Eidgenössische Technische Hochschule Zürich

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

Adaptive wavelet methods for elliptic partial differential equations with random operators*

C.J. Gittelson

Research Report No. 2011-37
May 2011

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

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

ADAPTIVE WAVELET METHODS FOR ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS WITH RANDOM OPERATORS

CLAUDE JEFFREY GITTELSON

ABSTRACT. We apply adaptive wavelet methods to boundary value problems with random coefficients, discretized by wavelets or frames in the spatial domain and tensorized polynomials in the parameter domain. Greedy algorithms control the approximate application of the fully discretized random operator, and the construction of sparse approximations to this operator. We suggest a power iteration for estimating errors induced by sparse approximations of linear operators.

INTRODUCTION

Uncertain coefficients in boundary value problems can be modeled as random variables or random fields. Stochastic Galerkin methods approximate the solution of the resulting random partial differential equation 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, such as a polynomial chaos basis, see [17, 39, 1, 36, 27, 20, 37, 38].

The primary obstacle in applying these methods is the construction of suitable spaces in which to compute an approximate solution. Sparse tensor product constructions have been shown to be highly effective in [35, 6, 5, 30]. Given sufficient prior knowledge on the regularity of the solution, these methods can be tuned to achieve nearly optimal complexity.

An adaptive approach, requiring less prior information, has been studied in [23, 24, 26]; see also *e.g.* [11] for complementary regularity results, and [7] for a similar approach for stochastic loading instead of a random operator. These methods use techniques from the adaptive wavelet algorithms [9, 10, 21] to select active polynomial chaos modes. Each of these is a deterministic function, and is approximated *e.g.* by adaptive finite elements.

Although these methods perform well in a model problem, the suggested equidistribution of error tolerances among all active polynomial chaos modes is only a heuristic. The theoretical analysis of these methods currently does not guarantee optimal convergence with respect to the full stochastic and spatial discretization.

In the present work, we apply adaptive wavelet methods simultaneously to the stochastic and spatial bases, omitting the former semidiscrete approximation stage. This takes full advantage of the adaptivity in these methods, and in particular their celebrated optimality properties apply to the fully discretized stochastic equation.

This paper is structured as follows. We define random operator equations in Section 1, and derive a weak formulation in the random parameters. In Section 2, we

Date: May 27, 2011.

2010 *Mathematics Subject Classification.* 35R60,47B80,60H35,65C20,65N12,65N22,65J10,65Y20.

Key words and phrases. partial differential equations with random coefficients, uncertainty quantification, stochastic finite element methods, operator equations, adaptive methods.

Research supported in part by the Swiss National Science Foundation grant No. 200021-120290/1. The author expresses his gratitude to Ch. Schwab for his many insightful remarks and suggestions.

construct a tensorized polynomial basis, and recast the random operator equation as an equivalent bi-infinite linear system.

In Section 3, we give an overview of adaptive wavelet methods, focusing in particular on a variant of the algorithm in [21]. Such methods can be used for an arbitrary bi-infinite linear system, provided that a suitable routine is available for approximating the action of the linear operator on a vector. If the operator is not positive, a similar routine for the adjoint operator is also required. In Sections 4 and 5, we present and analyze a generic adaptive application routine, based primarily on [19]. We use a greedy method to solve an optimization problem within this method, and provide a brief analysis of greedy methods in Appendix A.

This efficient approximate application hinges on a sequence of sparse approximations to the discrete operator, and uses estimates of their respective errors. Although convergence rates for such approximations have been shown *e.g.* in [33], explicit error bounds do not seem to be available. In Section 6, we consider a power method for approximating these errors in the operator norm. We provide an analysis of an idealized method, and suggest a practical variant using some ideas from adaptive wavelet methods. We note that this is different from [16] and references therein, where the smallest eigenvalue of *e.g.* a discretized differential operator is computed by an inverse iteration, in that we do not assume a discrete spectrum, and thus do not approximate an eigenvector, and in that we compute the maximum of the spectrum rather than the minimum.

In Section 7, we construct a sequence of sparse approximations of the discrete stochastic operator. This again makes use of a greedy algorithm. Section 7 discusses the abstract properties of s^*-compressibility and s^*-computability for this operator, which are used in the analysis of the adaptive application routine.

Finally, in Section 9, we present a brief example to illustrate our results. We compare the expected s^*-compressibility to approximation rates from [11]. The smaller of these determines the efficiency of adaptive wavelet methods applied to random boundary value problems.

