \mathbf{M} = \mathbf{I}$. Consider finding the smallest eigenvalue ($\lambda_1 \approx 32.18560954$) by Algorithm 2.1 and by Algorithm 3.1 with both decreasing and fixed tolerances. We take an initial vector $\mathbf{x}^{(0)}$ which is sufficiently close to the eigenvector $\mathbf{x}_1$ and consider the convergence behaviour of the outer iteration. We use preconditioned GMRES as the inexact solver and for the inexact solves with fixed tolerance we take $\tau^{(i)} = \tau = 0.3$ (case (b) in Corollary 3.3) and for the inexact solves with decreasing tolerance (case (a) in Corollary 3.3) we take

$$\tau^{(i)} = \min\{\tau, \|\mathbf{r}^{(i)}\|\}, \quad with \quad \tau = 0.3, \tag{3.16}$$

where the eigenvalue residual $\|\mathbf{r}^{(i)}\|$ is given by (2.9). As an overall stopping condition we use the norm of the eigenvalue residual, so that, once

$$\|\mathbf{r}^{(i)}\| < 10^{-10}$$

is satisfied, the computation stops.

The results are illustrated in Figure 3.1 which gives logarithmic plots for the norm of the error at step $i+1$ against the norm of the error at step i. The dotted lines indicate the slopes expected for linear and quadratic convergence. Clearly, exact inverse iteration indicated by a solid line yields quadratic convergence as expected, since it corresponds to Newton's method (see section 2). Also inexact inverse iteration with decreasing tolerance indicated by the dash-dotted

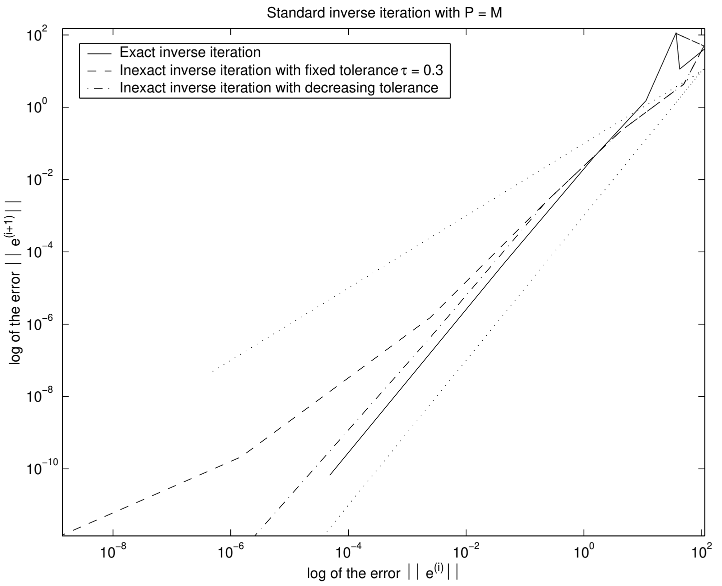

Figure 3.1: Numerical results for Example 3.1. The slopes of the solid, dashed and dashed-dotted lines indicate the rates of convergence achieved. The dotted lines indicate the slopes espected for linear and quadratic convergence.

line gives quadratic convergence as expected from Corollary 3.3 part (a). For inexact inverse iteration with fixed tolerance plotted in the dashed line we get only linear convergence as predicted in Corollary 3.3 part (b).

We remark that numerical results using $\mathbf{Z}(\lambda^{(i)}) = \frac{1}{\lambda^{(i)}}\mathbf{A}$ (so that $C_5 = 0$ in Corollary 3.2) using both exact solves and inexact solves with decreasing tolerance chosen as in (3.16) produce the expected quadratic convergence. We do not reproduce these results here.

4 Preconditioned Iterative Solvers.

In this section we consider the preconditioned iterative solution of the linear systems in inverse iteration algorithms and shall show how a modified right hand side as in (3.1) can arise.

