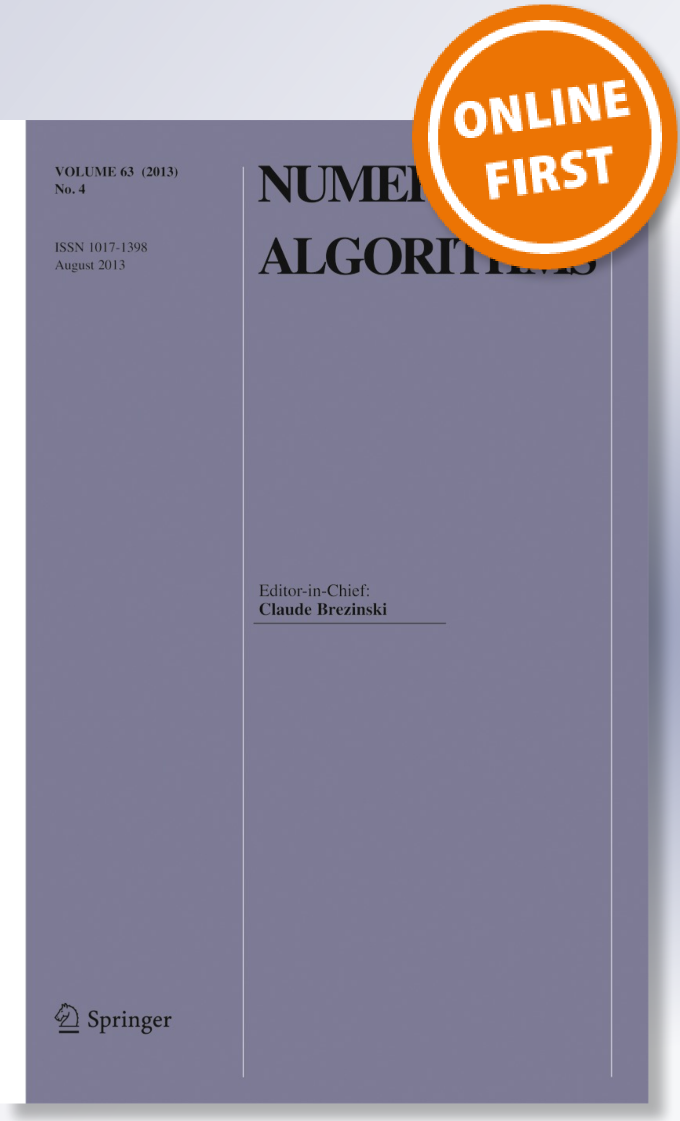
Numerical Algorithms

ISSN 1017-1398

Numer Algor
DOI 10.1007/s11075-013-9750-7



VOLUME 63 (2013)
No. 4

ISSN 1017-1398
August 2013

NUME
ALGORIT

ONLINE FIRST

Editor-in-Chief:
Claude Brezinski

Springer

Springer

Numer Algor
DOI 10.1007/s11075-013-9750-7

ORIGINAL PAPER

Chebyshev acceleration of iterative refinement

M. Arioli · J. Scott

Received: 4 April 2013 / Accepted: 26 July 2013

Abstract It is well known that the FGMRES algorithm can be used as an alternative to iterative refinement and, in some instances, is successful in computing a backward stable solution when iterative refinement fails to converge. In this study, we analyse how variants of the Chebyshev algorithm can also be used to accelerate iterative refinement without loss of numerical stability and at a computational cost at each iteration that is less than that of FGMRES and only marginally greater than that of iterative refinement. A component-wise error analysis of the procedure is presented and numerical tests on selected sparse test problems are used to corroborate the theory.

Keywords Chebyshev method · Iterative refinement · FGMRES · Gaussian elimination · Sparse matrices

Mathematics Subject Classification (2010) 65F05 · 65F50 · 65F10 · 65G50

1 Introduction

The combination of Gaussian elimination with partial pivoting followed by a few steps of iterative refinement can compute an approximate solution of a linear system of equations that is backward stable, i.e. the residual norm is less than or equal to

The work was supported by the EPSRC Grant EP/E053351/1

M. Arioli (✉) · J. Scott
Rutherford Appleton Laboratory, Chilton, Didcot, Oxon,
OX11 0QX, United Kingdom
e-mail: mario.arioli@stfc.ac.uk

J. Scott
e-mail: jennifer.scott@stfc.ac.uk

Published online: 10 August 2013

machine precision times the norm of the data. More precisely, iterative refinement produces a computed solution that is component-wise backward stable [1, 18], i.e. the computed solution is the exact solution of a perturbed system where the non zero entries of the original matrix are perturbed by a relative error bounded by machine precision. However, when threshold partial pivoting or static pivoting are used to limit fill-in in the factorization of large sparse systems, the number of iterative refinement steps needed to achieve the required accuracy can be large. Furthermore, when the factorization is computed in single precision and then a double precision backward stable approximate solution is recovered using iterative refinement, the number of refinement steps can also be very large and the cost prohibitive [11]. It is particularly important to limit the number of refinement steps on modern multicore architectures where the solve phase of a sparse direct solver can represent a potential bottleneck (see, for example, [12]).

We are seeking alternatives to iterative refinement that preserve two of its key properties:

- component-wise stability, and
- the absence of scalar products.

In a parallel environment, having no (or only a small number of) scalar products is important to limit the amount of communication needed between processors. The FGMRES algorithm [17] has been proposed in [2, 3] as an alternative to iterative refinement. For some problems, it is able to compute backward stable solutions when iterative refinement fails to converge. However, FGMRES does not preserve either of the above properties: FGMRES guarantees rapid convergence but it is only possible to prove that it is norm-wise stable, and its implementation involves scalar products.

Following the results of [7–9, 14–16], in this paper we consider variants of the Chebyshev algorithm and show that they can be used to accelerate the iterative refinement procedure while maintaining component-wise stability and at a computational cost for each iteration that is only marginally greater than that of the iterative refinement.

In the rest of this section, we introduce the notation that we will use throughout and then summarize iterative refinement and its properties and potential weakness. In Section 2, we describe two Chebyshev acceleration algorithm variants. We present an error analysis of the main algorithm in Section 3 and in Section 4 we discuss how to automatically choose some of the parameters. In Section 5, we present numerical results for sparse systems that arise from practical applications that corroborate the theoretical results of the previous sections and, in Section 6, we give some final comments. In the following, $||\cdot||$ will denote the Euclidean norm for $\mathbb{R}^n$ and the corresponding induced norm for the matrices.

Let $\mathbf{A} \in \mathbb{R}^{n\times n}$ and $b \in \mathbb{R}^n$, with Rank $(\mathbf{A}) = n$. The linear system

$$\mathbf{Ax} = \mathbf{b} \tag{1}$$

has a unique solution $\hat{\mathbf{x}}$. We assume Gaussian elimination is performed using floating-point arithmetic with relative precision ϵ. Thus, the computed factors $\widehat{\mathbf{L}}$ and $\widehat{\mathbf{U}}$ satisfy the relation [6, 10]

$$\mathbf{A} + \mathbf{F} = \widehat{\mathbf{L}}\widehat{\mathbf{U}} = \mathbf{M}, \tag{2}$$

where $\mathbf{F} \in \mathbb{R}^{n\times n}$ and

$$|\mathbf{F}| \leq c(n)\varepsilon|\widehat{\mathbf{L}}|\ |\widehat{\mathbf{U}}|, \tag{3}$$

with $c(n)$ a function of n (in practice $c(n)\varepsilon \approx \mathcal{O}(n\varepsilon)$). Hereafter, we assume that the perturbation $\mathbf{F}$ is sufficiently small such that the matrix $\mathbf{M}$ is non singular. Here and elsewhere, $|\mathbf{B}|$ denotes the matrix of entries equal to the absolute values of the corresponding entries in the matrix $\mathbf{B}$. From (2), it follows that

$$\mathbf{x} = \mathbf{M}^{-1}(\mathbf{b} + \mathbf{F}\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x} + \mathbf{M}\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{r}(\mathbf{x}) + \mathbf{M}\mathbf{x}), \tag{4}$$

where $\mathbf{r}(\mathbf{x})$ is the residual $\mathbf{b} - \mathbf{A}\mathbf{x}$. Thus, $\mathbf{x}$ is the fixed point of $\mathfrak{F}(\mathbf{x})$, where

$$\mathfrak{F}(\mathbf{x}) = \mathbf{x} + \mathbf{M}^{-1}\mathbf{r}(\mathbf{x}).$$

