SIAM J. NUMER. ANAL.
Vol. 41, No. 1, pp. 90–111

KRYLOV SUBSPACE ACCELERATION OF WAVEFORM RELAXATION*

ANDREW LUMSDAINE† AND DEYUN WU‡

Abstract. In this paper we describe and analyze Krylov subspace techniques for accelerating the convergence of waveform relaxation for solving time-dependent problems. A new class of accelerated waveform methods, convolution Krylov subspace methods, is presented. In particular, we give convolution variants of the CG algorithm and the GMRES algorithm and analyze their convergence behavior. We prove that the convolution Krylov subspace algorithms for initial value problems have the same convergence bounds as their linear algebra counterparts. Analytical examples are given to illustrate the operation of convolution Krylov subspace methods. Experimental results are presented which show the convergence behavior of traditional and convolution waveform methods applied to solving a linear initial value problem as well as the convergence behavior of static Krylov subspace methods applied to solving the associated linear algebraic equation.

Key words. convolution, dynamic iteration, Galerkin method, Krylov subspace methods, waveform relaxation

AMS subject classifications. 65L60, 65L05, 65R20, 65J10

PII. S0036142996313142

1. Introduction.

Dynamic iteration methods for initial value problems were first studied by Picard (1893) and Lindelöf (1894) in the context of existence and uniqueness of solutions to ODEs. In the early 1980's, dynamic iteration was reintroduced (with the name "waveform relaxation") as an efficient method for solving the large sparsely coupled differential equation systems generated by the simulation of integrated circuits [15, 43]. Since then, this method has been extended and applied to various other application areas [20, 24, 40]. Waveform relaxation continues to attract interest because of its natural medium-scale parallelism.

With the waveform approach, a dynamic system of equations is first decomposed spatially (i.e., at the equation level). Individual equations, or sets of equations taken together, are then solved iteratively by using values from previous iterates of other equations as input. Thus, the iterates are functions ("waveforms") rather than vectors.

Unfortunately, the convergence rate of standard waveform relaxation can be prohibitively slow for many problems of interest. As with relaxation-based approaches for linear algebra (e.g., Jacobi), application of appropriate acceleration is necessary to make the waveform approach practical. Previous approaches for accelerating the convergence of waveform relaxation include the shifted Picard iteration [34], multigrid [18, 41], SOR [23], Chebyshev acceleration [17], convolution SOR [30], $\mathbb{L}^2$ Krylov subspace methods [19], and adaptive window size selection [14].

Many of these waveform acceleration techniques are analogous to acceleration methods for iteratively solving linear systems of equations. However, in most cases, the generalizations of those approaches to waveform relaxation do not accelerate con-

*Received by the editors December 5, 1996; accepted for publication (in revised form) April 17, 2001; published electronically February 12, 2003. This work was supported in part by National Science Foundation grants CCR92-09815 and CCR95-02710.
http://www.siam.org/journals/sinum/41-1/31314.html
†Computer Science Department, Indiana University, Bloomington, IN 47405 (lums@cs.indiana.edu).
‡IdeaDavinci Lab, 1550 Alameda, Suite 303, San Jose, CA 95126 (deyun@ideadavinci.com).

vergence to the same degree as their linear algebra counterparts [23]. An analysis of why linear acceleration of waveform relaxation can, in general, be expected to be limited is given in [28].

One acceleration method for waveform relaxation that does, in fact, provide the same degree of acceleration as the analogous linear algebra method is convolution SOR, developed in [30, 31]. Inspired by convolution SOR, we use a convolution-based approach to develop an entirely new class of algorithms for accelerating the convergence of waveform relaxation, namely, convolution Krylov subspace methods. As particular exemplars of this new class of algorithms, we develop and analyze convolution GMRES (CGMRES) and biconvolution CG (BiCCG). Analysis of these methods shows that the convolution algorithms for linear differential equations and the corresponding algorithms for the associated linear algebraic equations have the same convergence rate bounds. In other words, the convolution Krylov subspace methods accelerate the convergence of waveform relaxation to the same degree as their linear algebra counterparts.

In the next two sections, we first review waveform relaxation and the $\mathbb{L}^2$ Krylov subspace techniques presented in [19]. The convolution Krylov subspace methods are then developed and analyzed. Experimental results comparing various waveform approaches are presented, and we conclude with a discussion and suggestions for further work.

2. Waveform relaxation. The mathematical description of waveform methods that we will be use throughout this paper is based on the model initial value problem:

$$\begin{cases} \frac{d}{dt}\boldsymbol{x}(t) + \boldsymbol{A}\boldsymbol{x}(t) = \boldsymbol{f}(t), \\ \boldsymbol{x}(0) = \boldsymbol{x}_0, \end{cases} \tag{2.1}$$

where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{f}(t) \in \mathbb{R}^n$ is a given input, and $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the unknown vector to be computed over an interval of interest $[0, T]$.

In (2.1), let $\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}$ be a splitting of $\boldsymbol{A}$. The waveform relaxation algorithm based on this splitting is expressed in matrix form as follows.

ALGORITHM 1 (waveform relaxation for linear systems).

1. *Initialize:* Pick $\boldsymbol{x}^0$.
2. *Iterate:* For waveform iteration $k = 0, 1, \ldots$

   Solve

$$\begin{cases} \frac{d}{dt}\boldsymbol{x}^{k+1}(t) + \boldsymbol{M}\boldsymbol{x}^{k+1}(t) = \boldsymbol{N}\boldsymbol{x}^k(t) + \boldsymbol{f}(t), \\ \boldsymbol{x}^{k+1}(0) = \boldsymbol{x}_0 \end{cases}$$

   for $\boldsymbol{x}^{k+1}(t)$ on $[0, T]$.

Using operator notation, the waveform relaxation iteration can be expressed as

$$\boldsymbol{x}^{k+1} = \mathcal{K}\boldsymbol{x}^k + \boldsymbol{\psi}, \tag{2.2}$$

where the variables are defined on $\mathbb{L}^2([0, T], \mathbb{R}^n)$. The operator

$$\mathcal{K} : \mathbb{L}^2([0, T], \mathbb{R}^n) \to \mathbb{L}^2([0, T], \mathbb{R}^n)$$

is defined by

$$(\mathcal{K}\boldsymbol{x})(t) = \int_0^t e^{-\mathbf{M}(t-s)}\boldsymbol{N}\boldsymbol{x}(s)ds, \tag{2.3}$$

and $\boldsymbol{\psi} \in \mathbb{L}^2([0,T],\mathbb{R}^n)$ is given by

$$\boldsymbol{\psi}(t) = e^{-\mathbf{M}t}\boldsymbol{x}_0 + \int_0^t e^{-\mathbf{M}(t-s)}\boldsymbol{f}(s)ds.$$

It is obvious then that $\boldsymbol{x}$ will satisfy

$$(\boldsymbol{I} - \mathcal{K})\boldsymbol{x} = \boldsymbol{\psi}, \tag{2.4}$$

where $\boldsymbol{I}$ is the identity operator.

2.1. Properties. In this section we briefly review some relevant properties of the operator $\mathcal{K}$.

Remark. Associated with the initial value problem (2.1) is a linear algebraic problem

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}. \tag{2.5}$$

Similarly, associated with the waveform relaxation equation (2.4) is a preconditioned linear system of equations

$$(\boldsymbol{I} - \boldsymbol{M}^{-1}\boldsymbol{N})\boldsymbol{x} = \boldsymbol{M}^{-1}\boldsymbol{b}. \tag{2.6}$$

In what follows, we relate properties (in particular, spectral properties) of $\boldsymbol{M}^{-1}\boldsymbol{N}$ to properties of $\mathcal{K}$ and relate the behavior of algorithms applied to (2.4) to the behavior of algorithms applied to (2.6).

LEMMA 2.1. *The operator $\mathcal{K}$ as defined in* (2.3) *is compact, has zero spectral radius, and has adjoint operator $\mathcal{K}^*$ given by*

$$(\mathcal{K}^*\boldsymbol{x})(t) = \int_t^T \left[e^{-\mathbf{M}(s-t)}\boldsymbol{N}\right]^T \boldsymbol{x}(s)ds.$$

Remark. In general, $\mathcal{K}$ is not self-adjoint with respect to the $\mathbb{L}^2$ inner product, even when $\boldsymbol{M}^{-1}\boldsymbol{N}$, the matrix for the corresponding linear system, is symmetric in $\mathbb{R}^n$ (or Hermitian in $\mathbb{C}^n$).

Since $\mathcal{K}$ is compact with zero spectral radius, a straightforward convergence result can be stated.

THEOREM 2.2. *The waveform relaxation algorithm* (2.2) *generates a sequence of iterates $\{\boldsymbol{x}^k\}$ such that $\boldsymbol{x}^k \to \boldsymbol{x}$ as $k \to \infty$.*

Although $\mathcal{K}$ has zero spectral radius, it is highly nonnormal and thus the characteristics of the operator are far from trivial. That is, the spectrum itself provides very little insight into the behavior of iterative methods involving the operator $\mathcal{K}$. One approach to understanding iterative methods involving $\mathcal{K}$ is to consider the case for $T \to \infty$, in which case spectral properties of the operator apparently do provide information about the behavior of iterative methods involving $\mathcal{K}$. A detailed analysis of waveform relaxation for the $T \to \infty$ case is given in [23].

Unfortunately, the spectrum of $\mathcal{K}$ is discontinuous as a function of T—for any finite T, the spectral radius of $\mathcal{K}$ is zero. Thus, the degree to which the results for infinite T apply to real problems (which necessarily use finite T) is problem-dependent. One tool for understanding the behavior of $\mathcal{K}$ for finite T, and one that in some sense unifies the two cases of finite and infinite T, is pseudospectral analysis [38].

Definition. Let $\mathbb{X}$ be a Banach space with norm $\|\cdot\|$. The ϵ-pseudospectrum of a densely defined closed linear operator $\mathcal{A} : \mathbb{X} \to \mathbb{X}$ is defined as

$$\Lambda_\epsilon(\mathcal{A}) \equiv \left\{\lambda \in \mathbb{C} : \|(\lambda \boldsymbol{I} - \mathcal{A})^{-1}\| \geq \epsilon^{-1}\right\},$$

where it is understood that $\|(\lambda \boldsymbol{I} - \mathcal{A})^{-1}\| = \infty$ for $\lambda \in \Lambda(\mathcal{A})$. Here $\Lambda(\mathcal{A})$ is the spectrum of $\mathcal{A}$.

The following result (the proof is given in [22]) shows that the pseudospectrum is continuous as $T \to \infty$.

THEOREM 2.3. *Let $\mathcal{K}_T$ and $\mathcal{K}_\infty$ denote the operator $\mathcal{K}$ on $\mathbb{L}^2([0,T],\mathbb{R}^n)$ and $\mathbb{L}^2([0,\infty),\mathbb{R}^n)$, respectively. Then, for $\varepsilon > 0$,*

$$cl \lim_{T\to\infty} \Lambda_\varepsilon(\mathcal{K}_T) = \Lambda_\varepsilon(\mathcal{K}_\infty).$$

2.2. Example. To aid our subsequent discussion, we provide a graphical illustration of the spectrum and pseudospectrum of $\mathcal{K}$ for the model problem (2.1). We take $\boldsymbol{A}$ to be a symmetric positive definite matrix:

$$\boldsymbol{A} = \frac{1}{\Delta x^2}\begin{bmatrix} 2 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ 0 & & & -1 & 2 \end{bmatrix}_{n\times n}, \tag{2.7}$$

which is obtained, e.g., from discretizing the one-dimensional heat equation with spatial discretization Δx. For this example, we use a Jacobi splitting to obtain $\boldsymbol{M}$ and $\boldsymbol{N}$ from $\boldsymbol{A}$, and we take $\Delta x = 1/16$, $n = 17$, and $\varepsilon = 10^{-3}$.

