Waveform Relaxation with Fast Direct Methods as Preconditioner

Jo Simoens
Stefan Vandewalle

Report TW 279, May 1998

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

Waveform Relaxation with Fast Direct Methods as Preconditioner

Jo Simoens
Stefan Vandewalle

Report TW 279, May 1998

Department of Computer Science, K.U.Leuven

Abstract

For a restricted class of parabolic PDEs one can devise a practical numerical solver with a parallel complexity that is theoretically optimal. The method uses a multidimensional FFT to decouple the unknowns in the spatial domain into independent scalar ODEs. These are discretized to give recurrence relations in the time dimension solved by parallel cyclic reduction. This is the *FFT/CR* algorithm.

We discuss the use of FFT/CR as a preconditioner to iteratively solve more general parabolic PDEs. This approach naturally leads to a waveform relaxation scheme. Waveform relaxation was developed as an iterative method for solving large systems of ODEs. It is the continuous-in-time analogue of stationary iterative methods for linear algebraic equations. Using the FFT/CR solver as a preconditioner preserves most of the potential for concurrency that accounts for the attractiveness of waveform relaxation with simple preconditioners like Jacobi or red-black Gauss-Seidel, while showing an important advantage: the convergence rate of the resulting iteration is independent of the mesh size used in the spatial discretization. The method can be accelerated by applying an appropriate scaling of the system before preconditioning.

Keywords : parabolic partial differential equations, waveform relaxation, dynamic iteration, fast Fourier transform
AMS(MOS) Classification : 65M20, 65F10, 65T10

WAVEFORM RELAXATION WITH FAST DIRECT METHODS AS PRECONDITIONER

JO SIMOENS[†‡] AND STEFAN VANDEWALLE[†§]

Abstract. For a restricted class of parabolic PDEs one can devise a practical numerical solver with a parallel complexity that is theoretically optimal. The method uses a multidimensional FFT to decouple the unknowns in the spatial domain into independent scalar ODEs. These are discretized to give recurrence relations in the time dimension solved by parallel cyclic reduction. This is the *FFT/CR* algorithm.

We discuss the use of FFT/CR as a preconditioner to iteratively solve more general parabolic PDEs. This approach naturally leads to a waveform relaxation scheme. Waveform relaxation was developed as an iterative method for solving large systems of ODEs. It is the continuous-in-time analogue of stationary iterative methods for linear algebraic equations. Using the FFT/CR solver as a preconditioner preserves most of the potential for concurrency that accounts for the attractiveness of waveform relaxation with simple preconditioners like Jacobi or red-black Gauss-Seidel, while showing an important advantage: the convergence rate of the resulting iteration is independent of the mesh size used in the spatial discretization. The method can be accelerated by applying an appropriate scaling of the system before preconditioning.

Key words. parabolic partial differential equations, waveform relaxation, dynamic iteration, fast Fourier transform

AMS subject classifications. 65M20, 65F10, 65T10

1. Introduction.

We consider the numerical solution of parabolic partial differential equations (PPDEs) of initial-boundary value type. A lower bound for the complexity of any solver of such a problem follows from an information-theoretical analysis that was elaborated by Worley in [30]: if N is the number of points in the space-time grid, the minimal sequential time complexity is $\mathcal{O}(N)$, and the parallel complexity is $\mathcal{O}(\log N)$. The *parallel complexity* of a parallel algorithm is defined here as the asymptotic time complexity assuming the available number of processors is unlimited and disregarding interprocessor communication.

Over the last few years, several parallel multigrid methods have been developed for the numerical solution of PPDEs. These algorithms have polylog parallel complexity [14, 28], but they are not optimal: their parallel complexity is strictly larger than the lower bound presented above. Yet, for a restricted class of parabolic problems, one can devise an algorithm that does indeed have optimal parallel complexity. This is the case when the parabolic operator is separable and has constant coefficients. The use of a multidimensional FFT to decouple the unknowns in the spatial domain, and parallel cyclic reduction to solve the recurrence relations in the time dimension, then results in a fast and asymptotically optimal parallel direct PPDE solver.

The method is inspired by fast direct methods for elliptic PDEs (EPDEs). Here *fast direct methods* refers to direct solvers with a sequential time complexity that is strictly less than quadratic in the number of unknowns. The reduced complexity of these methods comes in general at the cost of more limited applicability: they only apply to specific classes of problems. FFT-based fast Poisson solvers are classic examples. For a survey of the various algorithms of this kind, see e.g. [3, 5, 12,

[†]Katholieke Universiteit Leuven, Department of Computer Science, Celestijnenlaan 200A, B-3001 Heverlee, Belgium (`jo.simoens` and `stefan.vandewalle@cs.kuleuven.ac.be`). This paper presents research results of the Belgian Program on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture. The scientific responsibility rests with its authors.

[‡]Research Assistant of the Fund for Scientific Research – Flanders, Belgium (F.W.O.)

[§]Post-doctoral Research Fellow of the Fund for Scientific Research – Flanders, Belgium (F.W.O.)

27, 29]. The use of fast direct methods as preconditioners for solving more general elliptic problems emerged much at the same time as the methods themselves. The theory by D'yakonov and Gunn [6, 9] is the basis for the preconditioning done in [1] and [4]. Here, we will use a more recent theoretical framework.

The suggestion of applying the FFT methods developed for EPDEs to parabolic problems has been made by several authors [26, e.g.] and the method can even be traced back to Fourier himself [7]. More recently, however, Horton and Vandewalle [28] have drawn attention to its potential in the context of parallel computing. The parallel complexity of the resulting method was shown to be theoretically optimal. It is a logical next step to consider using fast direct PPDE solvers as preconditioners in solving more general parabolic problems. Doing this naturally leads to a waveform relaxation iteration.

The *waveform relaxation* (WR) method differs from most standard iterative techniques in that it iterates on functions of a continuous time variable. The method originates from electrical network simulation [20, 25]. Its convergence behavior has been studied extensively in a series of papers by Miekkala and Nevanlinna [22, 23, 24].

Keras [18, 19] proposes a WR iteration using symmetric Toeplitz matrices in the preconditioner. In particular, the author studies the numerical solution of PPDEs with preconditioners that are discretized parabolic operators with constant coefficients. The Toeplitz systems which arise when an implicit time integration scheme is employed can be solved by fast FFT-based methods. The fast solver is used as part of an additive Schwartz domain decomposition method [17]. Our work continues on these ideas. By using the literature on EPDEs, we are able to obtain more general results and simplify proofs. We show that some of the more sophisticated techniques that have been used in EPDE solvers—block decoupling and scaling—are applicable in WR methods for PPDEs as well. Furthermore, preconditioning directly with fast PPDE solvers allows us to fully exploit the waveform relaxation context by also introducing parallelism in the time dimension.

This paper is structured as follows. The model problem and its discretization is presented in §2. A section on fast direct methods for PPDEs follows, in which we discuss the underlying principles and identify the class of problems to which these methods apply. Section 4 reviews the basic WR convergence theory that will be used later on. In §5, the fast methods are employed as preconditioners for variable coefficient self-adjoint PPDEs in a WR iteration. We present a bound for the convergence factor that is independent of the mesh size. In §6 we discuss the selection of the preconditioner, and consider the effect of a scaling of the original problem in order to attempt to speed up convergence when dealing with more difficult problems. The computational complexity is studied in §7. The paper concludes in §8 with some numerical results.

2. Model problem and notations. We consider the numerical solution of the parabolic initial-boundary value problem

$$\begin{cases} \frac{\partial}{\partial t}u(x,t) + \mathcal{L}u(x,t) = f(x,t) \ , & x \in \Omega \ , \quad t \in \Omega_t \\ u(x,0) = u_0(x) \ , & x \in \Omega \end{cases} \tag{2.1}$$

with time-independent linear boundary conditions. The operator $\mathcal{L}$ is a (formally) self-adjoint linear second order elliptic operator with time-independent coefficients,

$$\mathcal{L} = -\frac{\partial}{\partial x_1}\big(a_1(x_1,x_2)\frac{\partial}{\partial x_1}\big) - \frac{\partial}{\partial x_2}\big(a_2(x_1,x_2)\frac{\partial}{\partial x_2}\big) + c(x_1,x_2) \ . \tag{2.2}$$

The time interval can be bounded, $\Omega_t = [0,T]$, or unbounded, $\Omega_t = [0,\infty)$. The spatial domain Ω is an open bounded domain, and $\bar{\Omega}$ denotes its closure. We here restrict ourselves to the case where Ω is a two-dimensional rectangle, $\Omega = \Omega_1 \times \Omega_2$. The extension to higher-dimensional problems is immediate.