Our motivation for choosing a different right hand side arises in the consideration of the performance of the iterative solver used in inexact inverse iteration. It was noted in [13] for the standard symmetric eigenvalue problem that it was advantageous to alter the right hand side in inverse iteration to reduce the number of iterations used by a Krylov solver. We explain the idea for the nonsymmetric case. The obvious way to implement the right-preconditioned solution of

$(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{M}\mathbf{x}^{(i)}$, with $\mathbf{P}^{-1}$ a suitable preconditioner, is

$$(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{P}^{-1}\widetilde{\mathbf{y}}^{(i)} = \mathbf{M}\mathbf{x}^{(i)}, \tag{4.1}$$

with $\mathbf{y}^{(i)}$ retrieved from $\mathbf{y}^{(i)} = \mathbf{P}^{-1}\widetilde{\mathbf{y}}^{(i)}$. However, the idea in [13] is to alter the right hand side of (4.1) to give

$$(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{P}^{-1}\widetilde{\mathbf{y}}^{(i)} = \mathbf{P}\mathbf{x}^{(i)}, \tag{4.2}$$

with the result that the right hand side of (4.2) is in the direction of the approximate null-vector of the iteration matrix $(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{P}^{-1}$. For left preconditioning, the approach suggests that instead of solving

$$\mathbf{P}^{-1}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{P}^{-1}\mathbf{M}\mathbf{x}^{(i)}, \tag{4.3}$$

one should solve

$$\mathbf{P}^{-1}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{x}^{(i)}. \tag{4.4}$$

Numerical experiments indicate that GMRES requires fewer steps to solve (4.2) than to solve (4.1) to a specified relative tolerance (see, for example, Table 4.2 in Example 4.2). This aspect of the costs of Krylov solvers applied to shifted linear systems in inverse iteration is further discussed in [1] (for symmetric problems) and in [2] for nonsymmetric problems. From the viewpoint of convergence theory both (4.2) and (4.4) reduce to the equation

$$(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{P}\mathbf{x}^{(i)}, \tag{4.5}$$

i.e. the equation analysed in Section 3, where $\mathbf{Z}(\lambda^{(i)}) = \mathbf{P}$. We shall see that for the form of inexact inverse iteration used here, where the approximate eigenvectors are scaled so that $\mathbf{c}^H\mathbf{x}^{(i)} = 1$, rather than using a norm, it is better to introduce a factor of $\frac{1}{\lambda^{(i)}}$ on the right hand side (as in (4.9) below).

In the following two subsections we will examine two different choices for the preconditioner $\mathbf{P}$ and compare the costs of solving (4.3) and (4.4).

4.1 Incomplete LU preconditioning

In this subsection we consider the use of an incomplete LU factorisation of $\mathbf{A}$ as a preconditioner. Assume

$$\mathbf{A} = \tilde{\mathbf{L}}\tilde{\mathbf{U}} + \mathbf{E}, \tag{4.6}$$

where $\tilde{\mathbf{L}}$ is a lower triangular matrix and $\tilde{\mathbf{U}}$ is an upper triangular matrix approximating the matrices $\mathbf{L}$ and $\mathbf{U}$ from the complete LU decomposition. $\mathbf{E}$ is the error matrix associated with the incomplete factorisation. We take as preconditioner the matrix

$$\tilde{\mathbf{P}} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}. \tag{4.7}$$

The previous discussion suggests two options for preconditioning. First, we may simply apply preconditioning in the obvious way to the standard inverse iteration method, that is,

$$\tilde{\mathbf{P}}^{-1}(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \tilde{\mathbf{P}}^{-1}\mathbf{M}\mathbf{x}^{(i)}, \tag{4.8}$$

which does not change the analysis of Section 3.1 for the outer rate of convergence of Algorithm 3.1. Alternatively, we may modify the right hand side and use

$$\tilde{\mathbf{P}}^{-1}(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{x}^{(i)}, \tag{4.9}$$

which, from the point of view of the outer convergence rate, is equivalent to $(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \frac{1}{\lambda^{(i)}}\tilde{\mathbf{P}}\mathbf{x}^{(i)}$ (cf. (4.5) but with the right hand side scaled by $\frac{1}{\lambda^{(i)}}$). This is motivated by the similar scaling given in (3.13). So, in the notation of Section 3, (4.9) is written as $(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}$ with $\mathbf{Z}(\lambda^{(i)}) = \frac{1}{\lambda^{(i)}}\tilde{\mathbf{P}}$. For this approach quadratic convergence is lost, but we do achieve a linear rate of convergence as is shown in the following corollary.