Figure 2.1 shows the spectra and pseudospectra of $\mathcal{K}_T$ and $\mathcal{K}_\infty$ for $\boldsymbol{A}$ given in (2.7), where $\mathcal{K}_T$ and $\mathcal{K}_\infty$, respectively, denote the operator $\mathcal{K}$ on $\mathbb{L}^2([0,T],\mathbb{R}^n)$ and $\mathbb{L}^2([0,\infty),\mathbb{R}^n)$. Since the matrix $\boldsymbol{M}^{-1}\boldsymbol{N}$ is normal, the pseudospectrum of $\mathcal{K}_\infty$

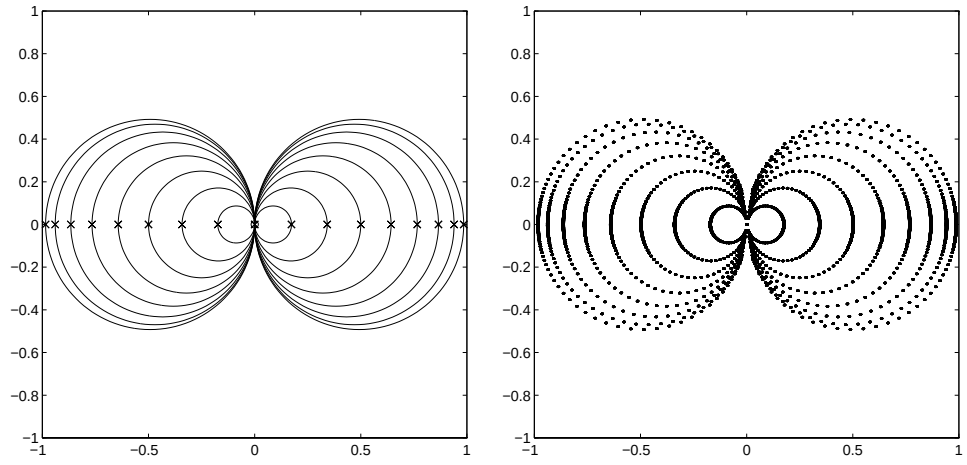


FIG. 2.1. *Spectrum and pseudospectrum of $\mathcal{K}_T$ and $\mathcal{K}_\infty$ for $\boldsymbol{A}$ given in* (2.7). *In the left figure, $\Lambda(\mathcal{K}_\infty)$ is the union of the interiors of the circles shown, $\Lambda(\mathcal{K}_T)$ is the origin (indicated with* o*), and the eigenvalues of $\boldsymbol{M}^{-1}\boldsymbol{N}$ are indicated with* x. *In the right figure, $\Lambda_\varepsilon(\mathcal{K}_\infty)$ is the union of the interiors of the dotted circles.*

is also very close to the spectrum. The spectra and pseudospectra were plotted using the formulae given in [22]. Note that $\Lambda_\varepsilon(\mathcal{K}_T)$ does not have a known formula (only bounds are known), and so it is not plotted.

3. Hilbert space acceleration methods. For solving linear algebra problems, Krylov subspace algorithms form sequences of approximate solutions $\{\boldsymbol{x}^k\}$ with

$$\boldsymbol{x}^k = \boldsymbol{x}^0 + \sum_{i=0}^{k-1} \alpha_i \boldsymbol{A}^i \boldsymbol{r}^0,$$

where $\boldsymbol{x}^0$ is the initial estimate for $\boldsymbol{x}$ and $\boldsymbol{r}^k = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}^k$ is the residual associated with the kth iterate. That is, each $\boldsymbol{x}^k$ is a member of the affine Krylov subspace

$$\boldsymbol{x}^k \in \boldsymbol{x}^0 + \mathbb{K}^k(\boldsymbol{A}, \boldsymbol{r}^0) = \boldsymbol{x}^0 + \operatorname{span}\{\boldsymbol{r}^0, \boldsymbol{A}\boldsymbol{r}^0, \ldots, \boldsymbol{A}^{k-1}\boldsymbol{r}^0\} \subset \mathbb{R}^n.$$

Algorithms for generating $\{\boldsymbol{x}^k\}$ typically do so by enforcing some type of Galerkin or minimal residual condition on the iterates. To enforce these conditions, it is only necessary that the underlying space has certain geometric properties, namely, that a notion of orthogonality exists. This is usually taken to mean Hilbert space, but, as we will see, it seems that Hilbert space geometry may be too strong and that weaker geometric conditions can yield effective methods (see the discussion in section 7). Thus, Krylov subspace algorithms can readily be extended from $\mathbb{R}^n$ to Hilbert space (a fact that has been known since the early development of Krylov subspace iterative methods for linear algebra [10]). By embedding (2.4) into an appropriate Hilbert space, it is a rather straightforward matter to develop Krylov subspace acceleration techniques for waveform relaxation. A natural Hilbert space for this problem is $\mathbb{L}^2([0,T],\mathbb{R}^n)$.

3.1. Waveform GMRES. By Lemma 2.1, it is obvious that $\mathcal{K}$ is not self-adjoint with respect to the $\mathbb{L}^2$ inner product. Thus, in order to accelerate waveform relaxation, we must restrict our attention to those Krylov subspace algorithms suitable for non-Hermitian linear systems. At present, the premier such algorithm is the GMRES algorithm of Saad and Schultz [33].

The waveform GMRES (WGMRES) algorithm is as follows.

ALGORITHM 2 (WGMRES).

1. *Start:* Set $\boldsymbol{r}^0 = \boldsymbol{\psi} - (\boldsymbol{I} - \mathcal{K})\boldsymbol{x}^0$, $\boldsymbol{v}^1 = \boldsymbol{r}^0/\|\boldsymbol{r}^0\|$, $\beta = \|\boldsymbol{r}^0\|$.
2. *Iterate:* For $k = 1, 2, \ldots,$ until satisfied do:

$$\begin{aligned} h_{jk} &= \langle (\boldsymbol{I} - \mathcal{K})\boldsymbol{v}^k, \boldsymbol{v}^j \rangle, \ j = 1, 2, \ldots, k \\ \widetilde{\boldsymbol{v}}^{k+1} &= (\boldsymbol{I} - \mathcal{K})\boldsymbol{v}^k - \textstyle\sum_{j=1}^k h_{jk}\boldsymbol{v}^j \\ h_{k+1,k} &= \|\widetilde{\boldsymbol{v}}^{k+1}\| \\ \boldsymbol{v}^{k+1} &= \widetilde{\boldsymbol{v}}^{k+1}/h_{k+1,k}. \end{aligned}$$

3. *Form approximate solution:*
   $\boldsymbol{x}^k = \boldsymbol{x}^0 + \boldsymbol{V}^k\boldsymbol{y}^k$, where $\boldsymbol{y}^k$ minimizes $\|\beta\boldsymbol{e}_1 - \bar{\boldsymbol{H}}^k\boldsymbol{y}^k\|$,
   $\boldsymbol{V}^k = (\boldsymbol{v}^1, \boldsymbol{v}^2, \ldots, \boldsymbol{v}^k)$, $\boldsymbol{e}_1 = (1, 0, \ldots, 0)^T$, and $\bar{\boldsymbol{H}}^k = (h_{ij})_{(k+1)\times k}$.

Remark. Symbolically, this algorithm is identical to the GMRES algorithm on $\mathbb{R}^n$. The difference is that the vectors and vector-space operations are understood to be defined on the Hilbert space $\mathbb{L}^2([0,T],\mathbb{R}^n)$ rather than on $\mathbb{R}^n$. That is, $\boldsymbol{x}^k$, $\boldsymbol{r}^k$, $\boldsymbol{\psi}$, $\boldsymbol{v}^k$, $\widetilde{\boldsymbol{v}}^k \in \mathbb{L}^2([0,T],\mathbb{R}^n)$, $h_{jk} \in \mathbb{C}$, and $\langle\cdot,\cdot\rangle$ is the $\mathbb{L}^2$ inner product.

3.2. Analysis of WGMRES. In order to analyze WGMRES, it is useful to recall that GMRES is a Galerkin method. The use of a Galerkin method over a Krylov space generated by $(\boldsymbol{I}-\mathcal{K})$ is discussed in [25] and [29], where the approach is called the method of moments (see also [42]). The following two results can be found in [19, 21].

THEOREM 3.1. *Let $\boldsymbol{X}$ be a Hilbert space and let $\mathcal{A} : \boldsymbol{X} \to \boldsymbol{X}$ be a bounded bijective linear operator. Let $\boldsymbol{X}^k \subset \boldsymbol{X}$ be a k-dimensional subspace with $\boldsymbol{X}^k \subset \boldsymbol{X}^{k+1}$ for all $k \in \mathbb{N}$. If $\boldsymbol{x}$ is in the closure of $\mathbb{S} = \bigcup_{k=1}^{\infty} \boldsymbol{X}^k$, then the Galerkin method for the operator equation $\mathcal{A}\boldsymbol{x} = \boldsymbol{f}$ converges.*

COROLLARY 3.2. *The Galerkin method for $(\boldsymbol{I}-\mathcal{K})\boldsymbol{x} = \boldsymbol{\psi}$ converges in the space $\mathbb{L}^2([0,T],\mathbb{R}^n)$, with finite dimensional subspaces $\mathbb{K}^m(\mathcal{K},\boldsymbol{\psi}) = \{\boldsymbol{\psi},\mathcal{K}\boldsymbol{\psi},\ldots,\mathcal{K}^{m-1}\boldsymbol{\psi}\}$ for all $m \in \mathbb{N}$.*

As an immediate consequence, we have that WGMRES converges.

Unfortunately, because of the nonnormality of the operator $\mathcal{K}$, nothing much can be said about the rate of convergence, particularly in relationship to the spectrum of $\boldsymbol{M}^{-1}\boldsymbol{N}$ (or, more to the point, in relationship to the behavior of GMRES applied to solving a linear system of equations based on $\boldsymbol{M}^{-1}\boldsymbol{N}$).

Although precise statements about the convergence behavior of GMRES cannot be made, certain (rather pessimistic) qualitative statements can be made. Nevanlinna considered the general case of linear acceleration of waveform relaxation in [28], with the conclusion that significant speedups (of the sort one sees for linear algebra problems) would not be achievable in general.

Intuitively, we can see that it is much more difficult for GMRES to be effective when applied to waveform relaxation. Figure 2.1 shows the spectrum and pseudospectrum of $\mathcal{K}$. For the matrix problem, the spectrum of $\boldsymbol{M}^{-1}\boldsymbol{N}$ consists of a set of distinct eigenvalues on the real axis. For the waveform problem, the pseudospectrum of $\mathcal{K}$ fills a two-dimensional region in the complex plane. Clearly, it is much more difficult to find a good minimizing polynomial for the waveform case than for the linear algebra case. In fact, by using conformal mapping techniques, it is possible to show that there is, in fact, no essential speedup possible for WGMRES for a large class of problems [37] (on the infinite interval).

However, all is not lost. Similar pessimistic results hold for waveform relaxation accelerated with SOR [23]. However, with the use of convolution techniques, Reichelt developed a variant of SOR for waveform relaxation that does for (2.4) what algebraic SOR does for (2.6) [31]. That convolution techniques can provide the desired rate of acceleration for SOR gives us hope that similar results can be achieved for Krylov subspace techniques. We develop such a class of algorithms and prove their rates of convergence in the next section.