Let $\Omega^h = \Omega_1^h \times \Omega_2^h$ be a rectangular grid defined by the ordered sets of points $\Omega_i^h = \{x_j^i\}_{j=1}^{n_i} \subset \Omega_i$. Write the grid as an $n_2 \times n_1$ matrix

$$U(t) = [u_1(t) \dots u_{n_1}(t)] \ , \quad u_i(t) = \left[u(x_i^1, x_1^2) \dots u(x_i^1, x_{n_2}^2)\right]^{\mathrm{T}} \ ,$$

and define the operator $\operatorname{vec}[u_1 \dots u_{n_1}] := [u_1^{\mathrm{T}} \dots u_{n_1}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^N$, $N = n_1 \cdot n_2$. Then the grid U is traversed in *lexicographical order* if $u^h(t) = \operatorname{vec} U(t)$. In the same way we denote the discrete right hand side $f^h(t) = \operatorname{vec} F(t)$. A spatial finite difference discretization of (2.1) on Ω^h yields the ODE system

$$\begin{cases} \frac{d}{dt} u^h(t) + A u^h(t) = f^h(t) \ , & t \in \Omega_t \ , \\ u^h(0) = u_0^h \ . \end{cases} \tag{2.3}$$

If $\mathcal{L}$ is a separable operator of the form (2.2), i.e., $\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2$, with

$$\mathcal{L}_i = -\frac{\partial}{\partial x_i}\big(a_i(x_i)\frac{\partial}{\partial x_i}\big) + c_i(x_i) \ ,$$

and is discretized as the sum of the discretizations of $\mathcal{L}_1$ and $\mathcal{L}_2$ on Ω^h, the lexicographical ordering of the unknowns allows us to write

$$A = \big(A_1 \otimes I_{n_2}\big) + \big(I_{n_1} \otimes A_2\big) = A_1 \oplus A_2 \ , \tag{2.4}$$

where A_1 and A_2 are the discretization matrices of $\mathcal{L}_1$ and $\mathcal{L}_2$ on Ω_1 and Ω_2, respectively. The symbols '$\otimes$' and '$\oplus$' denote the Kronecker product and the Kronecker sum of matrices [8], and I_n is the $n \times n$ identity matrix. We refer to (2.4) as a *separable discretization* of $\mathcal{L}$.

A practical algorithm for solving (2.3) will employ a numerical time integration method that is suited for solving stiff systems of ordinary differential equations. The precise nature of the ODE integration method is not essential to the remainder of the paper, although we will usually assume the use of an implicit linear multistep method (LMM) with a fixed a priori determined time-step.

3. Fast direct methods for PPDE.

3.1. A lower bound on the complexity. Worley [30] uses an information theoretical analysis to establish algorithm-independent lower bounds on the time complexity of parallel PDE solvers. This is done by estimating the minimal number of data values needed to compute an approximation of the solution at a given point to a fixed accuracy.

The parabolic PDE (2.1) satisfies the assumptions of [30, §3]. Let N be the number of spatial grid points and m the number of discrete time levels used by the time-integration method. The total number of points in the space-time grid equals Nm. The complexity of bringing the error down to discretization level on P processors is then bounded from below by an expression of the form

$$\mathcal{O}\bigg(\max\bigg\{\frac{Nm}{P}, \log_2 Nm\bigg\}\bigg) \ ,$$

measured in floating point operations. This includes the sequential setting. Hence a PDE solver is called *optimal* if its time complexity is $\mathcal{O}(Nm/P)$ for $P = \mathbf{o}(Nm)$ processors and is $\mathcal{O}(\log Nm)$ for $\mathcal{O}(Nm)$ processors or more. In the latter case the notion of parallel complexity is in order.

3.2. An optimal direct solver. An optimal parallel solver can be found for certain classes of parabolic PDEs of the form (2.1), as was suggested in [14]. Assume the discretization matrix A in (2.3) is diagonalizable by a fast transform based on the FFT. Section 3.3 will discuss for what A this is the case. Denote the ith eigenvalue of A by $\lambda_{A,i}$, and let the ith column of S be the corresponding eigenvector. Then $A = S\Lambda_A S^{-1}$.

The optimal parallel direct solver is composed of three sequential steps. By using the FFT, (2.3) is first transformed into a decoupled system of ODEs,

$$\frac{d}{dt}\big(S^{-1}u^h\big) + \Lambda_A\big(S^{-1}u^h\big) = S^{-1}f^h \ . \tag{3.1}$$

The equations of this system are then solved independently, leading to the transformed solution $S^{-1}u^h$. Finally, the solution u^h of the original equation is regained by the inverse transform.

Consider a separable discretization with matrix $A = A_1 \oplus A_2$ where, for the moment, we only assume A_1 to be diagonalizable by means of an FFT method, i.e., $A_1 = S_1\Lambda_{A_1}S_1^{-1}$ with $\Lambda_{A_1} = \mathrm{diag}[\lambda_{A_1,i}]_{i=1}^{n_1}$. We then have

$$\frac{d}{dt}\big(S_1^{-1}\otimes I_{n_2}\big)u^h + \big(\Lambda_{A_1}\oplus A_2\big)\big(S_1^{-1}\otimes I_{n_2}\big)u^h = \big(S_1^{-1}\otimes I_{n_2}\big)f^h \ .$$

The matrix $\Lambda_{A_1}\oplus A_2$ is block-diagonal. The transform S_1^{-1} acting on the rows of U decouples the equations into

$$\frac{d}{dt}\hat{u}^h + \big(\Lambda_{A_1}\oplus A_2\big)\hat{u}^h = \hat{f}^h \ , \tag{3.2}$$

with $\hat{u}^h = \big(S_1^{-1}\otimes I_{n_2}\big)\,\mathrm{vec}\,U = \mathrm{vec}\big(US_1^{-\mathrm{T}}\big)$ and an analogous notation for the right-hand side. Rewriting (3.2) by columns of $\hat{u}^h$ and $\hat{f}^h$, we now have n_1 independent systems, each of dimension n_2,

$$\frac{d}{dt}\hat{u}_i + \big(\lambda_{A_1,i}I_{n_2} + A_2\big)\hat{u}_i = \hat{f}_i \ .$$

The case where only A_2 is diagonalizable, i.e., $A_2 = S_2\Lambda_{A_2}S_2^{-1}$, is of course analogous. The matrix $A_1\oplus\Lambda_{A_2}$ is then block-diagonal after reordering.

When both A_1 and A_2 can be diagonalized, it immediately follows that system (2.3) can be fully decoupled and written as

$$\frac{d}{dt}\big(S_1^{-1}\otimes S_2^{-1}\big)u^h + \big(\Lambda_{A_1}\oplus\Lambda_{A_2}\big)\big(S_1^{-1}\otimes S_2^{-1}\big)u^h = \big(S_1^{-1}\otimes S_2^{-1}\big)f^h \ .$$

Full diagonalization is achieved through a two-dimensional transform. If n_1 and n_2 are both products of small primes, the forward or backward transform using P processors takes $\mathcal{O}(\max\{(Nm\log n_i)/P\,,\,\log n_i\})$ flops for each dimension.

Solving the decoupled ODEs is a trivially parallel task. With the use of a classical time-integration scheme, such as an implicit linear multistep method (LMM), additional parallelization is possible in the time dimension [14]. The recurrence relation to which a LMM reduces an ODE can be written as a banded lower triangular Toeplitz system, and can be solved by parallel cyclic reduction. Solving a single ODE discretized on by using m time-levels is then done in $\mathcal{O}(\log m)$ time, independent of the order of the recurrence relation. In all, the parallel complexity when doing full decoupling is $\mathcal{O}(\log Nm)$.

In case of partial decoupling in the x_1 dimension, using an implicit LMM means $n_2\times n_2$-systems have to be solved. This can be done in $\mathcal{O}(n_2)$ time if the matrix A_2 is a band matrix or periodic band matrix with $\mathcal{O}(1)$ bandwidth. Block recursive doubling could be considered to introduce additional parallelism in the time dimension. The parallel complexity of solving the block recurrence relations will, however, in general be larger than $\mathcal{O}(\log n_2 m)$ and will then no longer be optimal.

In both cases we will refer to this solver as the *FFT/CR algorithm*.

Remark. Partial decoupling is not restricted to discretization matrices of separable operators. Block-diagonalization of a matrix $A_1 \oplus A_2 = A_1 \otimes I_{n_2} + I_{n_1} \otimes A_2$ has an immediate generalization in the form

$$A_1 \otimes C + I_{n_1} \otimes A_2 = \left(S_1 \otimes I_{n_2}\right)\left(\Lambda_{A_1} \otimes C + I_{n_1} \otimes A_2\right)\left(S_1^{-1} \otimes I_{n_2}\right) .$$

The matrix $\Lambda_{A_1} \otimes C + I_{n_1} \otimes A_2$ is still block-diagonal, irrespective of C. We shall return to this once we have identified the class of discretized operators to which the fast methods that we consider are applicable.

