10-16-2008

Multigrid Algorithms for Inverse Problems with Linear Parabolic PDE Constraints

Santi S. Adavani
University of Pennsylvania, adavani@seas.upenn.edu

George Biros
University of Pennsylvania, biros@seas.upenn.edu

Multigrid Algorithms for Inverse Problems with Linear Parabolic PDE Constraints Santi S. Adavani and George Biros, SIAM J. Sci. Comput. 31, 369 (2008), DOI:10.1137/070687426

Multigrid Algorithms for Inverse Problems with Linear Parabolic PDE Constraints

Abstract

We present a multigrid algorithm for the solution of source identification inverse problems constrained by variable-coefficient linear parabolic partial differential equations. We consider problems in which the inversion variable is a function of space only. We consider the case of L-2 Tikhonov regularization. The convergence rate of our algorithm is mesh-independent-even in the case of no regularization. This feature makes the method algorithmically robust to the value of the regularization parameter, and thus useful for the cases in which we seek high-fidelity reconstructions. The inverse problem is formulated as a PDE-constrained optimization. We use a reduced-space approach in which we eliminate the state and adjoint variables, and we iterate in the inversion parameter space using conjugate gradients. We precondition the Hessian with a V-cycle multigrid scheme. The multigrid smoother is a two-step stationary iterative solver that inexactly inverts an approximate Hessian by iterating exclusively in the high-frequency subspace (using a high-pass filter). We analyze the performance of the scheme for the constant coefficient case with full observations; we analytically calculate the spectrum of the reduced Hessian and the smoothing factor for the multigrid scheme. The forward and adjoint problems are discretized using a backward-Euler finite-difference scheme. The overall complexity of our inversion algorithm is O(NtN + N log(2) N), where N is the number of grid points in space and N-t is the number of time steps. We provide numerical experiments that demonstrate the effectiveness of the method for different diffusion coefficients and values of the regularization parameter. We also provide heuristics, and we conduct numerical experiments for the case with variable coefficients and partial observations. We observe the same complexity as in the constant-coefficient case. Finally, we examine the effectiveness of using the reduced-space solver as a preconditioner for a full-space solver.

Keywords

inverse problems, heat equation, reaction-diffusion equations, multigrid, regularization

Comments

Multigrid Algorithms for Inverse Problems with Linear Parabolic PDE Constraints Santi S. Adavani and George Biros, SIAM J. Sci. Comput. 31, 369 (2008), DOI:10.1137/070687426

SIAM J. SCI. COMPUT.
Vol. 31, No. 1, pp. 369–397

MULTIGRID ALGORITHMS FOR INVERSE PROBLEMS WITH LINEAR PARABOLIC PDE CONSTRAINTS*

SANTI S. ADAVANI† AND GEORGE BIROS‡

Abstract. We present a multigrid algorithm for the solution of source identification inverse problems constrained by variable-coefficient linear parabolic partial differential equations. We consider problems in which the inversion variable is a function of space only. We consider the case of L^2 Tikhonov regularization. The convergence rate of our algorithm is mesh-independent—even in the case of no regularization. This feature makes the method algorithmically robust to the value of the regularization parameter, and thus useful for the cases in which we seek high-fidelity reconstructions. The inverse problem is formulated as a PDE-constrained optimization. We use a reduced-space approach in which we eliminate the state and adjoint variables, and we iterate in the inversion parameter space using conjugate gradients. We precondition the Hessian with a V-cycle multigrid scheme. The multigrid smoother is a two-step stationary iterative solver that inexactly inverts an approximate Hessian by iterating exclusively in the high-frequency subspace (using a high-pass filter). We analyze the performance of the scheme for the constant coefficient case with full observations; we analytically calculate the spectrum of the reduced Hessian and the smoothing factor for the multigrid scheme. The forward and adjoint problems are discretized using a backward-Euler finite-difference scheme. The overall complexity of our inversion algorithm is $\mathcal{O}(N_t N + N \log^2 N)$, where N is the number of grid points in space and N_t is the number of time steps. We provide numerical experiments that demonstrate the effectiveness of the method for different diffusion coefficients and values of the regularization parameter. We also provide heuristics, and we conduct numerical experiments for the case with variable coefficients and partial observations. We observe the same complexity as in the constant-coefficient case. Finally, we examine the effectiveness of using the reduced-space solver as a preconditioner for a full-space solver.

Key words. inverse problems, heat equation, reaction-diffusion equations, multigrid, regularization

AMS subject classifications. 34A55, 35R30, 65M55, 65M70, 35K50, 65F22, 65F10

DOI. 10.1137/070687426

1. Introduction.

We present multigrid algorithms for inverse problems constrained by parabolic partial differential equations (PDEs). As a model problem we consider the one-dimensional (1-D) heat equation and a source identification problem, the reconstruction of a heat source function given either full or partial observations of the temperature. Our method is designed for problems with unknown spatial variation and known temporal variation. Our model is motivated by inverse medium and data assimilation problems that are constrained by reaction–convection-diffusion equations. We use a PDE-constrained optimization formulation [5].

More precisely, we seek to reconstruct an unknown function $u(x)$ by solving the following minimization problem:

$$\min_{y,u} \mathcal{J}(y,u) := \frac{1}{2}\|Qy - y^*\|^2_{L^2(\Omega)\times L^2((0,T])} + \frac{\beta}{2}\|u\|^2_{L^2(\Omega)}$$

*Received by the editors April 4, 2007; accepted for publication (in revised form) March 11, 2008; published electronically October 16, 2008. This work was supported by the U.S. Department of Energy under grant DE-FG02-04ER25646, and the U.S. National Science Foundation grants CCF-0427985, CNS-0540372, IIS-0431070, and DMS-0612578.

http://www.siam.org/journals/sisc/31-1/68742.html

†Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia, PA 19104 (adavani@seas.upenn.edu).

‡Departments of Mechanical Engineering and Applied Mechanics, Bioengineering, and Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104 (biros@seas.upenn.edu).

subject to

$$\frac{\partial y(x)}{\partial t} - \nu\Delta y(x) = a(x,t)y(x) + b(x,t)u(x) \quad \text{in} \quad D, y = 0 \quad \text{on} \quad \partial\Omega, y(x,0) = 0 \quad \text{in} \quad \Omega,$$

where Q corresponds to the observation operator and D is defined as $\Omega \times (0,T]$. Here, y is the *state variable*, u is the *inversion variable*, $\nu > 0$ is the diffusion coefficient, and $\beta \geq 0$ is the regularization parameter. The objective is to reconstruct u by minimizing the misfit between the observed state y^* and the predicted state y. We assume that both $a(x,t)$ and $b(x,t)$ are known, smooth, and bounded functions.[1]

By forming a Lagrangian, by introducing the adjoint variables λ, and by requiring stationarity with respect to the state, inversion, and adjoint variables, we arrive at the first-order optimality conditions:

State

$$\frac{\partial y}{\partial t} - \nu\Delta y - ay - bu(x) = 0 \quad \text{in} \quad D,\ y = 0 \quad \text{on} \quad \partial\Omega,\ y(x,0) = 0 \quad \text{in} \quad \Omega.$$

Adjoint

$$-\frac{\partial \lambda}{\partial t} - \nu\Delta\lambda - a\lambda + (Q^TQy - Q^Ty^*) = 0 \quad \text{in} \quad D,\ \lambda = 0 \quad \text{on} \quad \partial\Omega, \lambda(x,T) = 0 \quad \text{in} \quad \Omega.$$

Inversion

$$\beta u - \int_T b\lambda\, d\,t = 0 \quad \text{in} \quad \Omega.$$

The above system of equations is also known as the *Karush–Kuhn–Tucker* optimality system or the *"KKT"* system. The corresponding linear operator can be written as

$$\begin{bmatrix} Q^TQ & 0 & -\frac{\partial}{\partial t} - \nu\Delta - a \\ 0 & \beta I & -\int_0^T b \\ \frac{\partial}{\partial t} - \nu\Delta - a & -b & 0 \end{bmatrix} = \begin{bmatrix} Q^TQ & 0 & J^T \\ 0 & \beta I & C^T \\ J & C & 0 \end{bmatrix}, \tag{1.1}$$

where J and C are the Jacobians of the constraints with respect to the state and the inversion variables, respectively. The KKT operator corresponds to a symmetric saddle point problem. For an excellent review on linear solvers for such problems, we refer the reader to [6]. In this paper, we will consider two methods: the so-called *full-space* and *reduced-space* [15]. In full-space methods, one solves directly (1.1), for example, using a Krylov-iterative method. In reduced-space methods, one solves for u by an iterative solver on the the Schur complement of u. To derive the Schur complement, we first eliminate y and λ using the state and adjoint equations, respectively, and then we substitute λ in the inversion equation. In this way, we obtain

$$Hu = g, \tag{1.2}$$

where

$$H = C^TJ^{-T}Q^TQJ^{-1}C + \beta I \tag{1.3}$$

is known as the *"reduced Hessian"* (or just *"Hessian"*). Since Q is positive semidefinite, H is a symmetric and strictly positive definite operator. The reduced gradient

[1] In the following, we suppress the notation for the explicit dependence on x and t.

g is defined by $g = -C^T J^{-T} Q^T y^*$. Each reduced Hessian matrix-vector product (herein after, "*matvec*") requires one exact forward solve and one adjoint solve which makes it expensive to solve (1.2). We focus our attention to the design of efficient solvers for reduced-space formulations. For completeness we include an example that shows how we can combine full- and reduced-space approaches.

Related work. Reduced-space methods are quite popular because one can iterate on the adjoint and state equation in sequence, they require less storage, and the conjugate gradient (CG) method can be used to invert H. The KKT matrix (1.1), is indefinite, ill-conditioned, and its size is more than twice as large as that of the forward problem. Most implementations avoid using H and instead use some approximation, for example, quasi-Newton. Such approaches, however, are not algorithmically scalable [1]. If H is to be used, direct solvers are not a viable option, since the reduced Hessian is a nonlocal and thus, dense operator. The preconditioned conjugate gradients (PCG) algorithm requires matvecs only, and thus can be used to solve (1.2).

If we fix the regularization parameter β to a positive value we can show that H is a compact perturbation of the identity and thus has a bounded (mesh-independent) condition number: It scales as $\mathcal{O}(1/\beta)$. Using CG to solve a linear system involving H requires $\mathcal{O}(1/\sqrt{\beta})$ iterations. Therefore, the overall scheme does not scale with vanishing β. We claim that, in mesh-refinement studies and scalability analyses for inverse problem solvers, having a fixed value of β can lead to wrong conclusions.

There are two reasons that drive the need to solve problems in refined meshes. The first reason is the need to resolve the forward and adjoint equations. In that case, one can use a mixed discretization in which u is discretized in a coarser grid, or one can use a large value for β. In the second case, which is pertinent to scalability of the inverse problem solver, we have high-quality observations[2] that allow for a high-resolution reconstruction of u. This implies that β cannot be fixed as we refine the mesh because we will not be able to recover the sought frequencies. There has been little work on a mesh-independent scheme for vanishing β [27].

Returning to the reduced Hessian, we observe that the deterioration of the condition number with decreasing β suggests the need for a preconditioning scheme. We cannot use standard preconditioning techniques like incomplete factorizations or Jacobi relaxations, as these methods need an assembled matrix [4]. In [7] a two-step stationary iterative method that does not need an assembled matrix was used to precondition the reduced Hessian.

Another alternative is to use multigrid methods. These methods have been developed mainly for linear systems arising from the discretization of elliptic and parabolic PDEs. The basic idea of multigrid is to accelerate the iterative solution of a PDE by computing corrections on a coarser grid and then interpolating them back to the original grid. The three important steps of multigrid scheme are *presmoothing, coarse-grid correction*, and *postsmoothing*. Smoothing is equivalent to taking a few iterations of an iterative method (*"smoother"*) that should selectively remove the high-frequency error components faster than low-frequency components. Besides the pioneering work of [12] for differential operators, and of [19] for second-kind Fredholm integral operators, there exists significant work on multigrid methods for optimal control problems. For example, see the work of [2] and [15] for a general discussion, and [9] and [10] for distributed control problems constrained by parabolic PDEs. An alternative to multigrid is domain decomposition; a promising work for problems similar to ours can

[2]If the data is not in the range of the inversion operator, e.g., due to noise, vanishing β will result in blow up for u as the discretization size goes to zero.

be found in [21]. There the author proposes a space-time Schur domain decomposition preconditioner for the KKT system. A nice feature of that method is that it can be readily parallelized in time. The context, however, is optimal control, where the value of the regularization parameter is relatively large because of the costs involved in applying the control.

In our case, the unregularized reduced Hessian is a Fredholm operator of the first kind. There has been little work on multigrid algorithms for such problems. In [20], multilevel and domain decomposition preconditioners were proposed for integral equations of the first kind. Multigrid solvers for Tikhonov-regularized ill-posed problems were discussed in [27]. In [26], a vanishing regularization parameter has been discussed. Such problems were further analyzed in [24] and [25]. A multigrid preconditioner based on that work was also used in [1] to solve the problems with million of inversion parameters with a relatively small but nonvanishing value of the regularization parameter. As we will discuss later in the paper, the methods described in [1] and [10] do not scale well in the case of a mesh-dependent regularization parameter. In real-time problems, the choice of a regularization parameter is driven by the magnitude of noise in the data. In numerical studies, since we assume zero noise, a mesh-dependent regularization parameter has to be chosen to study the scalability of the algorithm.

Contributions. Our main contribution is to derive a method for which we obtain a mesh-independent and β-independent convergence rate—including the case of $\beta = 0$. We design special smoothers that are used to build multigrid schemes for (1.2).