1. Random operator equations

1.1. **Pathwise definition.** 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 conjugate dual space of W, *i.e.* the space of continuous antilinear functionals on W. Furthermore, we define $\mathcal{L}(V, W^*)$ as the Banach space of bounded linear operators from V to W^*, endowed with the operator norm $\|\cdot\|_{V\to W^*}$, and abbreviate $\mathcal{L}(V) \coloneqq \mathcal{L}(V, V)$.

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

$$A\colon \Gamma \to \mathcal{L}(V, W^*) \qquad \text{and} \qquad f\colon \Gamma \to W^*\,, \tag{1.1}$$

we wish to determine

$$u\colon \Gamma \to V\,, \qquad A(y)u(y) = f(y) \quad \forall y \in \Gamma\,. \tag{1.2}$$

Let $\mathcal{B}(\Gamma)$ denote the Borel σ-algebra on Γ. Defining a probability measure π on $(\Gamma, \mathcal{B}(\Gamma))$, A, f and u become random variables. Although π is arbitrary in this section, we assume in Section 2 below that π is a countable product of probability measures on $[-1,1]$.

We decompose the operator A into deterministic and random components,

$$A(y) = D + R(y) \qquad \forall y \in \Gamma\,, \tag{1.3}$$

with $D \in \mathcal{L}(V, W^*)$ boundedly invertible and $R(y) \in \mathcal{L}(V, W^*)$ for all $y \in \Gamma$. Consequently, we also have the multiplicative decomposition

$$A(y) = D\Big(\mathrm{id}_V + D^{-1}R(y)\Big)\,, \quad y \in \Gamma\,. \tag{1.4}$$

Under the assumption

$$\left\|D^{-1}R(y)\right\|_{V\to V} \le \gamma < 1 \qquad \forall y \in \Gamma\,, \tag{1.5}$$

a Neumann series argument ensures existence and uniqueness of the solution $u(y)$ of (1.2) for all $y \in \Gamma$, and

$$\left\|A(y)\right\|_{V\to W^*} \le \|D\|_{V\to W^*}\,(1+\gamma) \qquad \forall y \in \Gamma\,, \tag{1.6}$$

$$\left\|A(y)^{-1}\right\|_{W^*\to V} \le \frac{1}{1-\gamma}\left\|D^{-1}\right\|_{W^*\to V} \qquad \forall y \in \Gamma\,. \tag{1.7}$$

As in *e.g.* [5, 6, 11, 35], we consider random components that are linear in $y \in \Gamma$,

$$R(y) = \sum_{m=1}^{\infty} y_m R_m \qquad \forall y = (y_m)_{m=1}^{\infty} \in \Gamma\,, \tag{1.8}$$

with $R_m \in \mathcal{L}(V, W^*)$ for all m. Such operators arise *e.g.* if A is a differential operator that depends affinely on a random field and this fields is expanded in a series. We assume that $(R_m)_m \in \ell^1(\mathbb{N}; \mathcal{L}(V, W^*))$ with

$$\sum_{m=1}^{\infty} \left\|D^{-1}R_m\right\|_{V\to V} \le \gamma < 1\,, \tag{1.9}$$

which implies (1.5) since $\left|y_m\right| \le 1$.

1.2. **Weak formulation.** Let the map $\Gamma \ni y \mapsto A(y)v(y)$ be measurable for any measurable $v\colon \Gamma \to V$. Then due to (1.6),

$$\mathcal{A}\colon L^2_\pi(\Gamma; V) \to L^2_\pi(\Gamma; W^*)\,, \quad v \mapsto [y \mapsto A(y)v(y)]\,, \tag{1.10}$$

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

$$K_m\colon L^2_\pi(\Gamma) \to L^2_\pi(\Gamma)\,, \quad v(y) \mapsto y_m v(y)\,, \quad m \in \mathbb{N}\,. \tag{1.11}$$

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

Identifying the Lebesgue–Bochner space $L^2_\pi(\Gamma; V)$ with the Hilbert tensor product $L^2_\pi(\Gamma) \otimes V$, and similarly for W^* in place of V, we expand $\mathcal{A}$ as $\mathcal{A} = \mathcal{D} + \mathcal{R}$ with

$$\mathcal{D} \coloneqq \mathrm{id}_{L^2_\pi(\Gamma)} \otimes D \qquad \text{and} \qquad \mathcal{R} \coloneqq \sum_{m=1}^{\infty} K_m \otimes R_m\,. \tag{1.12}$$

This sum converges in $\mathcal{L}(L^2_\pi(\Gamma; V), L^2_\pi(\Gamma; W^*))$ by the assumption that $(R_m)_{m=1}^{\infty}$ is absolutely summable in $\mathcal{L}(V, W^*)$.