3.3. Fast diagonalizable matrices. We now identify a class of discretization matrices than can be diagonalized by means of FFT-based techniques.

3.3.1. 1D discretizations. It is well-known that common discretizations of the operator

$$\mathcal{L} = -a\frac{\partial^2}{\partial x^2} + c \ , \quad a, c \in \mathbb{R} \ , \quad x \in [\alpha, \beta] \ , \tag{3.3}$$

combined with certain sets of boundary conditions, have the desired properties. We collect and summarize this knowledge, which is scattered over the literature, in the following proposition, and outline a proof.

PROPOSITION 3.1. *FFT-based methods apply to 1D operators of the form* (3.3) *in a centered-difference discretization of arbitrary order on a uniform mesh with homogeneous Dirichlet, homogeneous Neumann, periodic, anti-periodic or symmetric (i.e. Dirichlet–Neumann) boundary conditions.*

Proof. The eigenfunctions of operator $\mathcal{L}$ in (3.3) with the given boundary conditions are trigonometric functions restricted to the interval $[\alpha, \beta]$. They have analytic extensions into $\mathbf{C}^\infty(\mathbb{R})$ which are trigonometric functions on $\mathbb{R}$. These can be described by appropriate extensions of the domain, depending upon the boundary conditions:

boundary condition	*extension law*		
homogeneous Neumann in β	symmetric	$u(\beta + x) = u(\beta - x)$	
homogeneous Dirichlet in β	antisymmetric	$u(\beta + x) = -u(\beta - x)$	(3.4)
periodic	periodic	$u(\beta + x) = u(\alpha + x)$	
antiperiodic	antiperiodic	$u(\beta + x) = -u(\alpha + x)$	

Consider the vector space spanned by the extended eigenfunctions of $\mathcal{L}$. It is the space of analytic functions possessing the properties (3.4) corresponding to the boundary conditions of the problem at hand, and it will be denoted $\mathbf{C}^\infty_{\mathrm{BC}}(\mathbb{R})$.

We now show that centered-difference approximations of (3.3) considered as operators from $\mathbf{C}^\infty_{\mathrm{BC}}(\mathbb{R})$ to $\mathbf{C}^\infty(\mathbb{R})$ have the same eigenvectors as (3.3). The way the interval $[\alpha, \beta]$ is extended to $\mathbb{R}$ enforces the boundary conditions.

Let D denote the continuous derivative operator, and δ the centered finite difference operator on an infinite uniform grid of size h. We can formally write

$$\delta^2 = 4 \sinh^2 \frac{hD}{2} \ .$$

This is an even function of D, so eigenfunctions of D^2 are eigenfunctions of δ^2. On the other hand, every truncated power series expansion of

$$D^2 = \frac{4}{h^2} \operatorname{arcsinh}^2 \frac{\delta}{2} = h^{-2}\delta^2 \left(I - \frac{1}{12}\delta^2 + \frac{1}{90}\delta^4 - \dots \right)$$

gives a centered difference approximation $\widetilde{D^2}$ of D^2. Since each of these expressions is a polynomial in δ^2, the eigenfunctions of δ^2—and *a fortiori* those of D^2—are eigenfunctions of $\widetilde{D^2}$. Hence, the eigenfunctions of $\mathcal{L}$ are also eigenfunctions of its centered difference discretization $\widetilde{\mathcal{L}} := -a\widetilde{D^2} + c$.

We assume that the grid is such that both α and β coincide with a grid point. As an operator on discrete functions on the grid, $\widetilde{\mathcal{L}}$ is well-defined and has as eigenfunctions the eigenfunctions of $\widetilde{\mathcal{L}}$ on $\mathbf{C}^\infty_{\mathrm{BC}}(\mathbb{R})$ sampled on the (infinite) grid. A discretization on the restriction of the grid to $[\alpha, \beta]$ is found by eliminating values at points outside the interval using the relations (3.4).

The resulting vectors are sampled versions of trigonometric functions and for the boundary conditions listed above are linearly independent. There exist fast FFT-based transforms for the decomposition in the basis they form. ☐

Remark. In [26] it is shown how a nonself-adjoint operator with constant coefficients can be transformed into the desired form (3.3).

3.3.2. 2D discretizations. As a consequence of the above, the FFT method also applies to separable two-dimensional operators that reduce to the sum of such one-dimensional operators as (3.3). More specifically, the self-adjoint constant coefficient operator

$$\mathcal{L} = -a_1 \frac{\partial^2}{\partial x_1^2} - a_2 \frac{\partial^2}{\partial x_2^2} + c \tag{3.5}$$

allows a fully diagonalizable discretization. The algorithm outlined in §3.2 for this case corresponds to the FFT method for EPDEs proposed by Boris and Roberts [2].

The more general separable operator

$$\mathcal{L} = -a_1 \frac{\partial^2}{\partial x_1^2} - \frac{\partial}{\partial x_2}\big(a_2(x_2)\frac{\partial}{\partial x_2}\big) + c(x_2) \ , \tag{3.6}$$

whose coefficients are constant in the x_1 dimension, can be partially decoupled. It has a discretization which is block-diagonalizable by a fast transform in the x_1 dimension, depending upon the boundary conditions. We then need another solver for the decoupled subsystems—for classic second-order discretizations a simple tridiagonal systems solver is sufficient. This method corresponds to that used by Hockney [10] for 5-point discretizations of the Poisson equation. As a sequential algorithm, it is faster than full diagonalization. The *FACR* method [11], consisting in a clever combination of Fourier analysis (FA) and cyclic reduction (CR), is even more efficient.

Finally, the nonseparable operator

$$\mathcal{L} = -\frac{\partial}{\partial x_1}\big(a_1(x_2)\frac{\partial}{\partial x_1}\big) - \frac{\partial}{\partial x_2}\big(a_2(x_2)\frac{\partial}{\partial x_2}\big) + c(x_2) \tag{3.7}$$

leads to a discretization of the form

$$A = A_1 \otimes C + I_{n_1} \otimes A_2 \ ,$$

where $C = \mathrm{diag}\big[a_1(x_j^2)\big]_{j=1}^{n_2}$, and where A_1 and A_2 are the discretizations of

$$\frac{\partial^2}{\partial x_1^2} \quad \text{c.q.} \quad \frac{\partial}{\partial x_2}\big(a_2(x_2)\frac{\partial}{\partial x_2}\big) + c(x_2) \ .$$

Hence, following the discussion in §3.2, we may conclude that partial FFT-based decoupling is also possible for discretization of operators of the form (3.7). With this $\mathcal{L}$, the PPDE (2.1) is an anisotropic diffusion equation with diffusion coefficients which vary only in x_2. Equations of this type have physical significance: they model diffusion in layered media. A numerical example follows in §8.

4. A review of waveform relaxation results. This section briefly recalls some results on the convergence of waveform relaxation for initial-value problems of the form

$$\begin{cases} B\frac{d}{dt}u^h(t) + Au^h(t) = f^h(t) \ , \quad t \in \Omega_t \\ u^h(0) = u_0^h \ , \end{cases}$$

which is slightly more general than (2.3). The method is defined by choosing a splitting of the matrices $B = M_B - N_B$ and $A = M_A - N_A$, and consists of an iteration of the form

$$M_B \frac{d}{dt} u^{(\nu+1)}(t) + M_A u^{(\nu+1)}(t) = N_B \frac{d}{dt} u^{(\nu)}(t) + N_A u^{(\nu)}(t) + f^h(t) \ . \tag{4.1}$$

As a starting value one usually takes $u^{(0)}(t) \equiv u_0^h$. Iteration (4.1) is also called a *dynamic iteration* as opposed to the corresponding *static iteration* for solving $Ax = b$,

$$M_A x^{(\nu+1)} = N_A x^{(\nu)} + b \ , \quad \nu \geq 0 \ . \tag{4.2}$$

It is shown in [15] how iteration (4.1) can be written explicitly as a successive approximation scheme

$$u^{(\nu+1)} = \mathcal{K} u^{(\nu)} + \varphi \ . \tag{4.3}$$

The operator $\mathcal{K}$ is called the *waveform-relaxation operator*. It is well-known that the iteration (4.3) converges if and only if the spectral radius $\rho(\mathcal{K})$ of the WR operator is smaller than unity. The asymptotic convergence factor then equals $\rho(\mathcal{K})$. For the reader's convenience, we recall the formulae for the spectral radii of the WR operator. An essential distinction is to be made between the cases of finite and infinite time intervals. On a finite time interval $[0, T]$, we denote the WR operator with $\mathcal{K}_T$, and on an infinite time interval with $\mathcal{K}_\infty$.