There are several challenges in designing multigrid schemes for the reduced space. As we mentioned, we have to design a matrix-free smoother with no access to the diagonal or off-diagonal terms of the Hessian operator. The reduced Hessian (with $\beta = 0$) is a compact operator. Its dominant eigenvalues correspond to low-frequency components, and for such operators standard smoothers fail. Finally, every matrix-vector multiplication with the reduced Hessian is equivalent to a forward and an adjoint solve; hence, it is important to design algorithms that require a minimum number of matvecs in the fine grid.

We first propose preconditioners that can be used as smoothers in multigrid, so that multigrid can be used as a preconditioner to an outer CG solver. This uses an approximate filtering operator that restricts the smoother to the high-frequency Krylov subspace. We show numerical results that indicate good behavior. The method is easy to implement, but it cannot be extended to more general cases like variable-coefficient problems and partial observations. For this reason we propose a second smoother that is more expensive, but for which we can provide complexity estimates.

The main components of the proposed algorithm are: (1) *a reduced Hessian that is a composition of spectral filtering with an approximate Hessian operator based on inexact forward and adjoint solves; and* (2) *a smoothing scheme that uses a stationary second-order method targeted in the high-frequency components of* u. *It is crucial that the effectiveness of the smoother in the high-frequency regime is mesh-independent; our method fulfills this requirement. The multigrid scheme (a V-cycle) can be used as solver or as a preconditioner for a Krylov-iterative method.*

The forward and adjoint problems are discretized using a backward-Euler scheme in time and a standard three-point Laplacian (Dirichlet BCs) in space. We conduct numerical experiments to test (1) the effects of semicoarsening (only space coarsening) and standard-coarsening; (2) different smoothing techniques; and (3) the effects of using non-Galerkin coarse-grid operators. We analyze and experimentally measure convergence factors. Also, we present results for the more general case in which multigrid is used as a preconditioner and not as a solver. In addition, we test the

algorithm for the case of variable coefficients (resembling reaction-diffusion equations that admit traveling wave solutions) and partial observations. Finally, we include a discussion on full-space methods, and we propose a multigrid scheme for (1.1) along with numerical results that illustrate its performance.

Limitations. In this paper, we have assumed noiseless and attainable observations in all of the numerical experiments. Due to this assumption, we are allowed to take very small regularization parameters in all of the numerical studies. The presence of noise and the choice of regularization parameter will significantly affect the quality of reconstruction and the behavior of the solvers. However, investigations into the error propagation due to the presence of noise and the choice of regularization parameters are beyond the scope of this paper.

1.1. Organization of the paper. In section 2, we derive the spectral properties of the 1-D analytic and discretized reduced Hessian for the case of constant coefficients. In section 3, we discuss multigrid, and we give details on the coarse-grid operator. In section 3.2, we discuss standard smoothers and present the construction of novel smoothers based on the idea of subspace decomposition. In section 4, we present appropriate preconditioners that can be used as smoothers in multigrid. Numerical results on this approach are presented in section 4.1. In section 5, we present our main contribution, a multigrid variant which is based on exact subspace projections, and we present results for both the constant and the variable-coefficient cases. Finally, in section 6 we discuss full-space methods.

2. Spectral properties of the reduced Hessian. We start by calculating the spectrum of the 1-D reduced Hessian. We show that, in its general form, the source inversion is an ill-posed problem with algebraically decaying eigenvalues. Let K be the Green's operator for the forward problem, so that K maps functions from the inversion variable space to the state variable space and K corresponds to $-QJ^{-1}C$ in (1.3). Using K, we can eliminate the constraint and obtain an unconstrained variational problem for u:

$$\min \tilde{\mathcal{J}}(u) = \frac{1}{2}\int_\Omega \int_T (Ku - y^*)^2 \, d\,\Omega d\,t + \frac{\beta}{2}\int_\Omega u^2 \, d\,\Omega. \tag{2.1}$$

Taking the variations of (2.1), with respect to u, we get an Euler–Lagrange equation for u:

$$\frac{\partial \tilde{\mathcal{J}}}{\partial u}\hat{u} = \int_\Omega \int_T K^T(Ku - y^*)\hat{u} \, d\,\Omega d\,t + \beta \int_\Omega u\hat{u} \, d\,\Omega = 0 \ \forall \hat{u}, \tag{2.2}$$

where K^T is the adjoint of K. Here K corresponds to a forward solve, and K^T corresponds to an adjoint solve. Then the strong form of the optimality conditions is given by

$$\left(K^T K + \beta I\right) u = K^T y^* \qquad \text{or} \qquad Hu = K^T y^*, \tag{2.3}$$

where H is the reduced Hessian. If $a = 0$, then K (for homogeneous Dirichlet BCs) is given by

$$\begin{aligned} y &= \int_\Omega \int_0^t k(x - x', t - \tau) b(x', t) u(x') \, d\,x' d\,\tau \\ &= 2\sum_{k=1}^{\infty} \int_\Omega \int_0^t e^{-k^2\pi^2(t-\tau)} S_k(x) S_k(x') b(x', t) u(x') \, d\,x' d\,\tau, \end{aligned} \tag{2.4}$$

where $S_k(x) = \sin(k\pi x)$. If we assume that $b = 1(t)$, expand $u(x) = \sum_{j=1}^{\infty} u_j S_j(x)$, and use orthogonality we get that

$$y = \sum_{k=0}^{\infty} S_k(x) y_k(t) \quad \text{with} \quad y_k(t) = \int_0^t e^{-k^2\pi^2(t-\tau)} 1(\tau) u_k \, d\tau. \tag{2.5}$$

The adjoint operator K^T is given by

$$\lambda(x, T-t) = K^T z = \int_\Omega \int_0^t k(x - x', t - \tau)\, z(x', T - \tau)\, d\, x' d\, \tau. \tag{2.6}$$

Using (2.5) and (2.6) in (2.3), and setting $\beta = 0$, the *eigenvalues* (σ_k) and *eigenvectors* (v_k) of the reduced Hessian ($H = K^T K$) are given by

$$\sigma_k = \frac{2k^2\pi^2 T + 4e^{-k^2\pi^2 T} - 2e^{-2k^2\pi^2 T} - 3}{2k^6\pi^6} \text{ and } v_k = S_k. \tag{2.7}$$

If we discretize in space using the three-point Laplacian approximation, the corresponding eigenvalues and eigenvectors of the reduced Hessian (H^h) are given by

$$\sigma_k = \frac{2\lambda_k T + 4e^{-\lambda_k T} - 2e^{-2\lambda_k T} - 3}{2\lambda_k^3} \text{ and } v_k = S_k^h, \tag{2.8}$$

where $\lambda_k = 4\nu N^2 \sin^2(\frac{k\pi}{2N})$ is the kth eigenvalue of the discrete Laplacian and ν is the diffusion coefficient (see Table 1). The discrete sine function is represented by S_k^h, with N being discretization size. If we use a backward-Euler scheme for time, the eigenvalues of the discrete reduced Hessian (H^h) are given by

$$\sigma_k^\delta = \delta^3 \sum_{j=1}^{N_t} \sum_{l=0}^{N_t - j} \frac{\sum_{r=0}^{l+j-1} \frac{1}{(1+\lambda_k\delta)^r}}{(1+\lambda_k\delta)^{l+1}}, \tag{2.9}$$

where N_t is the number of time steps and δ the time step.

From (2.7) and (2.8) it is evident that $k \to \infty \Rightarrow \sigma_k \to 0$. Furthermore $\sigma_{\max} \approx \frac{T}{(\lambda_1)^2}$ for a large-enough time-horizon T. If $\beta \neq 0$, then $\sigma_{\min} = \beta$ and the *condition number* of the reduced Hessian is given by $\kappa = \frac{\sigma_{\max}+\beta}{\beta}$, and it is bounded. For small β, however, the reduced Hessian is a highly ill conditioned operator (see Figure 1).

The number of CG iterations required for convergence is proportional to the square root of the condition number of the underlying operator. Therefore, for a mesh-independent condition number, we obtain a mesh-independent number of iterations. It may be the case, however, that the data fidelity allows a quite small regularization parameter.

In Table 2, we report results from a numerical experiment in which we study the number of CG iterations for two cases of the regularization parameter. One can observe that, for constant β, the the number of iterations is mesh-independent. This is not the case when β is related to the mesh size. The goal of the present work is to use multigrid ideas to address the problem of a β-independence number of CG iterations, at least for the source inversion problem for the heat equation.

3. Reduced-space multigrid. In this section, we summarize the algorithmic issues related to multigrid for the reduced Hessian. Here, and in the rest of the paper, we use the superscript h to denote the fine discretization level, and we use

TABLE 1

The eigenvalues and eigenvectors of Laplacian and reduced Hessian. The kth eigenvalue and eigenvector of the operator are represented by σ_k and v_k, respectively. The diffusion coefficient and total time interval are ν and T, respectively. Here $S_k = \sin(k\pi x)$. Discrete operators and functions are denoted with a superscript h.

Operator	σ_k	v_k
$-\nu\Delta$	$\lambda_k = \nu k^2\pi^2$	S_k
$-\nu\Delta^h$	$\lambda_k^h = 4\nu N^2 \sin(\frac{k\pi}{2N})$	S_k^h
H	$\sigma_k = \frac{2\lambda_k T + 4e^{-\lambda_k T} - 2e^{-2\lambda_k T} - 3}{2\lambda_k^3}$	S_k
H^h	$\sigma_k = \frac{2\lambda_k^h T + 4e^{-\lambda_k^h T} - 2e^{-2\lambda_k^h T} - 3}{2(\lambda_k^h)^3}$	S_k^h
$H^{h,\delta}$	$\sigma_k^\delta = \delta^3 \sum_{j=1}^{N_t} \sum_{l=0}^{N_t-j} \frac{\sum_{r=0}^{l+j-1} \frac{1}{(1+\lambda_k\delta)^r}}{(1+\lambda_k\delta)^{l+1}}$	S_k^h

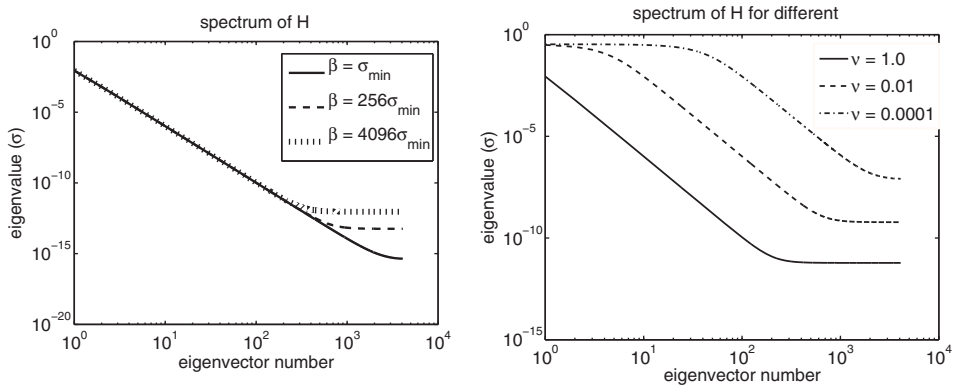

FIG. 1. *The effect of regularization on the spectrum of the reduced Hessian. Here we report the spectrum of the eigenvalues (σ) of the reduced Hessian H for $\nu = 1.0$ and $T = 1$. Three cases of regularization parameter $\beta = \sigma_{\min}$, $256\sigma_{\min}$, and $4096\sigma_{\min}$ are plotted. The right plot shows the spectrum of H for three different diffusion coefficients $\nu = 1, 0.01$, and 0.0001.*

TABLE 2

CG mesh-dependence. Here we report the performance of CG as a function of the mesh size and the value of the regularization parameter. The number of CG iterations does not change with an increase in the problem size N; β is the regularization parameter, and in parentheses the number of recovered frequencies; Iters *corresponds to the number of CG iterations for a relative residual reduction $\|r\|/\|r^0\| \leq 10^{-12}$; a maximum number of iterations is $2N$. Two cases of the regularization parameter are considered: $\beta = \sigma_{20}$ and $\beta = \sigma_{100}$. Additional parameters used in this numerical experiment are $\nu = 1$ and $T = 1$. One observes that the number of CG iterations are mesh-independent only in the case of constant β.*

N	β ($\sigma > \beta$)		Iters	
512	6e-08 (19)	1e-10 (99)	69	725
1024	6e-08 (19)	1e-10 (99)	70	781
2048	6e-08 (19)	1e-10 (99)	68	763
4096	6e-08 (19)	1e-10 (99)	71	713

$2h$ to denote the coarse level—in the case of a two-grid scheme. For example, we denote the discrete reduced Hessian at resolution h by H^h. The key steps of a multigrid algorithm for $H^h u^h = g^h$ are given in Algorithm 1. In the presmoothing and postsmoothing steps, high-frequency error components are attenuated, whereas the

ALGORITHM 1 Multigrid (MG).

1: *Presmoothing:* Smoother iterations on $H^h u^h = g^h$;
2: *Restriction:* $r^h = g^h - H^h u^h$ and $r^{2h} = I_h^{2h} r^h$;
3: *Coarse-grid correction:* Solve $H^{2h} e^{2h} = r^{2h}$;
4: *Prolongation:* $e^h = I_{2h}^h e^{2h}$;
5: *Update:* $u^h \leftarrow u^h + e^h$;
6: *Postsmoothing*: Smoother iterations on $H^h u^h = g^h$.