COROLLARY 4.1. *Assume*

$$\mathbf{Z}(\lambda^{(i)}) = \frac{1}{\lambda^{(i)}}\tilde{\mathbf{L}}\tilde{\mathbf{U}} \tag{4.10}$$

in Algorithm 3.1 and let the conditions of Theorem 3.1 hold. Then we obtain the following linear rates of convergence, depending on the tolerance $\tau^{(i)}$.

a) Decreasing tolerance. *If $\tau^{(i)} \le C_2\,\|\mathbf{r}^{(i)}\|$ in step (1) of Algorithm 3.1 then, for a close enough starting guess, Algorithm 3.1 achieves linear convergence with*

$$\begin{aligned}\|\mathbf{e}^{(i+1)}\| \le & \frac{\beta}{1-\delta}\frac{\|\mathbf{E}\mathbf{x}^{(i)}\|}{|\lambda^{(i)}|}\,\|\mathbf{e}^{(i)}\| \\ & + \frac{\beta}{1-\delta}\left\{\|\mathbf{M}\| + C_1C_2\,\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\| + \frac{C_1}{|\lambda^{(i)}|}\right\}\|\mathbf{e}^{(i)}\|^2,\end{aligned} \tag{4.11}$$

where $\mathbf{e}^{(i)} = \mathbf{z}^{(i)} - \mathbf{z}_1$.

b) Fixed tolerance. *If $\tau^{(i)} = \tau$ in step (1) of Algorithm 3.1, where τ is fixed but small enough, then, for a close enough starting guess, Algorithm 3.1 converges linearly with*

$$\begin{aligned}\|\mathbf{e}^{(i+1)}\| \le & \frac{\beta}{1-\delta}\left\{\frac{1}{|\lambda^{(i)}|}\,\|\mathbf{E}\mathbf{x}^{(i)}\| + \tau\,\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)}\|\right\}\|\mathbf{e}^{(i)}\| \\ & + \frac{\beta}{1-\delta}\left\{\|\mathbf{M}\| + \frac{C_1}{|\lambda^{(i)}|}\right\}\|\mathbf{e}^{(i)}\|^2\,.\end{aligned} \tag{4.12}$$

PROOF. With $\mathbf{Z}(\lambda^{(i)})$ defined by (4.10) and using (4.6) we may write

$$\begin{aligned}\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)} &= \frac{1}{\lambda^{(i)}}(\mathbf{A}-\mathbf{E})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)} \\ &= \frac{1}{\lambda^{(i)}}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{x}^{(i)} - \frac{1}{\lambda^{(i)}}\mathbf{E}\mathbf{x}^{(i)}.\end{aligned} \tag{4.13}$$

Hence, with (2.9) we get

$$\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)}\| = \frac{1}{|\lambda^{(i)}|}\|\mathbf{r}^{(i)} - \mathbf{E}\mathbf{x}^{(i)}\|, \tag{4.14}$$

and using (2.10) we obtain

$$\|\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} - \mathbf{M}\mathbf{x}^{(i)}\| \leq \frac{C_1}{|\lambda^{(i)}|}\|\mathbf{e}^{(i)}\| + \frac{1}{|\lambda^{(i)}|}\|\mathbf{E}\mathbf{x}^{(i)}\|. \tag{4.15}$$

Now using (3.9), and with $\tau^{(i)}$ satisfying (3.10), we obtain the convergence result (4.11) which shows linear convergence. Similarly, if $\tau^{(i)}$ satisfies a fixed tolerance (i.e. $\tau^{(i)} = \tau$) then (4.12) is derived from (3.9). □

This proof illustrates the importance of the scaling factor $\frac{1}{\lambda^{(i)}}$ on the right hand side of (4.9). Inequalities (4.11) and (4.12) show the dominant terms that influence the linear rate of convergence for a decreasing and a fixed tolerance respectively. The following example illustrates that (4.11) does indeed describe what is observed in practice.

EXAMPLE 4.1. *We use the same matrix* $\mathbf{A}$ *as in Example 3.1 but here* $\mathbf{M}$ *is a symmetric tridiagonal matrix with* $2/3$ *as diagonal and* $1/6$ *as sub- and superdiagonals. Again, we seek the smallest eigenvalue, which in this case is given by* $\lambda_1 \approx 32.17511440$.