Lemma 1.1. $\left\|\mathcal{D}^{-1}\mathcal{R}\right\|_{L^2_\pi(\Gamma;V)\to L^2_\pi(\Gamma;V)} \le \gamma < 1$.

Proof. We note that $\mathcal{D}$ is invertible with $\mathcal{D}^{-1} = \mathrm{id}_{L^2_\pi(\Gamma)} \otimes D^{-1}$, and as in (1.10), $(\mathcal{D}^{-1}\mathcal{R}v)(y) = D^{-1}R(y)v(y)$ for all $v \in L^2_\pi(\Gamma; V)$ and $y \in \Gamma$. Therefore, using (1.5), for all $v \in L^2_\pi(\Gamma; V)$,

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

□

Proposition 1.2. *The operator* $\mathcal{A}$ *from* (1.10) *is boundedly invertible,* $(\mathcal{A}^{-1}g)(y) = A(y)^{-1}g(y)$ *for any* $g \in L^2_\pi(\Gamma; W^*)$*, and*

$$\left\|\mathcal{A}^{-1}\right\|_{L^2_\pi(\Gamma;W^*)\to L^2_\pi(\Gamma;V)} \le \frac{1}{1-\gamma}\left\|D^{-1}\right\|_{W^*\to V}\,. \tag{1.13}$$

Proof. As in (1.4), 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 1.1, $\mathcal{A}$ 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 $A(y)^{-1}$. The estimate (1.13) follows from Lemma 1.1. □

Corollary 1.3. *The solution u of* (1.2) *is in $L^2_\pi(\Gamma;V)$, and u is the unique element of this space satisfying*

$$\mathcal{A}u = f \,. \tag{1.14}$$

The operator equation (1.14) in $L^2_\pi(\Gamma;W^*)$ can be reformulated as

$$\int_\Gamma \langle A(y)u(y), w(y)\rangle \,\mathrm{d}\pi(y) = \int_\Gamma \langle f(y), w(y)\rangle \,\mathrm{d}\pi(y) \qquad \forall w \in L^2_\pi(\Gamma;W) \,. \tag{1.15}$$

2. Discretization of random operator equations

2.1. **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

$$\pi = \bigotimes_{m=1}^{\infty} \pi_m \tag{2.1}$$

for probability measures π_m on $([-1,1],\mathcal{B}([-1,1]))$; see *e.g.* [4, 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 $P^m_{-1} \coloneqq 0$, $P^m_0 \coloneqq 1$ and

$$\beta^m_n P^m_n(\xi) \coloneqq (\xi - \alpha^m_{n-1})P^m_{n-1}(\xi) - \beta^m_{n-1}P^m_{n-2}(\xi) \,, \quad n \in \mathbb{N} \,, \tag{2.2}$$

with

$$\alpha^m_n \coloneqq \int_{-1}^{1} \xi P^m_n(\xi)^2 \,\mathrm{d}\pi_m(\xi) \qquad \text{and} \qquad \beta^m_n \coloneqq \frac{c^m_{n-1}}{c^m_n} \,, \tag{2.3}$$

where c^m_n is the leading coefficient of P^m_n, $\beta^m_0 \coloneqq 1$, 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

$$\Lambda \coloneqq \left\{\mu \in \mathbb{N}_0^{\mathbb{N}} \,;\, \#\operatorname{supp}\mu < \infty\right\} \,, \tag{2.4}$$

where the support is defined by

$$\operatorname{supp}\mu \coloneqq \{m \in \mathbb{N} \,;\, \mu_m \neq 0\} \,, \quad \mu \in \mathbb{N}_0^{\mathbb{N}} \,. \tag{2.5}$$

Then countably infinite tensor product polynomials are given by

$$\boldsymbol{P} \coloneqq \left(P_\mu\right)_{\mu\in\Lambda} \,, \quad P_\mu \coloneqq \bigotimes_{m=1}^{\infty} P^m_{\mu_m} \,, \quad \mu \in \Lambda \,. \tag{2.6}$$

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

$$P_\mu(y) = \prod_{m=1}^{\infty} P^m_{\mu_m}(y_m) = \prod_{m\in\operatorname{supp}\mu} P^m_{\mu_m}(y_m) \,, \quad \mu \in \Lambda \,, \tag{2.7}$$

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

Theorem 2.1. *$\boldsymbol{P}$ is an orthonormal basis of $L^2_\pi(\Gamma)$.*

We refer to *e.g.* [25, Theorem 2.8] for a proof of Theorem 2.1.

