Eidgenössische Technische Hochschule Zürich

Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Adaptive stochastic Galerkin methods: beyond the elliptic case

C.J. Gittelson

Research Report No. 2011-12
February 2011

Seminar für Angewandte Mathematik
Eidgenössische Technische Hochschule
CH-8092 Zürich
Switzerland

Adaptive Stochastic Galerkin Methods: Beyond the Elliptic Case

Claude Jeffrey Gittelson*

February 28, 2011

Abstract

We derive an adaptive solver for random boundary value problems, building on adaptive wavelet algorithms for non-elliptic problems. Replacing wavelets by polynomials of the random parameters leads to a modular solver for the parameter dependence, which may be combined with any discretization on the spatial domain. We show optimality properties of this solver, and present numerical computations, including a comparison of various adaptive methods.

Introduction

Stochastic Galerkin methods approximate the solution of a boundary value problem depending on random data by a Galerkin projection onto a finite dimensional space of random fields. This requires the solution of a single coupled system of deterministic equations for the coefficients of the Galerkin projection with respect to a predefined set of basis functions on the parameter domain, see [DBO01, XK02, BTZ04, WK05, MK05, FST05, WK06, TS07, BS09, BAS10].

The main difficulty in applying these methods is the construction of suitable spaces in which to compute approximate solutions. In [Git11b], we suggest an adaptive method for random elliptic boundary value problems, using techniques from the adaptive wavelet methods [CDD01, GHS07, DSS09].

We extend this approach to more general equations, building on ideas from the adaptive wavelet method [CDD02]. In place of wavelets, we use an orthonormal polynomial basis on the parameter domain. The coefficients of the random solution with respect to this basis are deterministic functions on the physical domain. An arbitrary discretization can be used to approximate these; our method can be combined with any solver for the deterministic counterpart of the random boundary value problem.

The resulting algorithm is quite different from that in [Git11b]. Instead of computing Galerkin projections on a sequence of subspaces, an iterative method is applied

*Research supported in part by the Swiss National Science Foundation grant No. 200021-120290/1.

directly to the full equation. Individual substeps of this iteration, such as applications of the stochastic operator, are replaced by approximate counterparts, realized by suitable adaptive algorithms. These keep track of errors entering the computation, ensuring convergence of the algorithm, and providing an upper bound on the error of the approximate solution.

In Section 1, we present three random boundary value problems that can be solved by our method: an elliptic equation, a saddle point problem, and a parabolic equation. An abstract equation is introduced in Section 2, for which we derive a weak formulation. We construct an orthonormal polynomial basis, and recast the equation as a bi-infinite operator matrix equation for the coefficients of the solution with respect to this basis.

We present the first version of our adaptive solver in Section 3, which is essentially a perturbed block Jacobi method. For symmetric probability distributions, we suggest a method for reducing the computational cost while maintaining the same accuracy.

In Section 4, we add a coarsening step the the initial solver. This prevents unimportant coefficients from accumulating in the approximate solution. We show an optimality property for this method.

Finally, in Section 5, we apply these adaptive solvers to a simple model problem. Numerical computations demonstrate the convergence of the algorithms, and compare them to the adaptive method from [Git11b] and the sparse tensor product construction in [BAS10]. We discuss the empirical convergence behavior in the light of the theoretical approximation results in [CDS10b, CDS10a].

1 Random Boundary Value Problems

1.1 The Isotropic Diffusion Equation

As an illustrative example, we consider the isotropic diffusion equation on a bounded Lipschitz domain $G \subset \mathbb{R}^d$ with homogeneous Dirichlet boundary conditions. For any uniformly positive $a \in L^\infty(G)$ and any $f \in L^2(G)$, we have

$$\begin{aligned} -\nabla \cdot (a(x)\nabla u(x)) &= f(x) \,, \quad x \in G \,, \\ u(x) &= 0 \,, \quad x \in \partial G \,. \end{aligned} \tag{1.1}$$

We view f as fixed, but allow a to vary.

We model the permeability a as a $L^\infty(G)$-valued random variable $\tilde{a}$ on a probability space $(\Omega, \mathcal{F}, P)$, which we assume to be uniformly bounded from above and away from 0,

$$0 < \check{a} \leq \tilde{a}(\omega, x) \leq \hat{a} < \infty \qquad \forall x \in G \,, \quad \forall \omega \in \Omega \,. \tag{1.2}$$

Let $\bar{a} \in L^\infty(G)$ be some uniformly positive deterministic approximation of $\tilde{a}$. For example, $\bar{a}$ can be the mean field

$$\bar{a} \colon G \to \mathbb{R} \,, \quad \bar{a}(x) \coloneqq \int_\Omega \tilde{a}(\omega, x) \, \mathrm{d}P(\omega) \,, \tag{1.3}$$

or simply a constant $\bar{a} := (\hat{a} + \check{a})/2$, $\bar{a} := \sqrt{\hat{a}\check{a}}$, or $\bar{a} := 1$.

Let $(\varphi_m)_{m=1}^\infty$ be a frame of $L^2(G)$ with dual frame $(\varphi_m^*)_{m=1}^\infty$, which we interpret also as a sequence in $L^2(G)$. Define the random variables

$$Y_m(\omega) := \frac{1}{\alpha_m} \int_G (\tilde{a}(\omega, x) - \bar{a}(x))\, \varphi_m^*(x)\, \mathrm{d}x \;, \quad m \in \mathbb{N} \;. \tag{1.4}$$

Note that Y_m is bounded due to Hölder's inequality and (1.2). We assume that α_m is chosen such that $Y_m(\Omega) \subset [-1, 1]$ for all $m \in \mathbb{N}$. For example, this holds for

$$\alpha_m := \sup_{\omega \in \Omega} \|\tilde{a}(\omega) - \bar{a}\|_{L^\infty(G)} \left\|\varphi_m^*\right\|_{L^1(G)} \;, \quad m \in \mathbb{N} \;. \tag{1.5}$$

Abbreviating $a_m := \alpha_m \varphi_m$, we have

$$\tilde{a}(\omega, x) = \bar{a}(x) + \sum_{m=1}^\infty Y_m(\omega) a_m(x) \tag{1.6}$$

for all $\omega \in \Omega$ with convergence in $L^2(G)$. Let $\Gamma := [-1, 1]^\infty$ and

$$a(y, x) := \bar{a}(x) + \sum_{m=1}^\infty y_m a_m(x) \;, \quad y = (y_m)_{m=1}^\infty \in \Gamma \;. \tag{1.7}$$

Then $\tilde{a}(\omega, x) = a(Y(\omega), x)$ for all $\omega \in \Omega$, where $Y := (Y_m)_{m=1}^\infty$.

We define the parametric operator

$$A(y) \colon H_0^1(G) \to H^{-1}(G) \;, \quad v \mapsto -\nabla \cdot (a(y) \nabla v) \;, \quad y \in \Gamma \;. \tag{1.8}$$

This leads to the parametric operator equation

$$A(y) u(y) = f \qquad \forall y \in \Gamma \;. \tag{1.9}$$

The solution of the random diffusion equation is $u(Y(\omega))$.

Due to linearity, we can expand $A(y)$ as

$$A(y) = D + R(y) \;, \quad R(y) := \sum_{m=1}^\infty y_m R_m \qquad \forall y \in \Gamma \;, \tag{1.10}$$

for

$$\begin{aligned} D &:= A_0(\bar{a}) \colon H_0^1(G) \to H^{-1}(G) \;, \quad v \mapsto -\nabla \cdot (\bar{a} \nabla v) \;, \\ R_m &:= A_0(a_m) \colon H_0^1(G) \to H^{-1}(G) \;, \quad v \mapsto -\nabla \cdot (a_m \nabla v) \;, \quad m \in \mathbb{N} \;. \end{aligned}$$

Note that $\|R_m\|_{H_0^1(G) \to H^{-1}(G)} \leq \|a_m\|_{L^\infty(G)}$, and thus convergence in (1.10) and (1.7) is assured if $(\|a_m\|_{L^\infty(G)})_{m=1}^\infty$ is summable.

1.2 The Stokes Equation

Other boundary value problems with random coefficients have a similar structure as the isotropic diffusion equation discussed above. We consider the Stokes equation

$$\begin{aligned} -\nabla \cdot (a(x)\nabla \boldsymbol{u}(x)) - \nabla p(x) &= \boldsymbol{f}(x) \,, \quad x \in G \,, \\ \nabla \cdot \boldsymbol{u}(x) &= 0 \,, \quad x \in G \,, \\ \boldsymbol{u}(x) &= 0 \,, \quad x \in \partial G \,, \end{aligned} \tag{1.11}$$

with a random kinematic viscosity $a(x)$, which we expand analogously to Section 1.1 as

$$a(y,x) \coloneqq \bar{a}(x) + \sum_{m=1}^{\infty} y_m a_m(x) \,, \quad y = (y_m)_{m=1}^{\infty} \in \Gamma = [-1,1]^{\infty} \,. \tag{1.12}$$

We define the parametric operator

$$A(y) \colon (H_0^1(G))^d \times L_*^2(G) \to (H^{-1}(G))^d \times L_*^2(G) \,, \quad (\boldsymbol{v}, q) \mapsto (-\nabla \cdot (a(y)\nabla \boldsymbol{v}) - \nabla q, \nabla \cdot \boldsymbol{v}) \tag{1.13}$$

for $y \in \Gamma$. Then the Stokes equation with viscosity (1.12) is

$$A(y)(\boldsymbol{u}(y), q) = (\boldsymbol{f}, 0) \qquad \forall y \in \Gamma \,. \tag{1.14}$$

The operator $A(y)$ can be expanded as in (1.10) for the deterministic operators

$$\begin{aligned} D &\colon (H_0^1(G))^d \times L_*^2(G) \to (H^{-1}(G))^d \times L_*^2(G) \,, \quad (\boldsymbol{v}, q) \mapsto (-\nabla \cdot (\bar{a}\nabla \boldsymbol{v}) - \nabla q, \nabla \cdot \boldsymbol{v}) \,, \\ R_m &\colon (H_0^1(G))^d \times L_*^2(G) \to (H^{-1}(G))^d \times L_*^2(G) \,, \quad (\boldsymbol{v}, q) \mapsto (-\nabla \cdot (a_m \nabla \boldsymbol{v}), 0) \,, \quad m \in \mathbb{N} \,. \end{aligned}$$

Note that D is boundedly invertible, assuming that $\bar{a}$ is bounded and uniformly positive, and R_m is bounded with $\|R_m\| \leq \|a_m\|_{L^\infty(G)}$.

1.3 The Heat Equation

As a third example, we consider a parabolic version of the isotropic diffusion equation,

$$\begin{aligned} \partial_t u(t,x) - \nabla \cdot (a(x)\nabla u(x)) &= f(x) \,, \quad (t,x) \in [0,T] \times G \,, \\ u(t,x) &= 0 \,, \quad (t,x) \in [0,T] \times \partial G \,, \\ u(0,x) &= u_0(x) \,, \quad x \in G \,, \end{aligned} \tag{1.15}$$

with a random diffusion coefficient of the form

$$a(y,t,x) \coloneqq \bar{a}(t,x) + \sum_{m=1}^{\infty} y_m a_m(t,x) \,, \quad y = (y_m)_{m=1}^{\infty} \in \Gamma = [-1,1]^{\infty} \,. \tag{1.16}$$

Let $I \coloneqq [0,T]$ and

$$\begin{aligned} V &\coloneqq L^2\Big(I; H_0^1(G)\Big) \cap H^1\Big(I; H^{-1}(G)\Big) \,, \\ W &\coloneqq L^2\Big(I; H_0^1(G)\Big) \times L^2(G) \,. \end{aligned}$$

We define the parametric operator

$$A(y)\colon V \to W^* , \quad v \mapsto (\partial_t v - \nabla \cdot (a(y)\nabla v), v(0)) , \quad y \in \Gamma . \tag{1.17}$$

Then the heat equation (1.15) can be recast as

$$A(y)u(y) = (f, u_0) \qquad \forall y \in \Gamma . \tag{1.18}$$

As in the previous examples, $A(y)$ can be expanded as (1.10) with deterministic operators

$$\begin{aligned} D\colon V \to W^* , &\quad v \mapsto (\partial_t v - \nabla \cdot (\bar{a}\nabla v), v(0)) , \\ R_m\colon V \to W^* , &\quad v \mapsto (-\nabla \cdot (a_m \nabla v), 0) , \quad m \in \mathbb{N} . \end{aligned}$$

Again, D is boundedly invertible by [SS09, Theorem 5.1] if $\bar{a}$ is bounded and uniformly positive, and the operators R_m satisfy $\|R_m\|_{V\to W^*} \le \|a_m\|_{L^\infty(I\times G)}$.

2 Abstract Setting

2.1 Pathwise Definitions

Let $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ and let V and W be separable Hilbert spaces over $\mathbb{K}$. We denote by W^* the space of all continuous antilinear functionals on W. Furthermore, $\mathcal{L}(V, W^*)$ is the Banach space of bounded linear maps from V to W^*.

We consider operator equations depending on a parameter in $\Gamma = [-1, 1]^\infty$. Given

$$A\colon \Gamma \to \mathcal{L}(V, W^*) \qquad \text{and} \qquad f\colon \Gamma \to W^* , \tag{2.1}$$

we wish to determine

$$u\colon \Gamma \to V , \qquad A(y)u(y) = f(y) \quad \forall y \in \Gamma . \tag{2.2}$$

Motivated by (1.10), we consider operators of the form

$$A(y) = D + R(y) \qquad \forall y \in \Gamma , \tag{2.3}$$

where $D \in \mathcal{L}(V, W^*)$ and

$$R(y) = \sum_{m=1}^{\infty} y_m R_m \qquad \forall y = (y_m)_{m=1}^\infty \in \Gamma , \tag{2.4}$$

as in *e.g.* [BAS10, BS09, CDS10b, CDS10a, TS07]. Here, each R_m is in $\mathcal{L}(V, W^*)$. We assume $(R_m)_m \in \ell^1(\mathbb{N}; \mathcal{L}(V, W^*))$, and we assume without loss of generality that the sequence $(\|R_m\|_{V\to W^*})_{m=1}^\infty$ is nonincreasing.