4. Convolution methods. The principle behind convolution SOR (CSOR), and, indeed, the convolution methods developed in this paper, is that rather than simply taking linear combinations of waveform iterates, the methods take sums weighted by a convolution kernel. The resulting algorithms thus circumvent the limitations of linear acceleration as described in [23] and [28]. In fact, CSOR and the convolution Krylov subspace algorithms developed here exhibit speedup precisely comparable to that in the associated linear algebra problem.

4.1. CSOR. A waveform relaxation algorithm using CSOR for solving (2.1) is shown in Algorithm 3. The algorithm takes an ordinary Gauss–Seidel waveform relaxation step to obtain a value for the intermediate variable $\widehat{x}_i^{k+1}$. The iterate x_i^{k+1} is obtained by adding a correction obtained by convolving $\widehat{x}_i^{k+1} - x_i^{k+1}$ with a kernel

function $\omega(t)$. This is in contrast to simple waveform SOR in which x_i^{k+1} is obtained by multiplying $\widehat{x}_i^{k+1} - x_i^{k+1}$ with a scalar parameter ω. With the convolution, the CSOR method correctly accounts for the temporal frequency-dependence of the spectrum of the Jacobi waveform relaxation operator (e.g., Jacobi waveform relaxation smoothes high frequency components of the error waveform more rapidly than low frequency components) by, in effect, using a different SOR parameter for each frequency [31].

ALGORITHM 3 (Gauss–Seidel waveform relaxation with CSOR acceleration).

1. *Initialize:* Pick vector waveform $\boldsymbol{x}^0(t) \in C^1([0,T],\mathbb{R}^n)$ with $\boldsymbol{x}^0(0) = \boldsymbol{x}_0$.
2. *Iterate:* For $k = 0, 1, \ldots$, until converged,
   - *Solve* for scalar waveform $\widehat{x}_i^{k+1}(t) \in C^1([0,T],\mathbb{R})$ with $\widehat{x}_i^{k+1}(0) = x_{0i}$,

$$\left(\tfrac{d}{dt} + a_{ii}\right)\widehat{x}_i^{k+1}(t) = f_i(t) - \sum_{j=1}^{i-1} a_{ij}x_i^{k+1}(t) - \sum_{j=i+1}^{n} a_{ij}x_i^k(t).$$

   - *Overrelax* to generate $x_i^{k+1}(t) \in C^1([0,T],\mathbb{R})$,

$$x_i^{k+1}(t) = x_i^k(t) + \int_0^t \omega(\tau)\cdot\left[\widehat{x}_i^{k+1}(t-\tau) - x_i^k(t-\tau)\right]d\tau. \tag{4.1}$$

4.2. Convolution Krylov subspace algorithms. In this section and the next, we incorporate convolution into the Krylov subspace approach for accelerating waveform relaxation. We begin by identifying some key spaces and associated operations to be used in what follows.

Assume $f, g \in \mathbb{L}^2(\mathbb{R},\mathbb{R})$ are functions, and $\boldsymbol{x} = (x_1,\ldots,x_n)^T$, $\boldsymbol{y} = (y_1,\ldots,y_n)^T \in \mathbb{L}^2(\mathbb{R},\mathbb{R}^n)$ are vectors of functions. Define

$$\begin{aligned}
(f \star g)(t) &= \int_{-\infty}^{\infty} f(s)g(t-s)ds \in \mathbb{L}^1(\mathbb{R},\mathbb{R}),\\
(f \star \boldsymbol{x})(t) &= ((f \star x_1)(t)\ldots(f \star x_n)(t))^T \in \mathbb{L}^1(\mathbb{R},\mathbb{R}^n),\\
\langle \boldsymbol{x}, \boldsymbol{y}\rangle_\star(t) &= \sum_{i=1}^{n}(x_i \star \widetilde{y}_i)(t) \in \mathbb{L}^1(\mathbb{R},\mathbb{R}),
\end{aligned}$$

where $\widetilde{f}(t) = f(-t)$, and $\widetilde{\boldsymbol{x}}(t) = \boldsymbol{x}(-t)$.

Remark. Although the above formulae are defined in general for $\mathbb{L}^2$ functions, one important subset of $\mathbb{L}^2$ that will figure prominently is $C_0([0,\infty),\mathbb{R})$, the compactly supported continuous functions, and we will be working with this space in what follows. The following technique is purely for the analysis of the algorithms, although the implementational computations are not related to it. For brevity we will indicate $\mathcal{K}_T$ by $\mathcal{K}$.

From the basic operational calculus [26, 27, 36], it is known that convolution induces a ring structure on $C_0([0,\infty),\mathbb{R})$. As with any ring structure, this ring structure, can be algebraically extended to a quotient field (in a manner similar to extending the ring of integers to the field of rational numbers). Define $\mathbb{Q}$ to be the set of ordered pairs ("fractions")

$$\mathbb{Q} = \{f/g : f, g \in C_0([0,\infty),\mathbb{R})\}.$$

By a result of Titchmarsh [36], there are no zero divisors; i.e., $f \star g = 0$ implies that either $f = 0$ or $g = 0$. The axiomatic operations required for field structure are then

readily defined. An operational calculus has been developed on this basis, adding a theoretical underpinning to the operational calculus of Heaviside [26, 27].

A result of Foiaş shows that the range of convolution is dense in $\mathbb{L}^1$, a result that was later extended to the continuous case [7, 35]. More recently, a constructive proof of these results has been given by Bäumer [2]. As a direct consequence of the injectivity and dense range properties of convolution, we have the following.

LEMMA 4.1. *Let $\mathbb{Q}$ be the quotient field induced by the convolution ring $\{C_0([0,\infty), \mathbb{R}), \star\}$. For any element $f/g \in \mathbb{Q}$, there is a sequence $\{\phi_i\}$ with $\phi_i \in C_0([0,\infty), \mathbb{R})$ such that*

$$\lim_{i\to\infty} \phi_i \star g = f.$$

Remark. $\mathbb{Q}$ is, in fact, a space of generalized functions; the limit $\lim_{i\to\infty} \phi_i$ may not necessarily exist in $C_0([0,\infty), \mathbb{R})$. For example, the element in $\mathbb{Q}$ identified with f/f is the Dirac δ-distribution. In this regard, $\mathbb{Q}$ is a completion of $C_0([0,\infty), \mathbb{R})$ with respect to the convolution operator. Notice that a function $f \in C_0([0,\infty), \mathbb{R})$ can be identified with any quotient of the form $(f\star g)/g$, $g \neq 0$; i.e., $f \in C_0([0,\infty), \mathbb{R})$ implies $f \in \mathbb{Q}$.

Using this definition of convolution between elements of $\mathbb{Q}$ and continuous functions, we can readily create a vector space over the field $\mathbb{Q}$ using convolution.

PROPOSITION 4.2. *Let $\mathbb{Q}$ be the quotient field induced by the convolution ring $\{C_0([0,\infty), \mathbb{R}), \star\}$. For $q \in \mathbb{Q}$ and $\boldsymbol{x} \in C_0([0,\infty), \mathbb{R}^n)$, let $\phi_i \to q$ in the sense of Lemma* 4.1. *With the convolution operation defined by*

$$q \star \boldsymbol{x} = \lim_{i\to\infty} \phi_i \star \boldsymbol{x},$$

$C_0([0,\infty), \mathbb{R}^n)$ forms a vector space over $\mathbb{Q}$.

Definition. Assume $\mathcal{A}$ is a bounded linear operator defined on $C_0([0,\infty), \mathbb{R}^n)$ and $\boldsymbol{r}^0 \in C_0([0,\infty), \mathbb{R}^n)$ is fixed. The m-dimensional convolution Krylov subspace generated by $\mathcal{A}$ and $\boldsymbol{r}^0$ is defined to be

$$\begin{aligned} \mathbb{K}^m_\star(\mathcal{A}, \boldsymbol{r}^0) &= \operatorname{span}_\star\{\boldsymbol{r}^0, \mathcal{A}\boldsymbol{r}^0, \ldots, \mathcal{A}^{m-1}\boldsymbol{r}^0\} \\ &= \left\{ \sum_{i=0}^{m-1} \alpha_i \star \mathcal{A}^i \boldsymbol{r}^0 : \alpha_i \in \mathbb{Q} \right\} \subset C_0([0,\infty), \mathbb{R}^n). \end{aligned}$$

Notice that this definition of the convolution Krylov subspace differs from the usual definition of a Krylov subspace due to the convolution operator; by $\operatorname{span}_\star$ we mean combinations under convolution. However, as with methods that use the traditional definition of a Krylov subspace, we seek to find the element of $\mathbb{K}^m_\star(\mathcal{A}, \boldsymbol{r}^0)$ that best satisfies (2.4).

4.3. Examples. Using convolutions in Krylov subspace methods may seem counterintuitive. To demonstrate the general operation of convolution Krylov subspace methods, we present two examples.

Example. Consider the following initial value problem:

$$\left\{ \begin{array}{r} \frac{d}{dt}\boldsymbol{x}(t) + \boldsymbol{A}\boldsymbol{x}(t) = \boldsymbol{0}, \\ \boldsymbol{x}(0) = (0,1)^T, \end{array} \right. \tag{4.2}$$

where $\boldsymbol{A}$ is a 2×2 matrix

$$\boldsymbol{A} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

For splitting $\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}$ with $\boldsymbol{M} = 2\boldsymbol{I}$, in order to solve $(\boldsymbol{I} - \mathcal{K})\boldsymbol{x} = \boldsymbol{\psi}$, we choose

$$\boldsymbol{x} = \alpha_0 \star \boldsymbol{r}^0 + \alpha_1 \star (\boldsymbol{I} - \mathcal{K})\boldsymbol{r}^0.$$

By the Galerkin conditions,

$$\langle (\boldsymbol{I} - \mathcal{K})\boldsymbol{x}, \boldsymbol{r}^0 \rangle_\star = \langle \boldsymbol{\psi}, \boldsymbol{r}^0 \rangle_\star,$$

$$\langle (\boldsymbol{I} - \mathcal{K})\boldsymbol{x}, (\boldsymbol{I} - \mathcal{K})\boldsymbol{r}^0 \rangle_\star = \langle \boldsymbol{\psi}, (\boldsymbol{I} - \mathcal{K})\boldsymbol{r}^0 \rangle_\star,$$

we can find

$$\begin{cases} \alpha_0 = \frac{2}{3}[e^{-2t} \star (-2 + e^{-2t})]/[(e^{-2t} - 1) \star (e^{-2t} - 1)], \\ \alpha_1 = -\frac{2}{3}[-1 + 2e^{-2t})]/[(e^{-2t} - 1) \star (e^{-2t} - 1)] \end{cases}$$

and

$$\begin{cases} x_1 = -(te^{-2t})/(te^{-2t} - 1) = e^{-2t} \sinh t, \\ x_2 = -(e^{-2t})/(te^{-2t} - 1) = e^{-2t} \cosh t, \end{cases}$$

which is the analytic solution for (4.2).