2.2. **Tensor product frames.** Let H be a separable Hilbert space. A frame of H is a countable sequence $\boldsymbol{\Phi} \coloneqq (\varphi_\nu)_{\nu\in\Xi} \subset H$ for which the synthesis operator

$$T_{\boldsymbol{\Phi}} \colon \ell^2(\Xi) \to H\,, \quad \boldsymbol{c} = (c_\nu)_{\nu\in\Xi} \mapsto \sum_{\nu\in\Xi} c_\nu \varphi_\nu \tag{2.8}$$

is bounded and surjective. The adjoint of $T_{\boldsymbol{\Phi}}$ is

$$T^*_{\boldsymbol{\Phi}} \colon H^* \to \ell^2(\Xi)\,, \quad f \mapsto (f(\varphi_\nu))_{\nu\in\Xi}\,. \tag{2.9}$$

The upper frame bound of $\boldsymbol{\Phi}$ is $B_{\boldsymbol{\Phi}} \coloneqq \|T_{\boldsymbol{\Phi}}\|_{\ell^2(\Xi)\to H}$. The sequence $\boldsymbol{\Phi}$ is a Riesz basis of H if $T_{\boldsymbol{\Phi}}$ is injective, and an orthonormal basis if $T_{\boldsymbol{\Phi}}$ is unitary.

The frame operator of a frame $\boldsymbol{\Phi}$ in H is the self-adjoint linear map

$$S_{\boldsymbol{\Phi}} \coloneqq T_{\boldsymbol{\Phi}} T^*_{\boldsymbol{\Phi}} \colon H^* \to H\,, \quad f \mapsto \sum_{\nu\in\Xi} f(\varphi_\nu)\varphi_\nu\,. \tag{2.10}$$

It is an isomorphism of Hilbert spaces.

Let $\boldsymbol{\Phi}$ be a frame of H as above and let $\boldsymbol{\Psi} = (\psi_\lambda)_{\lambda\in\Theta}$ be a frame of a second separable Hilbert space L. Then the countable sequence $\boldsymbol{\Phi}\times\boldsymbol{\Psi} \coloneqq (\varphi_\nu \otimes \psi_\lambda)_{(\nu,\lambda)\in\Xi\times\Theta}$ is a frame of $H \otimes L$. Its synthesis operator is

$$T_{\boldsymbol{\Phi}\times\boldsymbol{\Psi}} = T_{\boldsymbol{\Phi}} \otimes T_{\boldsymbol{\Psi}}\,, \tag{2.11}$$

and consequently $B_{\boldsymbol{\Phi}\times\boldsymbol{\Psi}} = B_{\boldsymbol{\Phi}} B_{\boldsymbol{\Psi}}$. Furthermore $\boldsymbol{\Phi}\times\boldsymbol{\Psi}$ is a Riesz basis if and only if both $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are Riesz bases, and an orthonormal basis if and only if $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ are orthonormal bases.

We refer to [22, Chapter 2] and the references therein for details and proofs.

2.3. **Discrete operator equation.** Let $\boldsymbol{\Phi} = (\varphi_\iota)_{\iota\in\Xi}$ be a frame of V and $\boldsymbol{\Psi} = (\psi_\kappa)_{\kappa\in\Theta}$ a frame of W. We define the discrete operator

$$\boldsymbol{A} \coloneqq T^*_{\boldsymbol{P}\times\boldsymbol{\Psi}} \mathcal{A} T_{\boldsymbol{P}\times\boldsymbol{\Phi}} \colon \ell^2(\Lambda\times\Xi) \to \ell^2(\Lambda\times\Theta)\,, \tag{2.12}$$

which can be interpreted as a bi-infinite matrix. Similarly, we define the discretized operators

$$\boldsymbol{D} \coloneqq T^*_{\boldsymbol{\Psi}} D T_{\boldsymbol{\Phi}} \quad \text{and} \quad \boldsymbol{R}_m \coloneqq T^*_{\boldsymbol{\Psi}} R_m T_{\boldsymbol{\Phi}}\,, \quad m \in \mathbb{N}\,, \tag{2.13}$$

which are bounded linear operators from $\ell^2(\Xi)$ to $\ell^2(\Theta)$.