TABLE 3

The spectral properties of the restriction and prolongation operators. Let $s_k = \sin^2(\frac{k\pi x}{2})$, $c_k = \cos^2(\frac{k\pi x}{2})$, *and* $S_k = \sin(k\pi x)$ *for* $1 \le k \le N-1$. $I_h^{2h}\colon \Omega^h \to \Omega^{2h}$, $I_{2h}^h\colon \Omega^{2h} \to \Omega^h$, *and* $I - I_{2h}^h I_h^{2h}\colon V^h \to W^{2h}$, *where* V^h *is the fine space and* W^{2h} *is the space containing high-frequency components.* S_k^h *and* S_k^{2h} *are the eigenfunctions of the discrete Laplacian and the reduced Hessian in* Ω^h *and* Ω^{2h}, *respectively.* $1 \le k \le \frac{N}{2}$ *for all the rows in the table.*

Operator	Input function	Output function
I_h^{2h}	S_k^h	$c_k S_k^{2h}$
I_h^{2h}	S_{N-k}^h	$-s_k S_k^{2h}$
I_{2h}^h	S_k^{2h}	$c_k S_k^h - s_k S_{N-k}^h$
$I - I_{2h}^h I_h^{2h}$	S_k^h	$(1-c_k^2) S_k^h + c_k s_k S_{N-k}^h$
$I - I_{2h}^h I_h^{2h}$	S_{N-k}^h	$(1-s_k^2) S_{N-k}^h + c_k s_k S_k^h$

coarse-grid correction acts on low-frequency error components. We refer the reader to [29] for an excellent introduction to multigrid. The restriction (I_h^{2h}), prolongation (I_{2h}^h), coarse-grid (H^{2h}) operators, and the smoother are important components that determine the performance of the algorithm. In Table 3, we summarize the spectra of several restriction and prolongation-based operators. Key in a multigrid scheme is that for each grid level the majority of the work is in removing errors associated with high frequencies (at the specific grid level). In addition, as we move into the grid hierarchy, errors should not be reintroduced or amplified. The problems we are discussing here are pretty regular, so prolongation and restriction do not present particular challenges. Below we first discuss the coarse-grid operator representation, and then we discuss smoothing techniques.

3.1. Coarse-grid operator. There are two main ways to define the coarse-grid operator, given a grid hierarchy, the Galerkin, and the direct discretization. Using a variational principle and provided that $I_h^{2h} = c{I_{2h}^h}^T$, the *"Galerkin"* coarse-grid operator operator is defined by

$$H_G^{2h} = I_h^{2h} H^h I_{2h}^h,$$

where H_G^{2h} and H^h are the Galerkin coarse-grid operator and fine grid operators, respectively [13]. Another way of defining the coarse-grid operator is by discretizing directly the forward and adjoint problems in the coarse grid:

$$H^{2h} = (C^T J^{-T})^{2h} (Q^T Q)^{2h} (J^{-1} C)^{2h}.$$

In the classical multigrid theory, the finite-difference Laplacian operator on regular grids using standard restriction and prolongation operators[3] with constant-coefficients there is no difference between the two coarse-grid operators. In the case of reduced

[3]Using a full-weighting restriction operator and bilinear interpolation operator.

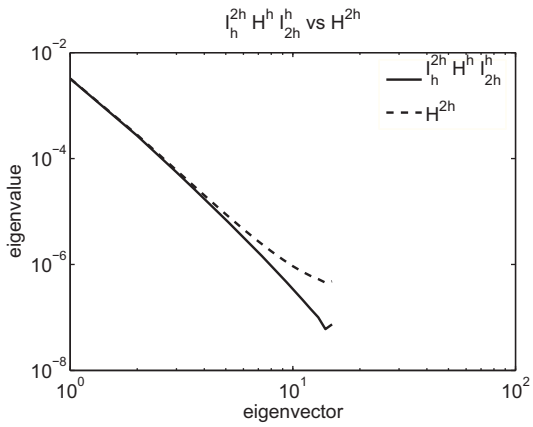

FIG. 2. *The spectrum of the coarse-grid reduced Hessian. Here we depict the difference between the Galerkin $I_h^{2h} H^h \; I_{2h}^h$ and the direct discretization of the reduced Hessian operators. We observe that H^{2h} does not satisfy the Galerkin condition, and thus inverting it will not eliminate the low-frequency components of the error. Due to this fact we use multigrid as a CG preconditioner.*

Hessian, however, they are quite different—especially in the high-frequency region of the coarse space (Figure 2). The difference in the spectra can be explained from the scaling of the eigenvalues of H^h due to the eigenstructure of the standard restriction and prolongation operators (Table 3). Therefore, the error components in certain intermediate eigenvector directions of the fine grid spectrum cannot be recovered if we use H^{2h}. So it is preferable to use the Galerkin coarse-grid operator for robustness and easily provable convergence. On the other hand every Galerkin coarse-grid matvec requires a fine-grid reduced Hessian matvec, which involves an exact forward and adjoint solve at h. Whereas, H^{2h} matvec requires an exact forward and adjoint solve at $2h$. This makes the Galerkin operator computationally more expensive. Therefore, we avoid using H_G^{2h} and use H^{2h}.

3.2. Smoothers. Classical smoothing schemes for the elliptic PDEs include iterative methods like Jacobi, Gauss–Seidel, and CG. A common characteristic of these methods is that they remove error components corresponding to large eigenvalues faster than error components corresponding to small eigenvalues.[4] This property makes these methods favorable for elliptic operators that have large eigenvalues for high-frequency eigenvectors. In our case, the (unregularized) reduced Hessian is a compact operator and behaves quite differently. As shown in Figure 1, the large eigenvalues of the reduced Hessian are associated with smooth eigenfunctions, and small eigenvalues are associated with rough or oscillatory eigenfunctions. Therefore, the above smoothing methods act as *roughers*. In addition, we do not have direct access to the entries of the reduced Hessian matrix, so there is no cheap way to apply smoothers like Jacobi or Gauss–Seidel.

We discuss in a greater detail why CG cannot be used as a smoother. Using (2.7) we will show that CG cannot be used as a smoother for this problem, as it acts on the high-energy (large eigenvalues) smooth components and acts as rougher instead. We neglect the exponential terms as they go to zero very fast. Let $\beta = 0$ and $T = \pi^4$, then the kth eigenvalue of the reduced Hessian is $\sigma_k = \frac{1}{k^4}$. At the ith iteration, CG

[4]CG works on both ends of the spectrum, but this is not so important in our context.

constructs an $(i-1)$th-degree polynomial to minimize $\|e_{(i)}\|_H$. Therefore the error at the ith iteration can be expressed as $e_{(i)} = P_i(\sigma)e_{(0)}$, where $P_i(\sigma)$ is the $(i-1)$th-degree polynomial, $e_{(0)}$ is the initial error, and $e_{(i)}$ is the error at ith iteration. $P_i(\sigma)$ is given by *Chebyshev polynomials*, where the Chebyshev polynomial $T_i(\omega)$ of degree i is $T_i(\omega) = \frac{1}{2}\left[(\omega + \sqrt{\omega^2-1})^i + (\omega - \sqrt{\omega^2-1})^i\right]$; see [28]. The polynomial $P_i(\sigma)$ is given by

$$P_i(\sigma) = \frac{T_i\left(\frac{\sigma_{\max}+\sigma_{\min}-2\sigma}{\sigma_{\max}-\sigma_{\min}}\right)}{T_i\left(\frac{\sigma_{\max}+\sigma_{\min}}{\sigma_{\max}-\sigma_{\min}}\right)}.$$

We have already seen that the reduced Hessian is a compact operator. Thus, neglecting $\sigma_{\min}$ ($\sigma_{\min} \ll \sigma_{\max}$) gives $P_i(\sigma_k) = T_i\left(1 - \frac{2}{k^4}\right)$. Without loss of generality, we can assume that the initial guess has error components in all of the eigenvector directions. Notice that $P_i(\sigma_k)$ is the amount of attenuation of the kth eigenvector at the ith CG iteration. For high-frequency error components, $P_i(\sigma_k) \approx T_i(1) = 1$; for small $k \approx P_i(\sigma_k) = 0.5$. Thus, the amplitude reduction of low-frequency error components is greater than that of high-frequency error components: CG can not be used as a smoother in the present problem.

This motivates a modification of the Hessian operator so that the low-frequency spectrum is screened out from CG. In this regard, we discuss the construction of smoothers based on the idea of decomposing the finite-dimensional space into relatively high-frequency and low-frequency subspaces given in [27] and [26]. This idea was further studied in [24] and [25]. Similar ideas of using the subspace decomposition are also used in the construction of efficient preconditioners in [18].

Our preconditioners will be based on a fine-coarse grid decomposition of the reduced Hessian. The "coarse" space V^{2h} is embedded into the "fine" space V^h. By $P^h : V^h \to V^{2h}$ we denote the L^2-orthogonal projection, and by $I - P^h : V^h \to W^{2h}$ the projection to high-frequency functions W^{2h}. We decompose $v \in V^h$ into a smooth component $v_s \in V^{2h}$ and an oscillatory component $w_o \in W^{2h}$. Then $H^h v = H^h v_s + H^h w_o$. If in addition, we assume that P^h coincides with the projection onto the subspace spanned by the eigenvectors of the reduced Hessian (as it is in the case of constant coefficients), we can write H^h as

$$(I - P^h + P^h)H^h(I - P^h + P^h) = (I - P^h)H^h(I - P^h) + P^h H^h P^h \tag{3.1}$$

assuming that P^h is the exact orthogonal projection operator i.e.,

$$(I - P^h)H^h P^h u = P^h H^h (I - P^h)u = 0 \; \forall u \in V^h. \tag{3.2}$$

Therefore we can write $H^h u = g$ as

$$(I - P^h)H^h(I - P^h)u + P^h H^h P^h u = (I - P^h)g + P^h g. \tag{3.3}$$

Hence $P^h H^h v_s = P^h g$, and

$$(I - P^h)H^h w_o = (I - P^h)g. \tag{3.4}$$

Since we are interested in removing the high-frequency error components while smoothing we solve (3.4). However, since in general, P^h will not correspond to the high-frequency spectrum of the Hessian, we can use it as an approximate projection. An alternative approach is to use Chebyshev-iterative methods and work on the spectrum

of interest, provided that we have eigenvalue estimates [3]. In principle, this method is quite similar to our approach (it is used for an entirely different problem.) It uses a number of reduced Hessian matvec operations and computes an exact decomposition. In the present case, we would like to avoid spectra calculations, if possible, as our main goal is to minimize the number of matrix-vector multiplications with the reduced Hessian. (For our smoother in section 5 we need only the spectra range, and *not all* of the eigenvalues.) Instead we approximate $I - P^h$ either by standard interpolation-restriction operators or by using Fourier transforms.

Based on these decompositions we present preconditioners that approximate H^h and $I_h^{2h} I_{2h}^h$ as an approximation to P^h (section 4). The advantage of this scheme is its straightforward implementation. However, it is hard to analyze its convergence for reasons that we will discuss in the following sections. We also present a second smoother in which we use a two-step stationary solver that acts exclusively on the high-frequency spectrum using exact frequency truncations for P^h (section 5).

4. Preconditioner as smoother and restriction-prolongation projection. We will consider V-cycle multigrid as a preconditioner for CG, since we are not using a Galerkin coarse-grid operator. In the multigrid preconditioner, we will use one application of the preconditioner described in this section as a smoothing step within multigrid. Our contribution here is the design of appropriate preconditioners for the smoothing step within multigrid. The preconditioner will be based on an inexact inversion of the $(I - P^h)H^h$. To that end we need to approximate P^h and H^h; P^h will be approximated by $I_h^{2h} I_{2h}^h$ since this approach generalizes to arbitrary meshes, and it is easy to implement. For H^h we will explore two approaches: one based on a level-dependent parameter (King preconditioner (K-CG)), and one based on inexact forward and adjoint solves (pointwise preconditioner).

King preconditioner. This approach was proposed by King in [26], where a multigrid method for the first kind of integral operator equations were developed. From Figure 1, we can see that if the regularization parameter β is sufficiently large, it can approximate most of the high-frequency spectrum. Therefore, the eigenvalues corresponding to the high-frequency eigenvectors will be β so that $\beta(I - P^h)I \approx (I - P^h)H^h$. Substituting this in the (3.4) we get a single-level preconditioner of the form $\beta^{-1}(I - P^h)$. In an additional approximation step, we substitute the orthogonal projection by standard interpolation-restriction operators. Therefore the single-level K-CG is given by $\beta^{-1}(I - I_{2h}^h I_h^{2h})$. In a more general case when β is not sufficiently large and when the eigenvalue distribution is compeletely known, β^{-1} is replaced by a level-dependent parameter ξ_j [27], which is defined as

$$\xi_j = \frac{0.9}{\sigma_{\frac{N+1}{2}}}, \tag{4.1}$$

where $\sigma_{\frac{N+1}{2}}$ is the $\frac{N+1}{2}$th eigenvalue, j is the level, and N is the number of grid points. In Table 3, we summarize the spectral properties of the restriction operator I_h^{2h}, prolongation operators I_{2h}^h, and the orthogonal decomposition operator $I - I_{2h}^h I_h^{2h}$.