We apply inexact inverse iteration to the problem with modified right hand side (4.9) and we use the algorithm with decreasing tolerance (3.10). Again, the initial vector $\mathbf{x}^{(0)}$ *is chosen to be sufficiently close to the eigenvector* $\mathbf{x}_1$*, the decreasing tolerance* $\tau^{(i)}$ *and the stopping condition are chosen as in Example 3.1. Hence (4.11) holds.*

We apply GMRES to (4.9) where $\tilde{\mathbf{P}} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}$*, with* $\tilde{\mathbf{L}}$ *and* $\tilde{\mathbf{U}}$ *chosen as in (4.6). Note, the preconditioner is only applied to the left hand side of the equation and the right hand side is scaled by the computed eigenvalue. Furthermore we perform the incomplete* LU *factorization of* $\mathbf{A}$ *with different drop tolerances ranging from* 10^{-2} *to* 10^{-4} *giving an increasingly better preconditioner.*

In Figure 4.1 the results for the convergence rate of the outer iteration with decreasing tolerance are given. As predicted in (4.11) we observe linear convergence in each experiment. Table 4.1 gives values of $\|\mathbf{E}\|_\infty$ and the numerical values of the slopes of the corresponding lines in Figure 4.1. We see that as the drop tolerance decreases by a factor of 10 the slope reduces approximately by a factor of 10 as predicted by (4.11).

The next experiment compares the costs of solving (4.8) with the costs of solving (4.9), where $\tilde{\mathbf{P}} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}$.

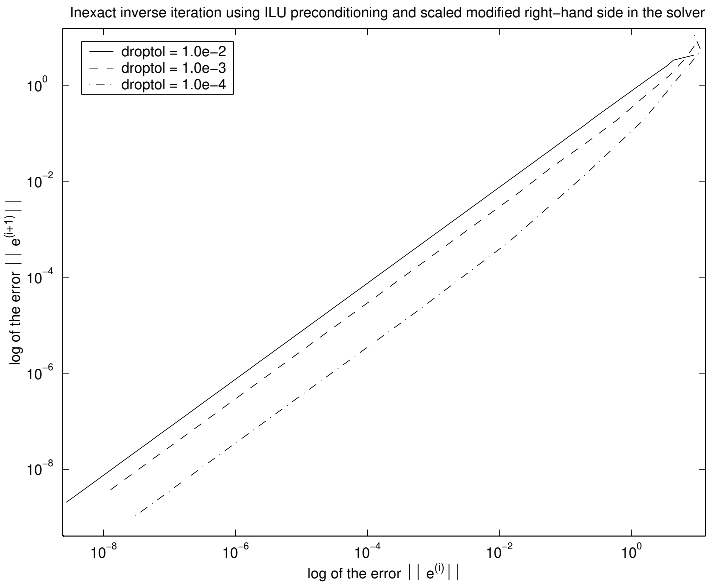

Figure 4.1: Numerical results for Example 4.1. Outer convergence rate are shown for inexact inverse iteration using ILU preconditioned GMRES with different drop tolerances and scaled modified right hand side

Table 4.1: Results for Example 4.1. The table gives values for $\|\mathbf{E}\|_\infty$ and the numerical values of the slopes of the corresponding lines in Figure 4.1 for different drop tolerances of the preconditioner

Drop Tolerance	$\|\mathbf{E}\|_\infty$	Slope in Figure 4.1
1.0e-2	1.672e+2	7.7e-1
1.0e-3	2.395e+1	3.0e-1
1.0e-4	2.846e+0	3.6e-2

Example 4.2. *Consider Example 4.1 with a drop tolerance of* 10^{-4}. *We now compare the perfomance, with respect to total number of inner iterations, of (4.8) with a decreasing tolerance (which exhibits quadratic convergence by Corollary 3.3) and (4.9) with a fixed tolerance (which has linear convergence by Corollary 4.1 no matter the choice of* $\tau^{(i)}$*). For the fixed tolerance we use* $\tau = 0.1$*, for the decreasing tolerance,* $\tau^{(i)} = \min\{0.1, \|\mathbf{r}^{(i)}\|\}$.