Example. Let us consider another initial value problem:

$$\begin{cases} \frac{d}{dt}\boldsymbol{x}(t) + \boldsymbol{A}\boldsymbol{x}(t) = \boldsymbol{0}, \\ \boldsymbol{x}(0) = (0, 1)^T, \end{cases} \tag{4.3}$$

where $\boldsymbol{A}$ is a 2×2 matrix

$$\boldsymbol{A} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

In this case, we take splitting $\boldsymbol{A} = \boldsymbol{M} - \boldsymbol{N}$ with $\boldsymbol{M} = 0$. Again choose

$$\boldsymbol{x} = \alpha_0 \star \boldsymbol{r}^0 + \alpha_1 \star (\boldsymbol{I} - \mathcal{K})\boldsymbol{r}^0.$$

As in the last example, by using the Galerkin conditions, we can find

$$\begin{cases} \alpha_0 = -(6t - t^3)/(6t + t^3), \\ \alpha_1 = 6/(6t + t^3) \end{cases}$$

and

$$\begin{cases} x_1 = t^3/(6t + t^3) = \sin t, \\ x_2 = (3t^2)/(6t + t^3) = \cos t, \end{cases}$$

which is the analytic solution for (4.3).

4.4. Fourier transform. Define the Fourier–Laplace transform on $\mathbb{L}^2(\mathbb{R},\mathbb{R})$ to be

$$\widehat{f}(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-izx} f(x) dx \quad \text{for } z \in \mathbb{C}.$$

It is well known that $\|\widehat{f}\|_{\mathbb{L}^2(\mathbb{R},\mathbb{R})} = \|f\|_{\mathbb{L}^2(\mathbb{R},\mathbb{R})}$.

Remarks.

1. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_\star \widehat{\ }(\xi) = \langle \widehat{\boldsymbol{x}}(\xi), \widehat{\boldsymbol{y}}(\xi) \rangle$, where $\langle \cdot, \cdot \rangle$ is the Hermitian inner product in $\mathbb{C}^n$, and $\xi \in \mathbb{R}$.
2. $\langle \boldsymbol{x}, \boldsymbol{y} \rangle_\star = \widetilde{\langle \boldsymbol{y}, \boldsymbol{x} \rangle_\star} = \langle \widetilde{\boldsymbol{y}}, \widetilde{\boldsymbol{x}} \rangle_\star$.
3. For $f, g \in C_0([0,\infty),\mathbb{R})$ $(f/g)\hat{\ } = \widehat{\frac{f}{g}}$ is a meromorphic function with discrete poles [1]. (Here $\frac{\cdot}{\cdot}$ indicates normal division.)
4. Under the convolution inner product "$\langle \cdot, \cdot \rangle_\star$," the convolution adjoint operator of $\mathcal{K}$ is the same as the $\mathbb{L}^2$ adjoint operator $\mathcal{K}^*$ of $\mathcal{K}$. Thus, even with real symmetric $\boldsymbol{A}$, $\mathcal{K}$ is not self-adjoint with respect to $\langle \cdot, \cdot \rangle_\star$.

Since convolution is closely related to the Fourier transform (which is isometric on the $\mathbb{L}^2(\mathbb{R},\mathbb{R})$ space), we restrict our analysis to $\mathbb{L}^2$ spaces. In this context, we view $C_0([0,\infty),\mathbb{R}^n)$ as a subspace of $\mathbb{L}^2(\mathbb{R},\mathbb{R}^n)$ by extending elements in $C_0([0,\infty),\mathbb{R}^n)$ trivially on $(-\infty, 0)$. Hence, for any $f \in C_0([0,\infty),\mathbb{R})$ we have $\|\widehat{f}\| = \|f\|$ in the $\mathbb{L}^2(\mathbb{R},\mathbb{R})$ norm.

4.5. The convolution GMRES algorithm. In this section, we introduce the convolution GMRES (CGMRES) algorithm. Analogous to GMRES for linear systems of equations, CGMRES is appropriate for operator systems where $\mathcal{A}$ is not self-adjoint with respect to the convolution inner product.

ALGORITHM 4 (CGMRES). Let $\mathcal{A} : C_0([0,\infty),\mathbb{R}^n) \to \underline{C_0([0,\infty),\mathbb{R}^n)}$ be a bounded linear operator. By Bäumer [2], $\mathcal{A}$ is extendable to $\overline{C_0([0,\infty),\mathbb{R}^n)}$, the vector-valued generalized function space, i.e., the completion of $C_0([0,\infty),\mathbb{R}^n)$ under operator $\mathcal{A}$, which is a vector space over $\mathbb{Q}$ due to the commutative of $\mathcal{A}$ and $\star$. Let $\boldsymbol{f} \in C_0([0,\infty),\mathbb{R}^n)$.

1. Pick $\boldsymbol{x}^0 \in C_0([0,\infty),\mathbb{R}^n)$ and compute $\boldsymbol{r}^0 = \boldsymbol{f} - \mathcal{A}\boldsymbol{x}^0$, $\beta = |\widehat{\boldsymbol{r}^0}|^\vee$, $\boldsymbol{v}^1 = \boldsymbol{r}^0/\beta$.
2. For $j = 1, \ldots,$ until converged,

$$\begin{aligned} \boldsymbol{w}^j &= \mathcal{A}\boldsymbol{v}^j \\ h_{ij} &= \langle \boldsymbol{w}^j, \boldsymbol{v}^i \rangle_\star, i = 1, \ldots, j \\ \boldsymbol{w}^j &= \boldsymbol{w}^j - \sum_{i \le j} h_{ij} \star \boldsymbol{v}^i \\ h_{j+1,j} &= |\widehat{\boldsymbol{w}^j}|^\vee, \quad \text{if } h_{j+1,j} \equiv 0, \text{ set } m = j \text{ and go to step 3} \\ \boldsymbol{v}^{j+1} &= \boldsymbol{w}^j / h_{j+1,j}. \end{aligned}$$

3. Compute $\boldsymbol{y}^m$, the minimizer of $\langle (\beta \star (\delta_0, 0, \ldots, 0)^T - \bar{\boldsymbol{H}}^m \star \boldsymbol{y}), (\beta \star (\delta_0, 0, \ldots, 0)^T - \bar{\boldsymbol{H}}^m \star \boldsymbol{y}) \rangle_\star$ and $\boldsymbol{x}^m = \boldsymbol{x}^0 + \boldsymbol{V}^m \star \boldsymbol{y}^m$, where $\bar{\boldsymbol{H}}^m = (h_{ij})_{(m+1)\times m}$, $\boldsymbol{V}^m = (\boldsymbol{v}^1, \boldsymbol{v}^2, \ldots, \boldsymbol{v}^m)$, and δ_0 is the Dirac δ-distribution.

Remarks.

1. In the above algorithms, "$|\cdot|$" is the Euclidean norm defined in $\mathbb{C}^n$, and "$\vee$" means the inverse Fourier transform.
2. By $\boldsymbol{x}/f$ for $\boldsymbol{x} \in C_0([0,\infty),\mathbb{R}^n)$ and $f \in C_0([0,\infty),\mathbb{R})$ we mean the vector $(x_1/f, x_2/f, \ldots, x_n/f)^T \in \mathbb{Q}^n$.

3. It is not hard to see that there is a compactly supported sequence $\beta_i \in C_0(-\infty,\infty)$ such that $\beta_i \to \beta \in \mathbb{L}^2(\mathbb{R})$. Rewrite $\boldsymbol{v}^1 = \boldsymbol{r}^0/\beta = (\boldsymbol{r}^0 \star h)/(\beta \star h)$, where $h \neq 0$ is any function in $C_0([0,\infty),\mathbb{R})$. Now $\beta_i \star h \in C_0([0,\infty),\mathbb{R})$. We view $\mathcal{A}\boldsymbol{v}^1 = \lim_{i\to\infty}(\mathcal{A}\boldsymbol{r}^0 \star h)/(\beta_i \star h)$ which is well-defined on the completion vector space $\overline{C_0([0,\infty),\mathbb{R}^n)}$ over $\mathbb{Q}$.

4.6. Convergence of CGMRES. To analyze the convergence of the CGMRES algorithm we begin with the convolution Petrov–Galerkin conditions

$$\langle \boldsymbol{f} - \mathcal{A}\boldsymbol{x}, \boldsymbol{w}\rangle_\star = 0 \quad \forall \boldsymbol{w} \in \mathcal{A}\mathbb{K},$$

where for brevity $\mathbb{K}$ indicates $\mathbb{K}^m_\star(\mathcal{A}, \boldsymbol{r}^0)$. Based on these conditions, we say that $\boldsymbol{f} - \mathcal{A}\boldsymbol{x}$ is *convolutionally perpendicular* to $\mathcal{A}\mathbb{K}$. Note that $\langle \boldsymbol{x}, \boldsymbol{y}\rangle_\star = 0$ if and only if $\langle \hat{\boldsymbol{x}}(\xi), \hat{\boldsymbol{y}}(\xi)\rangle_\star = 0$ for all $\xi \in \mathbb{R}$.

Therefore, after taking Fourier transforms, the CGMRES algorithm becomes the GMRES algorithm for the linear algebraic equation

$$\boldsymbol{A}(\xi)\boldsymbol{u} = \widehat{\boldsymbol{f}}(\xi)$$

at each fixed $\xi \in \mathbb{R}$, where $\boldsymbol{A}(\xi) = (i\xi + d)^{-1}(i\xi\boldsymbol{I} + \boldsymbol{A})$.

Notice that Fourier transform is isometric on $\mathbb{L}^2$. Therefore, by Proposition 6.15 in [32], we have the following result.

THEOREM 4.3. *Assume* $\boldsymbol{A} = \boldsymbol{X}\,\mathrm{diag}(\lambda_1,\dots,\lambda_n)\boldsymbol{X}^{-1}$ *has a splitting* $\boldsymbol{M} - \boldsymbol{N}$ *with* $\boldsymbol{M} = d\boldsymbol{I}$, $d > 0$. *Define*

$$\epsilon^{(k)} = \max_{\xi\in\mathbb{R}} \min_{\substack{p\,\in\,\mathbb{P}_k \\ p(0)\,=\,1}} \max_{j=1,\dots,n} \left| p\left(\frac{i\xi + \lambda_j}{i\xi + d}\right)\right|.$$

Then the residual $\boldsymbol{r}^k = \boldsymbol{\psi} - (\boldsymbol{I} - \mathcal{K})\boldsymbol{x}^k$ *obtained by the CGMRES algorithm satisfies*

$$\|\boldsymbol{r}^k\|_{\mathbb{L}^2} \le \kappa_2(\boldsymbol{X})\epsilon^{(k)}\|\boldsymbol{r}^0\|_{\mathbb{L}^2},$$

where $\kappa_2(\boldsymbol{X}) = \|\boldsymbol{X}\|_2\|\boldsymbol{X}^{-1}\|_2$ *is the condition number of* $\boldsymbol{X}$ *under the* 2*-norm.*

Proof. For each fixed $\xi \in \mathbb{R}$, it is known that

$$\left|\widehat{\boldsymbol{r}^k}(\xi)\right|_2 \le \kappa_2(\boldsymbol{X}) \min_{\substack{p\,\in\,\mathbb{P}_k \\ p(0)\,=\,1}} \max_{j=1,\dots,n} \left| p\left(\frac{i\xi + \lambda_j}{i\xi + d}\right)\right| \cdot \left|\widehat{\boldsymbol{r}^0}(\xi)\right|_2.$$

The right-hand side of the above inequality is bounded by $\kappa_2(\boldsymbol{X})\epsilon^{(k)}|\hat{\boldsymbol{r}}^0(\xi)|_2$. Integrating both sides in ξ, we get the desired inequality. ∎

In order to estimate $\epsilon^{(k)}$, we assume that eigenvalues of $\boldsymbol{A}$ are included in an ellipse $E(c,e,a)$, centered at c, with focal distance e and semimajor axis a, which excludes the origin.