Pointwise preconditioner. The pointwise preconditioner is based on a pointwise approximation of the reduced Hessian, combined with the high-frequency filtering described in the previous section. The approximate reduced Hessian $\tilde{H}^h$ should approximate well the high-frequency of the true Hessian (for $\beta = 0$) and should be easy to compute. Here we propose a simple waveform-Jacobi relaxation in time [22]. If we discretize in space using the standard three-point stencil for the Laplacian on a

uniform grid and introduce a space-Jacobi splitting a matrix-vector multiplication with the reduced Hessian (in the frequency domain) is given by

$$\begin{aligned} \frac{\partial y}{\partial t} + \frac{2\nu}{N^2} y - \frac{2\nu}{N^2} \cos \frac{k\pi}{N} y = u, \qquad & \text{solve for} \quad y, \\ \frac{\partial \lambda}{\partial t} + \frac{2\nu}{N^2} \lambda - \frac{2\nu}{N^2} \cos \frac{k\pi}{N} \lambda + y(T-t) = 0, \qquad & \text{solve for} \quad \lambda, \\ v = \int \lambda, \qquad & v = Hu. \end{aligned}$$

Here k is the wavenumber, and y, λ, and u represent the magnitude of the kth eigenvector. The approximate waveform-Jacobi relaxation is given by

$$\begin{aligned} \frac{\partial y_i}{\partial t} + \frac{2\nu}{N^2} y_i - \frac{2\nu}{N^2} \cos \frac{k\pi}{N} y_{i-1} = u, \qquad & i = 1 \dots M, \\ \frac{\partial \lambda_i}{\partial t} + \frac{2\nu}{N^2} \lambda_i - \frac{2\nu}{N^2} \cos \frac{k\pi}{N} \lambda_{i-1} + y_M(T-t) = 0, \qquad & i = 1 \dots M, \\ v = \int \lambda_M, \qquad & v = \tilde{H} u. \end{aligned}$$

The number of iterations M determines the quality of the preconditioner. So far we have only discretized in space. We use a backward-Euler scheme to discretize in time. The number of time steps equals the number of discretization points in space. We term this preconditioner "PFH" for projected Fourier Hessian.

4.1. Numerical experiments for preconditioner as smoother. We report numerical experiments in which we compare the the effectiveness of a V-cycle multigrid preconditioner for (1.2) (Table 4), since we are not using an exact coarse-grid operator. The V-cycle uses linear finite element-based interpolation and restriction operators, and the preconditioners described above as smoothers. Two cases of regularization parameter are considered:[5] $\beta = \frac{10^{-3}h^2}{\nu}, \frac{10^{-2}h^2}{\nu}$. We also study the effect of the diffusion coefficient. We consider two cases of diffusion coefficient: $\nu = 1$ and 0.01 are considered in the spectral domain. In all of the experiments, $\tilde{H}$ is constructed using 20 waveform-Jacobi iterations for the adjoint and forward problems. In Table 4, results are given for the number of PCG iterations with a V-cycle multigrid preconditioner. The pointwise preconditioner as smoother converges in a few iterations in all cases, whereas the K-CG takes an increasingly larger number of iterations in the case of $\beta = 10^{-3}h^2/\nu$. The pointwise preconditioner is faster than the K-CG in both the cases. In Figure 3, the decrease in the relative residual is compared for these two preconditioners as smoothers in the multigrid preconditioner for a PCG solver.

The pointwise preconditioner explicitly approximates the high-frequency spectrum using inexact forward and adjoint solves. Whereas, the K-CG uses a single eigenvalue, which is the mesh-dependent parameter ξ_j to approximate the high-frequency spectrum. Since the high-frequency spectrum has a wide distribution of eigenvalues, the pointwise preconditioner performs better than the K-CG, which uses just a single

[5]The regularization parameter is chosen to trade-off stability and fidelity to the data. In the present problem, the discretization error is of $\mathcal{O}(h^2)$ and acts as a noise to the problem. In these synthetic experiments (in which we commit several "inverse crimes", [14]) we know that the exact spectrum of the reduced Hessian, the level of noise, and our reconstructed solution is expected to be smooth. So the choice regularization is not an issue. In the general case, the choice of regularization is of paramount importance. But this is beyond the scope of this paper.

TABLE 4

The performance of a PCG as a solver with a multigrid preconditioner, with a preconditioner as a smoother in multigrid. N is the size of the problem, K-PCG corresponds to the PCG iterations with multigrid preconditioner $V(1,1)$, with a K-CG as smoother. A mesh-dependent parameter ξ_j as defined (4.1) is used in the K-CG. PF-PCG corresponds to the number of PCG iterations with multigrid preconditioner $V(1,1)$, with a pointwise preconditioner as the smoother. Stopping criterion for PCG is $\|r\|/\|r^0\| \leq 10^{-12}$. Two cases of the regularization parameter are considered: Case 1 is $\beta = 10^{-3}h^2/\nu$, and Case 2 is $\beta = 10^{-2}h^2/\nu$. Coarsest level is 16. Parameters used are $\nu = 1$ and $\nu = 0.01$, with $T = 1$, $N_t = Ns$, and $L = 1$.

$\nu = 1$

N	$\beta(\sigma > \beta)$		K-PCG		PF-PCG	
512	4e-09 (40)	4e-08 (22)	6	3	4	3
1024	1e-09 (57)	1e-08 (32)	9	3	4	3
2048	2e-10 (81)	2e-09 (45)	13	4	4	3
4096	6e-11 (114)	6e-10 (64)	16	6	4	3

$\nu = 0.01$

N	$\beta(\sigma > \beta)$		K-PCG		PF-PCG	
512	4e-07 (131)	4e-06 (72)	11	5	4	4
1024	1e-07 (183)	1e-06 (102)	15	5	4	4
2048	2e-08 (257)	2e-07 (144)	35	5	4	4
4096	6e-09 (363)	6e-08 (203)	24	7	5	4

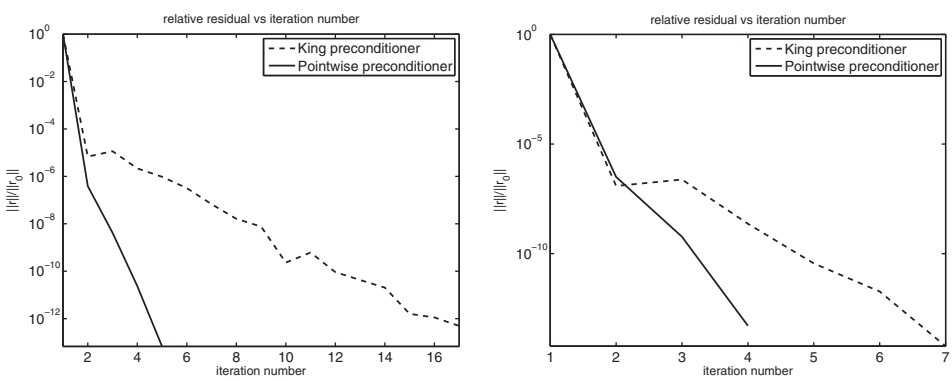

FIG. 3. *PCG residual w.r.t. the regularization parameter. The relative residual vs. iteration number for two cases of a regularization parameter is shown. In this figure, we compare the rate at which the residual decreases for different preconditioners used as smoothers within the multigrid preconditioner. Two cases of regularization parameters are considered. Case 1: $\beta = 10^{-3}h^2$ is shown on the left, and $\beta = 10^{-2}h^2$ is shown on the right, where $h = 1/4096$. King's preconditioner (K-CG) converges slower than the pointwise preconditioner in both of the cases. K-CG approximates the high-frequency spectrum using one eigenvalue, whereas the pointwise preconditioner approximates the whole distribution of eigenvalues in the high-frequency spectrum, which results in the faster convergence of the pointwise preconditioner.*

eigenvalue to approximate the high-frequency spectrum. Using both the preconditioners, we can solve the problem in $\mathcal{O}(1)$ iterations as shown in the numerical results. This is equivalent to solving the forward and adjoint problems a constant number of times independent of the mesh size and the regularization parameter.

The K-CG has negligible computational cost when compared to the actual reduced Hessian matvec. In the case of a pointwise preconditioner, there is an overhead associated in computing $\tilde{H}^{-1}$ every iteration. For a given residual reduction, it takes a constant number of CG iterations to invert $\tilde{H}^h$. Since the residual reduction is close to machine accuracy, $(\tilde{H}^h)^{-1}$ is a linear operator and creates no convergence

problems in the smoother. The computational cost of evaluating $\tilde{H}^{-1}$, however, is much higher than the cost associated with applying the K-CG. When the regularization parameter is large, the pointwise preconditioner is not necessary. However, the pointwise preconditioner approximates the high-frequency spectrum better than the K-CG. As seen from the numerical results, the pointwise preconditioner converges in fewer iterations for different cases of regularization parameters and diffusion coefficients than the K-CG. Therefore, the pointwise preconditioner can be used in cases where the regularization parameter is too small, though it has more computational overhead than the K-CG.

Solving the forward and adjoint problems has a computational complexity[6] of $\mathcal{O}(N^2)$ using multigrid algorithms to solve the linear algebraic system of equations at each time step in the case of linear problems, where N is the number of grid points. The pointwise preconditioner has an additional complexity of inverting an approximate reduced Hessian, which requires a few inexact forward and adjoint solves. In the worst case, this is equivalent to an additional reduced Hessian matvec at every pointwise preconditioner operation which has an $\mathcal{O}(N^2)$ computational cost. This would change the constants in the overall computational complexity of solving the problem.

The estimates of the eigenvalue distribution, in general, are computationally expensive. In such cases, we might have to use the regularization parameter for different levels. The pointwise preconditioner has a similar effect on the reduced Hessian at all of the levels for different regularization parameters. From Figure 4 for $\beta = 10^{-3}h^2$ and for $\beta = 10^{-2}h^2$, PFH has a significant clustering of eigenvalues near high-frequency eigenvectors. Whereas, for the K-CG, if we use the regularization parameter, there is little clustering of eigenvalues at coarser levels. Hence, using the regularization parameter instead of the level-dependent parameter would affect the performance of the K-CG.

5. Two-step stationary scheme as smoother and FFT filtering. As discussed above, the K-CG is slower than the pointwise preconditioner for smaller regularization parameters. Though the pointwise preconditioner is faster than the K-CG it has an overhead of computing the inverse of the approximate reduced Hessian at every iteration. The combination of multigrid with PCG and the pointwise preconditioner performs well, at least for the simple model. Our target application ultimately will involve variable-coefficient problems and partial observations. In those cases, we expect a higher number of iterations. Although we can use multigrid as a solver, it would be preferable to combine it with an outer-PCG acceleration. In the above multigrid preconditioner, there is no explicit iterative scheme working as a smoother. Therefore, this cannot be used as a robust preconditioner in variable-coefficient problems and partial observations.

As an alternative we propose to use an iterative two-step stationary scheme [16] (Algorithm 2) as a smoother. Then, in the constant coefficient case, one can derive exact smoothing factors. As in classical multigrid theory [12, 19], the analysis becomes approximate in the case of variable coefficients. One disadvantage of the two-step solver is that it requires estimates of extreme eigenvalues. To avoid computing eigenvalues we use a spectral cutoff and analytic spectrum estimates. In this manner, the smoother is forced to iterate on the high-frequency regime. In the following, we present the algorithm in detail, analyze its convergence factor, and conduct numerical experiments to test our hypothesis.

[6] $N_t = \mathcal{O}(N)$.

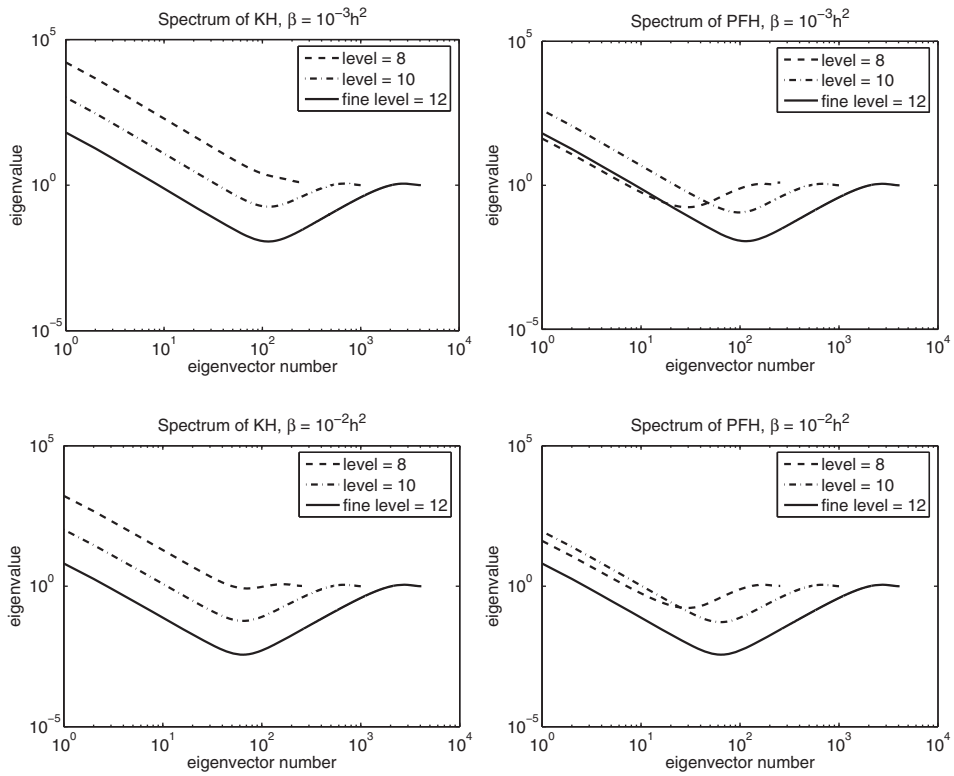

FIG. 4. *A comparison of the spectrum of the preconditioned reduced Hessian. We report the discrete spectrum of the preconditioned reduced Hessian with King preconditioner (KH) and pointwise (PFH) preconditioner in spectral domain for the finest level and different coarser levels. The PFH preconditioner has a similar clustering of eigenvalues at a high-frequency region at finer and coarser levels for different values of the regularization parameter. KH does not show a similar trend at finer and coarser levels for different values of regularization parameter. This would result in the better performance of PFH than KH for smaller regularization parameters if we use regularization parameter in the King preconditioner.*

ALGORITHM 2 Standard two-step stationary iterative scheme (Solve $Ad^T = d_{in}$).