If β denotes the scaled component-wise residual

$$\beta = \max_i \frac{|\mathbf{r}(\mathbf{x})|_i}{(|\mathbf{A}||\mathbf{x}| + |\mathbf{b}|)_i}, \tag{5}$$

where we assume that $\frac{0}{0} = 0$ owing to $|\mathbf{r}(\mathbf{x})| \leq |\mathbf{A}||\mathbf{x}| + |\mathbf{b}|$, then given a convergence tolerance $\eta > 0$, it is straightforward to write down the basic algorithm for iterative refinement:

Algorithm 1 Iterative refinement

```
Let $\mathbf{x}^{(0)} = \mathbf{M}^{-1}\mathbf{b}$ and $\mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(0)}$.

Initialise $k = 0$.
while $\beta^{(k)} > \eta$ do
    $\delta\mathbf{x} = \mathbf{M}^{-1}\mathbf{r}^{(k)}$;
    $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \delta\mathbf{x}$;
    $\mathbf{r}^{(k+1)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(k+1)}$;
    $\beta^{(k+1)} = \max_i |\mathbf{r}_i^{(k+1)}|/(|\mathbf{A}|\ |\mathbf{x}^{(k+1)}| + |\mathbf{b}|)_i$;
    $k = k + 1$.
end while
```

Note that if the scaled norm-wise residual is used as the stopping criteria, (5) is replaced by

$$\beta_n = \frac{\|\mathbf{r}(\mathbf{x})\|}{\|\mathbf{A}\|\,\|\mathbf{x}\| + \|\mathbf{b}\|}. \tag{6}$$

If the spectral radius $\sigma(\mathbf{M}^{-1}\mathbf{F})$ of $\mathbf{M}^{-1}\mathbf{F}$ satisfies

$$\sigma(\mathbf{M}^{-1}\mathbf{F}) < 1$$

in exact arithmetic, Algorithm 1 produces a sequence $\mathbf{x}^{(k)}$ that converges to $\hat{\mathbf{x}}$. Furthermore, from (4),

$$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} - \mathbf{A}\mathbf{M}^{-1}\mathbf{r}^{(k)} = \mathbf{F}\mathbf{M}^{-1}\mathbf{r}^{(k)}. \tag{7}$$

Therefore, if $\sigma(\mathbf{F}\mathbf{M}^{-1}) < 1$ the residuals converge to zero in exact arithmetic.

Remark 1 In [1, sec. 2.2, page 169], the case of a sparse exact solution $\mathbf{x}$ and a sparse right-hand side $\mathbf{b}$ is discussed . In this case, for some values of the index i the ratio

$$\frac{|\mathbf{r}(\mathbf{x})|_i}{(|\mathbf{A}||\mathbf{x}| + |\mathbf{b}|)_i},$$

can be close to one with both $|\mathbf{r}(\mathbf{x})|_i$ and $(|\mathbf{A}||\mathbf{x}| + |\mathbf{b}|)_i$ non zero but very small because of the presence of roundoff errors and in absence of exact cancellation. The technique used in [1] to cope with this situation can also be used during the Chebyshev acceleration we describe in the next section. We omit this to simplify the discussion but it is straightforward to modify the algorithms using [1].

Remark 2 In the following, we will use interchangeably both $\sigma(\mathbf{M}^{-1}\mathbf{F})$ and $\sigma(\mathbf{F}\mathbf{M}^{-1})$ since $\mathbf{M}^{-1}\mathbf{F}$ and $\mathbf{F}\mathbf{M}^{-1}$ have the same spectrum.

Remark 3 An alternative formulation for (1) based on relation (4) is

$$\mathbf{x} = \mathbf{M}^{-1}\mathbf{F}\mathbf{x} + \mathbf{M}^{-1}\mathbf{b}. \tag{8}$$

This is not of practical use, but it will be useful when developing the theory of Chebyshev acceleration that follows in Section 2.

Remark 4 Assume that $\sigma(\mathbf{M}^{-1}\mathbf{F}) = 0.5$. To achieve a reduction of three orders of magnitude in the initial residual, the required number of steps of iteration refinement is

$$iter = \left\lceil \frac{log_{10}(10^{-3})}{log_{10}(0.5)} \right\rceil,$$

which is approximatively 10. The cost of performing this number of iterations may be unacceptably high, for example, if the factors are held out-of-core. In the next section, we propose a variant of Chebyshev acceleration that may improve the rate of convergence.

2 Chebyshev acceleration

Chebyshev polynomials can be defined by the following 3-term recurrence formula (see [9, page 46]):

$$\begin{cases} T_0(z) = 1\,, \quad T_1(z) = z \\ T_{k+1}(z) = 2zT_k(z) - T_{k-1}(z) \qquad k \geq 1. \end{cases}$$

The optimal properties of Chebyshev polynomials given in Theorem 4.2.1 of [9, page 47] can be summarised as follows: let $d > 1$ and set

$$\mathcal{F}_k(z) = \frac{T_k(z)}{T_k(d)},$$

then $\mathcal{F}_k$ has minimum l_∞ norm on the interval $[-1, 1]$ over all polynomials Q_k of degree less than or equal to n and satisfying the condition $Q_k(d) = 1$, and

$$\max_{z\in[-1,1]} |\mathcal{F}_k(z)| = \frac{1}{T_k(d)}.$$

We now summarize some classical results. We refer the reader to [9, Chapters 4 and 12] and [14, 15] for further details. If all the eigenvalues of $\mathbf{M}^{-1}\mathbf{F}$ lie in the interior of an ellipse that is centred at the origin, symmetric with respect to the real axis (the matrix is real so the eigenvalues are either real or complex conjugate pairs) and has principal semi-axes a and b, i.e.

$$\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1, \tag{9}$$

then the following theorem holds (see [9, Theorem 12-2.1]).

Theorem 1 *Let $\mathcal{D}$ be the region enclosed by (9) where $0 < b < a < 1$. If $\mathcal{S}_j$ is the set of all real polynomials $p_j(z)$ of degree at most j such that $p_j(1) = 1$, then the polynomial*

$$\wp_j(z) = \frac{T_j(z/c)}{T_j(1/c)}, \qquad \textit{where } \; c^2 = a^2 - b^2,$$

is the unique polynomial in the set $\mathcal{S}_j$ such that

$$\max_{z\in\mathcal{D}} |\wp_j(z)| \le \max_{z\in\mathcal{D}} |p_j(z)|, \qquad p_j(z) \in \mathcal{S}_j.$$

Manteuffel [14] showed that this result cannot be extended to the case $0 < a < b < 1$. In this case, c is purely imaginary. However, the $\wp_j(z)$ are still real and the following weaker result can be proved [14]:

$$\lim_{j\to\infty} \left(\max_{z\in\mathcal{D}} |\wp_j(z)|\right)^{1/j} \le \lim_{j\to\infty} \left(\max_{z\in\mathcal{D}} |p_j(z)|\right)^{1/j}, \qquad p_j(z) \in \mathcal{S}_j. \tag{10}$$

From formula (10), we have that the polynomials $\wp_j(z)$ are asymptotically optimal and, furthermore, it has been noted [9, 14] that the asymptotic behaviour is very rapidly reached.