The number of inner solves per outer iteration are listed in Table 4.2. We observe that the method based on (4.9) needs more outer iterations, since the convergence is only linear whereas the method based on (4.8) and decreasing tolerance achieves quadratic convergence. However for (4.9) the number of inner iterations is approximately constant at each outer iteration whereas the number

Table 4.2: Iteration numbers for Example 4.2. Total number of iterations and number of inner iterations for inexact inverse iteration using either solves of (4.8) with decreasing tolerance or (4.9) with fixed tolerance

Outer Iterations	Example (4.9)	Example (4.8)
1	2	3
2	2	2
3	2	3
4	2	5
5	2	11
6	3	16
7	3	
8	4	
9	3	
10	3	
11	4	
total	30	40

of inner iterations per outer iteration increases for (4.8). This phenomena is discussed for symmetric problems in [1] and for nonsymmetric problems in [2]. For this example with this drop tolerance the linearly convergent method based on using (4.9) is seen to be more efficient than the quadratically convergent method based on (4.8). In other experiments on this example the quadratically convergent method was more efficient, so it is not clear which approach should be preferred in other applications.

Note that further experiments show that we can increase the fixed tolerance for (4.9) to about $\tau = 0.7$ without getting any significant change in the total number of iterations (which varies from 29 to 32). This can be explained by formula (4.12), where the two terms in $\frac{1}{|\lambda^{(i)}|} \parallel \mathbf{E}\mathbf{x}^{(i)} \parallel + \tau \parallel \mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} \parallel$ have similar orders of magnitude up to around $\tau = 0.7$.

4.2 Tuning the ILU preconditioner

In this section we consider the strategy of tuning the preconditioner to satisfy the condition $\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{A}\mathbf{x}^{(i)}$ and hence recover a quadratically convergent method (from Corollary 3.2 with $C_5 = 0$). We explain this idea using the ILU preconditioner derived from (4.6). Given $\mathbf{x}^{(i)}$ define

$$\mathbf{f}^{(i)} = \mathbf{E}\mathbf{x}^{(i)} \tag{4.16}$$

and introduce the "tuned" preconditioner

$$\mathbf{P}^{(i)} := \tilde{\mathbf{L}}\tilde{\mathbf{U}} + \mathbf{f}^{(i)}\mathbf{c}^H, \tag{4.17}$$

where $\mathbf{c}^H\mathbf{x}^{(i)} = 1$. $\mathbf{P}^{(i)}$ is tuned in the sense that $\mathbf{P}^{(i)}\mathbf{x}^{(i)} = \mathbf{A}\mathbf{x}^{(i)}$, and hence $\mathbf{Z}(\lambda^{(i)})\mathbf{x}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{P}^{(i)}\mathbf{x}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{A}\mathbf{x}^{(i)}$, and so $C_5 = 0$ in Corollary 3.2. Note that $\mathbf{P}^{(i)}$ is a rank-one change of $\tilde{\mathbf{P}}$, and so, using the Sherman-Morrison formula (see, for example, [4, p. 95]), the additional cost of calculating the action of ${\mathbf{P}^{(i)}}^{-1}$ compared with the action of $\tilde{\mathbf{P}}^{-1}$ for a given i, is merely one forward and one back substitution. The following corollary is easily proved.

Corollary 4.2. *Consider the use of*

$$ {\mathbf{P}^{(i)}}^{-1}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{x}^{(i)} \tag{4.18} $$

in step (2) of Algorithm 3.1, so that $\mathbf{Z}(\lambda^{(i)}) = \frac{1}{\lambda^{(i)}}\mathbf{P}^{(i)}$ *with* $\mathbf{P}^{(i)}$ *given by (4.17), and let the conditions of Theorem 3.1 hold. If* $\tau^{(i)} \leq C_2\, \|\mathbf{r}^{(i)}\|$ *in step (1) of Algorithm 3.1 then, for a close enough starting guess, Algorithm 3.1 achieves quadratic convergence.*

We now give a numerical example for the tuned preconditioner and compare it to the usual ILU preconditioner in Section 4.1.

Example 4.3. *We take the same eigenvalue problem as in Example 4.1 and use as solver preconditioned GMRES with either the usual ILU preconditioner,* $\tilde{\mathbf{P}}$ *given by (4.7), or the tuned preconditioner,* $\mathbf{P}^{(i)}$ *given by (4.17). We compare the costs of the following three methods.*

a) *"*$\mathbf{P}^{(i)}$*/modified-rhs": the tuned preconditioner is applied to the inverse iteration system with a modified right hand side, namely,*

$$ {\mathbf{P}^{(i)}}^{-1}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{x}^{(i)}, \quad \mathbf{P}^{(i)} = \tilde{\mathbf{L}}\tilde{\mathbf{U}} + \mathbf{f}^{(i)}\mathbf{c}^H, \tag{4.19} $$