A simple Neumann series argument shows existence and uniqueness of $u(y)$ for all $y \in \Gamma$. This uses the assumption that D is boundedly invertible, and

$$\left\|D^{-1}R(y)\right\|_{V\to V} \le \gamma < 1 \qquad \forall y \in \Gamma . \tag{2.5}$$

Then $A(y)$ can be decomposed as

$$A(y) = D(\mathrm{id}_V + D^{-1}R(y)) \,, \quad y \in \Gamma \,, \tag{2.6}$$

and consequently, using a Neumann series in $\mathcal{L}(V)$ to invert the second factor,

$$A(y)^{-1} = \left(\sum_{n=0}^{\infty} (-D^{-1}R(y))^n \right) D^{-1} \,, \quad y \in \Gamma \,. \tag{2.7}$$

In particular, (2.5) implies

$$\left\| A(y) \right\|_{V \to W^*} \leq \|D\|_{V \to W^*} (1 + \gamma) \qquad \forall y \in \Gamma \,, \tag{2.8}$$

$$\left\| A(y)^{-1} \right\|_{W^* \to V} \leq \frac{1}{1-\gamma} \left\| D^{-1} \right\|_{W^* \to V} \qquad \forall y \in \Gamma \,. \tag{2.9}$$

2.2 Weak Formulation

Let π be a probability measure on the parameter domain Γ with Borel σ-algebra $\mathcal{B}(\Gamma)$. In the examples from Section 1, π could be the image of the physical probability P under the map Y, or it may be any other probability measure. We derive a weak formulation of (2.2) by integrating over Γ with respect to π.

Let the map $\Gamma \ni y \mapsto A(y)v(y)$ be measurable for any measurable $v \colon \Gamma \to V$. Then due to (2.8),

$$\mathcal{A} \colon L^2_\pi(\Gamma; V) \to L^2_\pi(\Gamma; W^*) \,, \quad v \mapsto [y \mapsto A(y)v(y)] \,, \tag{2.10}$$

is well-defined and continuous with norm at most $(1+\gamma)\|D\|_{V \to W^*}$. We assume also that $f \in L^2_\pi(\Gamma; W^*)$.

We define the multiplication operators

$$K_m \colon L^2_\pi(\Gamma) \to L^2_\pi(\Gamma) \,, \quad v(y) \mapsto y_m v(y) \,, \quad m \in \mathbb{N} \,. \tag{2.11}$$

Since y_m is real and $|y_m|$ is less than one, K_m is symmetric and has norm at most one.

By separability of V, the Lebesgue–Bochner space $L^2_\pi(\Gamma; V)$ is isometrically isomorphic to the Hilbert tensor product $L^2_\pi(\Gamma) \otimes V$, and similarly for W^* in place of V. Using these identifications, we expand $\mathcal{A}$ as $\mathcal{A} = \mathcal{D} + \mathcal{R}$ with

$$\mathcal{D} \coloneqq \mathrm{id}_{L^2_\pi(\Gamma)} \otimes D \qquad \text{and} \qquad \mathcal{R} \coloneqq \sum_{m=1}^{\infty} K_m \otimes R_m \,. \tag{2.12}$$

This sum converges in $\mathcal{L}(L^2_\pi(\Gamma; V), L^2_\pi(\Gamma; W^*))$ by the assumption $(R_m)_m \in \ell^1(\mathbb{N}; \mathcal{L}(V, W^*))$.

Lemma 2.1. $\left\| \mathcal{D}^{-1}\mathcal{R} \right\|_{L^2_\pi(\Gamma;V) \to L^2_\pi(\Gamma;V)} \leq \gamma < 1$.

Proof. We note that $\mathcal{D}$ is invertible with $\mathcal{D}^{-1} = \mathrm{id}_{L^2_\pi(\Gamma)} \otimes D^{-1}$, and as in (2.10), $(\mathcal{D}^{-1}\mathcal{R}v)(y) = D^{-1}R(y)v(y)$ for all $v \in L^2_\pi(\Gamma; V)$ and $y \in \Gamma$. Therefore, using (2.5), for all $v \in L^2_\pi(\Gamma; V)$,

$$\left\| \mathcal{D}^{-1}\mathcal{R}v \right\|^2_{L^2_\pi(\Gamma;V)} = \int_\Gamma \left\| D^{-1}R(y)v(y) \right\|^2_V \mathrm{d}\pi(y) \leq \int_\Gamma \gamma^2 \left\| v(y) \right\|^2_V \mathrm{d}\pi(y) \leq \gamma^2 \|v\|^2_{L^2_\pi(\Gamma;V)} \,. \quad \square$$

Proposition 2.2. *The operator $\mathcal{A}$ from* (2.10) *is boundedly invertible,* $(\mathcal{A}^{-1}g)(y) = A(y)^{-1}g(y)$ *for any* $g \in L^2_\pi(\Gamma; W^*)$*, and*

$$\left\|\mathcal{A}^{-1}\right\|_{L^2_\pi(\Gamma;W^*)\to L^2_\pi(\Gamma;V)} \le \frac{1}{1-\gamma}\left\|D^{-1}\right\|_{W^*\to V} . \tag{2.13}$$

Proof. As in (2.6), we have

$$\mathcal{A} = \mathcal{D}(\mathrm{id}_{L^2_\pi(\Gamma;V)} + \mathcal{D}^{-1}\mathcal{R}) .$$

Therefore, by a Neumann series argument using Lemma 2.1, $\mathcal{A}$ is invertible, and $\mathcal{A}^{-1}$ can be represented as

$$\mathcal{A}^{-1} = \left(\sum_{n=0}^{\infty}(-\mathcal{D}^{-1}\mathcal{R})^n\right)\mathcal{D}^{-1} .$$

Since $(\mathcal{D}^{-1}g)(y) = D^{-1}g(y)$ by definition, and $(\mathcal{D}^{-1}\mathcal{R}v)(y) = D^{-1}R(y)v(y)$, this is just the Neumann series representation of $A(y)^{-1}$ from (2.7). The estimate (2.13) follows from Lemma 2.1. □

Corollary 2.3. *The solution u of* (2.2) *is in $L^2_\pi(\Gamma; V)$, and u is the unique element of this space satisfying*

$$\mathcal{A}u = f . \tag{2.14}$$

The operator equation (2.14) in $L^2_\pi(\Gamma; W^*)$ can be reformulated as

$$\int_\Gamma \langle A(y)u(y), w(y)\rangle \,\mathrm{d}\pi(y) = \int_\Gamma \langle f(y), w(y)\rangle \,\mathrm{d}\pi(y) \qquad \forall w \in L^2_\pi(\Gamma; W) . \tag{2.15}$$

2.3 Orthonormal Polynomial Basis

In order to construct an orthonormal polynomial basis of $L^2_\pi(\Gamma)$, we assume that π is a product measure. Let

$$\pi = \bigotimes_{m=1}^{\infty} \pi_m \tag{2.16}$$

for probability measures π_m on $([-1,1], \mathcal{B}([-1,1]))$; see *e.g.* [Bau02, Section 9] for a general construction of infinite products of probability measures. We assume for simplicity that the support of π_m in $[-1,1]$ has infinite cardinality.

For all $m \in \mathbb{N}$, let $(P^m_n)_{n=0}^\infty$ be an orthonormal polynomial basis of $L^2_{\pi_m}([-1,1])$, with $\deg P^m_n = n$. Such a basis is given by the three term recursion $P^m_{-1} := 0$, $P^m_0 := 1$ and

$$\beta^m_n P^m_n(\xi) := (\xi - \alpha^m_{n-1})P^m_{n-1}(\xi) - \beta^m_{n-1}P^m_{n-2}(\xi) , \quad n \in \mathbb{N} , \tag{2.17}$$

with

$$\alpha^m_n := \int_{-1}^{1} \xi P^m_n(\xi)^2 \,\mathrm{d}\pi_m(\xi) \qquad \text{and} \qquad \beta^m_n := \frac{c^m_{n-1}}{c^m_n} , \tag{2.18}$$

where c^m_n is the leading coefficient of P^m_n, $\beta^m_0 := 1$, and P^m_n is chosen as normalized in $L^2_{\pi_m}([0,1])$. This basis is unique *e.g.* if c^m_n is chosen to be positive.

We define the set of finitely supported sequences in $\mathbb{N}_0$ as

$$\Lambda := \left\{ \nu \in \mathbb{N}_0^{\mathbb{N}} \,;\, \#\operatorname{supp} \nu < \infty \right\} , \tag{2.19}$$

where the support is defined by

$$\operatorname{supp} \nu := \{ m \in \mathbb{N} \,;\, \nu_m \neq 0 \} \quad , \quad \nu \in \mathbb{N}_0^{\mathbb{N}} . \tag{2.20}$$

Then countably infinite tensor product polynomials are given by

$$\boldsymbol{P} := (P_\nu)_{\nu \in \Lambda} \quad , \quad P_\nu := \bigotimes_{m=1}^{\infty} P_{\nu_m}^m \quad , \quad \nu \in \Lambda . \tag{2.21}$$

Note that each of these functions depends on only finitely many dimensions,

$$P_\nu(y) = \prod_{m=1}^{\infty} P_{\nu_m}^m(y_m) = \prod_{m \in \operatorname{supp} \nu} P_{\nu_m}^m(y_m) \, , \quad \nu \in \Lambda \, , \tag{2.22}$$

since $P_0^m = 1$ for all $m \in \mathbb{N}$.

By *e.g.* [Git11c, Theorem 2.8], $\boldsymbol{P}$ is an orthonormal basis of $L^2_\pi(\Gamma)$. By Parseval's identity, this is equivalent to the statement that the map

$$T \colon \ell^2(\Lambda) \to L^2_\pi(\Gamma) \quad , \quad (c_\nu)_{\nu \in \Lambda} \mapsto \sum_{\nu \in \Lambda} c_\nu P_\nu \, , \tag{2.23}$$

is a unitary isomorphism. The inverse of T is

$$T^{-1} = T^* \colon L^2_\pi(\Gamma) \to \ell^2(\Lambda) \quad , \quad g \mapsto \left(\int_\Gamma g(y) \overline{P_\nu(y)} \, \mathrm{d}\pi(y) \right)_{\nu \in \Lambda} . \tag{2.24}$$

2.4 Bi-Infinite Operator Matrix Equation

We use the isomorphism T from (2.23) to recast the weak stochastic operator equation (2.14) as an equivalent discrete operator equation. Since T is a unitary map from $\ell^2(\Lambda)$ to $L^2_\pi(\Gamma)$, the tensor product operator $T \otimes \mathrm{id}_V$ is an isometric isomorphism from $\ell^2(\Lambda; V)$ to $L^2_\pi(\Gamma; V)$. By definition, $v \in L^2_\pi(\Gamma; V)$ and $\boldsymbol{v} = (v_\mu)_{\mu \in \Lambda} \in \ell^2(\Lambda; V)$ are related by $v = T \otimes \mathrm{id}_V \, \boldsymbol{v}$ if

$$v(y) = \sum_{\mu \in \Lambda} v_\mu P_\mu(y) \qquad \text{or} \qquad v_\mu = \int_\Gamma v(y) \overline{P_\mu(y)} \, \mathrm{d}\pi(y) \quad \forall \mu \in \Lambda \, , \tag{2.25}$$

and either of these properties implies the other. The series in (2.25) converges unconditionally in $L^2_\pi(\Gamma; V)$, and the integral can be interpreted as a Bochner integral in V. We abbreviate $T_V := T \otimes \mathrm{id}_V$ and $T_W := T \otimes \mathrm{id}_W$.

Let $\boldsymbol{A} := T_W^* \mathcal{A} T_V$ and $\boldsymbol{f} := T_W^* f$. Then $u = T_V \boldsymbol{u}$ for $\boldsymbol{u} \in \ell^2(\Lambda; V)$ with

$$\boldsymbol{A}\boldsymbol{u} = \boldsymbol{f} \tag{2.26}$$

due to (2.14).

By definition, $\boldsymbol{A}$ is a boundedly invertible linear map from $\ell^2(\Lambda; V)$ to $\ell^2(\Lambda; W^*)$. It can be interpreted as a bi-infinite operator matrix

$$\boldsymbol{A} = [A_{\nu\mu}]_{\nu,\mu\in\Lambda} , \quad A_{\nu\mu} \colon V \to W^* , \tag{2.27}$$

with entries

$$\begin{aligned} A_{\mu\mu} &= D + \sum_{m=1}^{\infty} \alpha^m_{\mu_m} R_m , \quad \mu \in \Lambda , \\ A_{\nu\mu} &= \beta^m_{\max(\nu_m,\mu_m)} R_m , \quad \nu, \mu \in \Lambda , \quad \nu - \mu = \pm\epsilon_m , \end{aligned} \tag{2.28}$$

and $A_{\nu\mu} = 0$ otherwise, where ϵ_m denotes the Kronecker sequence with $(\epsilon_m)_n = \delta_{mn}$. If π_m is a symmetric measure on $[-1, 1]$ for all $m \in \mathbb{N}$, then $\alpha^m_n = 0$ for all m and n, and thus $A_{\nu\nu} = D$. Similarly, the operator $\boldsymbol{R} \coloneqq T^*_W \mathcal{R} T_V$ can be interpreted as a bi-infinite operator matrix $\boldsymbol{R} = [R_{\nu\mu}]$ with $R_{\mu\mu} = A_{\mu\mu} - D$ and $R_{\nu\mu} = A_{\nu\mu}$ for $\nu \neq \mu$. The operator $\boldsymbol{D} \coloneqq T^*_W \mathcal{D} T_V$ is just the diagonal operator matrix with D on the diagonal. We refer to [Git11c, Git11a] for details.