Following along the lines of [9], we now describe the Chebyshev acceleration algorithm. The polynomials $\wp_j(z)$ are defined as follows:

$$\begin{cases} \wp_0 = 1, \qquad \wp_1 = z \\ \wp_{j+1}(z) = \varrho_{j+1} z \wp_j(z) + (1 - \varrho_{j+1})\wp_{j-1}(z) \\ \varrho_{j+1} = \frac{2}{c}\frac{T_j(1/c)}{T_{j+1}(1/c)} \end{cases}$$

The Chebyshev relations for problem (8) are then given by

$$\begin{cases} \mathbf{x}^{(-1)} = 0, \quad \mathbf{x}^{(0)} = \mathbf{M}^{-1}\mathbf{b}, \quad \varrho_1 = 1, \\ \mathbf{x}^{(j+1)} = \varrho_{j+1}\left(\mathbf{M}^{-1}\mathbf{F}\mathbf{x}^{(j)} + \mathbf{M}^{-1}\mathbf{b}\right) + (1 - \varrho_{j+1})\mathbf{x}^{(j-1)}, \quad j = 0, \ldots, . \end{cases}$$

Using (4), this can be simplified:

$$\begin{cases} \mathbf{x}^{(-1)} = 0, \quad \mathbf{x}^{(0)} = \mathbf{M}^{-1}\mathbf{b}, \quad \varrho_1 = 1, \\ \mathbf{x}^{(j+1)} = \varrho_{j+1}\left(\mathbf{M}^{-1}\mathbf{r}(\mathbf{x}^{(j)}) + \mathbf{x}^{(j)}\right) + (1-\varrho_{j+1})\mathbf{x}^{(j-1)}, \quad j = 0, \ldots, . \end{cases} \tag{11}$$

We observe that computing the ϱ_j is straightforward:

$$\varrho_{j+1} = \begin{cases} 1, & \text{if } j = 0, \\ \left(1 - \frac{1}{2}c^2\right)^{-1}, & \text{if } j = 1, \\ \left(1 - \frac{1}{4}c^2\varrho_j\right)^{-1}, & \text{if } j \geq 2\,. \end{cases} \tag{12}$$

From the maximum modulus principle and the analyticity of $\wp_j$, $\wp_j(z)$ will take its maximum on the ellipse (9). Moreover, we have [9] that

$$\max_{z\in\mathcal{D}} |\wp_j(z)| = \left[\frac{a+b}{1+\sqrt{1-c^2}}\right]^j. \tag{13}$$

Finally, assuming the ϱ_j have been precomputed, simple algebraic manipulations leads to the following algorithm:

Algorithm 2 Chebyshev acceleration of iterative refinement

```
Let $\mathbf{x}^{(-1)} = 0$, $\mathbf{x}^{(0)} = \mathbf{M}^{-1}\mathbf{b}$, $\mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(0)}$.
Initialise $k = 0$.

while $\beta^{(k)} > \eta$ do
    $\mathbf{w}^{(k)} = \mathbf{x}^{(k)} + \mathbf{M}^{-1}\mathbf{r}^{(k)}$;
    $\mathbf{x}^{(k+1)} = \varrho_{k+1}\mathbf{w}^{(k)} + (1-\varrho_{k+1})\mathbf{x}^{(k-1)}$;
    $\mathbf{r}^{(k+1)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(k+1)}$;
    $\beta^{(k+1)} = \max_i |\mathbf{r}_i^{(k+1)}|/(|\mathbf{A}|\,|\mathbf{x}^{(k+1)}| + |\mathbf{b}|)_i$;
    $k = k + 1$.
end while
```

We have followed the analysis and the evidence given in [7, 8, 16] and have chosen to compute the residuals explicitly. Recursive expressions can easily be computed but they can be less stable [7].

Remark 5 For the successful convergence of Algorithm 2 it is necessary to predict the equation of an ellipse that envelops the whole spectrum of $\mathbf{M}^{-1}\mathbf{F}$. If $\sigma(\mathbf{M}^{-1}\mathbf{F})$ lies outside the chosen ellipse and $c^2 \ll 1$ or the ellipse degenerates to a circle ($a = b$), the asymptotic behaviour of Algorithm 2 will be the same as that of iterative refinement and thus it will give no acceleration. In the first case, $\varrho_\infty = \lim_{j\to\infty}\varrho_j = 2/(1+\sqrt{1-c^2}) \approx 1$, while in the case $a = b$, $\varrho_j = 1\ \forall j$.

Remark 6 In Remark 4, we observed that if $\sigma(\mathbf{M}^{-1}\mathbf{F}) = 0.5$, iterative refinement requires approximately 10 steps to reduce the initial residual by three orders of magnitude. The asymptotic rate of convergence, i.e. the logarithm of the right-hand side of (13), describes the number of steps that Algorithm 2 needs to reduce the residual by one order of magnitude. The number of steps required to reduce it by p orders of magnitude is

$$iter = \left\lceil \frac{log_{10}(10^{-p})}{log_{10}\left(\frac{a+b}{1+\sqrt{1-c^2}}\right)} \right\rceil . \tag{14}$$

If, in our example, the ellipse

$$\left(\frac{x}{0.5}\right)^2 + \left(\frac{y}{0.05}\right)^2 = 1$$

contains all the eigenvalues, the Chebyshev accelerated algorithm will need approximately 6 steps to obtain a reduction of three orders of magnitude. This illustrates the potential savings offered by Algorithm 2.

We can also introduce a simple variant of Algorithm 2, based on (11):

Algorithm 3 Chebyshev acceleration of iterative refinement

```
Let $\Delta\mathbf{x}^{(0)} = 0$, $\mathbf{x}^{(0)} = \mathbf{M}^{-1}\mathbf{b}$, $\mathbf{r}^{(0)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(0)}$.
Initialise $k = 0$.

while $\beta^{(k)} > \eta$ do
    $\Delta\mathbf{x}^{(k+1)} = \varrho_{k+1}\mathbf{M}^{-1}\mathbf{r}^{(k)} - (1 - \varrho_{k+1})\Delta\mathbf{x}^{(k)}$;
    $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \Delta\mathbf{x}^{(k+1)}$;
    $\mathbf{r}^{(k+1)} = \mathbf{b} - \mathbf{A}\mathbf{x}^{(k+1)}$;
    $\beta^{(k+1)} = \max_i |\mathbf{r}_i^{(k+1)}|/(|\mathbf{A}|\ |\mathbf{x}^{(k+1)}| + |\mathbf{b}|)_i$;
    $k = k + 1$.
end while
```

This variant is slightly more awkward to analyse from a roundoff point of view. However, the numerical results do not differ significantly from those obtained using Algorithm 2, which we analyse in the next section.

3 Iterative refinement and Chebyshev error analysis

We assume finite precision arithmetic with relative precision ε is used, i.e. the arithmetic operations $\Diamond \in \{+, -, *, /\}$ satisfy

$$fl\,(g\Diamond r) = (1+\xi)g\Diamond r, \qquad |\xi| \leq \varepsilon\,,$$

where with $fl(\mathbf{.})$ denotes the actual results in finite precision. Taking into account formulae (12), we assume that the ϱ_j values have been computed using extended precision and that they are the correct rounded results to ε accuracy. The cost of this extended precision computations is negligible when we compare it with the other computations. From the formulae in Algorithm 2 and using standard techniques [10], we have that the computed values $\bar{\mathbf{r}}^{(k)}$ of $\mathbf{r}^{(k)}$ satisfy the relations

$$\left.\begin{array}{l}\bar{\mathbf{r}}^{(k)} = \left(\mathbf{I}+\Gamma_0^{(k)}\right)\left(\mathbf{b}-\left(\mathbf{A}+\mathbf{G}^{(k)}\right)\bar{\mathbf{x}}^{(k)}\right),\\ |\mathbf{G}^{(k)}| \le c_1\varepsilon\,|\mathbf{A}| \ll 1 \;\text{ and }\; |\Gamma_0^{(k)}| \le \varepsilon\,\mathbf{I},\end{array}\right\} \tag{15}$$

where $c_1 = \mathcal{O}(\nu)$ with ν the max number of non zero entries in a row of A, i.e. $\nu \le n$ and frequently $\nu << n$. Furthermore, from (15)

$$\left.\begin{array}{l}\bar{\mathbf{r}}^{(k)} = \mathbf{r}^{(k)} + \mathbf{g}^{(k)},\\ \mathbf{g}^{(k)} = \Gamma_0^{(k)}\mathbf{r}^{(k)} - \left(\mathbf{I}+\boldsymbol{\Gamma}_0^{(k)}\right)\mathbf{G}^{(k)}\bar{\mathbf{x}}^{(k)},\\ |\mathbf{g}^{(k)}| \le (2+c_1)\varepsilon\,\left(|\mathbf{b}|+|\mathbf{A}|\,|\bar{\mathbf{x}}^{(k)}|\right).\end{array}\right\} \tag{16}$$