Assume a kth order polynomial p satisfies $p(0) = 1$. Then, since $d > 0$, $\xi \in \mathbb{R}$, $i\xi + d \neq 0$,

$$q_\xi(z) = p\left(\frac{i\xi + z}{i\xi + d}\right)$$

is of degree k in z and $q_\xi(-i\xi) = 1$. For fixed ξ,

$$\{q_\xi \in \mathbb{P}_k :\ q_\xi(-i\xi) = 1\} = \{q \in \mathbb{P}_k :\ q(-i\xi) = 1\}.$$

In fact, assume $q(z) = 1 + a_1(z + i\xi) + \cdots + a_k(z + i\xi)^k$. Define kth order polynomials

$$p(z) = 1 + a_1(d + i\xi)z + \cdots + a_k(d + i\xi)^k z^k,$$
$$q_\xi(z) = p\left(\frac{i\xi + z}{i\xi + d}\right),$$

and then $q(z) = q_\xi(z)$. Therefore, the above two sets are equal. Hence,

$$\begin{aligned}\epsilon^{(k)} &\le \max_{\xi\in\mathbb{R}} \min_{\substack{p\,\in\,\mathbb{P}_k\\ p(0)\,=\,1}} \max_{z\in E(c,e,a)} \left|p\left(\frac{i\xi + z}{i\xi + d}\right)\right| \\ &= \max_{\xi\in\mathbb{R}} \min_{\substack{q_\xi\,\in\,\mathbb{P}_k\\ q_\xi(-i\xi)\,=\,1}} \max_{z\in E(c,e,a)} |q_\xi(z)| \\ &= \max_{\xi\in\mathbb{R}} \min_{\substack{q\,\in\,\mathbb{P}_k\\ q(-i\xi)\,=\,1}} \max_{z\in E(c,e,a)} |q(z)|.\end{aligned}$$

By an estimate in [32, p. 192],

$$\min_{q\in\mathbb{P}_k,\, q(-i\xi)=1} \max_{z\in E(c,e,a)} |q(z)| = \frac{C_k\left(\frac{a}{e}\right)}{\left|C_k\left(\frac{c+i\xi}{e}\right)\right|},$$

where C_k is the kth order Chebyshev polynomial. Hence,

$$\epsilon^{(k)} \le \max_{\xi\in\mathbb{R}} \frac{C_k\left(\frac{a}{e}\right)}{\left|C_k\left(\frac{c+i\xi}{e}\right)\right|}.$$

The following is a key lemma for analyzing the convergence of CGMRES.

LEMMA 4.4. *If c, e are real and positive, then*

$$\min_{\xi\in\mathbb{R}} \left|C_k\left(\frac{c + i\xi}{e}\right)\right| = C_k\left(\frac{c}{e}\right).$$

Proof. By the definition of the Chebyshev polynomial $C_k(z)$, or by noticing that $C_k(z)$ has real coefficients, we have $C_k(\overline{z}) = \overline{C_k(z)}$. As an immediate consequence,

$$\min_{\xi\in\mathbb{R}} \left|C_k\left(\frac{c + i\xi}{e}\right)\right| = \min_{\xi\ge 0} \left|C_k\left(\frac{c + i\xi}{e}\right)\right|.$$

In order to find the minimum, define

$$f(\xi) = \left|C_k\left(\frac{c + i\xi}{e}\right)\right|^2.$$

Let $w = \rho e^{i\theta}$ such that

$$\frac{1}{e}(c + i\xi) = \frac{1}{2}(w + w^{-1}).$$

Then

$$(\rho + \rho^{-1})\cos\theta = \frac{2c}{e}, \quad (\rho - \rho^{-1})\sin\theta = \frac{2\xi}{e}.$$

By a computation, we get

$$f'(\xi) = \frac{k}{2e\beta^2}[(\rho^{2k} - \rho^{-2k})(\rho + \rho^{-1})\sin\theta - 2(\rho - \rho^{-1})\cos\theta\sin(2k\theta)],$$

where $\beta^2 = |\rho e^{i\theta} - \rho^{-1}e^{-i\theta}|^2$.

If $\rho = 1$ or $\theta = 0$, then $\xi = 0$ and $f'(\xi) = 0$. However, in our case, since $c > e$, $\rho \neq 1$. Otherwise, $\cos\theta = \frac{c}{e} > 1$, which is impossible.

By the relation $(\rho - \rho^{-1})\sin\theta = \frac{2\xi}{e}$, if $0 < \theta < \frac{\pi}{2}$, then $\rho > 1$; if $-\frac{\pi}{2} < \theta < 0$, then $\rho < 1$. By symmetry, in order to prove that $f'(\xi) > 0$ for $\xi > 0$, it is enough to prove that

$$g(\rho, \theta) = (\rho^{2k} - \rho^{-2k})(\rho + \rho^{-1})\sin\theta - 2(\rho - \rho^{-1})\cos\theta\sin(2k\theta) > 0$$

for $0 < \theta < \frac{\pi}{2}$ and $\rho > 1$. By introduction, one can prove that $\sin(m\theta) \leq m\sin\theta$ for $0 < \theta < \frac{\pi}{2}$ and $m \in \mathbb{N}$. Also, notice that $\rho + \rho^{-1} > 2$. Hence, in order to prove $g(\rho, \theta) > 0$, it is sufficient to prove that

$$g(\rho) = \rho^{2k} - \rho^{-2k} - k(\rho^2 - \rho^{-2}) > 0$$

for $\rho > 1$. However, $g(\rho)$ is a strictly increasing function for $\rho > 1$, which implies $g(\rho) > g(1) = 0$. This proves the lemma. ∎

Notice that the Chebyshev polynomial $C_k(z)$ has properties $|C_k(z)| = |C_k(\overline{z})| = |C_k(-z)| = |C_k(-\overline{z})|$. Therefore, by the last lemma, we have the following consequence.

COROLLARY 4.5. *If c is a complex number and $e \neq 0$ is real, then*

$$\min_{\xi\in\mathbb{R}} \left| C_k\left(\frac{c + i\xi}{e}\right)\right| = C_k\left(\frac{|Re\, c|}{|e|}\right).$$

Finally, we have the following convergence result for CGMRES.

THEOREM 4.6. *Assume $\boldsymbol{A} = \boldsymbol{X}\Lambda\boldsymbol{X}^{-1} = \boldsymbol{M} - \boldsymbol{N}$, $\boldsymbol{M} = d\boldsymbol{I}$, $d > 0$. The spectrum of $\boldsymbol{A}$ is included in ellipse $E(c, e, a)$, centered at c, with focal distance e and semimajor axis a, which excludes origin. Then the residual $\boldsymbol{r}^k = \boldsymbol{\psi} - (\boldsymbol{I} - \mathcal{K})\boldsymbol{x}^k$ obtained by the CGMRES algorithm satisfies the estimate*

$$\|\boldsymbol{r}^k\|_{\mathbb{L}^2} \leq \kappa_2(\boldsymbol{X})\frac{C_k\left(\frac{a}{e}\right)}{C_k\left(\frac{c}{e}\right)}\|\boldsymbol{r}^0\|_{\mathbb{L}^2}.$$

Remark. Thus, CGMRES applied to (2.4) is bounded by the same rate of convergence as GMRES applied to the associated problem (2.6).

5. Biconvolution acceleration methods. The CGMRES algorithm demonstrates that the use of convolution can accelerate waveform relaxation in a manner similar to GMRES applied to the associated linear algebra problem. However, there is still something unsatisfying about the algorithm in that even for Hermitian $\boldsymbol{A}$ one must use CGMRES. In this section, we turn our attention to a method that exploits Hermitian properties of the matrix $\boldsymbol{A}$, namely, biconvolution CG (BiCCG). The development of BiCCG will use a convolution bilinear form in place of the convolution inner product used by CGMRES.

We begin by defining the following bilinear form:

$$[\boldsymbol{x}, \boldsymbol{y}]_\star(t) = \sum_{i=1}^{n}(x_i \star y_i)(t) \in \mathbb{L}^1(\mathbb{R}, \mathbb{R}).$$

Remarks.

1. Convolution between a function and a vector of functions is the same as defined in section 4. However, the convolution bilinear form $[\cdot,\cdot]_\star$ is different from the previously defined inner product $\langle\cdot,\cdot\rangle_\star$.
2. Notice that $[\boldsymbol{x},\boldsymbol{y}]_\star\widehat{\ }(\xi) = [\widehat{\boldsymbol{x}}(\xi),\widehat{\boldsymbol{y}}(\xi)]$. Here $[\cdot,\cdot]$ is the bilinear form in $\mathbb{C}^n$ defined by
$$[\boldsymbol{z},\boldsymbol{w}] = \sum_{i=1}^{n} z_i w_i$$
for $\boldsymbol{w},\boldsymbol{z}\in\mathbb{C}^n$. Note that this is *not* the typical inner product in $\mathbb{C}^n$. It is, however, the bilinear form used in the BiCG algorithm [8], hence the name "biconvolution CG."
3. Since, by assumption, functions and vectors are compactly supported, $f\star\boldsymbol{x}$ and $[\boldsymbol{x},\boldsymbol{y}]_\star$ are in fact in $\mathbb{L}^2$.

5.1. The BiCCG algorithm. The CG algorithm is a popular and effective iterative method for solving symmetric positive definite systems of equations [11, 13]. Waveform extensions (using scalar parameters) of the CG algorithm are not well-defined, even for symmetric positive definite $\boldsymbol{A}$, since, in general, the operator $\mathcal{K}$ is not self-adjoint with respect to the $\mathbb{L}^2$ inner product. On the other hand, as we will see below, it is possible to develop a well-defined waveform extension to CG using convolution, i.e., the BiCCG algorithm.

Definition. An operator $\mathcal{A}:\ \mathbb{L}^2([0,T],\mathbb{R}^n)\to\mathbb{L}^2([0,T],\mathbb{R}^n)$ is called *convolution self-adjoint* if for any $\boldsymbol{u},\ \boldsymbol{v}\in\mathbb{L}^2([0,T],\mathbb{R}^n),\ [\mathcal{A}\boldsymbol{u},\boldsymbol{v}]_\star = [\boldsymbol{u},\mathcal{A}\boldsymbol{v}]_\star$.

Remark. If $\boldsymbol{A}$ is Hermitian, the operator $\mathcal{K}$ is convolution self-adjoint with respect to the convolution bilinear form "$[\cdot,\cdot]_\star$."

Definition. An operator $\mathcal{A}:\ \mathbb{L}^2([0,T],\mathbb{R}^n)\to\mathbb{L}^2([0,T],\mathbb{R}^n)$ is called *convolution definite* if for any nonzero $\boldsymbol{u}\in\mathbb{L}^2([0,T],\mathbb{R}^n),\ [\mathcal{A}\boldsymbol{u},\boldsymbol{u}]_\star\neq 0$ in the $\mathbb{L}^2$ sense.

Since $(s+d)^{-1}(s\boldsymbol{I}+\boldsymbol{A})$ is symmetric positive definite for positive real s and symmetric positive definite $\boldsymbol{A}$, the following lemma follows immediately from the above definitions and the equivalence (via the Fourier transform) between $(\boldsymbol{I}-\mathcal{K})$ and $(s+d)^{-1}(s\boldsymbol{I}+\boldsymbol{A})$.

LEMMA 5.1. *If $\boldsymbol{A}$ is real symmetric with splitting $\boldsymbol{M}-\boldsymbol{N},\ \boldsymbol{M}=d\boldsymbol{I},\ d>0$, then $(\boldsymbol{I}-\mathcal{K})$ is convolution self-adjoint. Furthermore, if $\boldsymbol{A}$ is positive definite, then $(\boldsymbol{I}-\mathcal{K})$ is also convolution definite.*

If $\mathcal{A}$ is convolution self-adjoint and convolution definite on $\mathbb{L}^2([0,T],\mathbb{R}^n)$, then we can define the following BiCCG algorithm (analogous to CG).