where $\tilde{\mathbf{U}}$ *and* $\tilde{\mathbf{L}}$ *are given by the ILU decomposition, and with* $\mathbf{f}^{(i)}$ *defined by (4.16).*

b) *"*$\mathbf{P}^{(i)}$*/standard-rhs": the tuned preconditioner is applied to the standard inverse iteration system, namely,*

$$ {\mathbf{P}^{(i)}}^{-1}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = {\mathbf{P}^{(i)}}^{-1}\mathbf{M}\mathbf{x}^{(i)}, \quad \mathbf{P}^{(i)} = \tilde{\mathbf{L}}\tilde{\mathbf{U}} + \mathbf{f}^{(i)}\mathbf{c}^H. \tag{4.20} $$

c) *"*$\tilde{\mathbf{P}}$*/standard-rhs": the usual ILU preconditioner is applied to the standard inverse iteration system, namely,*

$$ \tilde{\mathbf{P}}^{-1}(\mathbf{A}-\lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \tilde{\mathbf{P}}^{-1}\mathbf{M}\mathbf{x}^{(i)}, \quad \tilde{\mathbf{P}} = \tilde{\mathbf{L}}\tilde{\mathbf{U}}. \tag{4.21} $$

In each case we use the decreasing tolerance $\tau^{(i)} = \min\{\tau, \|\mathbf{r}^{(i)}\|\}$ *with* $\tau = 0.2$*. So all three methods have quadratic convergence using Corollaries 3.3 and 4.2.*

In Figure 4.2 we give logarithmic plots of the errors obtained from method "$\mathbf{P}^{(i)}$/modified-rhs" for two different drop tolerances. The dotted lines indicate the slopes expected for linear and quadratic convergence. As predicted in Corollary 4.2 we achieve quadratic convergence for this method with the tuned preconditioner, as distinct from the linear convergence seen in Figure 4.1 achieved by the corresponding ILU preconditioner.

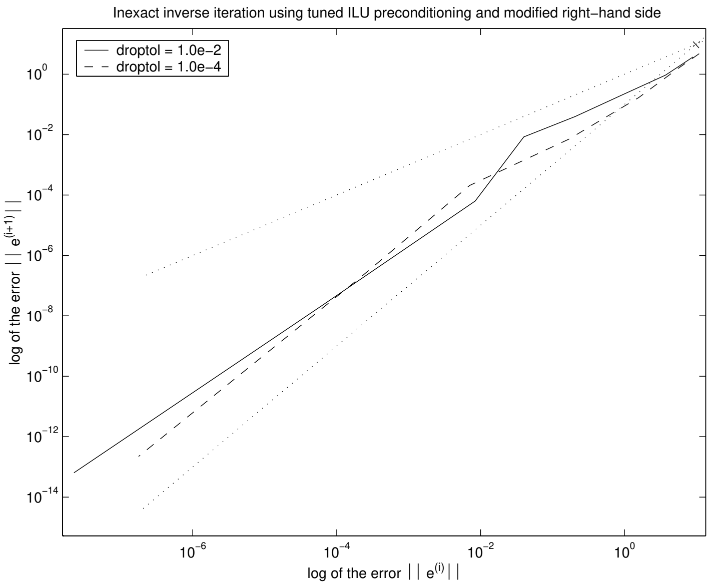

Figure 4.2: Numerical results for Example 4.3. The quadratic outer convergence rate for method "$\mathbf{P}^{(i)}$/modified-rhs" with different drop tolerances is readily observed.

Table 4.3: Iteration numbers for Example 4.3. Total number of iterations and number of inner iterations for the three methods using (4.19), (4.20) or (4.21) with decreasing tolerance. In each method the drop tolerances were 10^{-2} and 10^{-4}.