In [15] it is shown that if all eigenvalues of $M_B^{-1} M_A$ have positive real parts,

$$\rho(\mathcal{K}_T) = \rho(M_B^{-1} N_B) \tag{4.4}$$

and

$$\rho(\mathcal{K}_\infty) = \sup_{\mathrm{Re}\, z \geq 0} \rho(K(z)) = \sup_{\xi \in \mathbb{R}} \rho(K(i\xi)) \ , \tag{4.5}$$

where

$$K(z) := (M_B z + M_A)^{-1} (N_B z + N_A) \tag{4.6}$$

is the *dynamic iteration matrix* of $\mathcal{K}_\infty$. If $N_B = 0$, convergence is superlinear [22], i.e., $\rho(\mathcal{K}_T) = 0$. Note that $\rho(K(0)) = \rho(M_A^{-1} N_A)$ is the asymptotic convergence factor of the static iteration (4.2). Hence the dynamic iteration cannot converge faster, asymptotically, than the corresponding static iteration.

Although waveform relaxation is a continuous-time method, in practice the ODE system (4.1) is discretized in time. For systems that are semi-discretized PPDEs, the use of an A(α)-stable LMM with fixed timestep is in general satisfactory. Let τ denote the timestep and m the (possibly infinite) number of time levels. Time discretization of (4.1) with an implicit k-step LMM leads to

$$\sum_{j=0}^{k} \left(\frac{1}{\tau}\alpha_j M_B + \beta_j M_A\right) u_{n+j}^{(\nu+1)} = \sum_{j=0}^{k} \left(\frac{1}{\tau}\alpha_j N_B + \beta_j N_A\right) u_{n+j}^{(\nu)} + \sum_{j=0}^{k} \beta_j f_{n+j}^h \ , \quad -k < n \leq m-k \ , \tag{4.7}$$

where $\beta_k \neq 0$. The k starting values $u_j^{(\nu+1)}, -k < j \leq 0$ are assumed to be given.

To simplify notations, we write $x_\tau = [\, x(j\tau)\,]_{j=1}^m, T = m\tau$ on finite time intervals $[0,T]$ and $x_\tau = \{\, x(j\tau)\,\}_{j=1}^\infty$ on an infinite time interval. The discrete-time analogue of (4.3) is then

$$u_\tau^{(\nu+1)} = \mathcal{K}^\tau u_\tau^{(\nu)} + \varphi_\tau \ .$$

As with continuous-time WR operators, we distinguish between discrete-time WR operators on finite and on infinite intervals by writing them as $\mathcal{K}_T^\tau$ and $\mathcal{K}_\infty^\tau$, respectively.

Assuming that all poles of the dynamic iteration matrix $K(z)$ are in the interior of the scaled stability region $\frac{1}{\tau}S$ of the LMM, one has that [16]

$$\rho\big(\mathcal{K}_\infty^\tau\big) = \sup_{z \notin \frac{1}{\tau}\operatorname{int} S} \rho\big(K(z)\big) = \sup_{z \in \frac{1}{\tau}\partial S} \rho\big(K(z)\big).$$

By using the maximum principle, one can show that the supremum is attained on the border of the scaled stability region.

In numerical practice, one finds that the convergence of finite-interval discrete-time WR is better described by $\rho\big(\mathcal{K}_\infty^\tau\big)$ than by $\rho\big(\mathcal{K}_T^\tau\big)$. This behavior is explained by looking at the pseudospectra of $\mathcal{K}_T^\tau$ [21]. Furthermore, A-stable LMMs have the property that $\rho\big(\mathcal{K}_\infty^\tau\big) \leq \rho\big(\mathcal{K}_\infty\big)$. In many cases the equality holds, an example of which will be given in §5.2. This demonstrates that results about the infinite-interval, continuous-time WR operator $\mathcal{K}_\infty$ are significant in real-life computations, where the time domain is neither infinite nor continuous.

5. Preconditioning in waveform relaxation. We choose as preconditioner M_A a discretization matrix that is obtained by the same discretization scheme applied to an easier problem

$$\frac{\partial}{\partial t}u + \widetilde{\mathcal{L}}u = f \ ,$$

where $\widetilde{\mathcal{L}}$ is chosen as an approximation of $\mathcal{L}$ in the original problem that allows for a faster solution. The appropriate approximation criterion is given by the convergence factor of the preconditioned iterative method.

5.1. The static iteration. Convergence results in the field of preconditioning with fast direct solvers for EPDEs date back to the theory developed by D'yakonov and Gunn [6, 9]. They consider the following iteration

$$Mu^{(\nu+1)} = Mu^{(\nu)} - \omega(Au^{(\nu)} - f) \ ,$$

where A and M are symmetric positive definite (SPD) matrices and $\omega > 0$ is an over-relaxation parameter. For use in waveform relaxation we reframe these results in the theory developed by Miekkala and Nevanlinna [22]. The following theorem gives a convergence factor for the static iteration (4.2). It is an extension of a theorem due to Bank [1]. In [18], other arguments lead to an analoguous result for isotropic operators with no constant term.

THEOREM 5.1. *Let*

$$\mathcal{L} = -\frac{\partial}{\partial x_1}\big(a_1(x_1,x_2)\frac{\partial}{\partial x_1}\big) - \frac{\partial}{\partial x_2}\big(a_2(x_1,x_2)\frac{\partial}{\partial x_2}\big) + c(x_1,x_2) \ ,$$
$$\widetilde{\mathcal{L}} = -\frac{\partial}{\partial x_1}\big(\widetilde{a}_1(x_1,x_2)\frac{\partial}{\partial x_1}\big) - \frac{\partial}{\partial x_2}\big(\widetilde{a}_2(x_1,x_2)\frac{\partial}{\partial x_2}\big) + \widetilde{c}(x_1,x_2) \ ,$$

with $a_1, a_2, \widetilde{a}_1, \widetilde{a}_2, c, \widetilde{c} > 0$ *over* $\bar{\Omega}$. *Let* A *and* M_A *be the discretization matrices of* $\mathcal{L}$ *c.q.* $\widetilde{\mathcal{L}}$, *define* $N_A := M_A - A$, *and assume that the boundary conditions and the* *discretization scheme are linear and are such that A and M_A are symmetric positive definite whenever $\mathcal{L}$ and $\widetilde{\mathcal{L}}$ are positive definite. Then if*

$$\alpha := \max_{x\in\bar\Omega}\left\{ \left|\frac{a_1(x)-\widetilde{a}_1(x)}{\widetilde{a}_1(x)}\right| , \left|\frac{a_2(x)-\widetilde{a}_2(x)}{\widetilde{a}_2(x)}\right| , \left|\frac{c(x)-\widetilde{c}(x)}{\widetilde{c}(x)}\right| \right\} ,$$

it follows that $\rho\big(M_A^{-1}N_A\big) \le \alpha$.

Proof. Let

$$\mu_1 := \min_{(x_1,x_2)\in\bar\Omega}\left\{ \frac{a_1(x_1,x_2)}{\widetilde{a}_1(x_1,x_2)}, \frac{a_2(x_1,x_2)}{\widetilde{a}_2(x_1,x_2)}, \frac{c(x_1,x_2)}{\widetilde{c}(x_1,x_2)} \right\} ,$$

$$\mu_2 := \max_{(x_1,x_2)\in\bar\Omega}\left\{ \frac{a_1(x_1,x_2)}{\widetilde{a}_1(x_1,x_2)}, \frac{a_2(x_1,x_2)}{\widetilde{a}_2(x_1,x_2)}, \frac{c(x_1,x_2)}{\widetilde{c}(x_1,x_2)} \right\} ;$$

then $\mathcal{L}-\lambda_1\widetilde{\mathcal{L}}$ and $\lambda_2\widetilde{\mathcal{L}}-\mathcal{L}$ are positive definite for all $\lambda_1<\mu_1$ and $\lambda_2>\mu_2$. Using our assumptions on the discretization scheme, we find that for all $v\in\mathbb{R}^N$ with $\|v\|_2=1$ one has

$$\langle Av\,,v\rangle - \lambda_1\langle M_Av\,,v\rangle > 0 \quad \text{and} \quad \langle Av\,,v\rangle - \lambda_2\langle M_Av\,,v\rangle < 0 ,$$

which through a limit argument leads to

$$\frac{\langle Av\,,v\rangle}{\langle M_Av\,,v\rangle} \in [\,\mu_1\,,\,\mu_2\,] \subset \mathbb{R}_0^+ .$$

Since $[\mu_1,\mu_2]\subseteq[1-\alpha,1+\alpha]$ it then follows that

$$\max_{\|v\|_2=1}\left|\frac{\langle N_Av\,,v\rangle}{\langle M_Av\,,v\rangle}\right| \le \alpha .$$

On the other hand, since M_A is SPD and N_A is symmetric, we also have that

$$\max_{\|v\|=1}\left|\frac{\langle N_Av\,,v\rangle}{\langle M_Av\,,v\rangle}\right| = \rho(M_A^{-1}N_A) ,$$

which completes the proof. □

The established bound is independent of the mesh size. A somewhat sharper bound—which does depend on the mesh size—is obtained when μ_1 and μ_2 are taken as the minimum and maximum not over the whole of $\bar\Omega$ but merely over the points at which the respective coefficients are evaluated in the difference scheme.