Lemma 2.2. *For all $m \in \mathbb{N}$, $\boldsymbol{K}_m \coloneqq T^*_{\boldsymbol{P}} K_m T_{\boldsymbol{P}} \in \mathcal{L}(\ell^2(\Lambda))$ has the form*

$$(\boldsymbol{K}_m \boldsymbol{c})_\mu = \beta^m_{\mu_m+1} c_{\mu+\epsilon_m} + \alpha^m_{\mu_m} c_\mu + \beta^m_{\mu_m} c_{\mu-\epsilon_m}\,, \quad \mu \in \Lambda\,, \tag{2.14}$$

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

$$\|\boldsymbol{K}_m\|_{\ell^2(\Lambda)\to\ell^2(\Lambda)} = \|K_m\|_{L^2_\pi(\Gamma)\to L^2_\pi(\Gamma)} \le 1\,. \tag{2.15}$$

Proof. The identity (2.14) follows from the three term recursion (2.2). The rest of the claim is a consequence of $T^{-1}_{\boldsymbol{P}} = T^*_{\boldsymbol{P}}$ since K_m is self-adjoint and has norm at most one. □

Proposition 2.3. *The discrete operator $\boldsymbol{A}$ from* (2.12) *satisfies*

$$\boldsymbol{A} = \boldsymbol{I} \otimes \boldsymbol{D} + \sum_{m=1}^{\infty} \boldsymbol{K}_m \otimes \boldsymbol{R}_m \tag{2.16}$$

with convergence in $\mathcal{L}(\ell^2(\Lambda\times\Xi), \ell^2(\Lambda\times\Theta))$.

Proof. The claim follows from (1.12) and (2.11), using the definitions (2.13), the definition of $\boldsymbol{K}_m$ in Lemma 2.2, and $T_{\boldsymbol{P}}^* T_{\boldsymbol{P}} = \boldsymbol{I}$. □

We define the discrete right hand side

$$\boldsymbol{f} \coloneqq T_{\boldsymbol{P}\times\boldsymbol{\Psi}}^* f \in \ell^2(\Lambda \times \Theta) \ . \tag{2.17}$$

Theorem 2.4. *A $\boldsymbol{u} \in \ell^2(\Lambda \times \Xi)$ solves the bi-infinite matrix equation*

$$\boldsymbol{A}\boldsymbol{u} = \boldsymbol{f} \tag{2.18}$$

if and only if it is related to the solution u of (1.2) *by*

$$u = T_{\boldsymbol{P}\times\boldsymbol{\Phi}} \boldsymbol{u} \ . \tag{2.19}$$

In particular, a solution $\boldsymbol{u}$ of (2.18) *exists, and it is unique up to an element of* $\ker T_{\boldsymbol{P}\times\boldsymbol{\Phi}} \subset \ell^2(\Lambda \times \Xi)$.

Proof. Applying $T_{\boldsymbol{P}\times\boldsymbol{\Psi}}^* \mathcal{A}$ to (2.19) and inserting (1.14) and (2.17), it is evident that (2.18) follows from (2.19).

If $\boldsymbol{u} \in \ell^2(\Lambda \times \Xi)$ satisfies (2.18), then applying $T_{\boldsymbol{P}\times\boldsymbol{\Psi}}$ leads to

$$S_{\boldsymbol{P}\times\boldsymbol{\Psi}} \mathcal{A} T_{\boldsymbol{P}\times\boldsymbol{\Phi}} \boldsymbol{u} = T_{\boldsymbol{P}\times\boldsymbol{\Psi}} \boldsymbol{A}\boldsymbol{u} = T_{\boldsymbol{P}\times\boldsymbol{\Psi}} \boldsymbol{f} = S_{\boldsymbol{P}\times\boldsymbol{\Psi}} f \ .$$

Since $S_{\boldsymbol{P}\times\boldsymbol{\Psi}}$ is boundedly invertible, it follows that $T_{\boldsymbol{P}\times\boldsymbol{\Phi}}\boldsymbol{u}$ satisfies (1.14), and therefore (2.19) holds.

Consequently, $\boldsymbol{u}$ is characterized by (2.19). A solution of (2.19) exists since $T_{\boldsymbol{P}\times\boldsymbol{\Phi}}$ is surjective, and it is unique up to an element of $\ker T_{\boldsymbol{P}\times\boldsymbol{\Phi}}$. □

By Theorem 2.4, the solution $\boldsymbol{u}$ of (2.18) is unique if $\boldsymbol{\Phi}$ is a Riesz basis of V.

2.4. **The discrete adjoint operator.** If $\boldsymbol{A}$ is not symmetric positive definite, it is useful to consider the discrete normal equations

$$\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{u} = \boldsymbol{A}^*\boldsymbol{f} \ . \tag{2.20}$$

Here, $\boldsymbol{A}^*$ is the discrete adjoint operator

$$\boldsymbol{A}^* = T_{\boldsymbol{P}\times\boldsymbol{\Phi}}^* \mathcal{A}^* T_{\boldsymbol{P}\times\boldsymbol{\Psi}} \colon \ell^2(\Lambda \times \Theta) \to \ell^2(\Lambda \times \Xi) \ . \tag{2.21}$$

Note that $\boldsymbol{A}^*$ is only injective if $\boldsymbol{\Psi}$ is a Riesz basis of W.