Let $\boldsymbol{K}_m = T^* K_m T \in \mathcal{L}(\ell^2(\Lambda))$. Due to the three term recursion (2.17),

$$(\boldsymbol{K}_m \boldsymbol{c})_\mu = \beta^m_{\mu_m+1} c_{\mu+\epsilon_m} + \alpha^m_{\mu_m} c_\mu + \beta^m_{\mu_m} c_{\mu-\epsilon_m} , \quad \mu \in \Lambda , \tag{2.29}$$

for $\boldsymbol{c} = (c_\mu)_{\mu\in\Lambda} \in \ell^2(\Lambda)$, where $c_\mu \coloneqq 0$ if $\mu_m < 0$ for any $m \in \mathbb{N}$. Furthermore, $\boldsymbol{K}^*_m = \boldsymbol{K}_m$ and $\|\boldsymbol{K}_m\|_{\ell^2(\Lambda)\to\ell^2(\Lambda)} \leq 1$. Using the maps $\boldsymbol{K}_m$, $\boldsymbol{R}$ can be written succinctly as

$$\boldsymbol{R} = \sum_{m=1}^{\infty} \boldsymbol{K}_m \otimes R_m , \tag{2.30}$$

with unconditional convergence in $\mathcal{L}(\ell^2(\Lambda; V), \ell^2(\Lambda; W^*))$.

3 A General Iterative Procedure

3.1 Adaptive Application of the Stochastic Operator

Following [Git11b], we construct a sequence of approximations of $\boldsymbol{R}$ by truncating the series (2.30). For all $M \in \mathbb{N}$, let

$$\boldsymbol{R}_{[M]} \coloneqq \sum_{m=1}^{M} \boldsymbol{K}_m \otimes R_m , \tag{3.1}$$

and $\boldsymbol{R}_{[0]} \coloneqq 0$. For all $M \in \mathbb{N}$, let $\bar{e}_{\boldsymbol{R},M}$ be given such that

$$\left\| \boldsymbol{R} - \boldsymbol{R}_{[M]} \right\|_{\ell^2(\Lambda;V)\to\ell^2(\Lambda;W^*)} \leq \bar{e}_{\boldsymbol{R},M} . \tag{3.2}$$

For example, these bounds can be chosen as

$$\bar{e}_{\boldsymbol{R},M} \coloneqq \sum_{m=M+1}^{\infty} \|R_m\|_{V\to W^*} \ . \tag{3.3}$$

We assume that $(\bar{e}_{\boldsymbol{R},M})_{M=0}^{\infty}$ is nonincreasing and converges to 0, and also that the sequence of differences $(\bar{e}_{\boldsymbol{R},M} - \bar{e}_{\boldsymbol{R},M+1})_{M=0}^{\infty}$ is nonincreasing.

We consider the approximation of a vector $\boldsymbol{w} \in \ell^2(\Lambda)$ by a sum $\boldsymbol{w}_{[1]} + \cdots + \boldsymbol{w}_{[P]}$ for sections $\boldsymbol{w}_{[p]} \coloneqq \boldsymbol{w}|_{\Lambda_p}$, $p = 1, \dots, P$, with mutually disjoint sets $\Lambda_p \subset \Lambda$. The section $\boldsymbol{w}_{[1]}$ can be thought of as containing the largest elements of $\boldsymbol{w}$, $\boldsymbol{w}_{[2]}$ the next largest, and so on.

Such a partitioning can be constructed by the approximate sorting algorithm

$$\texttt{BucketSort}[\boldsymbol{w}, \epsilon] \mapsto \left[(\boldsymbol{w}_{[p]})_{p=1}^{P}, (\Lambda_p)_{p=1}^{P}\right] \ , \tag{3.4}$$

which, given a finitely supported $\boldsymbol{w} \in \ell^2(\Lambda)$ and a threshold $\epsilon > 0$, returns index sets

$$\Lambda_p \coloneqq \left\{\mu \in \Lambda \ ; \ \left|v_\mu\right| \in (2^{-p/2}\,\|\boldsymbol{w}\|_{\ell^\infty} \,, 2^{-(p-1)/2}\,\|\boldsymbol{w}\|_{\ell^\infty}]\right\} \tag{3.5}$$

and $\boldsymbol{w}_{[p]} \coloneqq \boldsymbol{w}|_{\Lambda_p}$, see [Met02, Bar05, GHS07, DSS09]. The integer P is minimal with

$$2^{-P/2}\,\|\boldsymbol{w}\|_{\ell^\infty(\Lambda)}\,\sqrt{\#\operatorname{supp}\boldsymbol{w}} \le \epsilon \ . \tag{3.6}$$

By [GHS07, Rem. 2.3] or [DSS09, Prop. 4.4], the number of operations and storage locations required by a call of $\texttt{BucketSort}[\boldsymbol{w}, \epsilon]$ is bounded by

$$\#\operatorname{supp}\boldsymbol{w} + \max(1, \lceil \log(\|\boldsymbol{w}\|_{\ell^\infty(\Lambda)}\,\sqrt{\#\operatorname{supp}\boldsymbol{w}}/\epsilon)\rceil) \ , \tag{3.7}$$

which is faster than exact comparison-based sorting algorithms.

The routine $\texttt{Apply}_{\boldsymbol{R}}[\boldsymbol{v}, \epsilon]$ adaptively approximates $\boldsymbol{R}\boldsymbol{v}$ in three distict steps. First, the elements of $\boldsymbol{v}$ are grouped according to their norm. Elements smaller than a certain tolerance are discarded. This truncation of the vector $\boldsymbol{v}$ induces an error of at most $\delta \le \epsilon/2$.

Next, a greedy algorithm is used to assign to each segment $\boldsymbol{v}_{[p]}$ of $\boldsymbol{v}$ an approximation $\boldsymbol{R}_{[M_p]}$ of $\boldsymbol{R}$. Starting with $\boldsymbol{R}_{[M_p]} = \boldsymbol{0}$ for all $p = 1, \dots, \ell$, these approximations are refined iteratively until an estimate of the error is smaller than $\epsilon - \delta$.

Finally, the operations determined by the previous two steps are performed. Each multiplication $R_m v_\mu$ is performed just once, and copied to the appropriate entries of $\boldsymbol{z}$.

Proposition 3.1. *For any finitely supported $\boldsymbol{v} \in \ell^2(\Lambda; V)$ and any $\epsilon > 0$, $\texttt{Apply}_{\boldsymbol{R}}[\boldsymbol{v}, \epsilon]$ produces a finitely supported $\boldsymbol{z} \in \ell^2(\Lambda; W^*)$ with*

$$\#\operatorname{supp}\boldsymbol{z} \le 3\sum_{p=1}^{\ell} M_p \#\Lambda_p \tag{3.8}$$

$\texttt{Apply}_{\boldsymbol{R}}[v,\epsilon] \mapsto z$

$[\cdot,(\Lambda_p)_{p=1}^P] \longleftarrow \texttt{BucketSort}\left[(\left\|v_\mu\right\|_V)_{\mu\in\Lambda}, \frac{\epsilon}{2\bar{e}_{\boldsymbol{R},0}}\right]$

for $p = 1,\dots,P$ **do** $v_{[p]} \longleftarrow (v_\mu)_{\mu\in\Lambda_p}$

Compute the minimal $\ell \in \{0,1,\dots,P\}$ s.t. $\delta := \bar{e}_{\boldsymbol{R},0}\left\|v - \sum_{p=1}^{\ell} v_{[p]}\right\|_{\ell^2(\Lambda;V)} \le \frac{\epsilon}{2}$

for $p = 1,\dots,P$ **do** $M_p \longleftarrow 0$

while $\sum_{p=1}^{\ell} \bar{e}_{\boldsymbol{R},M_p}\left\|v_{[p]}\right\|_{\ell^2(\Lambda;V)} > \epsilon - \delta$ **do**

  $q \longleftarrow \operatorname{argmax}_{p=1,\dots,\ell}(\bar{e}_{\boldsymbol{R},M_p} - \bar{e}_{\boldsymbol{R},M_p+1})\left\|v_{[p]}\right\|_{\ell^2(\Lambda;V)}/\#\Lambda_p$

  $M_q \longleftarrow M_q + 1$

$z = (z_\nu)_{\nu\in\Lambda} \longleftarrow \mathbf{0}$

for $p = 1,\dots,\ell$ **do**

  forall $\mu \in \Lambda_p$ **do**

    for $m = 1,\dots,M_p$ **do**

      $w \longleftarrow R_m v_\mu$

      $z_{\mu+\epsilon_m} \longleftarrow z_{\mu+\epsilon_m} + \beta^m_{\mu_m+1} w$

      if $\mu_m \ge 1$ **then** $z_{\mu-\epsilon_m} \longleftarrow z_{\mu-\epsilon_m} + \beta^m_{\mu_m} w$

      if $\alpha^m_{\mu_m} \ne 0$ **then** $z_\mu \longleftarrow z_\mu + \alpha^m_{\mu_m} w$

and

$$\|Rv - z\|_{\ell^2(\Lambda;W^*)} \le \delta + \eta_{\boldsymbol{M}} \le \epsilon\,, \quad \eta_{\boldsymbol{M}} := \sum_{p=1}^{\ell} \bar{e}_{\boldsymbol{R},M_p}\left\|v_{[p]}\right\|_{\ell^2(\Lambda;V)}\,, \tag{3.9}$$

where M_p refers to the final value of this variable in the call of $\texttt{Apply}_{\boldsymbol{R}}$. The total number of products $R_m v_\mu$ computed in $\texttt{Apply}_{\boldsymbol{R}}[v,\epsilon]$ is $\sigma_{\boldsymbol{M}} := \sum_{p=1}^{\ell} M_p \#\Lambda_p$. Furthermore, the vector $\boldsymbol{M} = (M_p)_{p=1}^{\ell}$ is optimal in the sense that if $\boldsymbol{N} = (N_p)_{p=1}^{\ell}$ with $\sigma_{\boldsymbol{N}} \le \sigma_{\boldsymbol{M}}$ then $\eta_{\boldsymbol{N}} \ge \eta_{\boldsymbol{M}}$, and if $\eta_{\boldsymbol{N}} \le \eta_{\boldsymbol{M}}$, then $\sigma_{\boldsymbol{N}} \ge \sigma_{\boldsymbol{M}}$.

We refer to [Git11b, Proposition 2.1] for a proof of Proposition 3.1.

3.2 An Adaptive Solver

Lemma 2.1 suggests a simple iterative procedure for computing u. Beginning with $u_0 := 0 \in L^2_\pi(\Gamma;V)$, we set

$$u_k := \mathcal{D}^{-1}(f - \mathcal{R}u_{k-1})\,, \quad k \in \mathbb{N}\,. \tag{3.10}$$

Then

$$u - u_k = -\mathcal{D}^{-1}\mathcal{R}(u - u_{k-1})\,, \quad k \in \mathbb{N}\,, \tag{3.11}$$

and therefore

$$\|u - u_k\|_{L^2_\pi(\Gamma;V)} \le \gamma^k \|u\|_{L^2_\pi(\Gamma;V)} \le \gamma^k \frac{1}{1-\gamma} \left\|D^{-1}\right\|_{W^*\to V} \left\|f\right\|_{L^2_\pi(\Gamma;W^*)} , \quad k \in \mathbb{N} . \tag{3.12}$$

We transfer this iteration to the representation $\boldsymbol{u} \in \ell^2(\Lambda;V)$ of u, and make it computationally accessible by allowing errors in the computation of f, the evaluation of $\mathcal{A}$ and the inversion of $\mathcal{D}$.

The adaptive approximate application of $\boldsymbol{R}$ is discussed in Section 3.1, and realized by the routine $\texttt{Apply}_{\boldsymbol{R}}$. We assume that a routine

$$\texttt{RHS}_{\boldsymbol{f}}[\epsilon] \mapsto \tilde{\boldsymbol{f}} \tag{3.13}$$

is available to compute approximations $\tilde{\boldsymbol{f}} = (\tilde{f}_\nu)_{\nu\in\Lambda}$ of $\boldsymbol{f}$ with $\#\operatorname{supp}\tilde{\boldsymbol{f}} < \infty$ and

$$\left\|\boldsymbol{f} - \tilde{\boldsymbol{f}}\right\|_{\ell^2(\Lambda;W^*)} \le \epsilon \tag{3.14}$$

for any $\epsilon > 0$. Furthermore, let $\texttt{Solve}_D$ be a solver for D such that for any $g \in W^*$ and any $\epsilon > 0$,

$$\texttt{Solve}_D[g,\epsilon] \mapsto v , \quad \left\|v - D^{-1}g\right\|_V \le \epsilon . \tag{3.15}$$

For example, $\texttt{Solve}_D$ could be an adaptive wavelet method, see *e.g.* [CDD01, CDD02, GHS07], an adaptive frame method, see *e.g.* [Ste03, DFR07, DRW$^+$07], or a finite element method with a posteriori error estimation, see *e.g.* [Dör96, MNS00, BDD04].