Remark. A one-dimensional second-order centered-difference scheme on a regular grid with homogeneous Dirichlet boundary conditions as discussed in §3.3 satisfies the assumptions of Theorem 5.1.

The same result is easily generalized to accommodate periodic boundary conditions, where the discretization matrix can have a zero eigenvalue. The result also immediately applies to separable higher-dimensional discretizations as the eigenvalues of a Kronecker sum of two matrices are pairwise sums of the eigenvalues of the respective matrices.

5.2. The dynamic iteration. We now present a result which relates the convergence factor of a WR iteration with a SPD discretization matrix A to that of the corresponding static iteration. The theorem is given for the general formulation of the problem (4.1), and as such will allow us to study waveform relaxation for scaled PPDEs in §6.2.

THEOREM 5.2. *Let B, A, M_B, M_A be symmetric positive definite matrices. Then the continuous infinite-interval waveform relaxation operator in* (4.3) *satisfies*

$$\rho(\mathcal{K}_\infty) = \max\{\, \rho(M_A^{-1}N_A)\,,\, \rho(M_B^{-1}N_B)\,\}\ .$$

Proof. The first part follows along the lines of [22, Theorem 3.3]. If M_B and M_A are SPD, $M_B^{-1}M_A$ has only real positive eigenvalues. According to (4.5), the spectral radius $\rho(\mathcal{K}_\infty)$ is given by the supremum of $\rho\big(K(i\xi)\big)$ over all $\xi \in \mathbb{R}$. Let $\lambda(\xi)$ be an eigenvalue of $K(i\xi)$ with the eigenvector $x(\xi) \in \mathbb{C}^N$:

$$(M_B i\xi + M_A)^{-1}(N_B i\xi + N_A)\, x(\xi) = \lambda(\xi)\, x(\xi)\ .$$

Then an expression for $\lambda(\xi)$ can be found as

$$\langle N_B x\,, x\rangle i\xi + \langle N_A x\,, x\rangle = \lambda(\xi)\big(\langle M_B x\,, x\rangle i\xi + \langle M_A x\,, x\rangle\big)\ ,$$
$$\text{or}\quad \lambda(\xi) = \frac{\langle N_A x\,, x\rangle + i\langle N_B x\,, x\rangle\,\xi}{\langle M_A x\,, x\rangle + i\langle M_B x\,, x\rangle\,\xi}$$

since M_B and M_A are positive definite. All matrices are symmetric, so the inner products are real and

$$|\lambda(\xi)| = \sqrt{\frac{\langle N_A x\,, x\rangle^2 + \langle N_B x\,, x\rangle^2\xi^2}{\langle M_A x\,, x\rangle^2 + \langle M_B x\,, x\rangle^2\xi^2}}\ .$$

In this last expression the eigenvector x also depends on ξ. We have, however,

$$\frac{\langle N_A x\,, x\rangle^2 + \langle N_B x\,, x\rangle^2\xi^2}{\langle M_A x\,, x\rangle^2 + \langle M_B x\,, x\rangle^2\xi^2} \le \max\left\{ \frac{\langle N_A x\,, x\rangle^2}{\langle M_A x\,, x\rangle^2}\,, \frac{\langle N_B x\,, x\rangle^2}{\langle M_B x\,, x\rangle^2} \right\}\ ,$$

and the terms in the right hand side are bounded by $\rho(M_A^{-1}N_A)^2$ and $\rho(M_B^{-1}N_B)^2$, respectively. We conclude that $|\lambda(\xi)| \le \max\{\rho(M_A^{-1}N_A), \rho(M_B^{-1}N_B)\}$.

Conversely,

$$\rho(\mathcal{K}_\infty) = \sup_{\xi\in\mathbb{R}} \rho\big(K(i\xi)\big) \ge \rho\big(K(0)\big) = \rho\big(M_A^{-1}N_A\big)\ ,$$

and likewise

$$\rho(\mathcal{K}_\infty) \ge \lim_{\xi\to\infty} \rho\big(K(i\xi)\big) = \rho\big(M_B^{-1}N_B\big)\ ,$$

which proves the theorem. ☐

The convergence analysis of the dynamic iteration is thus reduced to the convergence of the static iteration, and we can make use of Theorem 5.1.

COROLLARY 5.3. *When in Theorem 5.2 it holds that $\rho(M_A^{-1}N_A) \ge \rho(M_B^{-1}N_B)$, then the supremum of $\rho\big(K(z)\big)$ over the right half plane is attained at the origin.*

The result of Corollary 5.3 is *a fortiori* true for the case where $B = M_B = I$ en $N_B = 0$. This was already pointed out by Miekkala and Nevanlinna [22].

COROLLARY 5.4. *When in Theorem* (5.2) *$\rho(M_A^{-1}N_A) \ge \rho(M_B^{-1}N_B)$, then for any A-stable convergent linear multistep method $\rho\big(\mathcal{K}_\infty^\tau\big) = \rho\big(\mathcal{K}_\infty\big)$.*

This property still holds true when the LMM is only A(α)-stable with sufficiently large α. How large that is, however, depends on the operator $\mathcal{K}_\infty$ and is not easily determined.

6. Preconditioning with fast direct methods.

6.1. Choosing the preconditioner. Having established what operators $\mathcal{L}$ in the parabolic problem (2.1) lend themselves to the direct PPDE solvers of §3.2, it is straightforward to choose the preconditioner minimizing the bound of Theorem 5.1.

For the broadest class of approximations, viz. nonseparable operators (3.7), the bound is minimized by setting

$$\begin{aligned} \widetilde{a}_1(x_2) &= \frac{1}{2}\left(\min_{x_1\in\bar{\Omega}_1} a_1(x_1,x_2) + \max_{x_1\in\bar{\Omega}_1} a_1(x_1,x_2)\right) \ , \\ \widetilde{a}_2(x_2) &= \frac{1}{2}\left(\min_{x_1\in\bar{\Omega}_1} a_2(x_1,x_2) + \max_{x_1\in\bar{\Omega}_1} a_2(x_1,x_2)\right) \ , \\ \widetilde{c}(x_2) &= \frac{1}{2}\left(\min_{x_1\in\bar{\Omega}_1} c(x_1,x_2) + \max_{x_1\in\bar{\Omega}_1} c(x_1,x_2)\right) \ . \end{aligned} \tag{6.1}$$

Then α gives the maximum relative deviation of the coefficients of the original operator $\mathcal{L}$ with respect to the coefficients of $\widetilde{\mathcal{L}}$ in the preconditioner. While the choice (6.1) minimizes the upper bound for the convergence factor, it may not be the best choice in the sense of guaranteeing the fastest convergence. In the context of a similar preconditioning, Concus and Golub [4] argue that, e.g., a weighted mean of $c(x_i^1,x_j^2)$ can be a better value for $\widetilde{c}(x_j^2)$.

In the same fashion the closest separable operator of the form (3.6) amenable to partial decoupling by FFT is characterized by

$$\widetilde{a}_1 = \frac{1}{2}\left(\min_{(x_1,x_2)\in\bar{\Omega}} a_1(x_1,x_2) + \max_{(x_1,x_2)\in\bar{\Omega}} a_1(x_1,x_2)\right)$$

and $\widetilde{a}_2$, $\widetilde{c}$ as in (6.1).

Finally, if $\widetilde{a}_2$ and $\widetilde{c}$ are also chosen to be constant, a constant coefficient operator (3.5) is recovered which allows full decoupling. This minimax constant coefficient preconditioner has already been proposed for isotropic problems by Keras in [18, 19], where also convergence results analoguous to Theorem 5.1 are given.

Note that while the value of α is independent of discretization mesh size, it does greatly vary with the coefficients of the operator $\mathcal{L}$. If these are close to constant, α is small and convergence will be fast. If however the coefficients vary over several orders of magnitude, convergence may be extremely slow, and one will have to resort to more sophisticated methods.