In Algorithm 2, the linear system

$$\mathbf{M}\mathbf{z}^{(k)} = \bar{\mathbf{r}}^{(k)} \tag{17}$$

must be solved. Taking into account the properties of forward and backward substitutions, the computed solution $\bar{\mathbf{z}}^{(k)}$satisfies

$$(\mathbf{M}+\mathbf{E}_k)\,\bar{\mathbf{z}}^{(k)} = \bar{\mathbf{r}}^{(k)}, \qquad |\mathbf{E}_k| \le c_0(n)\varepsilon\,|\widehat{\mathbf{L}}||\widehat{\mathbf{U}}|, \quad ||\mathbf{E}_k\mathbf{M}^{-1}|| < 1,$$

so that $\mathbf{M}+\mathbf{E}_k$ is non singular for all k. Setting $\widetilde{\mathbf{M}}_k = \mathbf{M}+\mathbf{E}_k$, and denoting by $\bar{\mathbf{x}}^{(k)}$ and $\bar{\mathbf{w}}^{(k)}$ the computed values of $\mathbf{x}^{(k)}$ and $\mathbf{w}^{(k)}$, we have

$$\left.\begin{array}{l}\bar{\mathbf{w}}^{(k+1)} = \left(\mathbf{I}+\boldsymbol{\Gamma}_1^{(k)}\right)\left(\widetilde{\mathbf{M}}_k^{-1}\bar{\mathbf{r}}^{(k)}+\bar{\mathbf{x}}^{(k)}\right),\\ |\Gamma_1^{(k)}| \le \varepsilon\,\mathbf{I},\end{array}\right\} \tag{18}$$

and, finally,

$$\left.\begin{array}{l}\bar{\mathbf{x}}^{(k+1)} = \left(\mathbf{I}+\boldsymbol{\Gamma}_4^{(k)}\right)\left[\varrho_{k+1}\left(\mathbf{I}+\boldsymbol{\Gamma}_2^{(k)}\right)\bar{\mathbf{w}}^{(k)}+\right.\\ \qquad\qquad \left.(1-\varrho_{k+1})\left(\mathbf{I}+\boldsymbol{\Gamma}_3^{(k)}\right)\,\bar{\mathbf{x}}^{(k-1)}\right],\\ |\boldsymbol{\Gamma}_i^{(k)}| \le \varepsilon\,\mathbf{I}, \quad i=2,3,4.\end{array}\right\} \tag{19}$$

From (18) and (19), we deduce that

$$\begin{aligned}\bar{\mathbf{x}}^{(k+1)} &= \varrho_{k+1}\left(\mathbf{I}+\widehat{\boldsymbol{\Gamma}}_1^{(k)}\right)\bar{\mathbf{w}}^{(k)} + (1-\varrho_{k+1})\left(\mathbf{I}+\widehat{\boldsymbol{\Gamma}}_2^{(k)}\right)\,\bar{\mathbf{x}}^{(k-1)} &(20)\\ &= \varrho_{k+1}\left(\mathbf{I}+\widehat{\boldsymbol{\Gamma}}_3^{(k)}\right)\left(\widetilde{\mathbf{M}}_k^{-1}\bar{\mathbf{r}}^{(k)}+\bar{\mathbf{x}}^{(k)}\right)\\ &\quad +(1-\varrho_{k+1})\left(\mathbf{I}+\widehat{\boldsymbol{\Gamma}}_2^{(k)}\right)\,\bar{\mathbf{x}}^{(k-1)}, &(21)\end{aligned}$$

with $|\widehat{\boldsymbol{\Gamma}}_i^{(k)}| \lesssim 3\varepsilon\,\mathbf{I}$ for all k and $i = 1,2,3$. Although they are uniformly bounded, the $\widehat{\boldsymbol{\Gamma}}_i^{(k)}$ and $\mathbf{E}_k$ depend non-linearly on $\bar{\mathbf{w}}^{(k)}$ and $\bar{\mathbf{x}}^{(k)}$. Furthermore, from (16), (20) and (21), the exact residual $\mathbf{r}^{(j)} = \mathbf{b} - \mathbf{A}\bar{\mathbf{x}}^{(j)}$ satisfies

$$\begin{aligned}
\mathbf{r}^{(k+1)} &= \varrho_{k+1}\left[\mathbf{b} - \mathbf{A}\left(\mathbf{I} + \widehat{\boldsymbol{\Gamma}}_3^{(k)}\right)\left(\widetilde{\mathbf{M}}_k^{-1}\bar{\mathbf{r}}^{(k)} + \bar{\mathbf{x}}^{(k)}\right)\right] \\
&\quad +(1-\varrho_{k+1})\left[\mathbf{b} - \mathbf{A}\left(\mathbf{I} + \widehat{\boldsymbol{\Gamma}}_2^{(k)}\right)\ \bar{\mathbf{x}}^{(k-1)}\right] \\
&= \varrho_{k+1}\left[\mathbf{r}^{(k)} - \mathbf{A}\left(\mathbf{I} + \widehat{\boldsymbol{\Gamma}}_3^{(k)}\right)\widetilde{\mathbf{M}}_k^{-1}\bar{\mathbf{r}}^{(k)} - \mathbf{A}\widehat{\boldsymbol{\Gamma}}_3^{(k)}\bar{\mathbf{x}}^{(k)}\right)\right] \\
&\quad +(1-\varrho_{k+1})\left[\mathbf{r}^{(k-1)} - \mathbf{A}\widehat{\boldsymbol{\Gamma}}_2^{(k)}\bar{\mathbf{x}}^{(k-1)}\right] \\
&= \varrho_{k+1}\left[\mathbf{I} - \mathbf{A}\left(\mathbf{I} + \widehat{\boldsymbol{\Gamma}}_3^{(k)}\right)\widetilde{\mathbf{M}}_k^{-1}\right]\mathbf{r}^{(k)} + (1-\varrho_{k+1})\mathbf{r}^{(k-1)} \\
&\quad -\varrho_{k+1}\left(\mathbf{g}^{(k)} + \mathbf{A}\widehat{\boldsymbol{\Gamma}}_3^{(k)}\bar{\mathbf{x}}^{(k)}\right) - (1-\varrho_{k+1})\mathbf{A}\widehat{\boldsymbol{\Gamma}}_2^{(k)}\bar{\mathbf{x}}^{(k-1)}.
\end{aligned}$$

Therefore, we have the following recursive expression

$$\mathbf{r}^{(k+1)} = \varrho_{k+1}\mathbf{H}_k\mathbf{r}^{(k)} + (1-\varrho_{k+1})\mathbf{r}^{(k-1)} + \mathbf{f}^{(k+1)}, \tag{22}$$

where

$$\begin{aligned}
\mathbf{H}_k &= \mathbf{I} - \mathbf{A}\left(\mathbf{I} + \widehat{\boldsymbol{\Gamma}}_3^{(k)}\right)\widetilde{\mathbf{M}}_k^{-1}, \\
\mathbf{f}^{(k+1)} &= -\varrho_{k+1}\left(\mathbf{g}^{(k)} + \mathbf{A}\widehat{\boldsymbol{\Gamma}}_3^{(k)}\bar{\mathbf{x}}^{(k)}\right) - (1-\varrho_{k+1})\mathbf{A}\widehat{\boldsymbol{\Gamma}}_2^{(k)}\bar{\mathbf{x}}^{(k-1)},
\end{aligned}$$

and, from the bounds in (16), (18), and (19), it follows that

$$|\mathbf{f}^{(k+1)}| < 3\varepsilon\,|1-\varrho_{k+1}|\ |\mathbf{A}||\bar{\mathbf{x}}^{(k-1)}| + (5+c_1)\varepsilon\,\varrho_{k+1}\left(|\mathbf{A}||\bar{\mathbf{x}}^{(k)}| + |\mathbf{b}|\right).$$