A realization of the iteration (3.10) using the above approximations is given in $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$. The initial values can be set to

$$\tilde{\boldsymbol{u}}^{(0)} := \boldsymbol{0} \quad \text{and} \quad \delta_0 := (1-\gamma)^{-1} \left\|D^{-1}\right\|_{W^*\to V} \left\|\boldsymbol{f}\right\|_{\ell^2(\Lambda;W^*)} . \tag{3.16}$$

Note that δ_0 is an upper bound for $\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(0)}\right\|_{\ell^2(\Lambda;V)}$. Values for the other arguments are given below.

Theorem 3.2. *For any $\epsilon > 0$ and any finitely supported $\tilde{\boldsymbol{u}}^{(0)} \in \ell^2(\Lambda;V)$, if $\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(0)}\right\|_{\ell^2(\Lambda;V)} \le \delta_0$, $\alpha > 0$, $\beta_0, \beta_1 > 0$ and $\alpha + \beta_0 + \beta_1 + \gamma < 1$, then $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(0)}, \delta_0, \epsilon, \alpha, \beta_0, \beta_1, \gamma]$ terminates with*

$$\|\boldsymbol{u} - \boldsymbol{u}_\epsilon\|_{\ell^2(\Lambda;V)} \le \bar{\epsilon} \le \epsilon . \tag{3.17}$$

Furthermore, for all $k \in \mathbb{N}$ reached in the iteration,

$$\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2(\Lambda;V)} \le \min(\delta_k, \bar{\delta}_k) \le (\alpha + \beta_0 + \beta_1 + \gamma)^k \delta_0 . \tag{3.18}$$

Proof. We show that for all $k \in \mathbb{N}$,

$$\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2(\Lambda;V)} \le \min(\delta_k, \bar{\delta}_k) .$$

Let $\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k-1)}\right\|_{\ell^2(\Lambda;V)} \le \min(\delta_{k-1}, \bar{\delta}_{k-1})$. Since $\boldsymbol{D}\boldsymbol{u} = \boldsymbol{f} - \boldsymbol{R}\boldsymbol{u}$,

$$\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} = \boldsymbol{D}^{-1}(\boldsymbol{f} - \boldsymbol{R}\boldsymbol{u}) - \boldsymbol{A}^{-1}(\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)}) + \boldsymbol{A}^{-1}(\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)} - \boldsymbol{g}^{(k)}) + \boldsymbol{D}^{-1}\boldsymbol{g}^{(k)} - \tilde{\boldsymbol{u}}^{(k)} .$$

$\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(0)}, \delta_0, \epsilon, \alpha, \beta_0, \beta_1, \gamma] \mapsto [\boldsymbol{u}_\epsilon, \bar{\epsilon}]$

for $k = 1, 2, \dots$ **do**

$\eta_k \longleftarrow \delta_{k-1} \left\| D^{-1} \right\|^{-1}_{W^* \to V}$

$\boldsymbol{g}^{(k)} = (g_\mu^{(k)})_{\mu \in \Lambda} \longleftarrow \texttt{RHS}_{\boldsymbol{f}}[\beta_0 \eta_k] - \texttt{Apply}_{\boldsymbol{R}}[\tilde{\boldsymbol{u}}^{(k-1)}, \beta_1 \eta_k]$

$\zeta_k \longleftarrow \alpha \delta_{k-1} (\#\operatorname{supp} \boldsymbol{g}^{(k)})^{-1/2}$

forall $\mu \in \operatorname{supp} \boldsymbol{g}^{(k)}$ **do** $\tilde{u}_\mu^{(k)} \longleftarrow \texttt{Solve}_D[g_\mu^{(k)}, \zeta_k]$

$\tilde{\boldsymbol{u}}^{(k)} \longleftarrow (\tilde{u}_\mu^{(k)})_{\mu \in \Lambda}$

$\bar{\delta}_{k-1} \longleftarrow (1-\gamma)^{-1} \left(\left\| \tilde{\boldsymbol{u}}^{(k)} - \tilde{\boldsymbol{u}}^{(k-1)} \right\|_{\ell^2(\Lambda;V)} + (\alpha + \beta_0 + \beta_1) \delta_{k-1} \right)$

$\delta_k \longleftarrow (\alpha + \beta_0 + \beta_1) \delta_{k-1} + \gamma \min(\delta_{k-1}, \bar{\delta}_{k-1})$

if $\delta_k \le \epsilon$ **then break**

$\boldsymbol{u}_\epsilon \longleftarrow \tilde{\boldsymbol{u}}^{(k)}$

$\bar{\epsilon} \longleftarrow \delta_k$

By Lemma 2.1,

$$\left\| \boldsymbol{D}^{-1}(\boldsymbol{f} - \boldsymbol{R}\boldsymbol{u}) - \boldsymbol{A}^{-1}(\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)}) \right\|_{\ell^2(\Lambda;V)} = \left\| \boldsymbol{D}^{-1}\boldsymbol{R}(\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k-1)}) \right\|_{\ell^2(\Lambda;V)} \le \gamma \min(\delta_{k-1}, \bar{\delta}_{k-1}) \,.$$

Furthermore, by definition of $\boldsymbol{g}^{(k)}$, using $\left\| \boldsymbol{D}^{-1} \right\|_{\ell^2(\Lambda;W^*) \to \ell^2(\Lambda;V)} = \left\| D^{-1} \right\|_{W^* \to V}$,

$$\left\| \boldsymbol{A}^{-1}(\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)} - \boldsymbol{g}^{(k)}) \right\|_{\ell^2(\Lambda;V)} \le \left\| D^{-1} \right\|_{W^* \to V} (\beta_0 \eta_k + \beta_1 \eta_k) = (\beta_0 + \beta_1) \delta_{k-1} \,.$$

Finally, due to (3.15) and $\zeta_k = \alpha \delta_{k-1} (\#\operatorname{supp} \boldsymbol{g}^{(k)})^{-1/2}$,

$$\left\| \boldsymbol{D}^{-1} \boldsymbol{g}^{(k)} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \le \left(\sum_{\mu \in \operatorname{supp} \boldsymbol{g}^{(k)}} \alpha^2 \delta_{k-1}^2 (\#\operatorname{supp} \boldsymbol{g}^{(k)})^{-1} \right)^{1/2} = \alpha \delta_{k-1} \,.$$

By triangle inequality, the above estimates imply

$$\left\| \boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \le \gamma \min(\delta_{k-1}, \bar{\delta}_{k-1}) + (\beta_0 + \beta_1) \delta_{k-1} + \alpha \delta_{k-1} = \delta_k \,.$$

The residual at iteration k is $\boldsymbol{r}^{(k)} := \boldsymbol{f} - \boldsymbol{A}\tilde{\boldsymbol{u}}^{(k)} = \boldsymbol{A}(\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)})$. We observe that

$$\left\| \boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \le \left\| (\boldsymbol{D}^{-1}\boldsymbol{A})^{-1} \right\| \left\| \boldsymbol{D}^{-1} \boldsymbol{r}^{(k)} \right\|_{\ell^2(\Lambda;V)} \le \frac{1}{1-\gamma} \left\| \boldsymbol{D}^{-1} \boldsymbol{r}^{(k)} \right\|_{\ell^2(\Lambda;V)} \,.$$

Furthermore, $\boldsymbol{D}^{-1}\boldsymbol{r}^{(k)}$ can be approximated by known quantities since, similarly to above,

$$\left\| \tilde{\boldsymbol{u}}^{(k+1)} - \tilde{\boldsymbol{u}}^{(k)} - \boldsymbol{D}^{-1}\boldsymbol{r}^{(k)} \right\|_{\ell^2(\Lambda;V)} = \left\| \tilde{\boldsymbol{u}}^{(k+1)} - \boldsymbol{D}^{-1}(\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k)}) \right\|_{\ell^2(\Lambda;V)} \le (\alpha + \beta_0 + \beta_1) \delta_k \,.$$

Consequently,

$$\left\| \boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \le \frac{1}{1-\gamma} \left(\left\| \tilde{\boldsymbol{u}}^{(k+1)} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} + (\alpha + \beta_0 + \beta_1) \delta_k \right) = \bar{\delta}_k \,.$$

Equation (3.18) follows since $\delta_k \le (\alpha + \beta_0 + \beta_1 + \gamma) \delta_{k-1}$. □

Remark 3.3. The error bounds in $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$ can be improved if each of the subroutines $\texttt{RHS}_{\boldsymbol{f}}$, $\texttt{Apply}_{\boldsymbol{R}}$ and $\texttt{Solve}_D$ returns an estimate of the error it attaines. These values can replace $\alpha\delta_{k-1}$, $\beta_0\delta_{k-1}$ and $\beta_1\delta_{k-1}$ in the definitions of $\bar{\delta}_{k-1}$ and δ_k. For better legibility, we refrain from making this explicit. ⌟

3.3 Alternating Subspace Correction for Symmetric Distributions

In the case that π_m is a symmetric measure for all $m \in \mathbb{N}$, *i.e.* if π is invariant under the transformation $y \mapsto -y$, we suggest an alternative to $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$. Since $\alpha_n^m = 0$ for all m and n, the operator $\boldsymbol{R}$ has the form

$$(\boldsymbol{R}\boldsymbol{v})_\nu = \sum_{m=1}^{\infty} R_m(\beta^m_{\nu_m+1} v_{\nu+\epsilon_m} + \beta^m_{\nu_m} v_{\nu-\epsilon_m}) \tag{3.19}$$

for any $\boldsymbol{v} = (v_\mu)_{\mu\in\Lambda} \in \ell^2(\Lambda; V)$, where $v_\mu := 0$ if $\mu_m < 0$ for any $m \in \mathbb{N}$. If $|\nu| := \|\nu\|_{\ell^1(\mathbb{N})}$ is even, then all the indices $\mu \in \Lambda$ that appear on the right hand side of (3.19) have odd $\big|\mu\big|$; similarly, $\big|\mu\big|$ is always even if $|\nu|$ is odd.

Let $[n] := n + 2\mathbb{Z}$ denote the equivalence class modulo two of $n \in \mathbb{Z}$, *i.e.* $[n] = [m]$ if $n - m$ is even. We define the index sets

$$\Lambda^{[n]} := \left\{\mu \in \Lambda \,;\, [\big|\mu\big|] = [n]\right\} \,, \quad n \in \mathbb{Z} \,, \tag{3.20}$$

where $\big|\mu\big| = \big\|\mu\big\|_{\ell^1(\mathbb{N})}$. Then

$$\Lambda = \Lambda^{[0]} \sqcup \Lambda^{[1]} \,. \tag{3.21}$$

We call $\mu \in \Lambda$ *even* if $\mu \in \Lambda^{[0]}$ and *odd* if $\mu \in \Lambda^{[1]}$.

Let $\Pi^{[n]}$ denote the orthogonal projection in $\ell^2(\Lambda; V)$ onto $\ell^2(\Lambda^{[n]}; V)$; we use the same notation also in $\ell^2(\Lambda; W^*)$. Due to (3.19), $\boldsymbol{R}$ maps even indices onto odd indices and odd indices onto even indices, *i.e.*

$$\Pi^{[n]}\boldsymbol{R} = \boldsymbol{R}\Pi^{[n+1]} \,, \quad n \in \mathbb{Z} \,, \tag{3.22}$$

and $\Pi^{[n]}\boldsymbol{D} = \boldsymbol{D}\Pi^{[n]}$ by definition.

We assume that routines $\texttt{RHS}_{\Pi^{[0]}\boldsymbol{f}}$ and $\texttt{RHS}_{\Pi^{[1]}\boldsymbol{f}}$ are available similar to $\texttt{RHS}_{\boldsymbol{f}}$ from (3.13) to construct approximations of $\Pi^{[0]}\boldsymbol{f}$ and $\Pi^{[1]}\boldsymbol{f}$, and that these approximations have supports in $\Lambda^{[0]}$ and $\Lambda^{[1]}$, respectively. The method $\texttt{Apply}_{\boldsymbol{R}}$ from Section 3.1 already respects even and odd indices in the sense that (3.22) holds also with this approximate application of the operator $\boldsymbol{R}$.