6.2. Scaling. By means of a transformation of the dependent variable, one can sometimes transform the original PDE into an equivalent problem whose coefficients show less variation. Writing $u(x_1,x_2) = d(x_1,x_2)w(x_1,x_2)$, with d positive over $\bar{\Omega}$ and twice continuously differentiable, the PPDE in the new variable w becomes

$$d^2\frac{\partial}{\partial t}w + \left(-\frac{\partial}{\partial x_1}(a_1 d^2\frac{\partial}{\partial x_1}) - \frac{\partial}{\partial x_2}(a_2 d^2\frac{\partial}{\partial x_2}) + p\right) w = d\,f \ , \tag{6.2}$$

where $p = d\mathcal{L}d$. Instead of scaling the discretized problem, we discretize the scaled problem. A similar approach has been applied to elliptic problems by Bank [1]. The PPDE (6.2) is discretized to give

$$\bar{B}\frac{d}{dt}w^h + \bar{A}w^h = Df^h \ , \quad D = \operatorname{diag}\operatorname{vec}[d(x_i^1,x_j^2)]_{i,j} \ , \quad \bar{B} = D^2 \ . \tag{6.3}$$

We consider preconditioning (6.3) with a fast direct PPDE solver for

$$\widetilde{d}^2\frac{\partial}{\partial t}w + \left(-\frac{\partial}{\partial x_1}(\widetilde{a}_1\frac{\partial}{\partial x_1}) - \frac{\partial}{\partial x_2}(\widetilde{a}_2\frac{\partial}{\partial x_2}) + \widetilde{c}\right) w = d\,f \ , \tag{6.4}$$

with discretization

$$\bar{M}_B \frac{d}{dt} w^h + \bar{M}_A w^h = D f^h \ , \quad \bar{M}_B = \operatorname{diag}\operatorname{vec}[\widetilde{d}^2(x_i^1, x_j^2)]_{i,j} \ . \tag{6.5}$$

Defining $\bar{N}_B = \bar{M}_B - \bar{B}$ and $\bar{N}_A = \bar{M}_A - \bar{A}$, we know by Theorem 5.2 that the WR operator after scaling satisfies

$$\rho(\mathcal{K}_\infty) = \max\{\, \rho(\bar{M}_A^{-1}\bar{N}_A) \, , \, \rho(\bar{M}_B^{-1}\bar{N}_B) \,\} \ .$$

The first term is bounded by Theorem 5.1, the second equals

$$\rho(\bar{M}_B^{-1}\bar{N}_B) = \max_{i,j} \left\{ \left| \frac{\widetilde{d}^2(x_i^1, x_j^2) - d^2(x_i^1, x_j^2)}{\widetilde{d}^2(x_i^1, x_j^2)} \right| \right\} \ .$$

If $\widetilde{d}$ and the other coefficients in (6.4) are chosen as constants, the system (6.5) is fully diagonalizable by FFT.

In [4] scaling with $d(x_1, x_2) = a^{-1/2}(x_1, x_2)$ is used to transform an elliptic operator $\mathcal{L} = -\nabla\big(a(x_1,x_2)\nabla\big)$ into an operator $\mathcal{M} = \big(-\Delta + p(x_1, x_2)\big)$ whose differential part is the Laplacian. A D'yakonov-Gunn iteration is then applied with as preconditioner the constant-coefficient approximation $\widetilde{\mathcal{M}} = \big(-\Delta + \widetilde{p}\big)$ for which FFT methods can be used.

Observe that this approach when applied to parabolic operators merely moves the difficulty from the static to the dynamic term. If for $h \to 0$, the unscaled problem has

$$\rho(M_A^{-1}N_A) \to C \quad \text{and} \quad \rho(M_B^{-1}N_B) = 0 \ ,$$

then after scaling with $d = a^{-1/2}$ it has

$$\rho(\bar{M}_A^{-1}\bar{N}_A) \geq 0 \quad \text{and} \quad \rho(\bar{M}_B^{-1}\bar{N}_B) \to C$$

for the optimal choice of $\widetilde{p}$. Thus in continuous-time waveform relaxation, where $\rho(\mathcal{K}_\infty)$ is given by Theorem 5.2, the presence of the dynamic term severely limits the potential benefits of scaling, and the function d should be chosen such that $\rho(\bar{M}_A^{-1}\bar{N}_A) \approx \rho(\bar{M}_B^{-1}\bar{N}_B)$.

In the discrete-time setting, however, there can be some advantage to scaling. As we recalled in §4, the supremum of $\rho\big(K(z)\big)$ over the right half complex plane gives the spectral radius $\rho\big(\mathcal{K}_\infty\big)$, and the supremum over the exterior $S^C := \mathbb{C}\setminus \frac{1}{\tau}S$ of the scaled stability region gives $\rho\big(\mathcal{K}_\infty^\tau\big)$. The closure of S^C is a bounded domain containing the origin, and lying in the right half plane if the LMM is A-stable. We then have, under the conditions of Theorem 5.2,

$$\rho(M_A^{-1}N_A) \leq \rho\big(\mathcal{K}_\infty^\tau\big) \leq \max\big\{\rho(M_A^{-1}N_A)\, , \, \rho(M_B^{-1}N_B)\big\} = \rho\big(\mathcal{K}_\infty\big) \ .$$

Reducing the spectral radius of the dynamic iteration matrix around the origin at the cost of increasing it at infinity is now less problematic. We will illustrate these issues in §8.

7. Computational complexity. The convergence factor of waveform relaxation preconditioned with an appropriate direct solver was shown to be essentially independent of the number of grid points in both space and time. Therefore the number of iterations needed to reduce the error by a given factor ε^{-1} is $\mathcal{O}(-\log\varepsilon)$ independently of the grid size. Bringing the error down to discretization level takes $\mathcal{O}(\log Nm)$ iteration steps.

Recalling the complexity of a single iteration step from §3.2, one sees that the overall complexity of the WR method with the preconditioners we have proposed is $\mathcal{O}((Nm\log N)\log Nm)$ sequentially and $\mathcal{O}\big((\log Nm)^2\big)$ in parallel.

TABLE 7.1

Comparison of the parallel complexity of some iterative PPDE solvers for an $N \times m$ space-time grid, assuming a variable-coefficient parabolic operator.

method	**parallel complexity**
full multigrid time-stepping	$\mathcal{O}\big(m \log^2 N\big)$
full multigrid waveform relaxation [14]	$\mathcal{O}\big(\log m \log^2 N\big)$
FFT/CR waveform relaxation	$\mathcal{O}\big(\log^2 mN\big)$
full space-time multigrid [13]	$\mathcal{O}\big(\log^2 mN\big)$
theoretical optimum	$\mathcal{O}(\log mN)$

Table 7.1 compares with the parallel complexity of other numerical solution methods for parabolic PDEs. Compared to time-stepping methods, waveform relaxation leaves more room for concurrency because of its greater volume of computations per iteration step. This potential is effectively exploited when a fast parallel PPDE solver is chosen as preconditioner. Classic WR preconditioners such as Jacobi or red–black Gauss–Seidel are also highly concurrent: when recursive doubling is used for time integration, both have a degree of concurrency that is $\mathcal{O}(Nm)$ and a parallel complexity of $\mathcal{O}(m)$ for a single iteration step. The convergence of the latter simple WR iterations however deteriorates fast as the spatial grid is refined. The asymptotic convergence factor as a function of the grid size h is known to be $1 - \mathcal{O}\big(h^2\big)$ for both methods. The preconditioner proposed in this paper attains the same degree of concurrency while offering a convergence rate that is essentially independent of the discretization grid size.