$\sigma_1 = \sigma_{\min}(A)$ and $\sigma_n = \sigma_{\max}(A)$
$\rho = \frac{1-\sigma_1/\sigma_n}{1+\sigma_1/\sigma_n}$, $\alpha = \frac{2}{1+(1-\rho^2)^{1/2}}$, $\xi_0 = \frac{2}{\sigma_1+\sigma_n}$, $\xi = \frac{2\alpha}{\sigma_1+\sigma_n}$,
$r = -d_{in}$, $d_0 = 0$, $d_1 = \xi_0 r$
for $i = 1 \ldots L$ **do**
  $r = Ad_1 - d_{in}$
  $d = \alpha d_1 + (1-\alpha)d_0 - \xi r$
  $d_0 = d_1,\ d_1 = d$
end for

Since we are interested in removing the high-frequency error components while smoothing, we iterate on (3.4) in the smoothing step. In (3.4), the projection operator $I - P^h$ can be defined as a filter which removes the eigenvector components corresponding to small wave numbers. Let us denote the filtering operation by $W = I - P^h$. In the present problem, the eigenvectors are sines. Therefore, we can

Algorithm 3 Projection using Sine Transform.

1: Let u be the input vector, and let $v = Wu$.
2: $\hat{u} = \mathrm{DST}(u)$ — transform into spectral domain.
3: $\hat{u}_k = 0 \quad 1 < k < \frac{N-1}{2}$ — filtering in spectral domain.
4: $v = \mathrm{IDST}(\hat{u})$ — transform back to spatial domain.

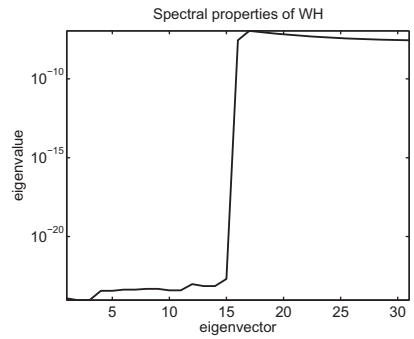

Fig. 5. *Eigenvalues for the spectrally filtered reduced Hessian. Here we report the magnitude of the eigenvalues of the reduced Hessian. Here the W operator represents an exact high-pass filter. During multigrid smoothing, the composite operator WH is inexactly inverted using a two-step stationary iterative solver.*

use discrete sine transforms to filter the low-frequency components of an input vector (Algorithm 3).[7]

The problem that we solve during the smoothing iterations is

$$WH^h u = Wg. \tag{5.1}$$

Since the null space of W is nontrivial, (5.1) is singular. However, it is proved that a positive semidefinite system of the form (5.1) can be solved by PCG as long as the right-hand side is consistent [23]. The two-step iterative scheme (see Figure 5) requires that all of the eigenvalues of the matrix (WH^h) be positive (see section 5.2.3 in [4]). Let $W = ZZ^T$, where $Z = [v_{\frac{N+1}{2}}, \ldots, v_{N-1}]$ in which v_k correspond to the kth eigenvector of H^h. The subspace spanned by the eigenvectors $[v_1, \ldots, v_{\frac{N-1}{2}}]$ is invariant and does not influence the convergence rate of the two-step iterative solver.

We define one smoothing step as one iteration of the two-step scheme. In case of a nonzero initial guess, the error e^l after l smoothing iterations is given by

$$e^l = \left(\left(\alpha - \frac{2\alpha WH}{\sigma_1 + \sigma_n}\right)\left(1 - \frac{2WH}{\sigma_1 + \sigma_n}\right) + (1-\alpha)\right)^l e^0, \tag{5.2}$$

where e^0 is the initial error and α is defined in Algorithm 2; σ_1 and σ_n in (5.2) are defined later. Let $e^0 = \Sigma_{k=1}^{N-1} m_k v_k$, where v_k are the eigenvectors of H and m_k are the corresponding error magnitudes. Assuming that W results in an exact decomposition (3.2) eigenvalues of WH are $\sigma(WH) = \{0, 0, \ldots, \sigma_{N+1/2}, \ldots, \sigma_{N-1}\}$, where σ_k correspond to the kth eigenvalue of the reduced Hessian H. Substitute e^0 in (5.2) and take one smoothing iteration, we get $e_k^1 = m_k v_k \ \forall\, 1 < k < \frac{N-1}{2}$, where e_k^1 is the error component in the kth eigenvector direction after one smoothing iteration.

[7] This is true only in the present case.

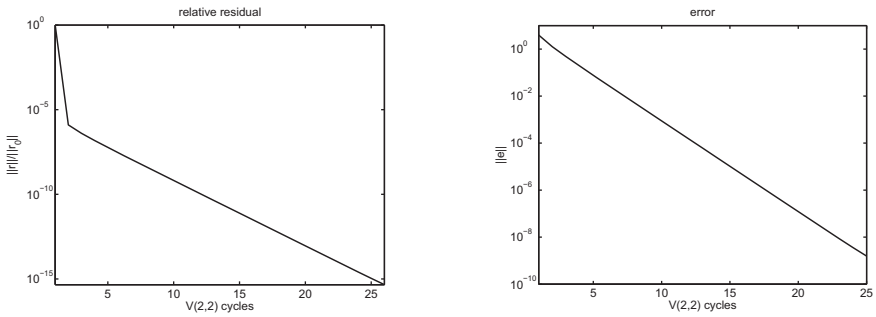

FIG. 6. *Residual reduction in multigrid. Relative residual and error vs. the number of two-level $V(2,2)$ cycles for $\nu = 1$ and level six and using H_G^{2h} as the coarse-grid operator.*

Similarly,

$$e_k^1 = \underbrace{\left(\left(\alpha - \frac{2\alpha\sigma_k}{\sigma_1+\sigma_n}\right)\left(1 - \frac{2\sigma_k}{\sigma_1+\sigma_n}\right) + (1-\alpha)\right)}_{\mu_k} m_k v_k \ \forall \ \frac{N+1}{2} < k < N-1.$$

Here μ_k is the amplification factor of the error component in the kth eigenvector direction. The eigenvalues $\sigma_1, \ldots, \sigma_{N-1/2}$ do not affect the iteration. The smoothing factor μ is given by $\max_k(\mu_k)$. To estimate μ we need estimates of σ_1 and σ_n, respectively. For the constant coefficient case we have computed these values analytically by (2.9): We fix the values of σ_1 and σ_n to be σ_N and $\sigma_{(N+1)/2}$, respectively. Then, since $\sigma_1 \leq \sigma_k \leq \sigma_n$, we have $\mu_k < 1 \ \forall \ \frac{N+1}{2} < k < N-1$. Using the exact spectrum we can also show that the ratio $\sigma_1/\sigma_n = 1/4$, and it is mesh-independent (for $\beta = 0$). In the variable-coefficient case, we use a heuristic. We estimate the σ_n of the unregularized Hessian using a Krylov method on the reduced Hessian. Then, guided by the constant coefficient case, we set $\sigma_1 = \sigma_n/4$. For this ratio the smoothing factor μ is 0.288 for $\nu = 1, 0.01$. In Figure 6, we can see that relative residual drops suddenly in the first V-cycle and maintains a constant ratio thereafter. The reduction in the error is constant, which is expected. The sudden drop in the relative residual is because of the coarse-grid correction where the low-frequency error components are removed.

According to the spectrum of H^h low-frequency error components correspond to large eigenvalues and since $r = He$, there is a sudden drop in the residual. After the first V-cycle, the reduction in the residual is less than the first V-cycle. We report the number of V-cycles to get a relative residual of 10^{-8} in Table 5 for different mesh sizes and two diffusion coefficients $\nu = 1, 0.01$. The number of V-cycles is mesh-independent.

5.1. Multigrid preconditioner. As we have mentioned, one difficulty in designing a multigrid scheme for the reduced Hessian operator is the choice of the coarse-grid operator. If we use H^{2h} instead H_G^{2h}, we cannot remove certain error components that belong to the intermediate frequency range (of the fine-grid). These error components are removed neither by the two-step scheme nor by the coarse-grid correction. To obtain a working scheme we use PCG as the primary solver and use the multigrid algorithm as a preconditioner. In this way, we can guarantee error removal in all frequencies.

We denote the multigrid preconditioner by M^{-1} (Algorithm 4), and the smoothing operator by $S(A, f, u)$, where A, f, and u, are the matvec operator, the right-hand

TABLE 5

Convergence for zero regularization parameter. Here we report the number of $V(2,2)$-cycles (two levels) to reduce the residual by eight orders of magnitude; V_1 and $V_{0.01}$ correspond to experiments with diffusion coefficients $\nu = 1$ and $\nu = 0.01$, respectively. All tests are given for the $\beta = 0$ case. The size of the problem is 2^{level}. The initial guess $u^0 = u^* + \Sigma_k \sin(k\pi x)$, where u^* is the exact solution. The theoretical smoothing factor is 0.288, and the numerical is 0.29. The smoothing factor is mesh-independent.

Level	V_1	$V_{0.01}$
4	16	22
5	22	27
6	25	32
7	26	34
8	24	33

ALGORITHM 4 Two-level Exact Multigrid Preconditioner ($u^h = M^{-1} f^h$).

1: $u^h = S(WH^h, Wf^h, 0)$	presmoothing with H^h.
2: $r^h = f^h - Hu^h$	residual evaluation.
3: $r^{2h} = I_h^{2h} r^h$	restriction.
4: $e^{2h} = (H^{2h})^{-1} r^{2h}$	coarse-grid correction.
5: $e^h = I_{2h}^h e^h$	prolongation.
6: $u^h = u^h + e^h$	correction.
7: $u^h = S(WH^h, Wf^h, u^h)$	postsmoothing with H^h.

side, and the initial guess, respectively. We denote the inexact multigrid preconditioner by $\tilde{M}^{-1}$. We term it "*inexact*" because, in the smoother, we replace the exact reduced Hessian H^h by an approximation $\tilde{H}^h$; $\tilde{H}^h$ is obtained by replacing exact forward and adjoint solves to do one reduced Hessian matvec by inexact forward and adjoint solves. In order to do this, we use a fixed number of Jacobi-waveform relaxation iterations [30]. The number of waveform relaxation steps have to be increased (with increasing problem size) in order to get a good preconditioner due to the convergence properties of the Jacobi-waveform relaxation method.

In the Jacobi-waveform relaxation, we solve ordinary differential equations at every spatial grid point, thus removing the spatial-coupling that arises from the discretization of the Laplacian operator. This is different from the standard spatial-weighted Jacobi scheme. The (high-frequency) convergence factors of the weighted-Jacobi method are mesh-independent—unlike the Jacobi-waveform relaxation that gives rise to a mesh-dependent convergence factor [30].[8]

5.2. Results and discussion. In this section, we present results for the constant and variable-coefficient case, as well as the case in which we have partial observations. We report PCG iterations with multigrid preconditioners M^{-1} and $\tilde{M}^{-1}$. We also show the sensitivity of the number of Jacobi-waveform relaxation steps on the number of PCG iterations. We present numerical results that interrogate the sensitivity of the scheme on the diffusion coefficient, the number of waveform relaxations, and the coarsening strategy (semicoarsening or space only vs. standard space-time coarsening).

[8] One could use standard time-marching schemes in which the exact inversions of the spatial operator can be replaced by an inexact solve, like weighted-Jacobi.

TABLE 6

PCG convergence using the exact high-frequency spectrum of the reduced Hessian. The number of PCG iterations with the two-level multigrid preconditioner with exact reduced Hessian in the smoother M^{-1} and inexact reduced Hessian in the smoother $\tilde{M}^{-1}$. Semicoarsening in space (subscript sec*) and standard coarsening (subscript* stc*) in space and time are considered. CG is terminated when $\|r\|/\|r_0\| < 10^{-8}$ or when the number of iterations is $2N$, where N is the size of the problem. The values in the brackets are the number of eigenvectors not filtered by the regularization. Here,* 16 *Jacobi-waveform relaxation sweeps are performed for forward and adjoint solves to do one matrix-vector operation of $\tilde{H}$.*

$\nu = 1$

N	β	M_{sec}^{-1}	$\tilde{M}_{\text{sec}}^{-1}$	M_{stc}^{-1}	$\tilde{M}_{\text{stc}}^{-1}$
31	5e-07 (31)	10	10	10	10
63	1e-07 (44)	9	9	9	9
127	3e-08 (63)	6	6	7	7
255	7e-09 (89)	4	4	4	4

$\nu = 0.01$

N	β	M_{sec}^{-1}	$\tilde{M}_{\text{sec}}^{-1}$	M_{stc}^{-1}	$\tilde{M}_{\text{stc}}^{-1}$
31	5e-03 (31)	14	14	13	13
63	1e-03 (44)	13	13	13	13
127	3e-04 (63	10	10	10	10
255	7e-05 (89)	7	7	7	7

5.2.1. Constant coefficients case. Results for two cases of diffusion coefficients $\nu = 1$ and 0.01 are given in Table 6. The convergence of PCG is mesh-independent. We are reporting results for both two-level and multiple V-cycle preconditioners. The condition number of H^h is $\kappa = \mathcal{O}(N^4)$, where N is the size of the problem. Therefore, without a preconditioner the number of CG iterations will be $\mathcal{O}(N^2)$. With the multigrid preconditioner, the number of CG iterations becomes mesh-independent $\mathcal{O}(1)$. Using backward-Euler time-stepping (combined with a fast elliptic solver) for the forward and adjoint problems, the amount of work done for each reduced Hessian matvec is $\mathcal{O}(NN_t + N\log^2 N)$, where the first term comes from the forward and adjoint solve with N_t time steps, and the second part comes from the multigrid sweeps (the square in the logarithm is related to the fast sine transforms). Therefore, the total amount of work done to solve the system is brought down from $\mathcal{O}(N^4)$ to $\mathcal{O}(N^2)$. *Therefore, the solution of the inverse problem requires solving the forward problem Q times, but Q is independent of the regularization parameter and the mesh size.*

5.2.2. Nonconstant coefficient case. We extend the above ideas to solve inverse problems in parabolic problems with nonconstant coefficients:

$$\frac{\partial y}{\partial t} - \Delta y = ay + bu \quad \text{in} \quad D, \quad y = 0 \quad \text{on} \quad \partial\Omega, \quad y(\Omega, 0) = 0 \quad \text{in} \quad \Omega.$$

Equations of this kind are obtained when a nonlinear reaction-diffusion equation is linearized. In this case, sines are not the eigenvectors of the reduced Hessian. We assume that a and b are smooth and bounded. Therefore, the Fourier coefficients of a and b decay to zero relatively fast. From this assumption, the contribution of a and b to the spectrum of the forward problem in the high-frequency region is negligible. Using this observation and considering the computational cost of constructing the exact high-frequency eigenspace of the reduced Hessian, we use sine transforms to decompose the finite-dimensional space to get acceptable convergence. The numerical results that we next discuss indicate that our assumption is reasonable. The reconstructed source is depicted in Figure 8.