Theorem 3.4. *For any $\epsilon > 0$ and any finitely supported $\tilde{\boldsymbol{u}}^{(-1)} \in \ell^2(\Lambda^{[-1]}; V)$, if $\alpha + \beta_0 + \beta_1 + \gamma < 1$ and $\big\|\Pi^{[-1]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(-1)}\big\|_{\ell^2(\Lambda;V)} \le \delta_{-1}$, then $\texttt{SolveAlternate}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(-1)}, \delta_{-1}, \epsilon, \alpha, \beta_0, \beta_1, \gamma]$ terminates with*

$$\|\boldsymbol{u} - \boldsymbol{u}_\epsilon\|_{\ell^2(\Lambda;V)} \le \bar{\epsilon} \le \epsilon \,. \tag{3.23}$$

Furthermore, for all $k \in \mathbb{N}$ reached in the iteration,

$$\big\|\Pi^{[k]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)}\big\|_{\ell^2(\Lambda;V)} \le \delta_k \le (\alpha + \beta_0 + \beta_1 + \gamma)^{k+1}\delta_{-1} \,. \tag{3.24}$$

$\texttt{SolveAlternate}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(-1)}, \delta_{-1}, \epsilon, \alpha, \beta_0, \beta_1, \gamma] \mapsto [\boldsymbol{u}_\epsilon, \bar{\epsilon}]$

for $k = 0, 1, 2, \dots$ **do**
- $\eta_k \longleftarrow \delta_{k-1} \left\| D^{-1} \right\|^{-1}_{W^* \to V}$
- $\boldsymbol{g}^{(k)} = (g_\mu^{(k)})_{\mu \in \Lambda^{[k]}} \longleftarrow \texttt{RHS}_{\Pi^{[k]}\boldsymbol{f}}[\beta_0 \eta_k] - \texttt{Apply}_{\boldsymbol{R}}[\tilde{\boldsymbol{u}}^{(k-1)}, \beta_1 \eta_k]$
- $\zeta_k \longleftarrow \alpha \delta_{k-1} (\#\operatorname{supp} \boldsymbol{g}^{(k)})^{-1/2}$
- **forall** $\mu \in \operatorname{supp} \boldsymbol{g}^{(k)}$ **do** $\tilde{u}_\mu^{(k)} \longleftarrow \texttt{Solve}_D[g_\mu^{(k)}, \zeta_k]$
- $\tilde{\boldsymbol{u}}^{(k)} \longleftarrow (\tilde{u}_\mu^{(k)})_{\mu \in \Lambda^{[k]}}$
- $\delta_k \longleftarrow (\alpha + \beta_0 + \beta_1 + \gamma) \delta_{k-1}$
- **if** $\delta_{k-1}^2 + \delta_k^2 \leq \epsilon^2$ **then break**

$\boldsymbol{u}_\epsilon \longleftarrow \tilde{\boldsymbol{u}}^{(k-1)} + \tilde{\boldsymbol{u}}^{(k)}$

$\bar{\epsilon} \longleftarrow \sqrt{\delta_{k-1}^2 + \delta_k^2}$

Proof. Since $\boldsymbol{D}\Pi^{[k]}\boldsymbol{u} = \Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\Pi^{[k-1]}\boldsymbol{u}$,

$$\begin{aligned} \Pi^{[k]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} = {} & \boldsymbol{D}^{-1}(\Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\Pi^{[k-1]}\boldsymbol{u}) - \boldsymbol{D}^{-1}(\Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)}) \\ & + \boldsymbol{D}^{-1}(\Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)} - \boldsymbol{g}^{(k)}) + \boldsymbol{D}^{-1}\boldsymbol{g}^{(k)} - \tilde{\boldsymbol{u}}^{(k)} \,. \end{aligned}$$

Due to Lemma 2.1,

$$\left\| \boldsymbol{D}^{-1}(\Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\Pi^{[k-1]}\boldsymbol{u}) - \boldsymbol{D}^{-1}(\Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)}) \right\|_{\ell^2(\Lambda;V)} \leq \gamma \left\| \Pi^{[k-1]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k-1)} \right\|_{\ell^2(\Lambda;V)} \,.$$

By definition of $\boldsymbol{g}^{(k)}$, using $\left\| \boldsymbol{D}^{-1} \right\|_{\ell^2(\Lambda;W^*) \to \ell^2(\Lambda;V)} = \left\| D^{-1} \right\|_{W^* \to V}$,

$$\left\| \boldsymbol{D}^{-1}(\Pi^{[k]}\boldsymbol{f} - \boldsymbol{R}\tilde{\boldsymbol{u}}^{(k-1)} - \boldsymbol{g}^{(k)}) \right\|_{\ell^2(\Lambda;V)} \leq \left\| D^{-1} \right\|_{W^* \to V} (\beta_0 \eta_k + \beta_1 \eta_k) = (\beta_0 + \beta_1) \delta_{k-1} \,.$$

Also, by (3.15),

$$\left\| \boldsymbol{D}^{-1}\boldsymbol{g}^{(k)} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \leq \left(\sum_{\mu \in \operatorname{supp} \boldsymbol{g}^{(k)}} \alpha^2 \delta_{k-1}^2 (\#\operatorname{supp} \boldsymbol{g}^{(k)})^{-1} \right)^{1/2} = \alpha \delta_{k-1} \,.$$

Combining these estimates leads to

$$\left\| \Pi^{[k]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \leq \gamma(\alpha + \beta_0 + \beta_1) \delta_{k-1} + \left\| \Pi^{[k-1]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k-1)} \right\|_{\ell^2(\Lambda;V)} \,.$$

Consequently, if $\left\| \Pi^{[k-1]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k-1)} \right\|_{\ell^2(\Lambda;V)} \leq \delta_{k-1}$, then $\left\| \Pi^{[k]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \leq \delta_k$, and (3.24) follows by induction. □

Remark 3.5. Analogousy to $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$, the error bounds δ_k can be refined using an approximation of the residual. As in the proof of Theorem 3.2, it follows that

$$\left\| \boldsymbol{u} - (\tilde{\boldsymbol{u}}^{(k-1)} + \tilde{\boldsymbol{u}}^{(k)}) \right\|_{\ell^2(\Lambda;V)} \leq \frac{1}{1-\gamma} \left(\left\| \tilde{\boldsymbol{u}}^{(k+1)} - \tilde{\boldsymbol{u}}^{(k-1)} \right\|_{\ell^2(\Lambda;V)} + (\alpha + \beta_0 + \beta_1) \sqrt{\delta_{k-1}^2 + \delta_k^2} \right) \,. \tag{3.25}$$

This term can be used as an alternative upper bound for the errors $\left\|\Pi^{[k]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2(\Lambda;V)}$ and $\left\|\Pi^{[k-1]}\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k-1)}\right\|_{\ell^2(\Lambda;V)}$. However, since it applies to the total error instead of directly to the even or odd part, we expect it to be less useful than the bound $\bar{\delta}_k$ in $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$. ⌟

Remark 3.6. Comparing the convergence estimates (3.18) and (3.24), it appears that $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$ and $\texttt{SolveAlternate}_{\boldsymbol{A},\boldsymbol{f}}$ converge at the same rate. Therefore, since the latter method updates only half of the solution vector in each iteration, it should be roughly twice as efficient. However, Remark 3.5 suggests that $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$ may provide a sharper bound for the error. It is not clear a priori which of these effects is more significant; numerical computations presented in Section 5 indicate that the two solvers are equally efficient. ⌟

4 A Solver with Coarsening

4.1 Subspace Coarsening

Although the methods $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$ and $\texttt{SolveAlternate}_{\boldsymbol{A},\boldsymbol{f}}$ use an efficient adaptive routine for applying the stochastic operator, it is possible that the approximate solution $\tilde{\boldsymbol{u}}^{(k)}$ accumulates many small coefficients. These can be dropped periodically to maintain as sparse a representation as possible. Instead of simply removing the smallest coefficients $\tilde{u}_\mu^{(k)}$, we consider replacing these by approximations using fewer degrees of freedom, *e.g.* on a coarser finite element mesh.

We assume that, for any $v \in V$ constructed by $\texttt{Solve}_D$, there is a finite or countable sequence $(v^i)_i$ of approximations of v. These could be projections of v onto a sequence of subspaces of V, or intermediate approximate solutions if $\texttt{Solve}_D$ is an adaptive solver. For each v^i, let n^i denote the number of basis functions used to represent v^i. We assume that the approximation errors $\left\|v - v^i\right\|_V$ are nonincreasing and converge to zero, and the costs n^i are strictly increasing. Furthermore, we make use of the technical assumption

$$\frac{\left\|v - v^i\right\|_V^2 - \left\|v - v^{i+1}\right\|_V^2}{n^{i+1} - n^i} \geq \frac{\left\|v - v^j\right\|_V^2 - \left\|v - v^{j+1}\right\|_V^2}{n^{j+1} - n^j} \qquad \forall i \leq j\,. \tag{4.1}$$

Also, v^0 is always $0 \in V$, with $n^0 = 0$.

Proposition 4.1. *Let $\boldsymbol{w} \in \ell^2(\Lambda;V)$ be finitely supported. If $(w_\mu^i)_i$ is a sequence of approximations of w_μ as above for all $\mu \in \operatorname{supp}\boldsymbol{w}$, then for any $\epsilon > 0$, a call of $\texttt{Coarsen}[\boldsymbol{w},\epsilon]$ constructs a $\boldsymbol{v} = (w_\mu^{j_\mu})_\mu \in \ell^2(\Lambda;V)$ satisfying*

$$\varrho = \|\boldsymbol{w} - \boldsymbol{v}\|_{\ell^2(\Lambda;V)} \leq \epsilon\,. \tag{4.2}$$

Furthermore, $\sum_\mu n_\mu^{j_\mu}$ is minimal among all sequences $(j_\mu)_\mu$ for which $\boldsymbol{v} = (w_\mu^{j_\mu})_\mu$ satisfies (4.2).

Proof. Equation (4.2) follows from the termination criterion of `Coarsen`; convergence is ensured since $w_\mu^i \to w_\mu$ for all μ. For the optimality property of the greedy algorithm, we refer to the more general statement [Git11a, Theorem 4.1.5]. □

$\texttt{Coarsen}[\boldsymbol{w}, \epsilon] \mapsto [v, \varrho]$

forall $\mu \in \operatorname{supp} \boldsymbol{w}$ **do** $j_\mu \longleftarrow 0$

while $\sum_{\mu \in \operatorname{supp} \boldsymbol{w}} \left\| w_\mu - w_\mu^{j_\mu} \right\|_V^2 > \epsilon^2$ **do**

$$\mu \longleftarrow \operatorname*{argmax}_{\mu \in \operatorname{supp} \boldsymbol{w}} \frac{\left\| w_\mu - w_\mu^{j_\mu} \right\|_V^2 - \left\| w_\mu - w_\mu^{j_\mu+1} \right\|_V^2}{n_\mu^{j_\mu+1} - n_\mu^{j_\mu}}$$

$j_\mu \longleftarrow j_\mu + 1$

forall $\mu \in \operatorname{supp} \boldsymbol{w}$ **do** $v_\mu \longleftarrow w_\mu^{j_\mu}$

$\varrho \longleftarrow \left(\sum_{\mu \in \operatorname{supp} \boldsymbol{w}} \left\| w_\mu - v_\mu \right\|_V^2 \right)^{1/2}$

Remark 4.2. If $w_\mu^0 \coloneqq 0$ and $w_\mu^1 \coloneqq w_\mu$ for all $\mu \in \operatorname{supp} \boldsymbol{w}$, and $n_\mu^0 = 0$, $n_\mu^1 = 1$, then $\texttt{Coarsen}[\boldsymbol{w}, \epsilon]$ reduces to the restriction of $\boldsymbol{w}$ to a subset $\Xi \subset \Lambda$ with minimal $\#\Xi$ under the condition that the approximation error is at most ϵ. This simple special case of `Coarsen` ignores differences in the resolution of the coefficients w_μ, and forgoes the possibility to reduce the resolution without completely truncating an index. ⌟

4.2 A Refined Adaptive Solver

We combine $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$ with a coarsening step in the routine $\texttt{SolveCoarse}_{\boldsymbol{A},\boldsymbol{f}}$. Analogously, $\texttt{SolveAlternate}_{\boldsymbol{A},\boldsymbol{f}}$ could be used in place of $\texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$.

$\texttt{SolveCoarse}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(0)}, \delta_0, \epsilon, \alpha, \beta_0, \beta_1, \gamma, \vartheta, \chi] \mapsto [\boldsymbol{u}_\epsilon, \bar{\epsilon}]$

for $k = 1, 2, \ldots$ **do**

$\eta_k \longleftarrow \max(\vartheta \delta_{k-1}, \epsilon)$

$[\boldsymbol{w}^{(k)}, \zeta_k] \longleftarrow \texttt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(k-1)}, \delta_{k-1}, \chi\eta_k, \alpha, \beta_0, \beta_1, \gamma]$

$[\tilde{\boldsymbol{u}}^{(k)}, \varrho_k] \longleftarrow \texttt{Coarsen}[\boldsymbol{w}^{(k)}, (\chi^{-1} - 1)\zeta_k]$

$\delta_k \longleftarrow \zeta_k + \varrho_k$

if $\delta_k \leq \epsilon$ **then break**

$\boldsymbol{u}_\epsilon \longleftarrow \tilde{\boldsymbol{u}}^{(k)}$

$\bar{\epsilon} \longleftarrow \delta_k$

Proposition 4.3. *For any $\epsilon > 0$, if the assumptions of Theorem 3.2 are satisfied, $0 < \vartheta < 1$ and $0 < \chi \leq 1$, then $\texttt{SolveCoarse}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(0)}, \delta_0, \epsilon, \alpha, \beta_0, \beta_1, \gamma, \vartheta, \chi]$ terminates with*

$$\|\boldsymbol{u} - \boldsymbol{u}_\epsilon\|_{\ell^2(\Lambda;V)} \leq \bar{\epsilon} \leq \epsilon \,. \tag{4.3}$$

Furthermore, for all $k \in \mathbb{N}$ reached in the iteration, with the possible exception of the last k,

$$\left\| \boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)} \right\|_{\ell^2(\Lambda;V)} \leq \delta_k \leq \vartheta^k \delta_0 \,. \tag{4.4}$$

Proof. By Theorem 3.2, $\left\|\boldsymbol{u} - \boldsymbol{w}^{(k)}\right\|_{\ell^2(\Lambda;V)} \le \zeta_k$. Proposition 4.1 implies $\left\|\boldsymbol{w}^{(k)} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2(\Lambda;V)} \le \varrho_k$. Then (4.4) follows by triangle inequality using $\delta_k = \zeta_k + \varrho_k \le \chi^{-1}\zeta_k \le \eta_k$, and $\eta_k \le \vartheta\delta_{k-1}$ if k is not the last iterate. If $\vartheta\delta_{k-1} \le \epsilon$, then $\delta_k \le \eta_k \le \epsilon$, the iteration terminates, and (4.3) follows. □

4.3 Optimality Properties

We consider a semidiscrete version of $\texttt{SolveCoarse}_{\boldsymbol{A},\boldsymbol{f}}$ with no discretization in V. This uses an idealized $\texttt{Solve}_D$, which computes the exact solution of (3.15), and thus we can set $\alpha = 0$. Coarsening is done as in Remark 4.2.