ALGORITHM 5 (BiCCG). Let $\mathcal{A}: C_0([0,\infty),\mathbb{R}^n)\to C_0([0,\infty),\mathbb{R}^n)$ be a bounded linear operator. As before, $\mathcal{A}$ is extendable to $\overline{C_0([0,\infty),\mathbb{R}^n)}$, the vector-valued generalized function space, which is again a vector space over $\mathbb{Q}$. Let $\boldsymbol{f}\in C_0([0,\infty),\mathbb{R}^n)$.

1. Pick $\boldsymbol{x}^0\in C_0([0,\infty),\mathbb{R}^n)$ and compute $\boldsymbol{r}^0=\boldsymbol{f}-\mathcal{A}\boldsymbol{x}^0,\ \boldsymbol{p}^0=\boldsymbol{r}^0$.
2. For $j=0,1,\ldots$ until converged,
$$\begin{aligned}
\alpha_j &= [\boldsymbol{r}^j,\boldsymbol{r}^j]_\star/[\mathcal{A}\boldsymbol{p}^j,\boldsymbol{p}^j]_\star\\
\boldsymbol{x}^{j+1} &= \boldsymbol{x}^j+\alpha_j\star\boldsymbol{p}^j\\
\boldsymbol{r}^{j+1} &= \boldsymbol{r}^j-\alpha_j\star\mathcal{A}\boldsymbol{p}^j,\\
\beta_j &= [\boldsymbol{r}^{j+1},\boldsymbol{r}^{j+1}]_\star/[\boldsymbol{r}^j,\boldsymbol{r}^j]_\star\\
\boldsymbol{p}^{j+1} &= \boldsymbol{r}^{j+1}+\beta_j\star\boldsymbol{p}^j.
\end{aligned}$$

Note that the Fourier transform of the BiCCG algorithm for $\mathcal{A}=\boldsymbol{I}-\mathcal{K}$ is the BiCG algorithm (see [6, 8, 13]) for matrix $\boldsymbol{A}(\xi)=(i\xi\boldsymbol{I}+\boldsymbol{M})^{-1}(i\xi\boldsymbol{I}+\boldsymbol{A})$. Therefore,

we can view the BiCG algorithm (for a complex system) as a continuation of the CG algorithm (for a real system). Also, notice that by [5] there is a CG algorithm for matrix $\boldsymbol{A}(\xi) = (i\xi + d)^{-1}(i\xi \boldsymbol{I} + \boldsymbol{A})$.

5.2. Laplace and Fourier transforms. To analyze the convergence of the BiCCG algorithm, we first give some definitions for weighted Sobolev spaces (see [16]) and introduce some results related to Laplace and Fourier transforms.

Definition. For a real number $\lambda > 0$, $\alpha \geq 0$, define the weighted Sobolev space $\mathbb{H}^{\alpha}_{\lambda}(\mathbb{R}, \mathbb{R})$ according to

$$\mathbb{H}^{\alpha}_{\lambda}(\mathbb{R}, \mathbb{R}) = \{u \in \mathbb{L}^2(\mathbb{R}, \mathbb{R}) : \ (\lambda^2 + |\xi|^2)^{\alpha/2} |\widehat{u}(\xi)| \in \mathbb{L}^2(\mathbb{R}, \mathbb{R})\}.$$

In particular, if $\lambda = 1$, then the weighted $\mathbb{H}^{\alpha}_1$ space is the regular $\mathbb{H}^{\alpha}$ space [16]. Also notice that $\mathbb{H}^{\alpha}_{\lambda}$ can be defined by another equivalent norm as follows:

$$\mathbb{H}^{\alpha}_{\lambda}(\mathbb{R}, \mathbb{R}) = \{u \in \mathbb{L}^2(\mathbb{R}, \mathbb{R}) : \ (|\lambda| + |\xi|)^{\alpha} |\widehat{u}(\xi)| \in \mathbb{L}^2(\mathbb{R}, \mathbb{R})\}.$$

Now assume $\boldsymbol{A} = \boldsymbol{U} \operatorname{diag}(\lambda_1, \ldots, \lambda_n) \boldsymbol{U}^T = \boldsymbol{U} \Lambda \boldsymbol{U}^T$ is real symmetric positive definite, where $\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}$, and $0 < \lambda_1 \leq \cdots \leq \lambda_n$. Define a weighted $\mathbb{H}^{\alpha}_{\mathbf{A}}$ as follows:

$$\mathbb{H}^{\alpha}_{\mathbf{A}}(\mathbb{R}, \mathbb{R}^n) = \{\boldsymbol{u} = (u_1, \ldots, u_n)^T : (\lambda_j + |\xi|)^{\alpha} |(\boldsymbol{U}^T \widehat{u})_j(\xi)| \in \mathbb{L}^2(\mathbb{R}, \mathbb{R})\}.$$

By the definition, we can see that $\mathbb{H}^{\alpha}_{\mathbf{A}} = \mathbb{H}^{\alpha}_{\lambda_1} \times \mathbb{H}^{\alpha}_{\lambda_2} \times \cdots \times \mathbb{H}^{\alpha}_{\lambda_n}$. Also, on $\mathbb{H}^{\alpha}_{\mathbf{A}}$, we can define an inner product

$$\begin{aligned}(\boldsymbol{u}, \boldsymbol{v})_{\alpha} &= \int_{\mathbb{R}} \widehat{\boldsymbol{u}}^T(\xi) (|\xi| \boldsymbol{I} + \boldsymbol{A})^{2\alpha} \overline{\widehat{\boldsymbol{v}}(\xi)} d\xi \\ &= \int_{\mathbb{R}} \langle \widehat{\boldsymbol{u}}(\xi), \widehat{\boldsymbol{v}}(\xi) \rangle_{(|\xi| \mathbf{I} + \mathbf{A})^{2\alpha}} d\xi,\end{aligned}$$

which makes $\mathbb{H}^{\alpha}_{\mathbf{A}}$ a Hilbert space with norm

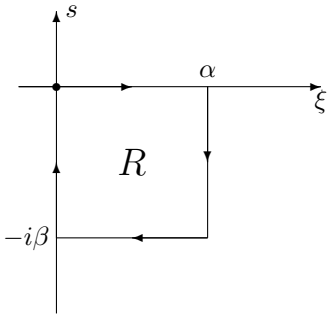
$$\|\boldsymbol{u}\|_{\mathbb{H}^{\alpha}_{\mathbf{A}}} = \left(\sum_{j=1}^{n} \int_{\mathbb{R}} (\lambda_j + |\xi|)^{2\alpha} |(\boldsymbol{U}^T \widehat{u})_j(\xi)|^2 d\xi \right)^{1/2}.$$

Now we want to study the relationship between the Laplace transform and the Fourier transform.

Since $f \in \mathbb{L}^2([0, T], \mathbb{R})$, and the trivial extension of f to $\mathbb{L}^2(\mathbb{R}, \mathbb{R})$ is compactly supported, the Fourier transform of f is therefore entire in $\mathbb{C}$. By Cauchy's theorem,

$$\int_R (\widehat{f}(z))^2 dz = 0,$$

where R is a rectangle as shown in the figure at the right. Therefore,



$$\begin{aligned}&\int_0^{\alpha} (\widehat{f}(\xi))^2 d\xi + \int_0^{-\beta} (\widehat{f}(\alpha + is))^2 d(is) \\ &+ \int_{\alpha}^{0} (\widehat{f}(\xi - i\beta))^2 d\xi + \int_{-\beta}^{0} (\widehat{f}(is))^2 d(is) = 0.\end{aligned}$$

For fixed β, let $\alpha \to \infty$. By a property of the Fourier transform (see [12]), $\lim_{\alpha\to\infty} \widehat{f}(\alpha + is) = 0$. Also, since $(\widehat{f}(\alpha + is))^2$ is absolutely integrable,

$$\lim_{\alpha\to\infty} \int_0^{-\beta} (\widehat{f}(\alpha + is))^2 d(is) = i \int_0^{-\beta} \lim_{\alpha\to\infty} (\widehat{f}(\alpha + is))^2 ds = 0.$$

The third term becomes $-\int_0^\infty (\widehat{f}(\xi - i\beta))^2 d\xi$. Let $\beta \to \infty$, and then by

$$\widehat{f}(\xi - i\beta) = \frac{1}{\sqrt{2\pi}} \int_0^\infty e^{-\beta x - i\xi x} f(x) dx \to 0$$

we get that

$$\lim_{\beta\to\infty} \int_0^\infty (\widehat{f}(\xi - i\beta))^2 d\xi = 0.$$

Hence, we conclude that

$$i \int_0^\infty (\widehat{f}(\xi))^2 d\xi = \int_0^\infty (\widehat{f}(-is))^2 ds \geq 0, \tag{5.1}$$

since $\widehat{f}(-is)$ is the Laplace transform and is real. We denote the Laplace transform by $\mathcal{L}f(s) = \widehat{f}(-is)$. This relationship between the Laplace and Fourier transforms plays an important role in what follows.

For subsequent analysis, we require the following definition (see [16]).

Definition.

$$\mathbb{H}_0^\alpha ([0,\infty), \mathbb{R}) = \text{closure of } C_0^\infty [0,\infty) \text{ in } \mathbb{H}^\alpha ([0,\infty), \mathbb{R}),$$

where $C_0^\infty[0,\infty) = \{f : f \in C^\infty[0,\infty) \text{ with compact support }\}$.

If we assume that $f \in \mathbb{H}_0^{1/2}([0,\infty), \mathbb{R})$, then by Cauchy's theorem again,

$$\int_R z(\widehat{f}(z))^2 dz = 0,$$

so that

$$-\int_0^\infty \xi(\widehat{f}(\xi))^2 d\xi = \int_0^\infty s(\mathcal{L}f(s))^2 ds \geq 0. \tag{5.2}$$

Combining (5.1) and (5.2) yields a key lemma.

LEMMA 5.2. *For $f \in \mathbb{H}_0^{1/2}([0,\infty), \mathbb{R})$, $\lambda \in \mathbb{C}$,*

$$i \int_0^\infty (i\xi + \lambda)(\widehat{f}(\xi))^2 d\xi = \int_0^\infty (s + \lambda)(\mathcal{L}f(s))^2 ds.$$

Because of this key equality, we can give the following definitions.

Definition. Assume $\boldsymbol{f} = (f_1, \ldots, f_n)^T$, $\boldsymbol{g} = (g_1, \ldots, g_n)^T \in \mathbb{H}_0^{1/2}([0,\infty), \mathbb{R}^n)$, and real symmetric and positive definite $\boldsymbol{A} = \boldsymbol{U}\,\mathrm{diag}(\lambda_1, \ldots, \lambda_n)\boldsymbol{U}^T$ with $\boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I}$ and $0 < \lambda_1 \leq \cdots \leq \lambda_n$. Define an $\boldsymbol{A}$-weighted inner product by

$$\begin{aligned}\langle \boldsymbol{f}, \boldsymbol{g} \rangle_{\mathbf{A}} &= i \int_0^\infty \langle \hat{\boldsymbol{f}}, \hat{\boldsymbol{g}} \rangle_{i\xi\mathbf{I}+\mathbf{A}} d\xi \\ &= \sum_{j=1}^n i \int_0^\infty (i\xi + \lambda_j) \left(\widehat{\boldsymbol{U}\boldsymbol{f}}\right)_j (\xi) \left(\widehat{\boldsymbol{U}\boldsymbol{g}}\right)_j (\xi) d\xi.\end{aligned}$$

Definition. For $\boldsymbol{f} \in \mathbb{H}_0^{1/2}([0,\infty),\mathbb{R}^n)$, define a norm by

$$\|\boldsymbol{f}\|_{\mathbb{K}_{\mathbf{A}}^{1/2}} = \langle \boldsymbol{f}, \boldsymbol{f}\rangle_{\mathbf{A}}^{1/2}.$$

The new normed space is again a Hilbert space and is denoted by $\mathbb{K}_{\mathbf{A}}^{1/2}([0,\infty),\mathbb{R}^n)$.