Theorem 2.5. *A $\boldsymbol{u} \in \ell^2(\Lambda \times \Xi)$ solves* (2.20) *if and only if it solves* (2.18)*.*

Proof. If $\boldsymbol{u} \in \ell^2(\Lambda \times \Xi)$ satisfies (2.18), then (2.20) follows by applying $\boldsymbol{A}^*$. Let $\boldsymbol{u} \in \ell^2(\Lambda \times \Xi)$ solve (2.20). Note that by (2.10),

$$\boldsymbol{A}^*\boldsymbol{A} = T_{\boldsymbol{P}\times\boldsymbol{\Phi}}^* \mathcal{A}^* S_{\boldsymbol{P}\times\boldsymbol{\Psi}} \mathcal{A} T_{\boldsymbol{P}\times\boldsymbol{\Phi}} \qquad \text{and} \qquad \boldsymbol{A}^*\boldsymbol{f} = T_{\boldsymbol{P}\times\boldsymbol{\Phi}}^* \mathcal{A}^* f \ .$$

Therefore, applying $T_{\boldsymbol{P}\times\boldsymbol{\Phi}}$ to (2.20) leads to

$$S_{\boldsymbol{P}\times\boldsymbol{\Phi}} \mathcal{A}^* S_{\boldsymbol{P}\times\boldsymbol{\Psi}} \mathcal{A} T_{\boldsymbol{P}\times\boldsymbol{\Phi}} \boldsymbol{u} = S_{\boldsymbol{P}\times\boldsymbol{\Phi}} \mathcal{A}^* S_{\boldsymbol{P}\times\boldsymbol{\Psi}} f \ .$$

Since $S_{\boldsymbol{P}\times\boldsymbol{\Phi}}$, $\mathcal{A}^*$ and $S_{\boldsymbol{P}\times\boldsymbol{\Psi}}$ are all invertible, it follows that $T_{\boldsymbol{P}\times\boldsymbol{\Phi}}\boldsymbol{u}$ satisfies (1.14), and the claim follows using Theorem 2.4. □

The discrete adjoint operator $\boldsymbol{A}^*$ has the same tensor product structure as $\boldsymbol{A}$, with $\boldsymbol{D}$ and R_m replaced by their adjoints.

Proposition 2.6. *The operator $\boldsymbol{A}^*$ has the form*

$$\boldsymbol{A}^* = \boldsymbol{I} \otimes \boldsymbol{D}^* + \sum_{m=1}^{\infty} \boldsymbol{K}_m \otimes \boldsymbol{R}_m^* \tag{2.22}$$

with convergence in $\mathcal{L}(\ell^2(\Lambda \times \Theta), \ell^2(\Lambda \times \Xi))$.

Proof. The claim follows from Theorem 2.3 since $\boldsymbol{I}^* = \boldsymbol{I}$ and $\boldsymbol{K}_m^* = \boldsymbol{K}_m$. □

3. Adaptive wavelet methods

3.1. **An adaptive Galerkin solver.** We consider a bounded linear operator $\boldsymbol{A} \in \mathcal{L}(\ell^2)$, which we interpret also as a bi-infinite matrix. For simplicity, we consider the index sets in the domain and codomain to be $\mathbb{N}$, although we will later tacitly substitute other countable sets.

We assume that $\boldsymbol{A}$ is positive symmetric and boundedly invertible, and consider the equation

$$\boldsymbol{A}\boldsymbol{u} = \boldsymbol{f} \tag{3.1}$$

for a $\boldsymbol{f} \in \ell^2$, as in (2.18) or (2.20). Let $\|\cdot\|_{\boldsymbol{A}}$ denote the norm on ℓ^2 induced by $\boldsymbol{A}$, which we will refer to as the energy norm.