3.1 Analysis of $\mathbf{H}_k$

We assume that numerical exceptions (overflows or underflows) do not occur during the execution of Algorithm 2. This is a necessary condition for continuous dependence of the errors on the data. Moreover, we assumed that

$$||\mathbf{E}_k\mathbf{M}^{-1}|| < 1. \tag{23}$$

With this assumption, $\widetilde{\mathbf{M}}_k$ is nonsingular and

$$\widetilde{\mathbf{M}}_k^{-1} = \mathbf{M}^{-1}\left(\mathbf{I} + \mathbf{E}_k\mathbf{M}^{-1}\right)^{-1} = \mathbf{M}^{-1}\left(\mathbf{I} - \mathbf{E}_k\mathbf{M}^{-1}\left(\mathbf{I} + \mathbf{E}_k\mathbf{M}^{-1}\right)^{-1}\right). \tag{24}$$

From (2), (3), (23), and (24), it follows that

$$\begin{aligned}
\mathbf{H}_k &= \mathbf{I} - \mathbf{A}\left(\mathbf{I} + \widehat{\boldsymbol{\Gamma}}_3^{(k)}\right)\mathbf{M}^{-1}\left(\mathbf{I} - \mathbf{E}_k\widetilde{\mathbf{M}_k}^{-1}\right) \\
&= \mathbf{I} - (\mathbf{M} - \mathbf{F})\,\mathbf{M}^{-1}\left(\mathbf{I} + \mathbf{M}\widehat{\boldsymbol{\Gamma}}_3^{(k)}\mathbf{M}^{-1}\right)\left(\mathbf{I} - \mathbf{E}_k\widetilde{\mathbf{M}_k}^{-1}\right) \\
&= \mathbf{F}\mathbf{M}^{-1} - \mathbf{M}\widehat{\boldsymbol{\Gamma}}_3^{(k)}\mathbf{M}^{-1} + \mathbf{E}_k\widetilde{\mathbf{M}_k}^{-1} + \mathcal{E},
\end{aligned}$$

where $\mathcal{E} = \mathcal{O}(\varepsilon^2)$. Thus, if for each k the matrices $\mathbf{F}\mathbf{M}^{-1}$, $\mathbf{M}\widehat{\boldsymbol{\Gamma}}_3^{(k)}\mathbf{M}^{-1}$, and $\mathbf{E}_k\widetilde{\mathbf{M}_k}^{-1}$ have Euclidean norm strictly less than 0.25, $||\mathbf{H}_k|| < 1$ and $\mathbf{I} - \mathbf{H}_k$ and $\mathbf{I} + \mathbf{H}_k$ are invertible.

3.2 Error bounds

Taking into account the previous results, and if we assume $\mathbf{H}_k$ and $\mathbf{f}^{(k)}$ depend continuously on the data, (21) has a fixed point on a large enough compact convex set of $\mathbb{R}^n$ (Generalized Brouwer Fixed Point Theorem [4, Theorem 3.2]). Assuming in finite precision arithmetic Algorithm 2 computes a sequence $\bar{\mathbf{x}}^{(k)}$ that converges to the point $\bar{\mathbf{x}}^{\infty}$, then the residuals $\mathbf{r}^{(k)}$ converge to $\mathbf{r}^{\infty}$. From (22) and taking into account that ϱ_k converges to a finite $\varrho_\infty = 2/(1+\sqrt{1-c^2})$, we have

$$\mathbf{r}^{\infty} = \varrho_\infty \mathbf{H}_\infty \mathbf{r}^{\infty} + (1-\varrho_\infty)\mathbf{r}^{\infty} + \mathbf{f}^{\infty}, \tag{25}$$

where

$$|\mathbf{f}^{\infty}| \leq 3\varepsilon\, |1-\varrho_\infty|\; |\mathbf{A}||\bar{\mathbf{x}}^{\infty}| + (5+c_1)\varepsilon\, \varrho_\infty \left(|\mathbf{A}||\bar{\mathbf{x}}^{\infty}| + |\mathbf{b}|\right).$$

From (25) it follows that

$$\varrho_\infty \left(\mathbf{I} - \mathbf{H}_\infty\right) \mathbf{r}^{\infty} = -\mathbf{f}^{\infty}$$

and thus

$$\mathbf{r}^{\infty} = -\frac{1}{\varrho_\infty} \left(\mathbf{I} - \mathbf{H}_\infty\right)^{-1} \mathbf{f}^{\infty}.$$

If $\sigma(\mathbf{H}_\infty) \ll 1$, we have

$$\begin{aligned} |\mathbf{r}^{\infty}| &\leq \tfrac{1}{\varrho_\infty} 3\varepsilon\, |1-\varrho_\infty|\; |\mathbf{A}||\bar{\mathbf{x}}^{\infty}| + (5+c_1)\varepsilon\; (|\mathbf{A}||\bar{\mathbf{x}}^{\infty}| + |\mathbf{b}|) + \mathcal{O}(\varepsilon^{2}) \\ &\leq (8+c_1)\varepsilon\; (|\mathbf{A}||\bar{\mathbf{x}}^{\infty}| + |\mathbf{b}|) + \mathcal{O}(\varepsilon^{2}). \end{aligned}$$

Thus if the sequence computed by Algorithm 2 converges, there exists k^* such that $\forall k > k^*$

$$|\mathbf{r}^{(k)}| \leq (8+c_1)\varepsilon\; \left(|\mathbf{A}||\bar{\mathbf{x}}^{(k)}| + |\mathbf{b}|\right) + \mathcal{O}(\varepsilon^{2}).$$

However, if $\sigma(\mathbf{H}_\infty) < 1$ the norm of $\mathbf{r}^{\infty}$ can be bounded by

$$||\mathbf{r}^{\infty}|| \leq \frac{(8+c_1)\varepsilon}{1-||\mathbf{H}_\infty||} \left(||\mathbf{A}||\; ||\bar{\mathbf{x}}^{\infty}|| + ||\mathbf{b}||\right) + \mathcal{O}(\varepsilon^{2}).$$

Hence $\forall k > k^*$

$$||\mathbf{r}^{(k)}|| \leq \frac{(8+c_1)\varepsilon}{1-||\mathbf{H}_k||} \left(||\mathbf{A}||\; ||\bar{\mathbf{x}}^{(k)}|| + ||\mathbf{b}||\right) + \mathcal{O}(\varepsilon^{2}). \tag{26}$$

Remark 7 If Algorithm 2 converges and $||\mathbf{H}_\infty|| \ll 1$, it converges to the solution of a linear system that is a perturbation of the original system (1). Therefore, if we choose $\eta = (8+c_1)\varepsilon\left(|\mathbf{A}||\bar{\mathbf{x}}^{(k)}| + |\mathbf{b}|\right)$, the computation will terminate with a vector $\bar{\mathbf{x}}$ that is the solution of

$$\begin{aligned} &(\mathbf{A} + \delta\mathbf{A})\, \bar{\mathbf{x}} = \mathbf{b} + \delta\mathbf{b} \\ &|\delta\mathbf{A}| \leq (8+c_1)\varepsilon\, |\mathbf{A}|, \quad |\delta\mathbf{b}| \leq (8+c_1)\varepsilon\, |\mathbf{b}|. \end{aligned}$$

Remark 8 If mixed precision is used in Algorithm 2 (the factorization is computed using arithmetic of relative precision ε_1 and all the other operations are performed using arithmetic of relative precision $\varepsilon_2 = \varepsilon_1^2$) then, with

$$\eta = \frac{9\varepsilon_2}{1 - ||\mathbf{H}_k||} \left(|\mathbf{A}||\bar{\mathbf{x}}^{(k^*)}| + |\mathbf{b}| \right)$$

and provided the condition numbers of **A** and **M** are less than ε_1^{-1}, the computation will terminate with a vector $\bar{\mathbf{x}}$ that satisfies

$$||\bar{\mathbf{x}} - \mathbf{x}|| \leq \frac{9\varepsilon_1 ||\mathbf{x}||}{1 - ||\mathbf{H}_k||} + \mathcal{O}(\varepsilon_2).$$

3.3 Best achievable accuracy

The inequality (26) can give the false impression that we can always achieve convergence after a finite number of steps to a residual lying within the ball of radius ε. When $\sigma(\mathbf{H}_\infty)$ is very close to 1, since $\sigma(\mathbf{H}_k) < ||\mathbf{H}_k||$ the ratios

$$\frac{\varepsilon}{1 - ||\mathbf{H}_k||}$$

will increase. Thus the 'best' achievable accuracy $\omega(\varepsilon)$ will be

$$\omega(\varepsilon) = \frac{\varepsilon}{1 - ||\mathbf{H}_\infty||}.$$

In practice, for $\sigma(\mathbf{H}_k)$ less than about 0.9, $\omega(\varepsilon) \approx \varepsilon$. However, for $\sigma(\mathbf{H}_k)$ greater than about 0.99, we see that $\omega(\varepsilon)$ can be much larger than ε.