	"$\mathbf{P}^{(i)}$/modified-rhs"		"$\mathbf{P}^{(i)}$/standard-rhs"		"$\tilde{\mathbf{P}}$/standard-rhs"	
OUTER IT.	10^{-2}	10^{-4}	10^{-2}	10^{-4}	10^{-2}	10^{-4}
1	2	2	2	2	3	2
2	3	2	2	2	3	2
3	3	2	3	2	6	2
4	3	2	3	2	9	4
5	4	2	7	5	17	9
6	6	3	12	9	32	18
7	10	5				
8	17	9				
total	48	27	29	22	70	37

Table 4.3 shows the number of inner iterations for the inexact solves of the three methods. We see that for both drop tolerances the tuned preconditioner applied to the standard inverse iteration formulation, method "$\mathbf{P}^{(i)}$/standard-rhs", requires fewer iterations than the other two methods. In particular, comparing

the results for "$\mathbf{P}^{(i)}$/standard-rhs" with "$\tilde{\mathbf{P}}$/standard-rhs" we see that the tuned preconditioner does better than the usual ILU preconditioner, with a significant reduction in overall costs when the drop tolerance is 10^{-2}. This has implications for the costs of implementing the action of the preconditioners; an aspect that we don't consider here. Method "$\mathbf{P}^{(i)}$/standard-rhs" requires two fewer outer iterations than "$\mathbf{P}^{(i)}$/modified-rhs". This may be explained by considering the constants in the convergence theory.

Hence, if the preconditioner is tuned as explained in this section, it appears to be best to apply it on the standard system $(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{M}\mathbf{x}^{(i)}$ rather than consider modifying the right hand side. This gives the best result in terms of the total number of iterations and the convergence rate. A simple, heuristic, explanation of this is as follows. We see from (4.20) that near convergence

$$\mathbf{P}^{(i)^{-1}}(\mathbf{A} - \lambda^{(i)}\mathbf{M})\mathbf{y}^{(i)} = \mathbf{P}^{(i)^{-1}}\mathbf{M}\mathbf{x}^{(i)} \approx \frac{1}{\lambda^{(i)}}\mathbf{P}^{(i)^{-1}}\mathbf{A}\mathbf{x}^{(i)} = \frac{1}{\lambda^{(i)}}\mathbf{x}^{(i)},$$

and so the right hand side $\mathbf{P}^{(i)^{-1}}\mathbf{M}\mathbf{x}^{(i)}$ is roughly in the desired direction, thus keeping the costs of the inner solves of GMRES low.

5 Conclusions.

We have analysed inexact inverse iteration for a generalised eigenvalue problem and we have shown that for an algebraically simple eigenvalue it is a modified Newton method. Using the convergence theory of the modified Newton method we obtained convergence rates for inexact solves with either fixed or decreasing tolerances. This approach is much simpler than previous approaches involving eigenvector expansions.

Using the same theoretical approach, we also analysed preconditioned iterative solves. Using an ILU preconditioner we have shown how modifying the right hand side may reduce the outer rate of convergence from quadratic to linear. We have also shown how the ILU preconditioner may be tuned to recover quadratic convergence in the modified right hand side formulation. Finally, we have shown that the application of the tuned preconditioner may be advantageous when applied to the standard inverse iteration formulation.

A Modified Newton Theory.

Here we are state a convergence theorem for modified Newton's method (see, for example, [5]) which is used in the proof of Theorem 3.1.

THEOREM A.1. *Assume* $\mathbf{F} : \mathcal{C}^n \longrightarrow \mathcal{C}^n$ *and let* $\mathbf{F}(\mathbf{z}^*) = \mathbf{0}$. *For some* $r > 0$, *define* $\mathcal{B} := \mathcal{B}(\mathbf{z}^*, r)$ *and assume* $\mathbf{J}(\mathbf{z}) \in Lip_\gamma\mathcal{B}$, *where* $\mathbf{J}(\mathbf{z})$ *is the Jacobian of* $\mathbf{F}(\mathbf{z})$. *Further, assume* $\|\mathbf{J}(\mathbf{z}^*)^{-1}\| \leq \beta$. *For each* $\mathbf{z}$ *let* $\widetilde{\mathbf{J}}(\mathbf{z})$ *be a complex* $n \times n$ *matrix satisfying, for some* δ, $0 \leq \delta < 1$,

$$\|\mathbf{J}(\mathbf{z}^*)^{-1}(\widetilde{\mathbf{J}}(\mathbf{z}) - \mathbf{J}(\mathbf{z}^*))\| \leq \delta. \tag{A.1}$$