8. Numerical experiments.

8.1. Presentation of the results. For each iterate $u_\tau^{(\nu)}$ we define the *discrete error* $e_\tau^{(\nu)} := u_\tau^{(\nu)} - u_\tau^*$ with u_τ^* the exact solution of the discretized problem. The scalar $\|e_\tau^{(\nu)}\|$ is the $\mathbf{l}_2$-norm of the discrete error over all spatial grid points and time levels in the space-time grid, divided by the total number of points $N \cdot m$. We divide by $N \cdot m$ to allow a comparison of the error across grids. The *observed convergence factor* is the ratio

$$\rho^{(\nu)} := \frac{\|e_\tau^{(\nu)}\|}{\|e_\tau^{(\nu-1)}\|} \ , \quad \nu \geq 1 \ .$$

8.2. Preconditioning with a constant coefficient operator. We apply a constant coefficient preconditioner to the PPDE

$$\frac{d}{dt}u - \nabla\big(a \, \nabla \, u\big) = 0 \ , \quad a(x_1, x_2) = e^{\beta \cdot (x_1^2 + x_2^2)} \ , \quad \beta \geq 0 \tag{8.1}$$

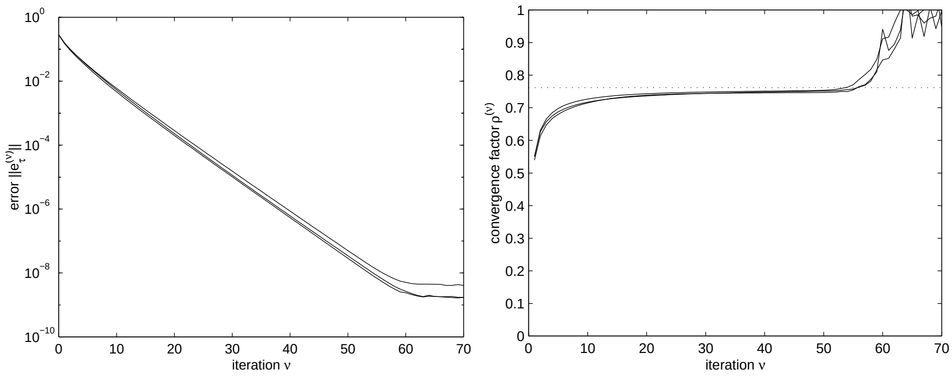
on the unit square with homogeneous Dirichlet boundary conditions. For this example the initial value u_0 consists of bandlimited white noise, so that all spatial frequencies appearing in the discretization are present. Assuming the coefficients of the preconditioner are chosen as in §6.1, the analysis of §5 gives

$$\alpha := \frac{e^{2\beta} - 1}{e^{2\beta} + 1}$$

as an upper bound for the spectral radius $\rho\big(\mathcal{K}_\infty\big)$. It is easy to see that in fact $\lim_{N \to \infty} \rho\big(\mathcal{K}_\infty\big) = \alpha$.

The numerical significance of this bound is verified for the observed convergence factor. It turns out that when the number of time-steps is large ($m \gg 10$), the iteration is clearly ruled by the pseudospectrum of $\mathcal{K}_T^\tau$ and the observed convergence factors are close to $\rho\big(\mathcal{K}_\infty^\tau\big)$. The asymptotic convergence factor $\rho\big(\mathcal{K}_T^\tau\big)$ does not set

FIG. 8.1. *Convergence history (left) and observed convergence factors (right) for problem* (8.1) *with* $\beta = 1$ *and a constant coefficient preconditioner. Results are plotted for spatial grids of size* 8×8, 16×16 *and* 32×32. *In all cases* $m = 1000$ *timesteps are taken with stepsize* $\tau = .001$.



in before machine precision is attained and convergence halts. Figure 8.1 shows the convergence history and observed convergence factors of problem (8.1) with $\beta = 1$ for various grid sizes. As discussed in §5.1, there is only a secondary dependence of the convergence rate on the grid size. For reference, the upper bound $\alpha = 0.7616$ is plotted as a dashed line.

8.3. Preconditioning with a nonseparable operator. When the coefficients in a diffusion problem are close to constant in one dimension only, partial decoupling may be more advantageous. As an example we solve the nonseparable isotropic diffusion equation

$$\frac{d}{dt}u - \nabla\big(a \nabla u\big) = f \ , \quad a(x_1, x_2) = \begin{cases} 1 + \alpha\cos(\pi x_1) \ , & x_2 \in (0.4, 0.6) \\ 101 + \alpha\cos(\pi x_1) & \text{elsewhere} \end{cases} \tag{8.2}$$

with $0 < \alpha \leq 1$. It models diffusion in a layered medium consisting of two regions of reasonably conducting material, separated by an isolating strip. A suitable preconditioner is obtained by replacing the coefficient a of the parabolic operator in (8.2) with an approximation $\widetilde{a}$ which is constant in x_1,

$$\widetilde{a}(x_2) = \begin{cases} 1 \ , & x_2 \in (0.4, 0.6) \\ 101 & \text{elsewhere} \ . \end{cases}$$

A PPDE with the resulting operator can be solved by the fast direct method described in §3.3.2. The convergence factor of the preconditioned WR iteration is bounded from above by the value of α. Figure 8.2 shows the convergence history for $\alpha = 0.2$. For the sake of comparison, preconditioning with a constant coefficient operator gives a convergence factor close to 0.98.

8.4. The effect of a scaling transformation. Returning to problem (8.1), it is clear that convergence can be made arbitrarily slow by increasing the parameter β. For $\beta = 4$, e.g., the value of α from §8.2 becomes a prohibitive 0.9993. We consider two scaling transformations. The first one gives an approximate minimization of the spectral radius of the continuous infinite-interval WR operator. Scaling with $d = a^{-\frac{1}{4}}$ one obtains

$$a^{-\frac{1}{2}}\frac{\partial}{\partial t}u - \nabla\big(a^{\frac{1}{2}} \nabla u\big) + \big(\beta + \frac{3}{4}\beta^2(x_1^2 + x_2^2)\big)\, a^{\frac{1}{2}}\, u = 0 \ . \tag{8.3}$$

FIG. 8.2. *Convergence history (left) and observed convergence factors (right) for problem* (8.2) *with* $\alpha = 0.2$ *and a nonseparable block-diagonalizable preconditioner. Results are plotted for spatial grids of size* 16×16, 32×32 *and* 64×64. *In all cases* $m = 1000$ *timesteps are taken with stepsize* $\tau = .001$.

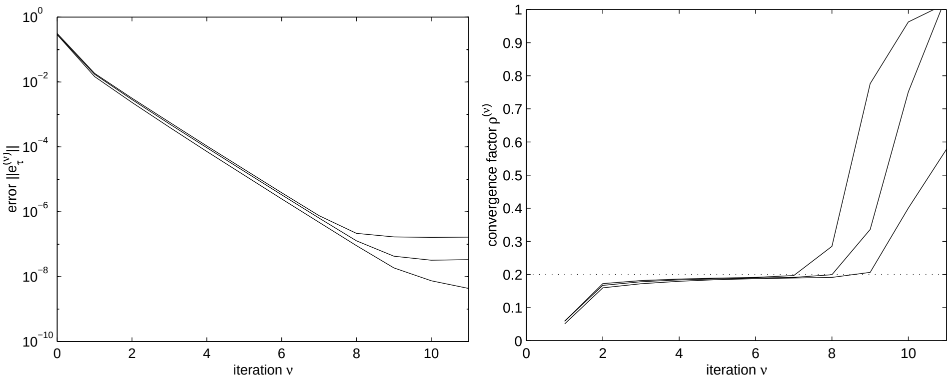
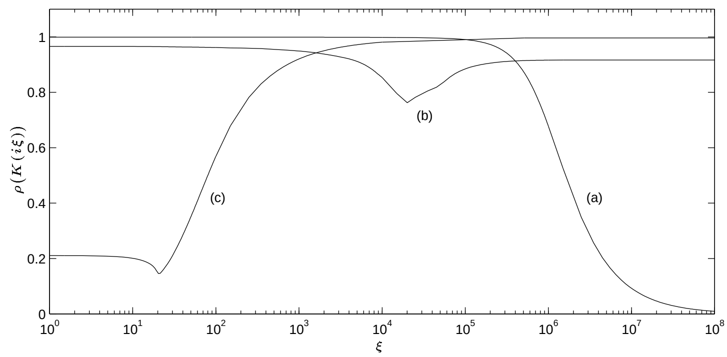


FIG. 8.3. *Comparing* $\rho\big(K(z)\big)$ *over the imaginary axis for* (a) *the unscaled problem* (8.1) *with* $\beta = 4$, (b) *the scaling* (8.3) *which aims to minimize* $\rho\big(\mathcal{K}_\infty\big)$, *and* (c) *the scaling* (8.4) *which reduces the spectral radius around the origin only.*



Secondly, scaling with $d = a^{-\frac{1}{2}}$ reduces the problem to

$$a^{-1}\frac{\partial}{\partial t}u - \Delta\, u + \big(2\beta + \beta^2(x_1^2 + x_2^2)\big)\, u = 0 \ . \tag{8.4}$$

This represents an approximate minimization of $\rho\big(M_A^{-1}N_A\big) = \rho\big(K(0)\big)$.

For the experiments the parameter value $\beta = 4$ is chosen. Figure 8.3 compares $\rho\big(K(z)\big)$ over the imaginary axis for each of the problems (8.1)–(8.4) using the constant coefficient preconditioners described in §6.1. The spectral radius of the continuous infinite-interval WR operator is the supremum over these values of z. In Figure 8.4, $\rho\big(K(z)\big)$ is shown over part of the complex plane. The dashed curves superimposed on the contour plot indicate the boundaries of the scaled stability regions $\frac{1}{\tau}S$ of the BDF-methods of orders 1 to 4. The spectral radius of a discrete-time infinite-interval WR operator is found as the supremum over the corresponding curve. It is apparent from these figures that while scaling does not yield an important reduction of $\rho\big(\mathcal{K}_\infty\big)$, it does accelerate the convergence of the discrete-time iteration. The numerical results are summarized in Table 8.1.