Remark 9 Throughout Section 3, we have assumed that **M** is the computed *LU*-factorization of **A** and the classical backward and forward substitution algorithm has been used in solving the corresponding triangular systems. We point out that it is straightforward to adapt the analysis to the general case where **M** is a generic preconditioner such that the solution of system (17) is backward stable, provided the condition

$$\sigma(\mathbf{M}^{-1}\mathbf{F}) < 1$$

holds, where $\mathbf{F} = \mathbf{A} - \mathbf{M}$ is no longer assumed to be of order ε.

4 How to choose the ellipse

In Remark 6 in Section 2, we illustrated how a simple choice of the ellipse in Chebyshev acceleration can significantly reduce the number of iterations required for convergence. As noted in the Introduction, we wish to limit the number of scalar products needed by the refinement process. To do this, we must introduce some strong assumptions on the parameters defying the ellipse and, unfortunately, we cannot use the adaptive method described in [15]. Because the eigenvalues of $\mathbf{M}^{-1}\mathbf{F}$ are either real or complex conjugate, the centre of the ellipse lies on the real axis. The real part of the eigenvalues corresponding to the spectral radius $\sigma(\mathbf{M}^{-1}\mathbf{F})$ can be either positive or negative. Therefore, we have opted to choose the centre of the ellipse to be zero.

If the spectral radius $\sigma(\mathbf{M}^{-1}\mathbf{F}) = 1 - \sigma(\mathbf{M}^{-1}\mathbf{A})$ lies between (0, 1), we can scale the ellipse so that

$$\left.\begin{array}{l} a = 1 - \sigma(\mathbf{M}^{-1}\mathbf{A}) \\ b = t * a \end{array}\right\},$$

with t chosen such that the spectrum is contained within the ellipse. This choice is based on our empirical experience: we have observed that Gaussian elimination frequently produces a matrix $\mathbf{M}^{-1}\mathbf{F}$ that has a spectrum characterised by a large cluster of very small eigenvalues of size ε while, for each of the few remaining eigenvalues, the absolute value of the real part is much larger than that of the imaginary part. This justifies our assumption that the major axis of the ellipse is on the real line. However, there is no proof that this is always the case; indeed, if $t \approx 1$, the ellipse degenerates to a circle and Algorithm 2 becomes iterative refinement. Finally, we point out that, if the spectral radius corresponds to eigenvalues for which the absolute value of the imaginary part is larger than that of the real part, the Chebyshev algorithm using our choice of the ellipse will diverge immediately, and we can then opt to exchange a and b, rotating the ellipse by $\pi/2$, and restart the algorithm.

Recall that the number of steps $iter$ to reduce the residual by p orders of magnitude is given by equation (14). In Figures 1 and 2, we present graphs of $iter$ for $p = 1$ and 8, respectively. Results are plotted for $t = 0$, 0.1, 0.01 and 1 (iterative refinement) and for $\sigma(\mathbf{M}^{-1}\mathbf{F})$ between 0 and 1. From the graphs, we can predict that if $\sigma(\mathbf{M}^{-1}\mathbf{F}) > 0.4$ and t is chosen to be 0.01, Algorithms 2 and 3 will require significantly fewer steps than iterative refinement. The reduction will potentially be very important for values of $\sigma(\mathbf{M}^{-1}\mathbf{A})$ close to 1, and the best achievable accuracy $\omega(\varepsilon)$ will be rapidly obtained.

We note that after the first step of Algorithm 2 (which is equal to the first step of iterative refinement), from (7), we can estimate the value of σ to be

$$\sigma(\mathbf{M}^{-1}\mathbf{F}) \lessapprox \rho_1 = \frac{||\bar{\mathbf{r}}_{IR}^{(1)}||}{||\bar{\mathbf{r}}_{IR}^{(0)}||}. \tag{27}$$

More generally, the ratio between the computed residuals $\bar{\mathbf{r}}_{IR}^{(k)}$ of the iterative refinement at the kth and $(k-1)th$ steps

$$\rho_k = \frac{||\bar{\mathbf{r}}_{IR}^{(k)}||}{||\bar{\mathbf{r}}_{IR}^{(k-1)}||} \tag{28}$$

may be used to estimate an upper bound of σ. We note that the ratio (28) approximates the largest singular value on $\mathbf{M}^{-1}\mathbf{F}$, which is an upper bound of $\sigma(\mathbf{M}^{-1}\mathbf{F})$. We frequently observe that the largest singular value is well separated from the rest of the singular value spectrum when Gaussian elimination is used. This explains why the number of steps k in (28) for convergence is seldom greater than 2 or 3. We also tested the power method which, using (7), gives directly an approximation of the $\sigma(\mathbf{M}^{-1}\mathbf{F})$ by computing

$$\frac{|\bar{\mathbf{r}}_{IR}^{(k)T}\bar{\mathbf{r}}_{IR}^{(k-1)}|}{||\bar{\mathbf{r}}_{IR}^{(k-1)}||}.$$

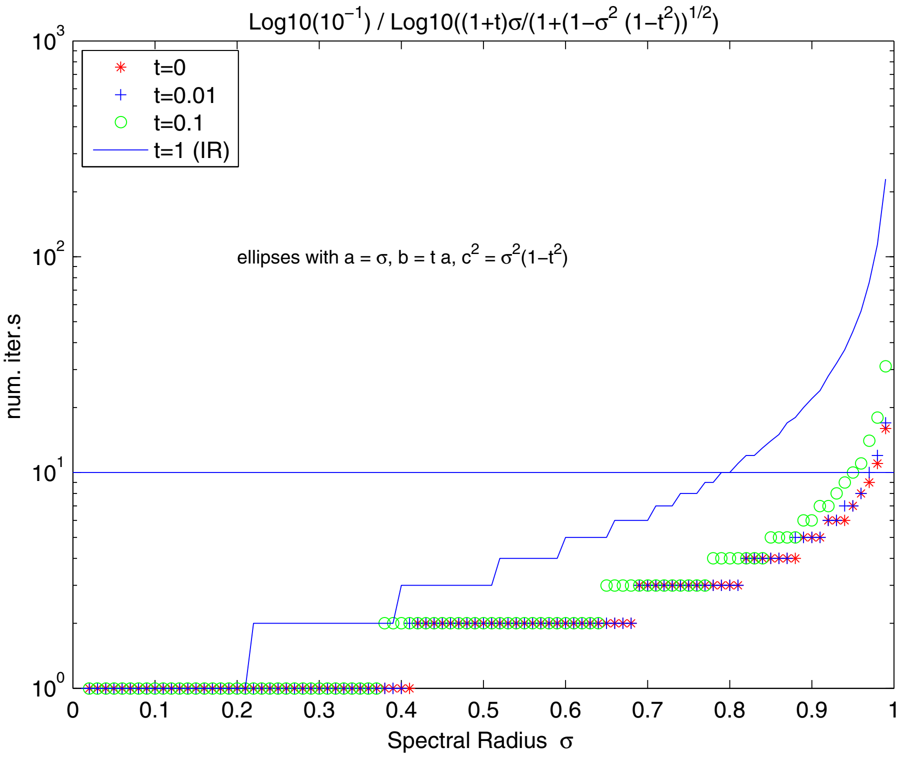


Fig. 1 Asymptotic rate of convergence for reducing the initial residual by 10^{-1}

However, in our tests the number of steps that this required to converge to an approximation of $\sigma(\mathbf{M}^{-1}\mathbf{F})$ was much larger than the number used by (28).