We briefly discuss a variant of the adaptive solver from [9, 21, 19] for (3.1). This method selects a nested sequence of finite sections of the infinite linear system, and solves these to appropriate tolerances. In each step, an approximation of the residual is computed in order to estimate the error and, if necessary, enlarge the set of active indices. For extensions of this method and alternative approaches, we refer to Section 3.4 below. We assume that the action of $\boldsymbol{A}$ can be approximated by a routine

$$\mathtt{Apply}_{\boldsymbol{A}}[v, \epsilon] \mapsto z\,, \quad \|\boldsymbol{A}v - z\|_{\ell^2} \le \epsilon\,, \tag{3.2}$$

for finitely supported vectors v. Similarly, we require a routine

$$\mathtt{RHS}_{\boldsymbol{f}}[\epsilon] \mapsto g\,, \quad \left\|\boldsymbol{f} - g\right\|_{\ell^2} \le \epsilon\,, \tag{3.3}$$

to approximate the right hand side $\boldsymbol{f}$ of (3.1) to an arbitrary precision ϵ. These building blocks are combined in $\mathtt{Residual}_{\boldsymbol{A},\boldsymbol{f}}$ to compute the residual up to an arbitrary relative error.

$\mathtt{Residual}_{\boldsymbol{A},\boldsymbol{f}}[\epsilon, v, \eta_0, \chi, \omega, \beta] \mapsto [\boldsymbol{r}, \eta, \zeta]$

$\zeta \longleftarrow \chi\eta_0$
repeat
  $\boldsymbol{r} \longleftarrow \mathtt{RHS}_{\boldsymbol{f}}[\beta\zeta] - \mathtt{Apply}_{\boldsymbol{A}}[v, (1-\beta)\zeta]$
  $\eta \longleftarrow \|\boldsymbol{r}\|_{\ell^2}$
  if $\zeta \le \omega\eta$ *or* $\eta + \zeta \le \epsilon$ **then break**
  $\zeta \longleftarrow \omega\frac{1-\omega}{1+\omega}(\eta + \zeta)$

Remark 3.1. The loop in $\mathtt{Residual}_{\boldsymbol{A},\boldsymbol{f}}$ terminates either if the residual is guaranteed to be smaller than ϵ, or if the tolerance ζ in the computation of the residual is less than a constant fraction ω of the approximate residual. If neither criterion is met, since $\zeta > \omega\eta$, the updated tolerance satisfies

$$\omega(\eta - \zeta) < \omega\frac{1-\omega}{1+\omega}(\eta + \zeta) < (1-\omega)\zeta\,. \tag{3.4}$$

This ensures a geometric decrease of ζ while also preventing ζ from becoming unnecessarily small. Since $\eta + \zeta$ and $\eta - \zeta$ are upper and lower bounds for the true residual, the updated tolerance ζ satisfies

$$\zeta \ge \omega\frac{1-\omega}{1+\omega}\left\|\boldsymbol{f} - \boldsymbol{A}v\right\|_{\ell^2} \ge \omega\frac{1-\omega}{1+\omega}(\eta - \zeta)\,, \tag{3.5}$$

which implies $\zeta \ge \frac{\omega(1-\omega)}{1+2\omega-\omega^2}\eta$. ⌟

Let $\|\boldsymbol{A}\| \le \hat{\alpha}$ and $\|\boldsymbol{A}^{-1}\| \le \check{\alpha}$. Then $\kappa_{\boldsymbol{A}} \coloneqq \hat{\alpha}\check{\alpha}$ is an upper bound for the condition number $\|\boldsymbol{A}\|\|\boldsymbol{A}^{-1}\|$ of $\boldsymbol{A}$. Furthermore, let $\|\boldsymbol{f}\|_{\ell^2} \le \lambda$.

$\texttt{Solve}_{\boldsymbol{A},\boldsymbol{f}}[\epsilon, \chi, \vartheta, \omega, \sigma, \beta] \mapsto [\boldsymbol{u}_\epsilon, \bar{\epsilon}]$