For $\boldsymbol{v} \in \ell^2(\Lambda;V)$ and $N \in \mathbb{N}_0$, let $P_N(\boldsymbol{v})$ be a best N-term approximation of $\boldsymbol{v}$, that is, $P_N(\boldsymbol{v})$ is an element of $\ell^2(\Lambda;V)$ that minimizes $\|\boldsymbol{v} - \boldsymbol{v}_N\|_{\ell^2(\Lambda;V)}$ over $\boldsymbol{v}_N \in \ell^2(\Lambda;V)$ with $\#\operatorname{supp}\boldsymbol{v}_N \le N$. For $s \in (0,\infty)$, we define

$$\|\boldsymbol{v}\|_{\mathcal{A}^s(\Lambda;V)} := \sup_{N\in\mathbb{N}_0} (N+1)^s \|\boldsymbol{v} - P_N(\boldsymbol{v})\|_{\ell^2(\Lambda;V)} \tag{4.5}$$

and

$$\mathcal{A}^s(\Lambda;V) := \left\{\boldsymbol{v} \in \ell^2(\Lambda;V) \;;\; \|\boldsymbol{v}\|_{\mathcal{A}^s(\Lambda;V)} < \infty\right\} . \tag{4.6}$$

By definition, an optimal approximation in $\ell^2(\Lambda;V)$ of $\boldsymbol{v} \in \mathcal{A}^s(\Lambda;V)$ with error tolerance $\epsilon > 0$ consists of $O(\epsilon^{-1/s})$ nonzero coefficients in V.

Proposition 4.4. *Let $\boldsymbol{v} \in \mathcal{A}^s(\Lambda;V)$ for an $s \in (0,\infty)$, and let $\boldsymbol{w} \in \ell^2(\Lambda;V)$ with*

$$\|\boldsymbol{v} - \boldsymbol{w}\|_{\ell^2(\Lambda;V)} \le \epsilon \tag{4.7}$$

for an $\epsilon > 0$. Furthermore, let $N \in \mathbb{N}_0$ be minimal with

$$\|\boldsymbol{w} - \boldsymbol{w}_N\|_{\ell^2(\Lambda;V)} \le 4\epsilon \tag{4.8}$$

for $\boldsymbol{w}_N := P_N(\boldsymbol{w})$. Then

$$\|\boldsymbol{v} - \boldsymbol{w}_N\|_{\ell^2(\Lambda;V)} \le 5\epsilon \le C\,\|\boldsymbol{v}\|_{\mathcal{A}^s(\Lambda;V)}\, N^{-s} , \tag{4.9}$$

$$\|\boldsymbol{w}_N\|_{\mathcal{A}^s(\Lambda;V)} \le C\,\|\boldsymbol{v}\|_{\mathcal{A}^s(\Lambda;V)} \tag{4.10}$$

with a constant C depending only on s.

Proof. Equation (4.9) is a straightforward generalization of [CDD01, Corollary 5.2] to $\ell^2(\Lambda;V)$. To show (4.10), we note that for all $n \le N-1$,

$$\begin{aligned}\|\boldsymbol{w}_N - P_n(\boldsymbol{w}_N)\|_{\ell^2(\Lambda;V)} &\le \|\boldsymbol{w}_N - P_n(\boldsymbol{v})\|_{\ell^2(\Lambda;V)} \le \|\boldsymbol{v} - \boldsymbol{w}_N\|_{\ell^2(\Lambda;V)} + \|\boldsymbol{v} - P_n(\boldsymbol{v})\|_{\ell^2(\Lambda;V)} \\ &\le \|\boldsymbol{v} - \boldsymbol{w}_N\|_{\ell^2(\Lambda;V)} + (n+1)^{-s}\,\|\boldsymbol{v}\|_{\mathcal{A}^s(\Lambda;V)} ,\end{aligned}$$

and the assertion follows using (4.9). □

Lemma 4.5. *Let the assumptions of Proposition 4.3 be satisfied, and $\chi = 1/5$. Then if $\boldsymbol{u} \in \mathcal{A}^s(\Lambda; V)$, there is a constant C depending only on s such that for all k reached by* $\texttt{SolveCoarse}_{\boldsymbol{A},f}$,

$$\#\operatorname{supp} \tilde{\boldsymbol{u}}^{(k)} \leq C \|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)} \delta_k^{-1/s} , \tag{4.11}$$

$$\left\|\tilde{\boldsymbol{u}}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq C \|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)} . \tag{4.12}$$

Proof. By Remark 4.2, $\tilde{\boldsymbol{u}}^{(k)}$ is a best N-term approximation of $\boldsymbol{w}^{(k)}$ with $N = \#\operatorname{supp} \tilde{\boldsymbol{u}}^{(k)}$. Also, $\left\|\boldsymbol{u} - \boldsymbol{w}^{(k)}\right\|_{\ell^2(\Lambda;V)} \leq \zeta_k$ by Theorem 3.2, and N is minimal with $\left\|\boldsymbol{w}^{(k)} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2(\Lambda;V)} \leq 4\zeta_k$. Then, using $\varrho_k \leq 4\zeta_k$, (4.9) implies

$$\#\operatorname{supp} \tilde{\boldsymbol{u}}^{(k)} = N \leq C \|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)} (5\zeta_k)^{-1/s} \leq C \|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)} \delta_k^{-1/s} .$$

Furthermore, (4.12) follows from (4.10). □

Proposition 4.6. *Let $s > 0$. If either*

$$\|R_m\|_{V \to W^*} \leq s\delta_{\boldsymbol{R},s}(m+1)^{-s-1} \qquad \forall m \in \mathbb{N} \tag{4.13}$$

or

$$\left(\sum_{m=1}^{\infty} \|R_m\|^{\frac{1}{s+1}}_{V \to W^*}\right)^{s+1} \leq \delta_{\boldsymbol{R},s} , \tag{4.14}$$

then

$$\left\|\boldsymbol{R} - \boldsymbol{R}_{[M]}\right\|_{\ell^2(\Lambda;V) \to \ell^2(\Lambda;W^*)} \leq \delta_{\boldsymbol{R},s}(M+1)^{-s} \qquad \forall M \in \mathbb{N}_0 . \tag{4.15}$$

We refer to [Git11b, Proposition 4.4] for a proof of Proposition 4.6.

Remark 4.7. If the assumptions of Proposition 4.6 are satisfied for all $s \in (0, s^*)$, then the operator $\boldsymbol{R}$ is s^*-compressible with sparse approximations $\boldsymbol{R}_{[M]}$. In this case, $\boldsymbol{R}$ is a bounded linear map from $\mathcal{A}^s(\Lambda; V)$ to $\mathcal{A}^s(\Lambda; W^*)$ for all $s \in (0, s^*)$, see [CDD01]. This carries over to the routine $\texttt{Apply}_{\boldsymbol{R}}$ in that if $\boldsymbol{v} \in \mathcal{A}^s(\Lambda; V)$ and $\boldsymbol{z}$ is the output of $\texttt{Apply}_{\boldsymbol{R}}[\boldsymbol{v}, \epsilon]$ for an $\epsilon > 0$, then

$$\#\operatorname{supp} \boldsymbol{z} \lesssim \|\boldsymbol{v}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)} \epsilon^{-1/s} , \tag{4.16}$$

$$\|\boldsymbol{z}\|_{\mathcal{A}^s(\Lambda;W^*)} \lesssim \|\boldsymbol{v}\|_{\mathcal{A}^s(\Lambda;V)} \tag{4.17}$$

with constants depending only on s and $\boldsymbol{R}$. Moreover, (4.16) is an upper bound for the total number of applications of operators R_m in $\texttt{Apply}_{\boldsymbol{R}}[\boldsymbol{v}, \epsilon]$. This follows as in the standard scalar case, see *e.g.* [DSS09]. ⌟

We make additional assumptions on the routine $\texttt{RHS}_f$. If $f \in \mathcal{A}^s(\Lambda; W^*)$ and $\tilde{f}$ is the output of $\texttt{RHS}_f[\epsilon]$ for an $\epsilon > 0$, then $\tilde{f}$ should satisfy

$$\#\operatorname{supp} \tilde{f} \lesssim \left\|f\right\|^{1/s}_{\mathcal{A}^s(\Lambda;W^*)} \epsilon^{-1/s} . \tag{4.18}$$

Note that if $\boldsymbol{u} \in \mathcal{A}^s(\Lambda; V)$ and $\boldsymbol{R}$ is s^*-compressible with $s < s^*$, then also $\boldsymbol{A}$ is s^*-compressible, and therefore $\left\|f\right\|_{\mathcal{A}^s(\Lambda;W^*)} \lesssim \|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}$.

Lemma 4.8. *Let $\epsilon > 0$ and let $\tilde{\boldsymbol{u}}^{(0)} \in \ell^2(\Lambda; V)$ be finitely supported with*

$$\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(0)}\right\|_{\ell^2(\Lambda;V)} \leq \delta_0\,, \quad \#\operatorname{supp}\tilde{\boldsymbol{u}}^{(0)} \leq c\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\,\delta_0^{-1/s} \quad \text{and} \quad \left\|\tilde{\boldsymbol{u}}^{(0)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq c\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}\,. \tag{4.19}$$

If $\beta_0, \beta_1 > 0$, $\beta_0 + \beta_1 + \gamma < 1$, and $\boldsymbol{R}$ is s^*-compressible with $s < s^*$, then for all k reached by the loop in $\texttt{SolveDirect}_{\boldsymbol{A},f}[\tilde{\boldsymbol{u}}^{(0)}, \delta_0, \epsilon, 0, \beta_0, \beta_1, \gamma]$,

$$\#\operatorname{supp}\tilde{\boldsymbol{u}}^{(k)} \leq C\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\,\delta_0^{-1/s}\,, \tag{4.20}$$

$$\left\|\tilde{\boldsymbol{u}}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq C\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}\,, \tag{4.21}$$

for a constant C depending only on c, s, $\boldsymbol{R}$, D, β_0, β_1, γ and ϵ/δ_0.

Proof. For $k = 0$, (4.20) and (4.21) hold by assumption. Let $k \in \mathbb{N}$ such that

$$\#\operatorname{supp}\tilde{\boldsymbol{u}}^{(k-1)} \leq C_{k-1}\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\,\delta_0^{-1/s} \quad \text{and} \quad \left\|\tilde{\boldsymbol{u}}^{(k-1)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq C_{k-1}\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}\,.$$

Due to Remark 4.7 and (4.18), there is a constant c_1 such that

$$\begin{aligned}\#\operatorname{supp}\boldsymbol{g}^{(k)} &\leq c_1(\beta_0^{-1/s}\left\|\boldsymbol{f}\right\|^{1/s}_{\mathcal{A}^s(\Lambda;W^*)} + \beta_1^{-1/s}\left\|\tilde{\boldsymbol{u}}^{(k-1)}\right\|^{1/s}_{\mathcal{A}^s(\Lambda;V)})\eta_k^{-1/s}\\ \left\|\boldsymbol{g}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;W^*)} &\leq c_1(\left\|\boldsymbol{f}\right\|_{\mathcal{A}^s(\Lambda;W^*)} + \left\|\tilde{\boldsymbol{u}}^{(k-1)}\right\|_{\mathcal{A}^s(\Lambda;V)})\,.\end{aligned}$$

By assumption, there is a constant c_2 such that $\left\|\boldsymbol{f}\right\|_{\mathcal{A}^s(\Lambda;W^*)} \leq c_2\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}$. Consequently, using $\operatorname{supp}\tilde{\boldsymbol{u}}^{(k)} = \operatorname{supp}\boldsymbol{g}^{(k)}$ and $\left\|\tilde{\boldsymbol{u}}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq \left\|D^{-1}\right\|\left\|\boldsymbol{g}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;W^*)}$, it follows that

$$\begin{aligned}\#\operatorname{supp}\tilde{\boldsymbol{u}}^{(k)} &\leq c_1(\beta_0^{-1/s}c_2^{1/s} + \beta_1^{-1/s}C_{k-1}^{1/s})\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\,\eta_k^{-1/s}\,,\\ \left\|\tilde{\boldsymbol{u}}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;V)} &\leq c_1(c_2 + C_{k-1})\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}\,.\end{aligned}$$

Furthermore, $\delta_0 \leq (\delta_0/\epsilon)\left\|D^{-1}\right\|\eta_k$, which implies the existence of a constant C_k such that

$$\#\operatorname{supp}\tilde{\boldsymbol{u}}^{(k)} \leq C_k\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\,\delta_0^{-1/s} \quad \text{and} \quad \left\|\tilde{\boldsymbol{u}}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq C_k\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)}\,.$$

The assertion follows since the loop terminates after at most $\lceil \log(\epsilon/\delta_0)/\log(\beta_0+\beta_1+\gamma)\rceil$ iterations. □

Remark 4.9. In $\texttt{SolveCoarse}_{\boldsymbol{A},f}$, the method $\texttt{SolveDirect}_{\boldsymbol{A},f}$ is always called with the same required relative error reduction $\chi\vartheta$, except in the last iteration, where this factor may be closer to one. If $\chi = 1/5$, then Lemma 4.5 implies that the constant c in (4.19) is uniform in k. Therefore, Lemma 4.8 implies that there is a constant C such that

$$\#\operatorname{supp}\boldsymbol{w}^{(k)} \leq C\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\,\delta_{k-1}^{-1/s} \quad \text{and} \quad \left\|\boldsymbol{w}^{(k)}\right\|_{\mathcal{A}^s(\Lambda;V)} \leq C\,\|\boldsymbol{u}\|_{\mathcal{A}^s(\Lambda;V)} \quad \forall k\,. \tag{4.22}$$

Note that (4.19) follows from Proposition 4.3 and Lemma 4.5. ⌟

Theorem 4.10. *Let $\beta_0, \beta_1 > 0$, $\beta_0 + \beta_1 + \gamma < 1$, $0 < \vartheta < 1$, $\chi = 1/5$, and let $\tilde{\boldsymbol{u}}^{(0)} \in \ell^2(\Lambda; V)$ with $\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(0)}\right\|_{\ell^2(\Lambda;V)} \leq \delta_0$. If $\boldsymbol{R}$ is s^*-compressible and $s \in (0, s^*)$ such that $\boldsymbol{u} \in \mathcal{A}^s(\Lambda; V)$ and (4.18) holds, then fora ny $\epsilon > 0$, the total number of evaluations of D^{-1} and R_m, $m \in \mathbb{N}$, in $\mathtt{SolveCoarse}_{\boldsymbol{A},\boldsymbol{f}}[\tilde{\boldsymbol{u}}^{(0)}, \delta_0, \epsilon, 0, \beta_0, \beta_1, \gamma, \vartheta, \chi]$ is bounded by $\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\, \epsilon^{-1/s}$ up to a constant factor depending only on $\tilde{\boldsymbol{u}}^{(0)}$, δ_0, s, $\boldsymbol{R}$, D, β_0, β_1, ϑ and γ.*

Proof. The number of calls of D^{-1} in $\mathtt{SolveCoarse}_{\boldsymbol{A},\boldsymbol{f}}$ is equal to the sum of $\#\operatorname{supp} \tilde{\boldsymbol{u}}^{(k)}$, $k \geq 1$, in all interior calls of $\mathtt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$. By Remark 4.7, this sum is also an upper bound for the total number of applications of R_m in all calls of $\mathtt{Apply}_{\boldsymbol{R}}$, up to a constant factor.