TABLE 7

Convergence comparisons for PCG using inexact approximations of the reduced Hessian. We report the number of PCG iterations using a multigrid preconditioner that employs (in the smoother) either an exact reduced Hessian M^{-1} or an inexact reduced Hessian $\tilde{M}^{-1}$. Semicoarsening in space (subscript sec*) and standard coarsening (subscript* stc*) in space and time are considered. PCG is terminated when $\|r\|/\|r_0\| < 10^{-8}$. The values in the brackets in the column β are the number of reconstructed eigenvectors (not filtered by the regularization). The size of the coarsest level problem is* 15*. Here* 16 *Jacobi-waveform relaxation steps are done for forward and adjoint solves to do one matrix-vector operation of $\tilde{H}$.*

$\nu = 1.0$

N	β	M_{sec}^{-1}	$\tilde{M}_{\text{sec}}^{-1}$	M_{stc}^{-1}	$\tilde{M}_{\text{stc}}^{-1}$
31	5e-07 (31)	10	10	10	10
63	1e-07 (44)	13	13	13	17
127	3e-08 (63)	13	14	14	16
255	7e-09 (89)	13	19	13	16
511	2e-09 (127)	15	18	15	17
1023	5e-10 (180)	15	17	15	17

$\nu = 0.01$

N	β	M_{sec}^{-1}	$\tilde{M}_{\text{sec}}^{-1}$	M_{stc}^{-1}	$\tilde{M}_{\text{stc}}^{-1}$
31	5e-03 (31)	14	14	13	13
63	1e-03 (44)	17	17	17	17
127	3e-04 (63)	20	20	19	21
255	7e-05 (89)	23	24	21	23
511	2e-05 (127)	24	26	24	27
1023	5e-06 (180)	25	28	25	30

Two cases of coarsening strategies are implemented: (1) semicoarsening in space and (2) standard-coarsening is space and time. The mesh-independent convergence of PCG with a multigrid preconditioner is observed in the case of M^{-1}, whereas the performance of $\tilde{M}^{-1}$ slightly deteriorates with mesh size. Standard coarsening does not perform as well as semicoarsening. This can be explained by the fact that the convergence factors of the Jacobi-waveform relaxation are mesh dependent, given by $1 - \mathcal{O}(h^2)$, and the convergence factors using standard-coarsening are worse than semicoarsening [22]. If we increase the number of Jacobi-waveform relaxation steps with the mesh size, then we could observe that the number of PCG iterations with a $\tilde{M}^{-1}$ preconditioner will tend to the number of iterations taken by M^{-1}. The results of PCG with multigrid preconditioners is shown in (Tables 7 and 8). The sensitivity of number of PCG iterations with increasing number of Jacobi-waveform relaxation steps is reported in Table 9. The number of PCG iterations taken by $\tilde{M}^{-1}$ decrease with an increase in Jacobi-waveform relaxation steps. A lower bound to the number of iterations taken by $\tilde{M}^{-1}$ is the number of iterations taken by M^{-1}. The overall computational complexity in using M^{-1} and $\tilde{M}^{-1}$ differs only by a constant if we use a sufficient number of Jacobi-waveform relaxation steps in $\tilde{M}^{-1}$.

In Table 10, we report the number of PCG iterations when the data is given at seven equally spaced points in space at all of the time steps. An exact multigrid preconditioner with a standard-coarsening of the exact reduced Hessian and an approximate multigrid preconditioner with semicoarsening of the approximate reduced Hessian are considered. Results in Table 10 show that the multigrid preconditioners presented here are robust even in practical situations when the data is sparse.

6. Full-space methods. A disadvantage of a reduced-space approach is the need to solve the forward and adjoint problems far from the optimum. In this section, we discuss full-space methods where the optimality system is solved for state, adjoint,

TABLE 8

Multigrid performance for the variable coefficient case. The number of CG iterations for the two-level preconditioner with exact reduced Hessian in the smoother M^{-1} and inexact reduced Hessian in the smoother $\tilde{M}^{-1}$. Semicoarsening in space (subscript sec*) and standard coarsening (subscript* stc*) in space and time are considered. CG is terminated when $\|r\|/\|r_0\| < 10^{-8}$ or when the number of iterations is $2N$, where N is the size of the problem. Case* I *has $a = \hat{u}$ and $b = \hat{y}$, and Case* II *has $a = 2\hat{y}\hat{u}$ and $b = \hat{y}^2$, where $\hat{y}$ is a traveling wave with a Gaussian shape (Figure 7) and $\hat{u} = \text{Gaussian}(0.2) + \sin(\pi x)$ (0.2 is the center of the Gaussian). Here 8 Jacobi-waveform relaxation sweeps are performed in all of the cases in $\tilde{M}^{-1}$.*

CASE I : $a = \hat{u}, b = \hat{y}$

N	β	M_{sec}^{-1}	$\tilde{M}_{\text{sec}}^{-1}$	M_{stc}^{-1}	$\tilde{M}_{\text{stc}}^{-1}$
31	2e-06	12	15	13	16
63	5e-07	11	11	13	13
127	1e-07	10	10	12	12
255	3e-08	8	9	10	10

CASE II : $a = 2\hat{y}\hat{u}, b = \hat{y}^2$

N	β	M_{sec}^{-1}	$\tilde{M}_{\text{sec}}^{-1}$	M_{stc}^{-1}	$\tilde{M}_{\text{stc}}^{-1}$
31	2e-06	13	15	14	15
63	5e-07	11	12	15	15
127	1e-07	11	11	11	11
255	3e-08	9	9	10	10

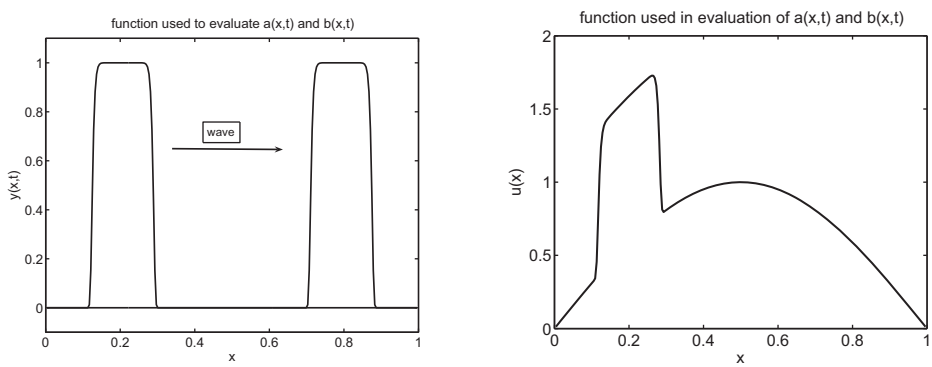

FIG. 7. *Parabolic PDE with variable coefficients. We have constructed a traveling wave solution to emulate solutions to reaction-diffusion equations. The function $\hat{y}(x,t)$ is used to evaluate $a(x,t)$ and $b(x,t)$, which are then used in numerical experiments. The inversion parameter u is depicted on the left panel.*

and inversion variables in one shot. The main advantage of a full-space method is that we can avoid solving the forward and adjoint problems at each reduced Hessian matrix-vector multiplication. On the other hand, the size of the KKT system is more than twice as big as the size of the forward problem. Furthermore, the KKT system is ill-conditioned and indefinite. For such systems, Krylov solvers are slow to converge. Therefore, a good preconditioner is required to make the full-space method efficient. A Lagrange–Newton–Krylov–Schur preconditioner (LNKS) has been proposed in [7] and [8] in the context of solving optimal control problems with elliptic PDE constraints. Space-time multigrid methods for a parabolic PDE have been considered in literature [22]. In the present problem, we have two coupled PDEs with opposite time orientation, something that poses a significant challenge to a multigrid scheme. These issues have been considered, and to a great extend addressed, in [9] in which a new time-split collective Gauss–Seidel method (TS-CGS) was introduced. In [9],

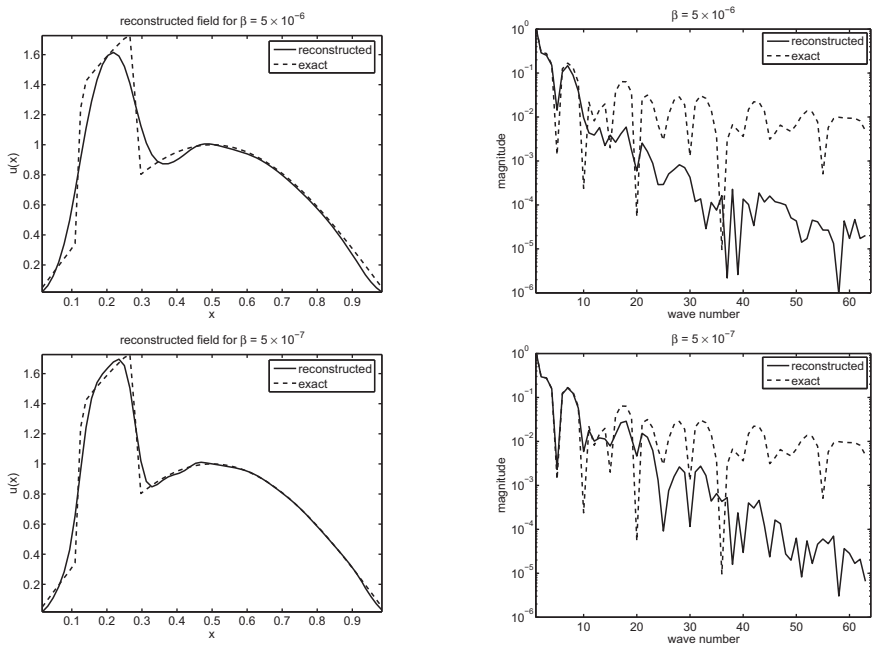

FIG. 8. *Reconstructed source. Here we show the reconstructed curves in the real (left column) and frequency domains (right column) for $N = 64$ and two values of the regularization parameter.*

TABLE 9

Dependence on the fidelity of the reduced Hessian approximation. The number of CG iterations for a multilevel preconditioner with exact reduced Hessian in the smoother M^{-1} and inexact reduced Hessian in the smoother $\tilde{M}^{-1}$. Semicoarsening in space (subscript sec*) and standard coarsening (subscript* stc*) in space and time are considered. CG is terminated when $\|r\|/\|r_0\| < 10^{-8}$ or when the number of iterations is $2N$, where N is the size of the problem. Case* I *has the $a = \hat{u}$ and $b = \hat{y}$, and Case* II *has $a = 2\hat{y}\hat{u}$ and $b = \hat{y}^2$, where $\hat{y}$ is a traveling wave with a Gaussian shape (Figure 7) and $\hat{u} = \text{Gaussian}(0.2) + \sin(\pi x)$ (0.2 is the center of the Gaussian). The number of Jacobi-waveform relaxation steps used in $\tilde{M}^{-1}$ is given in brackets.*

CASE I : $a = \hat{u}, b = \hat{y}$

N	β	M^{-1}_{sec}	$\tilde{M}^{-1}_{\text{sec}}(8)$	$\tilde{M}^{-1}_{\text{sec}}(16)$	$\tilde{M}^{-1}_{\text{sec}}(32)$	M^{-1}_{stc}	$\tilde{M}^{-1}_{\text{stc}}(8)$	$\tilde{M}^{-1}_{\text{stc}}(16)$	$\tilde{M}^{-1}_{\text{stc}}(32)$
31	2e-06	12	15	14	15	13	16	15	15
63	5e-07	13	16	15	14	14	16	16	16
127	1e-07	14	27	24	18	17	40	30	30
255	3e-08	18	52	37	26	23	-	-	-

CASE II : $a = 2\hat{y}\hat{u}, b = \hat{y}^2$

N	β	M^{-1}_{sec}	$\tilde{M}^{-1}_{\text{sec}}(8)$	$\tilde{M}^{-1}_{\text{sec}}(16)$	$\tilde{M}^{-1}_{\text{sec}}(32)$	M^{-1}_{stc}	$\tilde{M}^{-1}_{\text{stc}}(8)$	$\tilde{M}^{-1}_{\text{stc}}(16)$	$\tilde{M}^{-1}_{\text{stc}}(32)$
31	2e-06	13	15	15	15	14	15	15	15
63	5e-07	14	16	16	16	16	21	21	20
127	1e-07	16	27	28	21	18	-	30	23
255	3e-08	19	140	60	32	20	-	-	-

a space-time control variable was used, penalized by an L^2 term. The optimality condition with respect to the controls is simply a scalar relation between the control and Lagrange multipliers. In our case, the control equation is an algebraic-integral equation.