Remarks.

1. By integration over a wedge instead of over a rectangle as in Lemma 5.2, one can see that
$$ie^{i\theta}\int_0^\infty (ire^{i\theta}+\lambda)(\widehat{f}(re^{i\theta}))^2 dr = \int_0^\infty (s+\lambda)(\mathcal{L}f(s))^2 ds.$$
This means the integral
$$i\int_{R(\theta)} (iz+\lambda)(\widehat{f}(z))^2 dz = \int_{R(\theta+\pi/2)} (z+\lambda)(\mathcal{L}f(z))^2 dz$$
is invariant in θ, where $R(\theta) = \{re^{i\theta} : 0 \le r < +\infty\}$ is a ray starting at the origin.
2. The $\mathbb{K}_{\mathbf{A}}^{1/2}$ norm is bounded by the $\mathbb{H}_{\mathbf{A}}^{1/2}$ norm.
3. By Theorem 11.1, Chapter 1 in [16], $\mathbb{H}_0^{1/2}(\mathbb{R}^+,\mathbb{R}^n) = \mathbb{H}^{1/2}(\mathbb{R}^+,\mathbb{R}^n)$, where $\mathbb{R}^+ = \{x \in \mathbb{R} : x > 0\}$ is the positive half ray of $\mathbb{R}$.

5.3. Convergence of the BiCCG algorithm. By using the projection properties and Chebyshev polynomials, one can prove the following well-known theorem [32] for CG applied to the linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$.

THEOREM 5.3. *Assume matrix $\boldsymbol{A}$ is real symmetric and positive definite. Let $\{\boldsymbol{x}_m\}$ be the sequence of approximate solutions obtained by the CG algorithm and let $\boldsymbol{x}_*$ be the exact solution. Then the iterates satisfy*

$$\|\boldsymbol{x}_* - \boldsymbol{x}_m\|_{\mathbf{A}} \le 2\left[\frac{\sqrt{\kappa(\boldsymbol{A})}-1}{\sqrt{\kappa(\boldsymbol{A})}+1}\right]^m \|\boldsymbol{x}_* - \boldsymbol{x}_0\|_{\mathbf{A}},$$

where $\kappa(\boldsymbol{A}) = \frac{\lambda_{\max}}{\lambda_{\min}}$ is the condition number of the matrix $\boldsymbol{A}$.

For the BiCCG algorithm we can prove a similar convergence theorem.

THEOREM 5.4. *Let $\boldsymbol{M} - \boldsymbol{N}$ be a splitting of a real symmetric positive definite matrix $\boldsymbol{A}$ with $\boldsymbol{M} = d\boldsymbol{I}$, $d > 0$. Then the BiCCG algorithm applied to* (2.4) *generates a sequence of iterates $\{\boldsymbol{x}^m\}$ that satisfy the weighted $\frac{1}{2}$-estimates*

$$\|\boldsymbol{x}^* - \boldsymbol{x}^m\|_{\mathbb{K}_{\mathbf{A}}^{1/2}} \le 2\left[\frac{\sqrt{\kappa(\boldsymbol{A})}-1}{\sqrt{\kappa(\boldsymbol{A})}+1}\right]^m \|\boldsymbol{x}^* - \boldsymbol{x}^0\|_{\mathbb{K}_{\mathbf{A}}^{1/2}}, \tag{5.3}$$

where $\boldsymbol{x}^*$ is the exact solution.

Proof. By taking Fourier transform on the BiCCG algorithm, we can see that, for each m, $\boldsymbol{x}^* - \boldsymbol{x}^m \in \mathbb{L}^2([0,\infty),\mathbb{R}^n)$. Since $\boldsymbol{x}^* - \boldsymbol{x}^m$ is differentiable on $[0,T]$ and $(\boldsymbol{x}^* - \boldsymbol{x}^m)(0) = 0$, by Urysohn's lemma, we can assume that $\boldsymbol{x}^* - \boldsymbol{x}^m$ is also compactly supported on $[0,\infty)$. Therefore $\boldsymbol{x}^* - \boldsymbol{x}^m$ is in fact in $\mathbb{H}_0^{1/2}([0,\infty),\mathbb{R}^n)$. By taking Laplace transform on the BiCCG algorithm, we can see that $\mathcal{L}(\boldsymbol{x}^m)(s)$ is an approximate solution obtained by the CG algorithm applied to real matrix $(s+d)^{-1}(s\boldsymbol{I}+\boldsymbol{A})$. By Theorem 5.3, for each fixed $s \ge 0$,

$$\|\mathcal{L}(\boldsymbol{x}^* - \boldsymbol{x}^m)(s)\|_{\frac{s\mathbf{I}+\mathbf{A}}{s+d}} \le 2\left[\frac{\sqrt{\kappa(s\boldsymbol{I}+\boldsymbol{A})}-1}{\sqrt{\kappa(s\boldsymbol{I}+\boldsymbol{A})}+1}\right]^m \|\mathcal{L}(\boldsymbol{x}^* - \boldsymbol{x}^0)(s)\|_{\frac{s\mathbf{I}+\mathbf{A}}{s+d}}. \tag{5.4}$$

Since $\boldsymbol{A}$ is positive definite, there exists an unitary matrix $\boldsymbol{U}$ such that

$$\boldsymbol{A} = \boldsymbol{U}\Lambda\boldsymbol{U}^T,$$

where $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$, and $0 < \lambda_1 \leq \cdots \leq \lambda_n$. It is easy to see that

$$\kappa(s\boldsymbol{I} + \boldsymbol{A}) = \frac{s + \lambda_n}{s + \lambda_1}.$$

Notice that

$$\max_{s\in\mathbb{R}^+\cup\{0\}} \kappa(s\boldsymbol{I} + \boldsymbol{A}) = \max_{s\in\mathbb{R}^+\cup\{0\}} \left[\frac{s + \lambda_n}{s + \lambda_1}\right] = \kappa(\boldsymbol{A}).$$

Since $f(x) = \frac{\sqrt{x}-1}{\sqrt{x}+1}$ is an increasing function on $[0, +\infty)$, we have

$$\begin{aligned}\frac{\sqrt{\kappa(s\boldsymbol{I} + \boldsymbol{A})} - 1}{\sqrt{\kappa(s\boldsymbol{I} + \boldsymbol{A})} + 1} &\leq \frac{\sqrt{\max_{s\in\mathbb{R}^+\cup\{0\}} \kappa(s\boldsymbol{I} + \boldsymbol{A})} - 1}{\sqrt{\max_{s\in\mathbb{R}^+\cup\{0\}} \kappa(s\boldsymbol{I} + \boldsymbol{A})} + 1} \\ &= \frac{\sqrt{\kappa(\boldsymbol{A})} - 1}{\sqrt{\kappa(\boldsymbol{A})} + 1}.\end{aligned}$$

By (5.4) we get

$$\|\mathcal{L}(\boldsymbol{x}^* - \boldsymbol{x}^m)(s)\|^2_{\frac{s\mathbf{I}+\mathbf{A}}{s+d}} \leq 4\left[\frac{\sqrt{\kappa(\boldsymbol{A})} - 1}{\sqrt{\kappa(\boldsymbol{A})} + 1}\right]^{2m} \|\mathcal{L}(\boldsymbol{x}^* - \boldsymbol{x}^0)(s)\|^2_{\frac{s\mathbf{I}+\mathbf{A}}{s+d}}. \tag{5.5}$$

Multiplying both sides of (5.5) by $(s + d)$, we obtain

$$\|\mathcal{L}(\boldsymbol{x}^* - \boldsymbol{x}^m)(s)\|^2_{s\mathbf{I}+\mathbf{A}} \leq 4\left[\frac{\sqrt{\kappa(\boldsymbol{A})} - 1}{\sqrt{\kappa(\boldsymbol{A})} + 1}\right]^{2m} \|\mathcal{L}(\boldsymbol{x}^* - \boldsymbol{x}^0)(s)\|^2_{s\mathbf{I}+\mathbf{A}}. \tag{5.6}$$

Integrating both sides of (5.6) with respect to s and taking the square root gives the desired inequality (5.3). □

Remarks.

1. Thus, under the weighted $\mathbb{K}^{1/2}_{\mathbf{A}}$ norm, BiCCG applied to (2.4) is bounded by the same rate of convergence as CG applied to the associated problem (2.6).
2. Another important point to note is that, for the Laplace transform, we have the pointwise inequality (5.4). For the Fourier transform, it is not known whether a similar pointwise inequality is true or not. Such an inequality with the Fourier transform is unnecessary, however, because, by Theorem 5.4, it is the integral of the Fourier transforms of $\boldsymbol{x}^* - \boldsymbol{x}^m$ and $\boldsymbol{x}^* - \boldsymbol{x}^0$ which must satisfy a similar inequality.

Notice that, since $((\boldsymbol{I} - \mathcal{K})^n)\hat{} = ((\boldsymbol{I} - \mathcal{K})\hat{})^n$, and the CG algorithm terminates in finite steps, we have the following result regarding finite termination.

COROLLARY 5.5. *The BiCCG algorithm applied to* (2.4) *terminates in finite steps.*

Remark. For the algebraic equation (2.5), the CG algorithm terminates in finite steps. For the differential equation (2.1), we should expect that finite termination is possible [34]. Although the finite termination property is typically not important in practice, the fact that convolution Krylov subspace methods exhibit this property is another indication that they are the "right" generalization from linear algebra problems to waveform problems. The Hilbert space methods in section 3 do not exhibit finite termination.

6. Numerical experiments. In this section, we present preliminary experimental results using convolution Krylov subspace methods. For our model problem, we take the one-dimensional heat equation with unit spatial dimension and $T = 64$ for the temporal dimension. The problem is discretized with 64 spatial points and 32 temporal points and is integrated using backward-Euler.

Two convolution Krylov subspace methods are examined: BiCCG and the convolution variant of the generalized conjugate residual algorithm [4], CGCR. The CGCR algorithm is included rather than convolution GMRES because, although it is theoretically equivalent to GMRES, it is much simpler to implement. Thus, the CGCR results should be taken to be indicative of CGMRES.

The experimental code for BiCCG was written in C++, using the CG module from the IML++ class library [3]. Although IML++ was developed for solving linear systems of equations, by using it with a waveform class and by overloading the appropriate operators, the same CG code was able to be used for both linear algebra problems and waveform problems. The code for CGCR was similarly based on a GCR module (auxiliary to the IML++ library distribution). The experimental code for CSOR and WGMRES was written in C. The convolution kernel for CSOR was obtained according to the algorithms given in [30, 31].

Elements in $\mathbb{Q}$ were applied to functions by first convolving with the numerator and then deconvolving with the denominator. The deconvolution was implemented in a variety of ways, including computation of $(\frac{\hat{f}}{\hat{g}})^{\vee}$, inversion of the convolution based recurrence, and a Newton iteration. All approaches gave similar results.