Let K denote the total number of calls of $\mathtt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$. By Proposition 4.3, $K \leq \lceil \log(\epsilon/\delta_0)/\log \vartheta \rceil$. Due to the fixed relative error reduction $\chi\vartheta$, the number of iterations within $\mathtt{SolveDirect}_{\boldsymbol{A},\boldsymbol{f}}$ is bounded independently of k. In each one, by Lemma 4.8, the approximate solution has support size at most $C\,\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\, \delta_{k-1}^{-1/s}$. For all $k \leq K$, $\delta_{k-1} \geq \vartheta^{k-K}\delta_{K-1}$. Consequently, up to a constant factor, the total number of inversions of D and applications of R_m is bounded by

$$\sum_{k=1}^{K} \|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\, \delta_{k-1}^{-1/s} \leq \|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s(\Lambda;V)}\, \delta_{K-1}^{-1/s} \sum_{k=1}^{K} \vartheta^{-(k-K)/s}\,.$$

The sum in the second term is bounded by a constant depending only on s. Furthermore, by definition of K, $\delta_{K-1} \geq \epsilon$, which implies $\delta_{K-1}^{-1/s} \leq \epsilon^{-1/s}$. □

5 Numerical Examples

5.1 A Model Problem

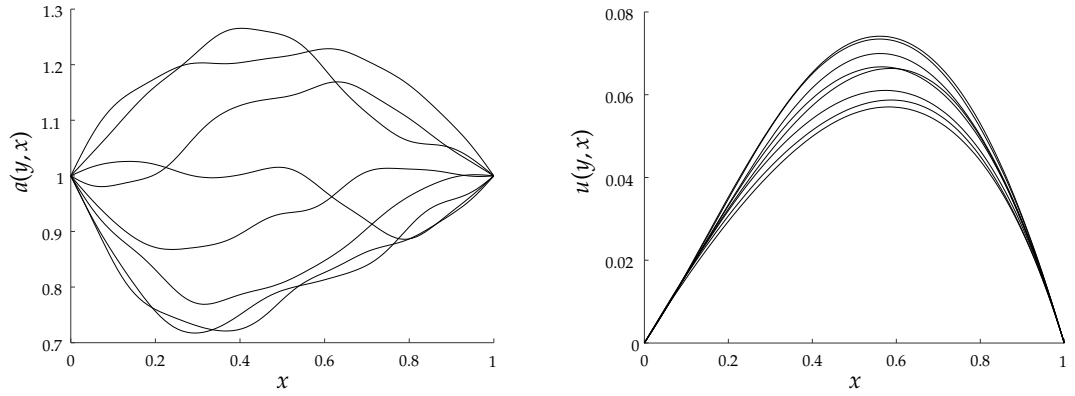

Figure 1: Realizations of $a(y, x)$ (left) and $u(y, x)$ (right).

We consider as a model problem the diffusion equation (1.1) on the one dimensional domain $G = (0,1)$. For two parameters k and γ, the diffusion coefficient has the form

$$a(y,x) = 1 + \frac{1}{c}\sum_{m=1}^{\infty} y_m \frac{1}{m^k}\sin(m\pi x)\,, \quad x \in (0,1)\,, \quad y \in \Gamma = [-1,1]^\infty\,, \tag{5.1}$$

where c is chosen as

$$c = \gamma \sum_{m=1}^{\infty} \frac{1}{m^k}\,, \tag{5.2}$$

such that $\left|a(y,x) - 1\right|$ is always less than γ. We set the parameters to $k = 2$ and $\gamma = 1/2$.

For the distribution of $y \in \Gamma$, we consider the countable product of uniform distributions on $[-1,1]$; the corresponding family of orthonormal polynomials is the Legendre polynomial basis. A few realizations of $a(y)$ and the resulting solutions $u(y)$ of (1.1) are plotted in Figure 1.

We use a multilevel finite element discretization with piecewise linear basis functions on uniform meshes. The residual-based a posteriori error estimator from [Git11b] is used to estimate the error in $\texttt{Solve}_D$. In order to isolate the stochastic discretization, we also consider a fixed spatial discretization, using linear finite elements on a uniform mesh of $(0,1)$ with 1024 elements to approximate all coefficients. We refer to these simpler versions of the numerical methods as single level discretizations. All computations were performed in Matlab on a workstation with an AMD Athlon™ 64 X2 5200+ processor and 4GB of memory.

5.2 Comparison of Solvers

We set the parameters of $\texttt{SolveDirect}_{A,f}$, $\texttt{SolveAlternate}_{A,f}$ and $\texttt{SolveCoarse}_{A,f}$ to $\alpha = 1/20$, $\beta_0 = 0$, $\beta_1 = 1/10$ and, for the last method, $\vartheta = 1/4$ and $\chi = 1/2$. The assumptions of Theorem 4.10 are not satisfied; however, $\texttt{SolveCoarse}_{A,f}$ performs substantially better with $\chi = 1/2$ than with $\chi = 1/5$.

These solvers are compared with $\texttt{SolveGalerkin}_{A,f}$ from [Git11b], with parameters $\chi = 1/8$, $\vartheta = 0.57$, $\omega = 1/4$, $\sigma = 0.01114$, $\alpha = 1/20$ and $\beta = 0$. These values were determined experimentally to maximize the efficiency of the solver. The optimality properties of $\texttt{SolveGalerkin}_{A,f}$ are not proven to hold for these parameters.

We compare the discretizations generated adaptively by the above methods to the heuristic a priori adapted sparse tensor product construction from [BAS10]. Using the notation of [SG11, Section 4], we set $\gamma = 2$ and $\eta_m = 1/(r_m + \sqrt{1 + r_m^2})$ for $r_m = cm^2/2$ and c from (5.2). These values are similar to those used in the computational examples of [BAS10]. The coarsest spatial discretization used in the sparse tensor product contains 16 elements.

Figures 2 and 3 show the convergence of $\texttt{SolveDirect}_{A,f}$ and $\texttt{SolveCoarse}_{A,f}$, respectively. On the left, the errors are plotted against the total number of basis functions used in the discretization. On the right, we plot the errors against an estimate of the computational cost. This estimate takes scalar products, matrix-vector multiplications

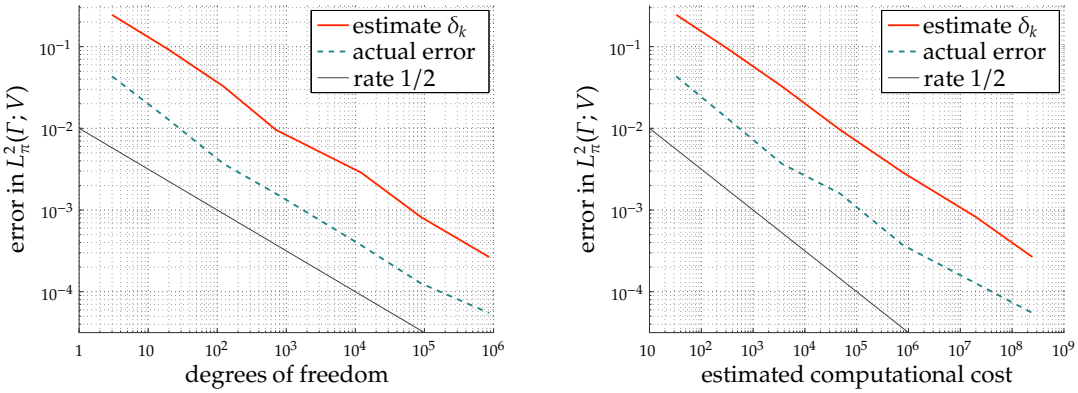

Figure 2: Convergence of $\texttt{SolveDirect}_{A,f}$.

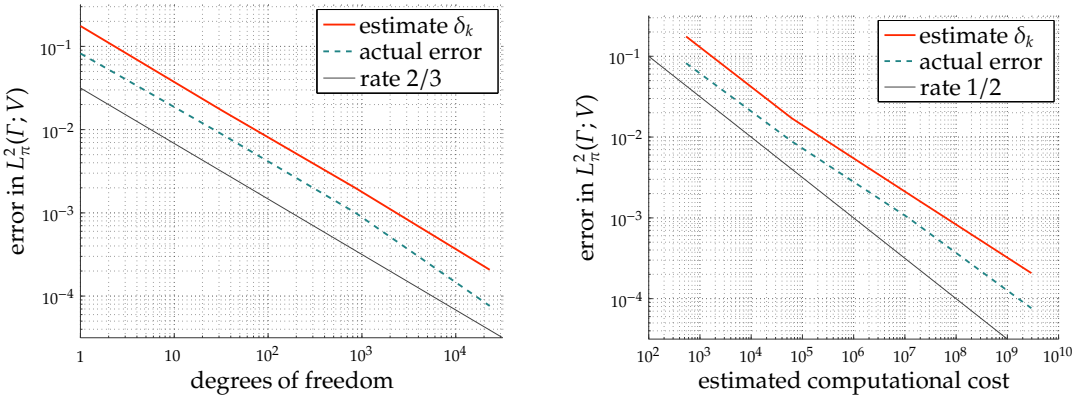

Figure 3: Convergence of $\texttt{SolveCoarse}_{A,f}$.

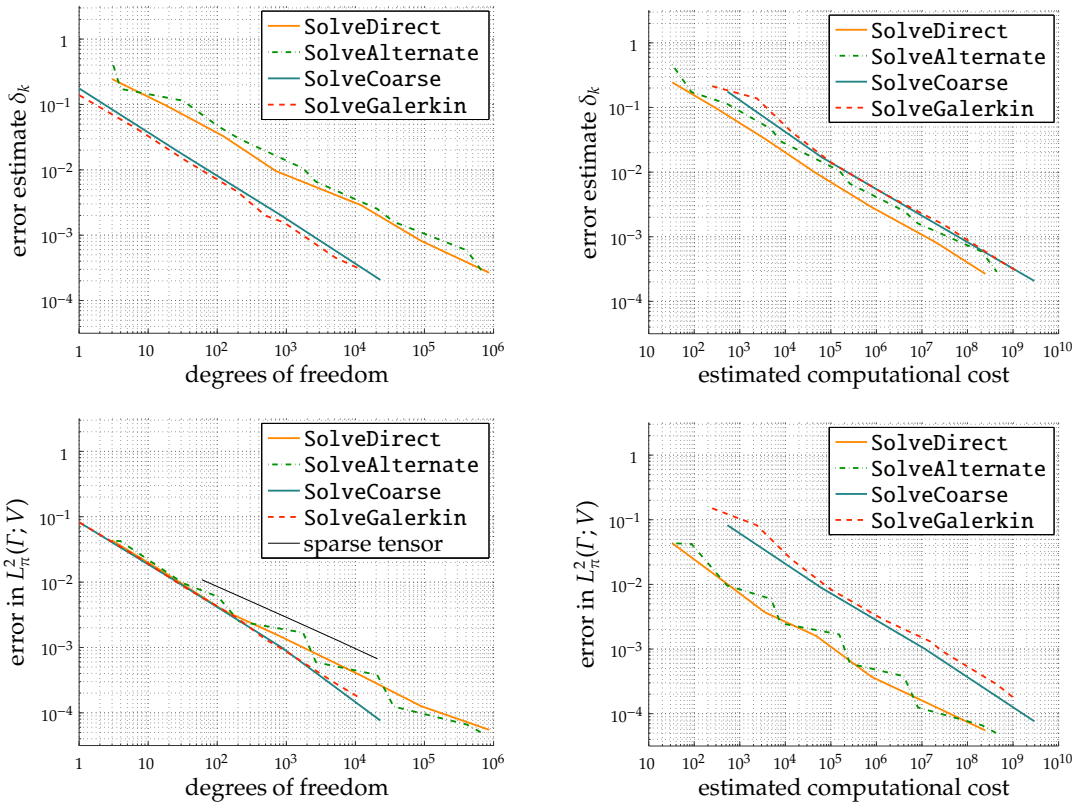

Figure 4: Comparison of numerical solvers.

and linear solves into account. The total number of each of these operations on each discretization level is tabulated during the computation, weighted by the number of degrees of freedom on the discretization level, and summed over all levels. The estimate is equal to seven times the resulting sum for linear solves, plus three times the value for matrix-vector multiplications, plus the sum for scalar products. These weights were determined empirically by timing the operations for tridiagonal sparse matrices in Matlab. The errors were computed by comparison with a reference solution, which has an error of approximately $5 \cdot 10^{-5}$.