Remark 10 From Figure 1 and Figure 2, we see that there is no significant advantage in choosing $t < 0.01$. Thus, we use $t = 0.01$ as the default value in the experiments reported on in the next section.

Remark 11 Frequently it is necessary to solve a sequence of linear systems with the same matrix $\mathbf{A}$ but different right-hand sides $\mathbf{b}$. The ellipse parameters need to be computed only once and the subsequent applications can reuse the information. In this situation, the adaptive technique of [15] may become attractive even though it uses scalar products since it is only needed for the first $\mathbf{b}$. This contrasts with FGMRES, which requires scalar products for each $\mathbf{b}$.

5 Tests on sparse linear systems

In our experiments on sparse systems, we factorize the matrix $\mathbf{A}$ using the single precision version of the new sparse multifrontal solver `HSL_MA97` [13], store the

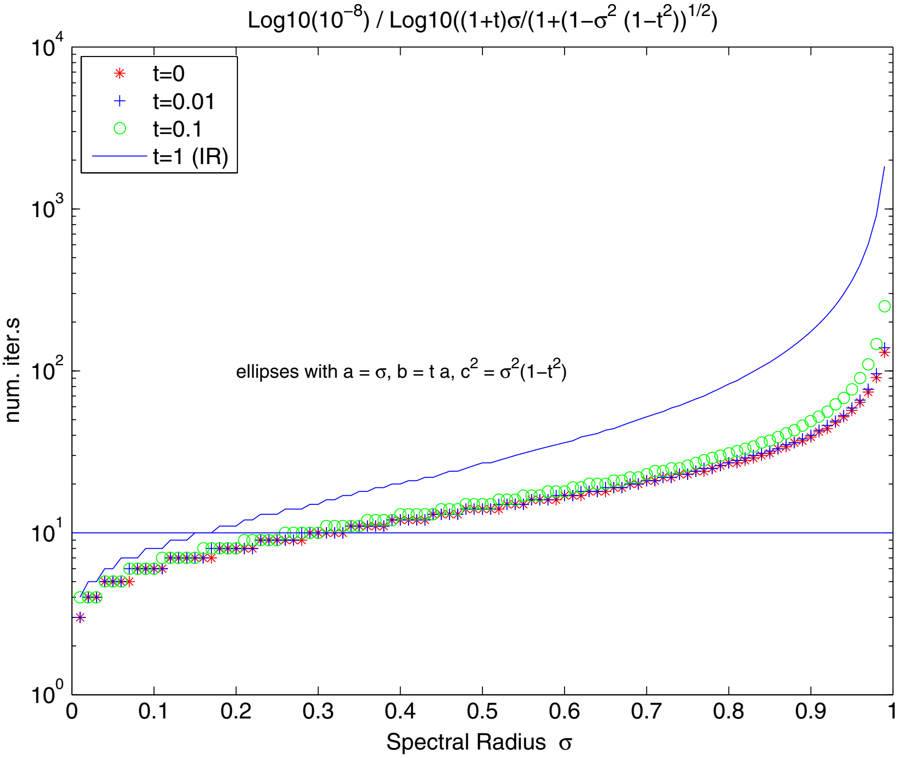


Fig. 2 Asymptotic rate of convergence for reducing the initial residual by 10^{-8}

computed factors in double precision and then perform refinement using double precision arithmetic. We ran this mixed precision approach on a large number of real symmetric problems taken from the University of Florida Sparse Matrix Collection [5]. For each example, the right-hand side vector $\mathbf{b}$ was generated by setting each component x_i of $\mathbf{x}$ to be a random number in the range $(-1, 1)$. In many cases, only one or two steps of iterative refinement were required to achieve a component-wise scaled residual β (see (5)) of less than $5*10^{-15}$ (see also the results in [11]). For some ill-conditioned problems, iterative refinement converged to the requested accuracy but required ten or more iterations. These problems are reported on in Table 1. Our expectation for these problems is that, with appropriately chosen ellipse parameters a and b, Chebyshev acceleration will be able to reduce the number of iterations.

In our tests, we set $b = 0.01 * a$ and experimented with a range of values of a. For some problems (including HB/nos2 and HB/nos7) setting a equal to the estimate (27) gives very good results. In some cases, ρ_k given by (28) rapidly converges and setting $a = \rho_k$ for a small value of $k > 1$ ($k = 2$ or 3) minimises the number of iterations. For example, for test example GHS_indef/bratu3d, using ρ_1 requires 20 iterations whereas using ρ_2 reduces the number of iterations to 17 (iterative refinement needs 25 steps). However, we also observed that for some examples, ρ_1 is very small and $a = \rho_1$ gives no improvement on iterative refinement, whereas using a later value of ρ_k can result in savings. This is illustrated by problem GHS_indef/cont-300. In this instance, $\rho_1 = 0.08$ and using $a = \rho_1$ requires 213 iterations (only 5 less than iterative refinement), while using $\rho_9 = 0.87$ reduces the iteration count to 65.

In Table 1, we present results for $a = \rho_1$ and for the a that in our tests resulted in the smallest number of iterations (where applicable, we indicate which ρ_k was used).

These results confirm our expectations that Chebyshev acceleration can significantly reduce the number of calls to the solve phase of the direct solver and also that the savings achieved can be very dependent on choosing appropriate ellipse parameters. In many cases, it is worthwhile to perform 2 or 3 steps of iterative refinement to obtain a suitable value for a and then to use Chebyshev acceleration. For comparison purposes, in Table 1, we include iteration counts for using FGMRES to perform the refinement. The mixed precision version of the restarted FGMRES algorithm that is described in [11] is used. This is essentially as given in [2] but it additionally uses an adaptive restart parameter that was found in numerical experiments to be more efficient than using a fixed restart parameter (that is, in general, it reduced the number of iterations required). Our choice of initial restart parameter of 4 is based on the results given in [11]. Note that the results are sensitive to this choice: using a larger value can lead to a larger total number of iterations because the termination conditions are only tested when the algorithm is restarted. We see that our efficient FGMRES implementation generally converges more rapidly than Chebyshev refinement but, as observed in the Introduction, FGMRES suffers from the disadvantage of requiring scalar products.

For problems with ρ_k almost equal to 1 for k sufficiently large, iterative refinement converges very slowly. For these examples, we are interested in seeing whether

Table 1 Comparison of the number of steps ($iter$) required by iterative refinement (IR), Chebyshev accelerated iterative refinement and restarted FGMRES. Chebyshev refinement is run with $a = \rho_1$ and the a that minimizes the number of iterations, denoted by a_{best} (if $a_{best} = \rho_k$, the value of k is given in parentheses)

Problem	*IR*	Chebyshev *IR*				FGMRES
		$a = \rho_1$		$a = a_{best}$		
	$iter$	$iter$	ρ_1	$iter$	a_{best}	
HB/nos2	23	12	0.43	12	0.43 (1)	4
HB/nos7	28	15	0.53	13	0.54	8
HB/bcsstm27	30	15	0.56	14	0.57	12
GHS_indef/bratu3d	25	20	0.34	17	0.25 (2)	12
Cylshell/s3rmt3m1	25	19	0.40	15	0.47 (4)	8
Cylshell/s3rmq4m1	14	10	0.25	10	0.25 (1)	8
GHS_indef/ncvxbqp1	29	25	0.28	19	0.40 (2)	12
GHS_indef/cont-300	218	213	0.08	65	0.87 (9)	28
Oberwolfach/gyro	25	19	0.28	13	0.46 (3)	12
GHS_indef/sparsine	38	21	0.51	20	0.50 (2)	12