FIG. 8.4. *Spectral pictures $\rho\big(K(z)\big)$ of the problems* (8.3) *(left) and* (8.4) *(right). The boundaries of the scaled stability regions of the* BDF *methods of step number* 1 *through* 4 *for steplength* $\tau = 0.01$ *are superposed on the plots in dashed lines.*

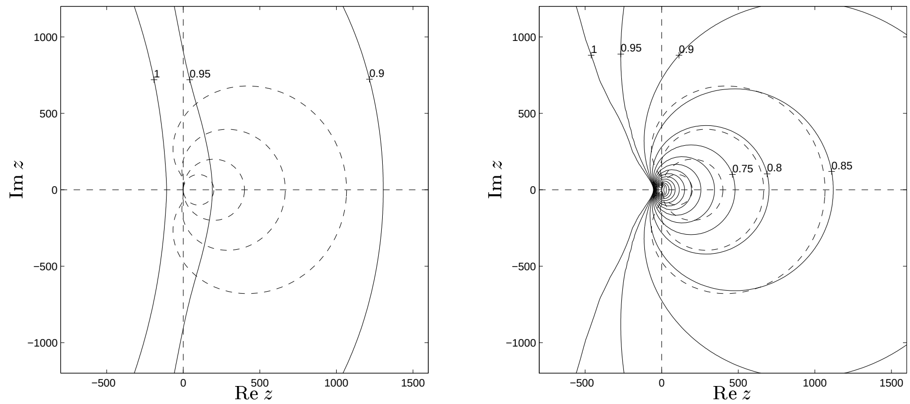


TABLE 8.1

Spectral radii of WR operators for problem (8.1) *with and without scaling. The results are for an* 8×8 *spatial grid. Time discretization uses* BDF *methods of step number* 1–3 *with a timestep of* 0.01.

	scaling	$\rho\big(M_A^{-1}N_A\big)$	$\rho\big(M_B^{-1}N_B\big)$	$\rho\big(\mathcal{K}_\infty\big)$	$\rho\big(\mathcal{K}_\infty^\tau\big)$		
					BDF1	BDF2	BDF3
(8.1)	1	.9993	.0000	.9993	.9993	.9993	.9993
(8.3)	$1/\sqrt[4]{a}$	.9660	.9167	.9660	.9660	.9660	.9660
(8.4)	$1/\sqrt{a}$	.2106	.9964	.9964	.6048	.7232	.7939

REFERENCES

[1] R. E. BANK, *Marching algorithms for elliptic boundary value problems, II: the variable coefficient case*, SIAM J. Numer. Anal., 14 (1977), pp. 950–970.

[2] J. BORIS AND K. V. ROBERTS, *The optimization of particle calculations in 2 and 3 dimensions*, J. Comp. Phys., 4 (1969), pp. 552–571.

[3] B. L. BUZBEE, G. H. GOLUB, AND C. W. NIELSON, *On direct methods for solving Poisson's equations*, SIAM J. Numer. Anal., 7 (1970), pp. 627–656.

[4] P. CONCUS AND G. H. GOLUB, *Use of fast direct methods for the efficient numerical solution of nonseparable elliptic equations*, SIAM J. Numer. Anal., 10 (1973), pp. 1103–1120.

[5] F. W. DORR, *The direct solution of the discrete Poisson equation on a rectangle*, SIAM Review, 12 (1970), pp. 248–263.

[6] E. G. D'YAKONOV, *An iteration method for solving systems of finite difference equations*, Dokl. Akad. Nauk SSSR, 138 (1961), pp. 522–525.

[7] J. FOURIER, *Théorie Analytique de la Chaleur*, vol. 1 of Oeuvres de Fourier, Gauthier–Villars, Paris, 1888.

[8] A. GRAHAM, *Kronecker Products and Matrix Calculus*, Ellis Horwood series in mathematics and its applications, Ellis Horwood Ltd., Chichester, England, 1981.

[9] J. E. GUNN, *The solution of elliptic difference equations by semi-explicit iterative techniques*, J. SIAM Numer. Anal. Ser.B, 2 (1964), pp. 24–45.

[10] R. W. HOCKNEY, *A fast direct solution of Poisson's equation using Fourier analysis*, J. Assoc. Comput. Mach., 12 (1965), pp. 95–113.

[11] ———, *The potential calculation and some applications*, in Methods in Computational Physics, B. Adler, S. Fernbach, and M. Rotenberg, eds., vol. 9, Academic Press, London, 1969, pp. 136–211.

[12] ———, *Rapid elliptic solvers*, in Numerical Methods in Applied Fluid Dynamics, B. Hunt, ed., Academic Press, London, 1980, pp. 1–48.

[13] G. HORTON AND S. VANDEWALLE, *A space-time multigrid method for parabolic P.D.E.s*, SIAM J. Sci. Comput., 16 (1995), pp. 848–864.

[14] G. HORTON, S. VANDEWALLE, AND P. WORLEY, *An algorithm with polylog parallel complexity for solving parabolic partial differential equations*, SIAM J. Sci. Comput., 16 (1995), pp. 531–541.

[15] J. JANSSEN AND S. VANDEWALLE, *Multigrid waveform relaxation on spatial finite element meshes: the continuous-time case*, SIAM J. Numer. Anal., 33 (1996), pp. 456–474.

[16] ———, *Multigrid waveform relaxation on spatial finite element meshes: the discrete-time case*, SIAM J. Sci. Comput., 17 (1996), pp. 133–155.

[17] S. KERAS, *Combining domain decomposition with waveform relaxation*, Numerical Analysis Report DAMTP 1995/NA7, Department of Applied Mathematics and Theoretical Physics, Cambridge University, Oct. 1995.

[18] ———, *Waveform relaxation method with Toeplitz operator splitting*, Numerical Analysis Report DAMTP 1995/NA4, Department of Applied Mathematics and Theoretical Physics, Cambridge University, Aug. 1995.

[19] ———, *Numerical Methods for Parabolic Partial Differential Equations*, PhD-thesis, Magdalene College, University of Cambridge, Cambridge, Sept. 1996.

[20] E. LELARASMEE, A. RUEHLI, AND A. SANGIOVANNI-VINCENTELLI, *The waveform relaxation method for time-domain analysis of large scale integrated circuits*, IEEE Trans. CAD of ICAS, CAD-1 (1982), pp. 131–145.

[21] A. LUMSDAINE AND D. WU, *Spectra and pseudospectra of waveform relaxation operators*, SIAM J. Sci. Comput., 18 (1997), pp. 286–304.

[22] U. MIEKKALA AND O. NEVANLINNA, *Convergence of dynamic iteration methods for initial value problems*, SIAM J. Sci. Stat. Comp., 8 (1987), pp. 459–482.

[23] ———, *Sets of convergence and stability regions*, BIT, 27 (1987), pp. 554–584.

[24] O. NEVANLINNA, *Remarks on Picard-Lindelöf iteration, II*, BIT, 29 (1989), pp. 535–562.

[25] A. NEWTON AND A. SANGIOVANNI-VINCENTELLI, *Relaxation-based electrical simulation*, SIAM J. Sci. Stat. Comp., 4 (1983), pp. 485–524.

[26] M. PICKERING, *An Introduction to Fast Fourier Transform Methods for Partial Differential Equations*, J. Wiley and Sons, New York, 1986.

[27] P. N. SWARZTRAUBER, *The methods of cyclic reduction, Fourier analysis and the FACR algorithm for the discrete solution of Poisson's equation on a rectangle*, SIAM Review, 19 (1977), pp. 490–501.

[28] S. VANDEWALLE AND G. HORTON, *Multicomputer-multigrid solution of parabolic partial differential equations*, in Multigrid Methods IV: Proceedings of the Fourth European Multigrid Conference, P. Hemker and P. Wesseling, eds., vol. 116 of International Series of Numerical Mathematics, Basel, 1994, Birkhauser Verlag, pp. 97–109.

[29] O. B. WIDLUND, *On the use of fast methods for separable finite difference equations for the solution of general elliptic problems*, in Sparse Matrices and Their Applications, D. Rose and R. Willoughby, eds., Plenum, 1972, pp. 121–131.

[30] P. Worley, *Limits on parallellism in the numerical solution of linear PDEs*, SIAM J. Sci. Stat. Comp., 12 (1991), pp. 1–35.