We note that, for both methods, the error estimate δ_k is an upper bound for the error in $L^2_\pi(\Gamma;V)$. However, this bound is much sharper for $\texttt{SolveCoarse}_{\boldsymbol{A},f}$ than for $\texttt{SolveDirect}_{\boldsymbol{A},f}$. Also, the convergence of $\texttt{SolveCoarse}_{\boldsymbol{A},f}$ with respect to the number of degrees of freedom is faster than with respect to the computational cost. A similar observation was made in [Git11b] for $\texttt{SolveGalerkin}_{\boldsymbol{A},f}$.

The convergence behavior of all of the above methods is compared in Figure 4. There are two distinct pairs of solvers. The convergence behavior of the two methods with some form of coarsening, $\texttt{SolveCoarse}_{\boldsymbol{A},f}$ and $\texttt{SolveGalerkin}_{\boldsymbol{A},f}$, is almost identical. Furthermore, the convergence of $\texttt{SolveAlternate}_{\boldsymbol{A},f}$ is very similar to that of $\texttt{SolveDirect}_{\boldsymbol{A},f}$, though less regular. Also, the error estimate δ_k seems to be even

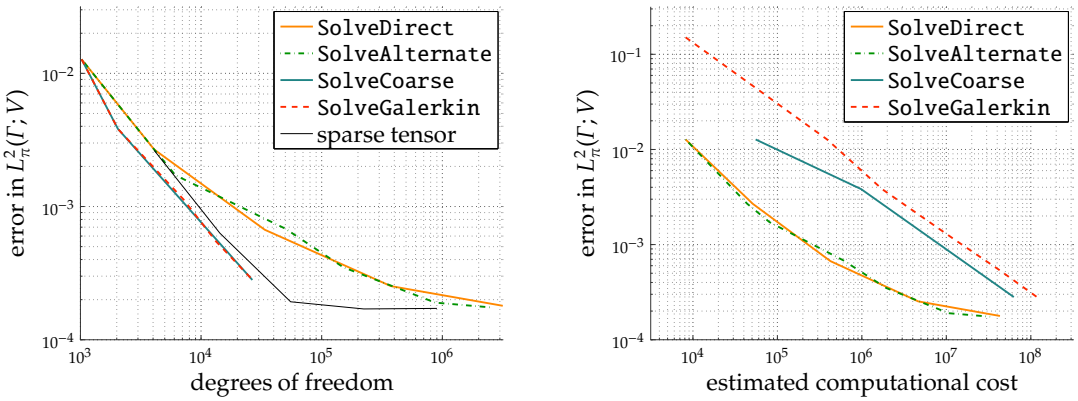

Figure 5: Convergence of single level methods.

coarser for $\texttt{SolveAlternate}_{A,f}$ than for $\texttt{SolveDirect}_{A,f}$, as anticipated by Remark 3.5.

The methods with coarsening steps converge faster than those without coarsening with respect to the number of degrees of freedom in the approximate solution. However, the methods without coarsening are more efficient with respect to the estimated computational cost. The latter effect is most pronounced for the actual error, since it is partially offset by the sharper error bound δ_k of the methods with coarsening. All of the adaptive methods generate discretizations that are more efficient than the sparse tensor product construction.

Figure 5 shows the convergence of the single level versions of the above solvers. The flattening of the convergence curves at an error of $2 \cdot 10^{-4}$ is due to the finite element discretization error, which begins to dominate at this point.

Again, the convergence of the adaptive methods seems to depend most on whether or not they have a coarsening step. The difference between the two classes seems more pronounced than with multilevel discretizations. In particular, the discretizations generated by the methods without coarsening are significantly less efficient than thos with coarsening. This is not surprising, since the superfluous coefficients, which are truncated in the other methods, are all discretized on the same fine grid. With a multilevel discretization, many of these would be approximated only on a coarse mesh.

In the single level setting, the sparse tensor product discretization is clearly more efficient than the discretizations generated adaptively without coarsening. The solvers with coarsening are only able to surpass the sparse tensor construction by a small margin. This suggests that the adaptive solvers are most useful in combination with a multilevel discretization, as opposed to a fixed spatial discretization.

5.3 Empirical Convergence Rates

We use the above convergence plots to empirically determine convergence rates of the adaptive solvers. For the methods without coarsening, the convergence rate with respect to the total number of degrees of freedom is 1/2. This is equal to the approximation

rate shown in [CDS10b, CDS10a], and to the convergence rate of the sparse tensor construction. With a coarsening step, this convergence rate improves to 2/3. However, in both cases, the convergence with respect to the estimated computational cost is just slightly under 1/2.

The solvers with fixed finite element meshes simulate semi-discrete methods with no spatial discretization. In this setting, [CDS10b, CDS10a] show an approximation rate of 3/2 with respect to the number of active indices in Λ. We observe a rate of slightly above 1 for the adaptive methods with coarsening, and for the sparse tensor construction. The solvers without coarsening have a convergence rate of only 1/2—the same rate as with a multilevel discretization, although with a larger error, and a positive spatial discretization error. The convergence rate of the single level methods without coarsening with respect to the estimated computational cost is also approximately 1/2, and that with coarsening is 2/3.

In principle, it is possible that $\texttt{SolveCoarse}_{A,f}$ and $\texttt{SolveGalerkin}_{A,f}$ do not converge with the optimal rate in this example since the parameters used in the computations do not satisfy the conditions of Theorem 4.10, and the analogous result for $\texttt{SolveGalerkin}_{A,f}$. Alternatively, due to large constants in the approximation estimates, the asymptotic rate may not be perceivable for computationally accessible tolerances in this example.

References

[Bar05] A. Barinka. *Fast Evaluation Tools for Adaptive Wavelet Schemes*. PhD thesis, RWTH Aachen, March 2005.

[BAS10] Marcel Bieri, Roman Andreev, and Christoph Schwab. Sparse tensor discretization of elliptic SPDEs. *SIAM J. Sci. Comput.*, 31(6):4281–4304, 2009/10.

[Bau02] Heinz Bauer. *Wahrscheinlichkeitstheorie*. de Gruyter Lehrbuch. [de Gruyter Textbook]. Walter de Gruyter & Co., Berlin, fifth edition, 2002.

[BDD04] Peter Binev, Wolfgang Dahmen, and Ron DeVore. Adaptive finite element methods with convergence rates. *Numer. Math.*, 97(2):219–268, 2004.

[BS09] Marcel Bieri and Christoph Schwab. Sparse high order FEM for elliptic sPDEs. *Comput. Methods Appl. Mech. Engrg.*, 198(37-40):1149–1170, 2009.

[BTZ04] Ivo Babuška, Raúl Tempone, and Georgios E. Zouraris. Galerkin finite element approximations of stochastic elliptic partial differential equations. *SIAM J. Numer. Anal.*, 42(2):800–825 (electronic), 2004.

[CDD01] Albert Cohen, Wolfgang Dahmen, and Ronald DeVore. Adaptive wavelet methods for elliptic operator equations: convergence rates. *Math. Comp.*, 70(233):27–75 (electronic), 2001.

[CDD02] A. Cohen, W. Dahmen, and R. DeVore. Adaptive wavelet methods. II. Beyond the elliptic case. *Found. Comput. Math.*, 2(3):203–245, 2002.

[CDS10a] A. Cohen, R. DeVore, and Ch. Schwab. Analytic regularity and polynomial approximation of parametric stochastic elliptic PDEs. Technical Report 2010-3, Seminar for Applied Mathematics, ETH Zürich, 2010. In review.

[CDS10b] Albert Cohen, Ronald DeVore, and Christoph Schwab. Convergence rates of best N-term Galerkin approximations for a class of elliptic sPDEs. *Found. Comput. Math.*, 10(6):615–646, 2010.

[DBO01] Manas K. Deb, Ivo M. Babuška, and J. Tinsley Oden. Solution of stochastic partial differential equations using Galerkin finite element techniques. *Comput. Methods Appl. Mech. Engrg.*, 190(48):6359–6372, 2001.

[DFR07] Stephan Dahlke, Massimo Fornasier, and Thorsten Raasch. Adaptive frame methods for elliptic operator equations. *Adv. Comput. Math.*, 27(1):27–63, 2007.

[Dör96] Willy Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM J. Numer. Anal.*, 33(3):1106–1124, 1996.

[DRW+07] Stephan Dahlke, Thorsten Raasch, Manuel Werner, Massimo Fornasier, and Rob Stevenson. Adaptive frame methods for elliptic operator equations: the steepest descent approach. *IMA J. Numer. Anal.*, 27(4):717–740, 2007.

[DSS09] Tammo Jan Dijkema, Christoph Schwab, and Rob Stevenson. An adaptive wavelet method for solving high-dimensional elliptic PDEs. *Constr. Approx.*, 30(3):423–455, 2009.

[FST05] Philipp Frauenfelder, Christoph Schwab, and Radu Alexandru Todor. Finite elements for elliptic problems with stochastic coefficients. *Comput. Methods Appl. Mech. Engrg.*, 194(2-5):205–228, 2005.

[GHS07] Tsogtgerel Gantumur, Helmut Harbrecht, and Rob Stevenson. An optimal adaptive wavelet method without coarsening of the iterands. *Math. Comp.*, 76(258):615–629 (electronic), 2007.

[Git11a] Claude Jeffrey Gittelson. *Adaptive Galerkin Methods for Parametric and Stochastic Operator Equations*. PhD thesis, ETH Zürich, 2011. ETH Dissertation No. 19533.

[Git11b] Claude Jeffrey Gittelson. An adaptive stochastic Galerkin method. Technical Report 2011-11, Seminar for Applied Mathematics, ETH Zürich, 2011.

[Git11c] Claude Jeffrey Gittelson. Stochastic Galerkin approximation of operator equations with infinite dimensional noise. Technical Report 2011-10, Seminar for Applied Mathematics, ETH Zürich, 2011.

[Met02] A. Metselaar. *Handling Wavelet Expansions in Numerical Methods*. PhD thesis, University of Twente, 2002.

[MK05] Hermann G. Matthies and Andreas Keese. Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations. *Comput. Methods Appl. Mech. Engrg.*, 194(12-16):1295–1331, 2005.

[MNS00] Pedro Morin, Ricardo H. Nochetto, and Kunibert G. Siebert. Data oscillation and convergence of adaptive FEM. *SIAM J. Numer. Anal.*, 38(2):466–488 (electronic), 2000.

[SG11] Christoph Schwab and Claude Jeffrey Gittelson. Sparse tensor discretization of high-dimensional parametric and stochastic PDEs. *Acta Numerica*, 2011. To appear.

[SS09] Christoph Schwab and Rob Stevenson. Space-time adaptive wavelet methods for parabolic evolution problems. *Math. Comp.*, 78(267):1293–1318, 2009.

[Ste03] Rob Stevenson. Adaptive solution of operator equations using wavelet frames. *SIAM J. Numer. Anal.*, 41(3):1074–1100 (electronic), 2003.

[TS07] Radu Alexandru Todor and Christoph Schwab. Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients. *IMA J. Numer. Anal.*, 27(2):232–261, 2007.

[WK05] Xiaoliang Wan and George Em Karniadakis. An adaptive multi-element generalized polynomial chaos method for stochastic differential equations. *J. Comput. Phys.*, 209(2):617–642, 2005.

[WK06] Xiaoliang Wan and George Em Karniadakis. Multi-element generalized polynomial chaos for arbitrary probability measures. *SIAM J. Sci. Comput.*, 28(3):901–928 (electronic), 2006.

[XK02] Dongbin Xiu and George Em Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644 (electronic), 2002.

Research Reports

No.	Authors/Title
11-12	*C.J. Gittelson* Adaptive stochastic Galerkin methods: beyond the elliptic case
11-11	*C.J. Gittelson* An adaptive stochastic Galerkin method
11-10	*C.J. Gittelson* Stochastic Galerkin approximation of operator equations with infinite dimensional noise
11-09	*R. Hiptmair, A. Moiola and I. Perugia* Error analysis of Trefftz-discontinuous Galerkin methods for the time-harmonic Maxwell equations
11-08	*W. Dahmen, C. Huang, Ch. Schwab and G. Welper* Adaptive Petrov-Galerkin methods for first order transport equations
11-07	*V.H. Hoang and Ch. Schwab* Analytic regularity and polynomial approximation of stochastic, parametric elliptic multiscale PDEs
11-06	*G.M. Coclite, K.H. Karlsen. S. Mishra and N.H. Risebro* A hyperbolic-elliptic model of two-phase flow in porous media - Existence of entropy solutions
11-05	*U.S. Fjordholm, S. Mishra and E. Tadmor* Entropy stable ENO scheme
11-04	*M. Ganesh, S.C. Hawkins and R. Hiptmair* Convergence analysis with parameter estimates for a reduced basis acoustic scattering T-matrix method
11-03	*O. Reichmann* Optimal space-time adaptive wavelet methods for degenerate parabolic PDEs
11-02	*S. Mishra, Ch. Schwab and J. Šukys* Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions
11-01	*V. Wheatley, R. Jeltsch and H. Kumar* Spectral performance of RKDG methods
10-49	*R. Jeltsch and H. Kumar* Three dimensional plasma arc simulation using resistive MHD
10-48	*M. Swärd and S. Mishra* Entropy stable schemes for initial-boundary-value conservation laws