In the following, we discuss LNKS and TS-CGS for our setting. We follow the notation in [9].

TABLE 10

Variable coefficients and partial observations. The number of PCG iterations for a multilevel preconditioner for seven observations on the spatial grid. Semicoarsening in space is represented by subscript sec *and standard coarsening is represented by subscript* stc*. CG is terminated when* $\|r\|/\|r_0\| < 10^{-8}$*. Case* I *has the* $a = \hat{u}$ *and* $b = \hat{y}$*, and Case* II *has* $a = 2\hat{y}\hat{u}$ *and* $b = \hat{y}^2$*, where* $\hat{y}$ *is a traveling wave with a Gaussian shape (Figure* 7*) and* $\hat{u} = \text{Gaussian}(0.2) + \sin(\pi x)$ *(*0.2 *is the center of the Gaussian).*

CASE I : $a = \hat{u}, b = \hat{y}$			
N	β	M_{stc}^{-1}	$\tilde{M}_{\text{sec}}^{-1}(32)$
31	2e-04	13	13
63	5e-05	15	15
127	1e-05	18	22
255	3e-06	19	23

CASE II : $a = 2\hat{y}\hat{u}, b = \hat{y}^2$			
N	β	M_{stc}^{-1}	$\tilde{M}_{\text{sec}}^{-1}(32)$
31	2e-04	13	13
63	5e-05	15	15
127	1e-05	18	21
255	3e-06	18	23

6.1. Lagrange–Newton–Krylov–Schur method (LNKS). First, we briefly discuss the LNKS method proposed in [7] and [8]. The LNKS method is based on the block factorization of the KKT system which is shown below. (Please refer to [7] for further details.)

$$K = \begin{bmatrix} I & 0 & J^T \\ 0 & \beta I & C^T \\ J & C & 0 \end{bmatrix} = \begin{bmatrix} J^{-1} & 0 & I \\ 0 & I & C^T J^{-T} \\ I & 0 & 0 \end{bmatrix} \begin{bmatrix} J & C & 0 \\ 0 & H & 0 \\ 0 & -J^{-1}C & J^T \end{bmatrix}. \tag{6.1}$$

The KKT preconditioner P is then defined as

$$\tilde{P} = \begin{bmatrix} 0 & 0 & I \\ 0 & I & C^T \tilde{J}^{-T} \\ I & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{J} & C & 0 \\ 0 & B & 0 \\ 0 & 0 & \tilde{J}^T \end{bmatrix}. \tag{6.2}$$

In $\tilde{P}$, exact forward J^{-1} and adjoint solves J^{-T} are replaced by inexact solves $\tilde{J}^{-1}$ and $\tilde{J}^{-T}$, respectively. The preconditioned KKT matrix is $\tilde{P}^{-1}K$, where

$$\tilde{P}^{-1} = \begin{bmatrix} \tilde{J}^{-1} & -\tilde{J}^{-1}CB^{-1} & 0 \\ 0 & B^{-1} & 0 \\ 0 & 0 & \tilde{J}^{-T} \end{bmatrix} \begin{bmatrix} 0 & 0 & I \\ -C^T\tilde{J}^{-T} & I & 0 \\ I & 0 & 0 \end{bmatrix}. \tag{6.3}$$

An outline of applying the LNKS preconditioner as a smoother to a vector is given in Algorithm 5. This preconditioner can be used as an accelerator for an iterative solver for liner systems. A popular method for solving large symmetric indefinite systems is MINRES [4]. One major disadvantage of MINRES is that it requires a symmetric positive definite preconditioner, despite the fact that the KKT is indefinite.

ALGORITHM 5 LNKS smoother.

1: Given y, u, $\boldsymbol{\lambda}$, and $f = [f_y, f_u, f_\lambda]$.
2: Evaluate $\tilde{f}_y = y + J^T\boldsymbol{\lambda} - f_y$, $\tilde{f}_u = u + C^T\boldsymbol{\lambda} - f_u$, $\tilde{f}_\lambda = c - f_\lambda$,
3: where $c = Jy + Cu$
4: $\tilde{H}p_u = \tilde{f}_u$ — $\tilde{H}$: pointwise preconditioner.
5: $\tilde{J}p_y = \tilde{f}_y - Cp_u$ — Inexact forward solve.
6: $\tilde{J}^T p_\lambda = \tilde{f}_\lambda - p_y$ — Inexact adjoint solve.
7: $y = y - p_y$, $u = u - p_u$, and $\boldsymbol{\lambda} = \boldsymbol{\lambda} - p_\lambda$. — Update.

Alternatively, the symmetric quasi-minimum residual method (SQMR) can be used with indefinite preconditioners but it requires two matvecs per Krylov iteration, and it does not take advantage of the fact that the KKT system is symmetric [17]. In all of the numerical experiments with LNKS we use SQMR.

We now discuss a multigrid scheme for the full KKT matrix. We use a V-cycle, with standard restriction and prolongation, and one application $\tilde{P}$ as smoother. The goal is to remove high-frequency error components in the state, adjoint, and inversion variables in each step of the smoother without doing exact forward or adjoint solves. Therefore, we use the waveform-Jacobi method. To update the inversion variables we use the pointwise preconditioner discussed in section 4.

6.2. Time-split Collective Gauss–Seidel (TS-CGS). In this method, we eliminate the inversion variables using the inversion equation (6.4),

$$\beta u - \int_T \lambda \, d\, t = 0 \quad \text{in} \quad \Omega. \tag{6.4}$$

(This cannot be done for $\beta = 0$.)

Therefore, we can rewrite the KKT system as follows:

$$\begin{aligned} &\frac{\partial y}{\partial t} - \Delta y = \frac{1}{\beta} \int_T \lambda \, d\, t, \quad y(\Omega, 0) = y_0, \quad y(\partial\Omega, t) = 0, \\ &-\frac{\partial \lambda}{\partial t} - \Delta \lambda = -(y - y^*), \quad \lambda(\Omega, 0) = 0, \lambda(\partial\Omega, t) = 0. \end{aligned} \tag{6.5}$$

Using finite differences for the Laplacian and backward-Euler scheme in time (6.5), the above system can be written as

$$[1 + 2\gamma] y_{im} - \gamma[y_{i-1m} + y_{i+1m}] - y_{im-1} = \frac{\delta t^2}{\beta} \sum_{k=1}^{N_t} \lambda_{ik} \tag{6.6}$$

$$[1 + 2\gamma] \lambda_{im} - \gamma[\lambda_{i-1m} + \lambda_{i+1m}] - \lambda_{im+1} = -\delta t (y_{im} - y^*_{im}), \tag{6.7}$$

where $\gamma = \frac{\delta t}{h^2}$ and i, m represent the spatial and temporal indices of the variables, respectively. In the case of a collective Gauss–Seidel iteration, let us denote the variables as $\phi_k = (y_k, \lambda_k)$ at each grid point. We can write (6.6), (6.7) as $E(\phi_{im}) = [f - A(\phi_{im})] = 0$ at the grid point im. Let E' be the Jacobian of E with respect to (y_k, λ_k). One step of the collective Gauss–Seidel scheme is given by $\phi^1_{im} = \phi^0_{im} - [E'(\phi^0_{im})]^{-1} E(\phi^0_{im})$. This scheme performs well for steady state problems [11], but it diverges in the case of an optimal control of a parabolic PDE because of the opposite time orientation of the state and adjoint equations. In order to overcome this problem, TS-CGS iteration was proposed in [9] (Algorithm 6). Following [9], we use Fourier mode analysis to analyze the convergence properties of the two-grid version of the inverse solver. Let the smoothing operator be S_k, and let the coarse-grid correction be given by CG_k^{k-1}. Fourier symbols are represented with a hat. On the fine grid consider the Fourier components $\phi(j, \boldsymbol{\theta}) = e^{ij \cdot \boldsymbol{\theta}}$, where i is the imaginary unit, $j = (j_x, j_t) \in \mathcal{Z} \times \mathcal{Z}$, $\theta = (\theta_x, \theta_t) \in [\pi, \pi)^2$, and $j \cdot \theta = j_x \theta_x + j_t \theta_t$. The frequency domain is spanned by $\boldsymbol{\theta}^{(0,0)} := (\theta_x, \theta_t)$ and $\boldsymbol{\theta}^{(1,0)} := (\bar{\theta}_x, \theta_t)$ $(\theta_x, \theta_t) \in ([-\pi/2, \pi/2) \times [-\pi, \pi))$, and $\bar{\theta_x} = \theta_x - \text{sign}(\theta_x)\pi$. Let $E^\theta_k = \text{span}[\phi_k(\cdot, \boldsymbol{\theta}^\alpha) : \alpha \in \{(0,0), (1,0)\}$. Assuming all multigrid components are linear and that A^{-1}_{k-1} exists, the Fourier symbol of the two-grid operator TG^{k-1}_k on the space $E^\theta_k \times E^\theta_k$ is given by

ALGORITHM 6 Time-split Collective Gauss–Seidel Method (TS-CGS).

```
Set φ^0 = φ̃
for m = 1, ..., N_t do
    for i in lexicographic order do
        y^1_im = y^0_im − [E'(φ_im)]^{−1} E(φ_im)|_y
        λ^1_{iN_t−m} = λ^0_{iN_t−m} − [E'(φ_im)]^{−1} E(φ_im)|_λ
    end for
end for
```

$$\hat{TG}_k^{k-1}(\boldsymbol{\theta}) = \hat{S}_k(\boldsymbol{\theta})^{m_2}\hat{CG}_k^{k-1}(\boldsymbol{\theta})\hat{S}_k(\boldsymbol{\theta})^{m_1}, \tag{6.8}$$

where m_1 and m_2 are the number of presmoothing and postsmoothing iterations, respectively. Using (6.6) and (6.7), the Fourier symbol of the smoothing operator is given by

$$\hat{S}(\boldsymbol{\theta}) = \mathrm{diag}\{\sigma(\boldsymbol{\theta}^{(0,0)}), \sigma(\boldsymbol{\theta}^{(1,0)}), \sigma(\boldsymbol{\theta}^{(0,0)}), \sigma(\boldsymbol{\theta}^{(1,0)})\},$$

where

$$\sigma(\boldsymbol{\theta}^{(p,q)}) = \frac{\beta\gamma(2\gamma+1)e^{i\theta_x^p}}{\delta t^3 \sum_{k=1}^{N_t} e^{i(k-m)\theta_t^q} + \beta(2\gamma+1)[1+2\gamma-\gamma e^{-i\theta_x^p} - e^{-i\theta_t^q}]}.$$

The smoothing property of the operator S_k is analyzed assuming a perfect coarse-grid correction that removes all of the low-frequency error components and leaves the high-frequency error components unchanged. The smoothing property of S_k is defined by

$$\mu = \max\{r(\hat{P}_k^{k-1}(\boldsymbol{\theta})S_k(\boldsymbol{\theta})) : \boldsymbol{\theta} \in ([-\pi/2, \pi/2) \times [-\pi, \pi))\},$$

where r is the spectral radius and P_k^{k-1} is the projection operator defined on E_k^θ by

$$P_k^{k-1}\phi(\boldsymbol{\theta}, \cdot) = \begin{cases} 0 & \text{if } \boldsymbol{\theta} = \boldsymbol{\theta}^{(0,0)}, \\ \phi(\cdot, \boldsymbol{\theta}) & \text{if } \boldsymbol{\theta} = \boldsymbol{\theta}^{(1,0)}. \end{cases}$$

The Fourier symbol for a full-weighting restriction operator is given by

$$\hat{I}_k^{k-1} = \frac{1}{2}\begin{bmatrix} 1+\cos(\theta_x) & 1-\cos(\theta_x) & 0 & 0 \\ 0 & 0 & 1+\cos(\theta_x) & 1-\cos(\theta_x) \end{bmatrix},$$

and the linear prolongation operator is given by $\hat{I}_{k-1}^k(\boldsymbol{\theta}) = \hat{I}_k^{k-1}(\boldsymbol{\theta})^T$. The symbol of the fine grid operator is

$$\hat{A}_k(\boldsymbol{\theta}) = \begin{bmatrix} a_y(\boldsymbol{\theta}^{(0,0)}) & 0 & -\delta t^2/\beta & 0 \\ 0 & a_y(\boldsymbol{\theta}^{(1,0)}) & 0 & -\delta t^2/\beta \\ \delta t & 0 & a_p(\boldsymbol{\theta}^{(0,0)}) & 0 \\ 0 & \delta t & 0 & a_p(\boldsymbol{\theta}^{(1,0)}) \end{bmatrix},$$

where

$$a_y(\boldsymbol{\theta}^{(p,q)}) = 2\gamma\cos(\theta_x^p) - e^{-i\theta_t^q} - 2\gamma - 1 \text{ and } a_p(\boldsymbol{\theta}^{(p,q)}) = 2\gamma\cos(\theta_x^p) - e^{i\theta_t^q} - 2\gamma - 1,$$

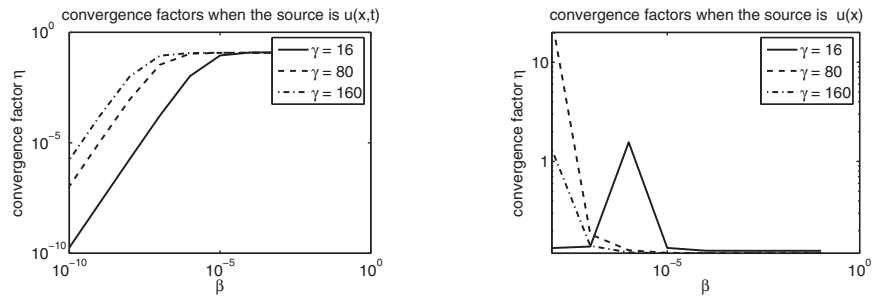

FIG. 9. *Convergence factors for TS-GCS. Convergence factor as a function of β and γ when (left) the source term is u* [9] *and (right) for inverse problem when the source term is u. When the source term is a function of space, convergence factors are greater than* 1 *for a certain range of β and γ.*

and the coarse-grid correction factor is given by

$$\hat{A}_{k-1}(\boldsymbol{\theta}) = \begin{bmatrix} b_y(\boldsymbol{\theta}^{(0,0)}) & -\delta t^2/\beta \\ \delta t & b_p(\boldsymbol{\theta}^{(0,0)}) \end{bmatrix},$$

where

$$b_y(\boldsymbol{\theta}^{(p,q)}) = \gamma\cos(2\theta_x^p)/2 - e^{-i\theta_t^q} - \gamma/2 - 1 \text{ and } b_p(\boldsymbol{\theta}^{(p,q)}) = \gamma\cos(2\theta_x^p)/2 - e^{i\theta_t^q} - \gamma/2 - 1.$$

Using (6.8) for the definition of the two-grid operator we can evaluate the convergence factor by

$$\eta(TG_k^{k-1}) = \sup\{r(\hat{TG}_k^{k-1}(\boldsymbol{\theta})) : \boldsymbol{\theta} \in ([-\pi/2, \pi/2) \times [-\pi, \pi))\}. \tag{6.9}$$

In [9], Fourier mode analysis was carried for a spatiotemporal course time, and the convergence factors were less sensitive to γ and β. In the present problem, this is not the case: for small values of β the method fails to converge (Figure 9).