Then $\widetilde{\mathbf{J}}(\mathbf{z})^{-1}$ *exists in* $\mathcal{B}$ *and* $\|\widetilde{\mathbf{J}}(\mathbf{z})^{-1}\| \leq \dfrac{\beta}{1-\delta}$, $\forall \mathbf{z} \in \mathcal{B}$. *Next consider the solution of*

$$\mathbf{F}(\mathbf{z}) = \mathbf{0}, \quad \mathbf{z} \in \mathcal{C}^n \tag{A.2}$$

using modified *Newton's method:*

$$\mathbf{z}^{(i+1)} = \mathbf{z}^{(i)} - \widetilde{\mathbf{J}}(\mathbf{z}^{(i)})^{-1}\mathbf{F}(\mathbf{z}^{(i)}), \quad \mathbf{z}^{(0)} \in \mathcal{B}. \tag{A.3}$$

If

$$\left\{ \frac{\beta\gamma r}{2(1-\delta)} + \delta \right\} =: \alpha < 1, \quad \forall \mathbf{z} \in \mathcal{B} \tag{A.4}$$

then, with $\mathbf{e}^{(i)} := \mathbf{z}^{(i)} - \mathbf{z}^*$,

a) modified Newton's method converges linearly *to* $\mathbf{z}^*$,

b) we have

$$\|\mathbf{e}^{(i+1)}\| \leq \frac{\beta}{(1-\delta)} \left\{ \frac{\gamma}{2} \|\mathbf{e}^{(i)}\| + \|\mathbf{J}(\mathbf{z}^{(i)}) - \widetilde{\mathbf{J}}(\mathbf{z}^{(i)})\| \right\} \|\mathbf{e}^{(i)}\|,$$

and

c) if $\|(\mathbf{J}(\mathbf{z}^{(i)}) - \widetilde{\mathbf{J}}(\mathbf{z}^{(i)}))\mathbf{e}^{(i)}\| \leq C\, \|\mathbf{e}^{(i)}\|^2$ *for some constant* C *then modified Newton's method converges* quadratically.

REFERENCES

1. J. Berns-Müller, I. G. Graham, and A. Spence, *Inexact inverse iteration for symmetric matrices*, 2005. Submitted to Linear Algebra Appl.

2. J. Berns-Müller and A. Spence, *Inexact inverse iteration and gmres*, 2005. In preparation.

3. ———, *Inexact inverse iteration and the nonsymmetric generalized eigenvalue problem*, 2005. Submitted to SIMAX.

4. J. W. Demmel, *Applied Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

5. J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, vol. 16 of Classics in Applied Mathematics, SIAM, Philadelphia, PA, 1996. Unabridged, corrected reprint of the 1983 original.

6. G. H. Golub and Q. Ye, *Inexact inverse iteration for generalized eigenvalue problems*, BIT, 40 (2000), pp. 671–684.

7. S. C. Hawkins and A. Spence, *The detection of Hopf bifurcations in discretization of certain nonlinear partial differential equations*, Notes on Numerical Fluid Mechanics, 74 (2000), pp. 145–154.

8. H. B. Keller, *Numerical solution of bifurcation and nonlinear eigenvalue problems*, in Applications of Bifurcation Theory, P. H. Rabinowitz, ed., Academic Press, New York, 1977, pp. 359–384.
9. A. V. Knyazev, *Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method*, SIAM Journal on Scientific Computing, 23 (2001), pp. 517–541.
10. Y.-L. Lai, K.-Y. Lin, and L. Wen-Wei, *An inexact inverse iteration for large sparse eigenvalue problems*, Numerical Linear Algebra with Applications, 1 (1997), pp. 1–13.
11. K. Neymeyr, *A geometric theory for preconditioned inverse iteration i: Extrema of the Rayleigh quotient*, Linear Algebra and its Applications, 322 (2001), pp. 61–85.
12. A. Ruhe and T. Wiberg, *The method of conjugate gradients used in inverse iteration*, BIT, 12 (1972), pp. 543–554.
13. V. Simoncini and L. Eldén, *Inexact Rayleigh quotient-type methods for eigenvalue computations*, BIT, 42 (2002), pp. 159–182.
14. P. Smit and M. H. C. Paardekooper, *The effects of inexact solvers in algorithms for symmetric eigenvalue problems*, Linear Algebra and its Applications, 287 (1999), pp. 337–357.
15. H. Symm and J. Wilkinson, *Realistic error bounds for a simple eigenvalue and its associated eigenvector*, Numerische Mathematik, 35 (1980), pp. 113–126.