Table 2 Comparison of the scaled residuals for iterative refinement and Chebyshev accelerated iterative refinement after 200 steps. These are denoted by $\beta_{IR}^{(200)}$ and $\beta_{C}^{(200)}(a)$, respectively; $\beta^{(0)}$ is the initial residual

Problem	$\beta^{(0)}$	$\beta_{IR}^{(200)}$	$\beta_{C}^{(200)}(a)$		
			$a = 0.99$	0.9999	0.999999
Boeing/crystk03	$1.42 * 10^{-7}$	$7.03 * 10^{-10}$	$9.98 * 10^{-11}$	$1.22 * 10^{-12}$	$7.83 * 10^{-12}$
Oberwolfach/t2dal	$1.13 * 10^{-7}$	$5.64 * 10^{-10}$	$8.01 * 10^{-11}$	$9.74 * 10^{-12}$	$3.18 * 10^{-12}$
Oberwolfach/t3dh_a	$3.10 * 10^{-7}$	$1.54 * 10^{-9}$	$2.18 * 10^{-10}$	$2.63 * 10^{-11}$	$1.55 * 10^{-11}$

Chebyshev acceleration is able to improve the rate of convergence. We experimented with setting $a = 0.99, 0.9999, 0.999999$ (again, with $b = 0.01 * a$) and ran iterative refinement and Chebyshev accelerated iterative refinement for 200 steps. Our findings are reported in Table 2. Here we give the initial scaled residual, the scaled residual for iterative refinement ($\beta_{IR}^{(200)}$) and for Chebyshev refinement ($\beta_{C}^{(200)}(a)$). As a approaches 1, $\beta_{C}^{(k)}(a)$ reduces more rapidly than $\beta_{IR}^{(k)}$. In particular, after 200 steps, the scaled residual for Chebyshev refinement with $a = 0.999999$ is two orders of magnitude smaller than for iterative refinement. Furthermore, with $a \geq 0.9999$, only 15 steps are required to reduce $\beta_{C}^{(k)}(a)$ below $\beta_{IR}^{(200)}$. However, for these examples, it is significantly better to use FGMRES. For the three problems in Table 2, FGMRES achieved the required accuracy in 12, 4 and 4 steps, respectively.

So far, our results have shown that, with an appropriate choice of ellipse, Chebyshev acceleration offers advantages over iterative refinement and, for problems where ρ_k is not close to 1, it requires a modest number of additional iterations compared with restarted FGMRES. However, we note that if we choose an ellipse that is too large, the performance of Chebyshev acceleration may be significantly worse than that of iterative refinement. For example, iterative refinement requires 25 steps for problem Oberwolfach/gyro; with $a = \rho_3 = 0.46$, Chebyshev acceleration reduced this to 13 iterations (which is one more than is required by FGMRES) but other values of a can require more iterations. This is illustrated in Table 3. As expected, for small a, the performance is as for iterative refinement while for sufficiently large a (in this case, $a > 0.8$), the performance is worse than for iterative refinement. Note that, although $a = \rho_3$ minimises the number of iterations, the precise choice of a is not critical: for a in the approximate range 0.4 to 0.7, Chebyshev acceleration offers worthwhile savings over iterative refinement.

Table 3 The number of iterations required for convergence of Chebyshev accelerated iterative refinement for problem Oberwolfach/gyro using a range of values of a. $a = \rho_3$ is in bold

ρ	0.1	0.2	0.4	0.44	**0.46**	0.5	0.6	0.7	0.8	0.9
$iter$	25	23	19	17	**13**	15	17	21	25	39

6 Conclusions

We have analysed Chebyshev accelerated iterative refinement from the point of view of roundoff and presented new results that prove the method is component-wise backward stable. We have presented numerical results for sparse linear systems arising from practical applications that support the theory. As our experiments illustrate, using an inexpensive estimate of the spectral radius obtained by performing a small number of steps of iterative refinement gives good convergence (albeit slower than FGMRES). Moreover, if the ellipse is chosen to be too small (so that it does not contain the complete spectrum), Chebyshev accelerated iterative refinement performs no worse than iterative refinement. We also point out that in all our numerical tests on real symmetric matrices, ellipses with $0 < b < a < 1$ were optimal. This suggests that for such problems $\sigma(\mathbf{M}^{-1}\mathbf{F})$ generally corresponds either to a real eigenvalue or to one with its real part much larger than its imaginary part. We could not find a theoretical justification of this phenomenon and it is possible to build small artificial examples where this is not the case. Finally, we have observed that while FGMRES often requires fewer iterations, its implementation involves scalar products that are inefficient when implemented in parallel. However, when $\sigma(\mathbf{M}^{-1}\mathbf{F})$ is close to or greater than 1, FGMRES is recommended.

Acknowledgments We are grateful to Michele Benzi for his encouragement and to an anonymous reviewer for carefully reading our manuscript and helping to improve the presentation.

References

1. Arioli, M., Demmel, J.W., Duff, I.S.: Solving sparse linear systems with sparse backward error. SIAM J. Matrix Anal. Appl. **10**(2), 165–190 (1989)
2. Arioli, M., Duff, I.S.: Using FGMRES to obtain backward stability in mixed-precision. Electron. Trans. Numer. Anal. **33**, 31–44 (2009)
3. Arioli, M., Duff, I.S., Gratton, S., Pralet, S.: A note on GMRES preconditioned by a perturbed LDL^T -decomposition with static pivoting. SIAM J. Sci. Comput. **29**, 2024–2044 (2007)
4. Brown, R.F.: A Topological introduction to nonlinear analysis. Birkhauser, Boston (1993)
5. Davis, T.A., Hu, Y.: The University of Florida Sparse Matrix Collection. ACM Trans. Math. Softw. **38**(1), 1–25 (2011)
6. Golub, G.H., Loan, C.F.V.: Matrix computations, 2nd edn. Johns Hopkins University Press, Baltimore (1989)
7. Gutknecht, M.H., Röllin, S.: The Chebyshev iteration revisited. Parallel Comput. **28**, 263–283 (2002)
8. Gutknecht, M.H., Strakoš, Z.: Accuracy of two three-term and three two-term recurrences for Krylov space solvers. SIAM J. Matrix Anal. Appl. **21**(1), 213–229 (2000)
9. Hageman, L.A., Young, D.M.: Applied iterative methods. Academic Press, New York (1981)
10. Higham, N.J.: Accuracy and stability of numerical algorithms, 2nd Edition. Society for Industrial and Applied Mathematics, Philadelphia (2002)
11. Hogg, J.D., Scott, J.A.: A fast and robust mixed precision solver for the solution of sparse symmetric linear systems. ACM Trans. Math. Softw. **37**, 17–24 (2010)

12. Hogg, J.D., Scott, J.A.: A note on the solve phase of a multicore solver. Tech. Rep. RAL-TR-2010-007, Rutherford Appleton Laboratory (2010)
13. Hogg, J.D., Scott, J.A.: `HSL_MA97`: A bit-compatible multifrontal code for sparse symmetric systems. Tech. Rep. RAL-TR-2011-024, Rutherford Appleton Laboratory (2011)
14. Manteuffel, T.A.: The Tchebychev iteration for nonsymmetric linear systems. Numer. Math. **28**, 307–327 (1977)
15. Manteuffel, T.A.: Adaptive procedure for estimating parameters for the nonsymmetric Tchebychev iteration. Numer. Math. **31**, 183–208 (1978)
16. Rutishauser, H.: Theory of gradient methods In: Refined iterative methods for computation of the solution and the Eigenvalues of self-adjoint boundary value problems, Nr. 8, pp. 24–49. Birkhauser, Basel (1959)
17. Saad, Y.: A flexible inner-outer preconditioned GMRES algorithm. SIAM J. Sci. Stat. Comput. **14**(2), 461–469 (1993)
18. Skeel, R.D.: Iterative refinement implies numerical stability for Gaussian elimination. Math. Comput. **35**, 817–832 (1980)