6.3. Numerical results. In Table 11, we report SQMR iterations using LNKS preconditioner P and the multigrid preconditioner MG with $\tilde{P}$ as a smoother for three values of the regularization parameter. SQMR converges to the required tolerance in constant iterations using P and MG. SQMR with MG preconditioner takes less iterations than with P preconditioner. In P, one exact forward and adjoint solve are performed during each application of P. In MG, only inexact forward and adjoint solves are done at every iteration at different levels of multigrid. One major advantage of solving the problem in full space and using the multigrid preconditioner, is that we avoid any forward or adjoint solves which are inevitable in reduced-space methods. Even in this case, the computational complexity is $\mathcal{O}(N^2)$, as we need to do a KKT matvec at every iteration.

In Table 12, convergence factors and residuals of the multigrid solver using TS-CGS smoother are given. The multigrid solver converges for $\beta = 10^{-2}, 10^{-4}$ and diverges in the case of $\beta = 10^{-6}$. This agrees with the convergence factors estimates obtained from Fourier mode analysis, which show that the multigrid solver using TS-CGS smoother has convergence factors greater than 1 for certain combination of β and γ.

TABLE 11

LNKS preconditioner. The performance of the preconditioned SQMR solver for a constant regularization parameter. Here P represents the number of SQMR iterations with the single-grid version of the LNKS preconditioner; MG corresponds to SQMR iterations using a multigrid LNKS preconditioner with $\tilde{P}$ as smoother. The stopping criterion for SQMR is $\|r\|/\|r_0\| \leq 10^{-8}$ in all cases. Three cases of regularization are considered, $\beta = 10^{-2}, 10^{-4}$, and 10^{-6}, respectively.

$N \times N_t$	β			P		
17 x 8	1e-02	1e-04	1e-06	10	21	36
33 x 16	1e-02	1e-04	1e-06	13	21	51
65 x 32	1e-02	1e-04	1e-06	14	21	54
129 x 64	1e-02	1e-04	1e-06	14	21	56
$N \times N_t$	β			MG		
17 x 8	1e-02	1e-04	1e-06	5	7	12
33 x 16	1e-02	1e-04	1e-06	7	8	16
65 x 32	1e-02	1e-04	1e-06	10	10	17
129 x 64	1e-02	1e-04	1e-06	12	12	17

TABLE 12

TS-CGS results for inverse problem. We observe that the convergence is not sensitive to a decrease in the regularization parameter for a larger regularization parameter. For a smaller regularization parameter, the multigrid solver diverged ($\beta < 10^{-4}$), which agrees with the Fourier mode analysis (see Figure 9).

$N \times N_t$	β	ρ	r_y	r_λ
17 x 8	1e-02	0.091	5e-10	1e-10
33 x 16	1e-02	0.101	3e-10	1e-10
65 x 32	1e-02	0.127	4e-09	5e-10
129 x 64	1e-02	0.130	5e-09	7e-10
17 x 8	1e-04	0.127	5e-08	3e-10
33 x 16	1e-04	0.134	2e-08	1e-10
65 x 32	1e-04	0.130	1e-08	7e-11
129 x 64	1e-04	0.131	2e-08	1e-10

7. Conclusions. In this paper, we presented multigrid algorithms for inverse problems with linear parabolic PDE constraints. Our algorithms are designed for the case in which the inversion variable depends only in space. Although there is prior work on multigrid for optimization problems, there is limited work on algorithms for vanishing regularization parameters. Assuming that we have sufficient information in the data and we need accurate reconstructions, most existing schemes will fail to deliver mesh-independent convergence rates. In this paper, our aim is to construct and analyze schemes that are robust to the vanishing regularization parameter and allow fast high-fidelity reconstructions. Our work is an extension of the work discussed in [26] and [24].

The novel component in our multigrid scheme is the smoother. We use a high-pass filter that allows an iterative solver to work exclusively in the high-frequency regime. A second novel component is the acceleration of the computation by using appropriate inexact versions of the reduced Hessian. By using an exact high-pass filter and a two-step stationary iterative solver as a preconditioner, we were able to analyze the behavior of the algorithm. The overall scheme uses a V-cycle multigrid to accelerate a CG solver that iterates in the reduced space. In addition, we proposed alternative smoothing strategies that use cheaper high-pass filters, and we conducted numerical experiments to examine the effects of the diffusion, and the effect of the coarse-grid operator. The high-frequency projections are preferable but are limited to the cases in which Fourier-type expansions can be carried through fast transforms.

Our numerical experiments gave promising results and justified the extension of our scheme to problems with variable-coefficient PDE constraints. Finally, we combined the reduced space with a full-space solver so that we avoid solving a forward and an adjoint problem at each optimization iteration. We also tested the method using partial measurements, but the 1-D problem may hide difficulties (associated with partial observations) that appear in higher dimensions.

All of the implementations were in `MATLAB`, and no effort was made to optimize the code. We refrained from reporting wall-clock times. We should emphasize, however, that, although the method has optimal complexity, the associated constants can be high. In fact, if the number of the sought frequencies in the reconstructed field is small, then the regularization parameter should be set to a relatively large value. In that case, one can use much cheaper solvers; for example, schemes based on the K-CG and inexact L^2 projections.

We would like to caution the reader that we have committed several "inverse crimes" by choosing attainable observations, the simplest possible regularization, and zero noise (besides the discretization error). These parameters significantly change the quality of the reconstruction and can potentially alter the behavior of the solvers. These topics, however, are beyond the scope of the present paper.

Spectral properties for the reduced Hessian and restriction and prolongation operators can be derived for higher dimensions using a tensorial notation. Algorithms 1, 4, and K-CG can be extended to higher dimensions by using the two-dimensional (2-D) or three-dimensional (3-D) analogue of the intergrid transfer operators presented in 3. Algorithms 2, 5, and 6 can be directly applied for higher dimensions. For Algorithm 3, 2-D and 3-D fast Fourier transforms have to be used to perform spectral filtering. Hence, different algorithms can be extended to higher dimensions though the implementation is not straightforward. We are currently working on parallel implementation in higher dimensions. Further complexity analysis and algorithmic tuning are required to implement an efficient and parallelizable scheme. Most important, an optimal method is highly problem dependent. For example, in the case of sparse partial observations, the full-space method has much higher storage requirements than the reduced-space approach (this is a reason we pursued reduced-space methods). Our method can be used for nonlinear problems, for example, within a Newton multigrid context. Alternatively, nonlinear multigrid methods can be considered.

Acknowledgments. We would like to thank Alfio Borzì and Omar Ghattas for their insightful comments on multigrid and inverse problems.

REFERENCES

[1] V. Akcelik, G. Biros, A. Draganescu, O. Ghattas, J. Hill, and B. V. B. Waanders, *Dynamic data driven inversion for terascale simulations: Real-time identification of airborne contaminants*, in Proceedings of the 2005 ACM/IEEE conference on Supercomputing, Seattle, WA, 2005.

[2] E. Arian and S. Ta'saan, *Multigrid One Shot Methods for Optimal Control Problems: Infinite Dimensional Control*, Technical report NASA-CR-194939, Institute for Computer Applications in Science and Engineering, Hampton, VA, 1994.

[3] M. Arioli and D. Ruiz, *A Chebyshev-based Two-stage Iterative Method as an Alternative to the Direct Solution of Linear Systems*, Technical report RAL-TR-2002-021, Rutherford Appleton Laboratory, Atlas Center, Didcot, Oxfordshire, England, 2002.

[4] O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, London, 1994.

[5] H. T. Banks and K. Kunisch, *Estimation Techniques for Distributed Parameter Systems*, Birkhauser, Boston, MA, 1989.

[6] M Benzi, G. H. Golub, and J. Liesen, *Numerical solution of saddle point problems*, Acta Numer., 14 (2005), p. 1.

[7] G. Biros and O. Ghattas, *Parallel Lagrange–Newton–Krylov–Schur methods for PDE-constrained optimization. Part* I: *The Krylov–Schur solver*, SIAM J. Sci. Comput., 27 (2005), pp. 687–713.

[8] G. Biros and O. Ghattas, *Parallel Lagrange–Newton–Krylov–Schur methods for PDE-constrained optimization. Part* II: *The Lagrange–Newton solver and its application to optimal control of steady viscous flows*, SIAM J. Sci. Comput., 27 (2005), pp. 714–739.

[9] A. Borzì, *Multigrid methods for parabolic distributed optimal control problems*, J. Comput. Appl. Math., 157 (2003), pp. 365–382.

[10] A. Borzì and R. Griesse, *Experiences with a space-time multigrid method for the optimal control of a chemical turbulence model*, Internat. J. Numer. Methods Fluids, 47 (2005), pp. 879–885.

[11] A. Borzì and K. Kunisch, *The numerical solution of the steady state solid fuel ignition model and its optimal control*, SIAM J. Sci. Comput., 22 (2000), pp. 263–284.

[12] A. Brandt, *Multi-level adaptive solutions to boundary-value problems*, Math. Comp., 31 (1977), pp. 333–390.

[13] W. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, SIAM, Philadelphia, PA, 2000.

[14] D. Colton and R. Kress, *Inverse Acoustic and Electromagnetic Scattering Theory*, 2nd ed., Appl. Math. Sci., Springer, New York, 1998.

[15] T. Dreyer, B. Maar, and V. Schulz, *Multigrid optimization and applications*, J. Comput. Appl. Math., 120 (2000), pp. 67–84.

[16] S. P. Frankel, *Convergence rates of iterative treatments of PDEs*, Math. Tables Aids Comput., 4 (1950), pp. 65–75.

[17] R. W. Freund and N. M. Nachtigal, *An implementation of the QMR method based on coupled two-term recurrences*, SIAM J. Sci. Comput., 15 (1994), pp. 313–337.

[18] L. Giraud, D. Ruiz, and A. Touhami, *A comparative study of iterative solvers exploiting spectral information for SPD systems*, SIAM J. Sci. Comput., 27 (2006), pp. 1760–1786.

[19] W. Hackbusch, *Multigrid Methods and Applications*, Springer Ser. Comput. Math., Springer, Berlin, 1985.

[20] M. Hanke and C. R. Vogel, *Two-level preconditioners for regularized inverse problems* I. *Theory*, Numer. Math., 83 (1999), pp. 385–402.

[21] M. Heinkenschloss, *A time-domain decomposition iterative method for the solution of distributed linear quadratic optimal control problems*, J. Comput. Appl. Math., 173 (2005), pp. 169–198.

[22] G. Horton and S. Vandewalle, *A space-time multigrid method for parabolic partial differential equations*, SIAM J. Sci. Comput., 16 (1995), pp. 848–864.

[23] E. Kaasschieter, *Preconditioned conjugate gradient for solving singular systems*, J. Comput. Appl. Math., 24 (1988), pp. 265–275.

[24] B. Kaltenbacher, *On the regularizing properties of a full multigrid method for ill-posed problems*, Inverse Problems, 17 (2001), pp. 767–788.

[25] B. Kaltenbacher, *V-cycle convergence of some multigrid methods for ill-posed problems*, Math. Comp., 72 (2003), pp. 1711–1730.

[26] J. T. King, *On the construction of preconditioners by subspace decomposition*, J. Comput. Appl. Math., 29 (1990), pp. 195–205.

[27] J. T. King, *Multilevel algorithms for ill-posed problems*, Numer. Math., 61 (1992), pp. 311–334.

[28] J. R. Shewchuk, *An introduction to the conjugate gradient method without agonizing pain*, CMU-CS-94-125, Technical report, 1994.

[29] U. Trottenberg, C. Oosterlee, and A. Schuller, *Chapter* 7, *Multigrid*, Academic Press, New York, 2001.

[30] S. Vandewalle and R. Piessens, *Efficient parallel algorithms for solving initial-boundary value problems and time-periodic parabolic Partial Differential Equations*, SIAM J. Sci. Statist. Comput., 13 (1992), pp. 1330–1346.