Figure 6.1 compares the convergence rates of waveform relaxation, WGMRES, CSOR, and BiCCG applied to solving the model problem. We also show the performance of CG applied to solving the corresponding linear algebra problem. For this experiment, BiCCG and CGCR have remarkably better convergence behavior than the other waveform methods and is, in fact, much better than CG itself. This be-

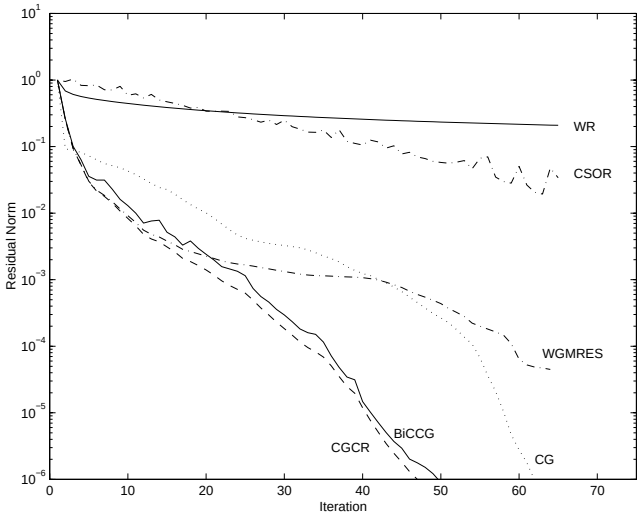


FIG. 6.1. *Convergence of waveform relaxation, WGMRES, CSOR, BiCCG, and CGCR applied to solving a model initial value problem. Also shown is the convergence of CG applied to the corresponding linear algebra problem.*

havior is typical of BiCCG and CGCR over a wide variety of experiments that we conducted (with a particular *caveat*, which we discuss below).

7. Discussion.

7.1. Deconvolution. It is well known that, in general, deconvolution is an ill-posed problem and that therefore numerical deconvolution will be ill-conditioned. This ill-posedness arises because the "divisor" in the deconvolution may have zero values at particular frequencies.

Theoretically, the convolution Krylov subspace algorithms avoid this ill-posedness naturally because of our choice of functions (i.e., compactly supported $\mathbb{L}^2$ functions). That is, the deconvolution is well-defined in the $\mathbb{L}^2$ sense.

Practically, however, the issue becomes somewhat more delicate, because we need to be concerned not simply with possible zero divisors, but with divisors that are small in a relative numerical sense. Since the convolution Krylov algorithms can be interpreted as being simultaneous iterative processes in the frequency domain, one at each frequency, small divisors can occur during the solution process if the residual values at particular frequencies are small relative to others.

The numerical ill-conditioning can be avoided in part by ensuring that each temporal frequency is present in the initial waveform $\boldsymbol{x}^0$. In our experiments, we effected this by setting $\boldsymbol{x}^0$ to have random values as a function of $t \neq 0$. Unfortunately, for problems having a large number of timepoints, higher temporal frequencies will tend to converge at a much higher rate than the lower frequencies, and numerical instabilities due to deconvolution may appear. Practical implementations of convolution Krylov subspace algorithms (if there turn out to be such things) should be able to circumvent this difficulty via windowing (which may be attractive for memory conservation reasons at any rate) or perhaps by restarting. Alternatively, it may be possible to modify the algorithms in such a way as to equalize the rates of convergence at all frequencies, or through the incorporation of some kind of regularization procedure.

7.2. Conclusion. As should be evident from this paper, convolution Krylov subspace methods are tremendously interesting, and we have scratched only the surface here. These methods appear to be the "right" generalization of linear algebra acceleration techniques to waveform relaxation. Moreover, they open some entirely new lines of inquiry about Krylov subspace iterations. For instance, the vector space defined by convolution with generalized function is seemingly more abstract than $\mathbb{R}^n$ or $\mathbb{L}^2$, where Krylov subspace algorithms are normally thought to be appropriate. The geometry of Hilbert space is explicitly present only in the transform domain. Finally, there have been a number of algorithms developed recently for the efficient iterative solution, large nonsymmetric linear systems of equations—QMR [9] and Bi-CGSTAB [39] to name just two. Adaptation of these and other methods to the convolution case should be relatively straightforward in terms of description and implementation (although, as with BiCCG and CGMRES, analysis may be somewhat less straightforward). However, a comprehensive experimental study of an assortment of convolution Krylov subspace methods, particularly if applied to practical application problems, would help to shed light on whether or not these methods will be practical in real life.

Acknowledgments. Convolution Krylov subspace algorithms have tantalized the authors for several years. During that time, we have had helpful discussions on this topic with Boris Bäumer, Roland Freund, Alex Himonas, Ken Jackson, Tom Manteuffel, Mark Reichelt, Stefan Vandewalle, and Jacob White and we wish to thank

them. The authors also wish to thank the referees for their careful reading and helpful suggestions for strengthening this paper.

REFERENCES

[1] L. V. Ahlfors, *Complex Analysis*, 3rd ed., McGraw-Hill, New York, 1979.
[2] B. Bäumer, *A Vector-Valued Operational Calculus and Abstract Cauchy Problems*, Ph.D. thesis, Louisiana State University, Baton Rouge, LA, 1997.
[3] J. Dongarra, A. Lumsdaine, X. Niu, R. Pozo, and K. Remington, *A sparse matrix library in C++ for high performance architectures*, in Proceedings of the Object Oriented Numerics Conference, Sun River, OR, 1994.
[4] H. C. Elman, *Iterative Methods for Large Sparse Nonsymmetric Systems of Linear Equations*, Ph.D. thesis, Computer Science Department, Yale University, New Haven, CT, 1982.
[5] V. Faber and T. Manteuffel, *Necessary and sufficient conditions for the existence of a conjugate gradient method*, SIAM J. Numer. Anal., 21 (1984), pp. 352–362.
[6] R. Fletcher, *Conjugate gradient methods for indefinite systems*, in Numerical Analysis, G. Watson, ed., Springer-Verlag, Berlin, New York, 1975.
[7] C. Foiaş, *Approximation des opérateurs de J. Mikusinski par des fonctions continues*, Studia Math., 21 (1961), pp. 73–74.
[8] R. W. Freund, *Conjugate gradient-type methods for linear systems with complex symmetric coefficient matrices*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 425–448.
[9] R. W. Freund and N. M. Nachtigal, *A quasi-minimal residual method for non-Hermition linear systems*, Numer. Math., 60 (91), pp. 315–339.
[10] R. M. Hayes, *Iterative methods of solving linear problems on Hilbert space*, in Contributions to the Solution of Systems of Linear Equations and the Determination of Eigenvalues, Nat. Bur. Standards Appl. Math. 39, O. Taussky, ed., U.S. Government Printing Office, Washington, D.C., 1954, pp. 71–103.
[11] M. R. Hestenes and E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.
[12] L. Hörmander, *The Analysis of Linear Partial Differential Operators*. I., 2nd ed., Springer-Verlag, Berlin, New York, 1990.
[13] C. Lanczos, *Solution of systems of linear equations by minimized iterations*, J. Research Nat. Bur. Standards, 49 (1952), pp. 33–53.
[14] B. Leimkuhler, *Estimating waveform relaxation convergence*, SIAM J. Sci. Comput., 14 (1993), pp. 872–889.
[15] E. Lelarasmee, A. E. Ruehli, and A. L. Sangiovanni-Vincentelli, *The waveform relaxation method for time domain analysis of large scale integrated circuits*, IEEE Trans. CAD, 1 (1982), pp. 131–145.
[16] J. L. Lions and E. Magenes, *Non-Homogeneous Boundary Value Problems and Applications*, Vol. 1, Springer-Verlag, New York, 1972.
[17] C. Lubich, *Chebyshev acceleration of Picard-Lindelöf iteration*, BIT, 32 (1992), pp. 535–538.
[18] C. Lubich and A. Osterman, *Multigrid dynamic iteration for parabolic problems*, BIT, 27 (1987), pp. 216–234.
[19] A. Lumsdaine, *Theoretical and Practical Aspects of Parallel Numerical Algorithms for Initial Value Problems, with Applications*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 1992.
[20] A. Lumsdaine, M. W. Reichelt, J. M. Squyres, and J. K. White, *Accelerated waveform methods for parallel transient simulation of semiconductor devices*, IEEE Trans. CAD, 15 (1996), pp. 716–726.
[21] A. Lumsdaine and J. K. White, *Accelerating dynamic iteration methods with application to parallel semiconductor device simulation*, Numer. Funct. Anal. Optim., 16 (1995), pp. 395–414.
[22] A. Lumsdaine and D. Wu, *Spectra and pseudospectra of waveform relaxation operators*, SIAM J. Sci. Comput., 18 (1997), pp. 286–304.
[23] U. Miekkala and O. Nevanlinna, *Convergence of dynamic iteration methods for initial value problems*, SIAM J. Sci. Statist. Comput., 8 (1987), pp. 459–482.
[24] U. Miekkala and O. Nevanlinna, *Iterative Solution of Systems of Linear Differential Equations*, Acta Numer. 5, Cambridge University Press, Cambridge, UK, 1996, pp. 259–307.
[25] G. Miel, *Iterative refinement of the method of moments*, Numer. Funct. Anal. Optim., 9 (1987/1988), pp. 1193–1200.
[26] J. Mikusiński, *Operational Calculus*, Vol. 1, 2nd ed., Pergamon Press, Oxford, UK, 1983.

[27] J. Mikusiński, *Operational Calculus*, Vol. 2, 2nd ed., Pergamon Press, Oxford, UK, 1987.
[28] O. Nevanlinna, *Linear acceleration of Picard-Lindelöf iteration*, Numer. Math., 57 (1990), pp. 147–156.
[29] P. Omari, *On the fast convergence of a Galerkin-like method for equations of the second kind*, Math. Z., 201 (1989), pp. 529–539.
[30] M. Reichelt, *Accelerated Waveform Relaxation Techniques for the Parallel Transient Simulation of Semiconductor Devices*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 1993.
[31] M. W. Reichelt, J. K. White, and J. Allen, *Optimal convolution SOR acceleration of waveform relaxation with application to parallel simulation of semiconductor devices*, SIAM J. Sci. Comput., 16 (1995), pp. 1137–1158.
[32] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS, Boston, MA, 1996.
[33] Y. Saad and M. H. Schultz, *GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 7 (1986), pp. 856–869.
[34] R. D. Skeel, *Waveform iteration and the shifted Picard splitting*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 756–776.
[35] K. Skornik, *On the Foiaş theorem on convolution of continuous functions*, in Complex Analysis and Applications '85, Publ. House Bulgar. Acad. Sci., Sofia, Bulgaria, 1986.
[36] E. C. Titchmarsh, *The zeros of certain integral functions*, Proc. London Math. Soc., 25 (1926), pp. 283–302.
[37] L. N. Trefethen, *private communication*, 1992.
[38] L. N. Trefethen, *Pseudospectra of matrices*, in Proceedings of 14th Dundee Biennial Conference on Numerical Analysis, 1991.
[39] H. A. van der Vorst, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 631–644.
[40] S. Vandewalle, *Parallel Multigrid Waveform Relaxation for Parabolic Problems*, Teubner-Skripten zur Numerik, B. G. Teubner, Stuttgart, Germany, 1993.
[41] S. Vandewalle and R. Piessens, *Efficient parallel algorithms for solving initial-boundary value and time-periodic parabolic partial differential equations*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 1330–1346.
[42] Y. V. Vorobyev, *Method of Moments in Applied Mathematics*, Gordon and Breach, New York, 1965.
[43] J. K. White and A. Sangiovanni-Vincentelli, *Relaxation Techniques for the Simulation of VLSI Circuits*, Engineering and Computer Science Series, Kluwer Academic Publishers, Norwell, MA, 1986.