$\Xi^{(0)} \longleftarrow \varnothing$
$\tilde{\boldsymbol{u}}^{(0)} \longleftarrow \boldsymbol{0}$
$\delta_0 \longleftarrow \check{\alpha}^{1/2}\lambda$
for $k = 0, 1, 2, \ldots$ **do**
  if $\delta_k \leq \epsilon$ **then break**
  $[\boldsymbol{r}_k, \eta_k, \zeta_k] \longleftarrow \texttt{Residual}_{\boldsymbol{A},\boldsymbol{f}}[\epsilon\check{\alpha}^{-1/2}, \tilde{\boldsymbol{u}}^{(k)}, \hat{\alpha}^{1/2}\delta_k, \chi, \omega, \beta]$
  $\bar{\delta}_k \longleftarrow \check{\alpha}^{1/2}(\eta_k + \zeta_k)$
  if $\bar{\delta}_k \leq \epsilon$ **then break**
  $[\Xi^{(k+1)}, \varrho_k] \longleftarrow \texttt{Refine}[\Xi^{(k)}, \boldsymbol{r}_k, \sqrt{\eta_k^2 - (\zeta_k + \vartheta(\eta_k + \zeta_k))^2}]$
  $\bar{\vartheta}_k \longleftarrow \big(\sqrt{\eta_k^2 - \varrho_k^2} - \zeta_k\big)/(\eta_k + \zeta_k)$
  $[\tilde{\boldsymbol{u}}^{(k+1)}, \tau_{k+1}] \longleftarrow \texttt{Galerkin}_{\boldsymbol{A},\boldsymbol{f}}[\Xi^{(k+1)}, \tilde{\boldsymbol{u}}^{(k)}, \sigma \min(\delta_k, \bar{\delta}_k)]$
  $\delta_{k+1} \longleftarrow \tau_{k+1} + \sqrt{1 - \bar{\vartheta}_k^2\kappa_{\boldsymbol{A}}^{-1}}\min(\delta_k, \bar{\delta}_k)$
$\boldsymbol{u}_\epsilon \longleftarrow \tilde{\boldsymbol{u}}^{(k)}$
$\bar{\epsilon} \longleftarrow \min(\delta_k, \bar{\delta}_k)$

The method $\texttt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ uses approximate residuals computed by $\texttt{Residual}_{\boldsymbol{A},\boldsymbol{f}}$ to adaptively select and iteratively solve a finite section of (3.1). For a finite $\Xi \subset \mathbb{N}$, a finitely supported $\boldsymbol{r} \in \ell^2$ and $\epsilon > 0$, the routine

$$\texttt{Refine}[\Xi, \boldsymbol{r}, \epsilon] \mapsto [\bar{\Xi}, \varrho] \tag{3.6}$$

constructs a set $\bar{\Xi} \supset \Xi$ such that $\varrho := \|\boldsymbol{r} - \boldsymbol{r}|_{\bar{\Xi}}\|_{\ell^2} \leq \epsilon$, and $\#\bar{\Xi}$ is minimal with this property, up to a constant factor $\hat{c}$. This can be realized with $\hat{c} = 1$ by sorting $\boldsymbol{r}$ and appending the indices i to Ξ for which $|r_i|$ is largest. Using an approximate sorting routine, `Refine` can be realized in linear complexity with respect to $\#\operatorname{supp}\boldsymbol{r}$ at the cost of a constant $\hat{c} > 1$.

The function

$$\texttt{Galerkin}_{\boldsymbol{A},\boldsymbol{f}}[\Xi, \boldsymbol{v}, \epsilon] \mapsto [\tilde{\boldsymbol{u}}, \tau] \tag{3.7}$$

approximates the solution of (3.1) restricted to the finite index set $\Xi \subset \mathbb{N}$ up to an error of at most $\tau \leq \epsilon$ in the energy norm, using as the initial approximation $\boldsymbol{v}$. For example, a conjugate gradient or conjugate residual method could be used to solve this linear system.

Remark 3.2. In the call of $\texttt{Galerkin}_{\boldsymbol{A},\boldsymbol{f}}$ in $\texttt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$, the previous approximate solution is used as an initial approximation. Alternatively, the approximate residual $\boldsymbol{r}_k$, which is readily available, may be used to compute one step of a linear iteration, such as a Richardson method, prior to calling $\texttt{Galerkin}_{\boldsymbol{A},\boldsymbol{f}}$. Although this may have quantitative advantages, we refrain from going into details in order to keep the presentation and analysis simple. ⌟

3.2. **Convergence analysis.** The convergence analysis of $\texttt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ is based on [9, Lemma 4.1], which is the following statement. We note that the solution of (3.1) restricted to a set $\Xi \subset \mathbb{N}$ is the Galerkin projection onto $\ell^2(\Xi) \subset \ell^2$.

Lemma 3.3. *Let $\Xi \subset \mathbb{N}$ and $\boldsymbol{v} \in \ell^2(\Xi)$ such that, for a $\vartheta \in [0, 1]$,*

$$\big\|(\boldsymbol{f} - \boldsymbol{A}\boldsymbol{v})|_\Xi\big\|_{\ell^2} \geq \vartheta\,\big\|\boldsymbol{f} - \boldsymbol{A}\boldsymbol{v}\big\|_{\ell^2}\ , \tag{3.8}$$

then the Galerkin projection $\bar{\boldsymbol{u}}$ of $\boldsymbol{u}$ onto $\ell^2(\Xi)$ satisfies

$$\|\boldsymbol{u} - \bar{\boldsymbol{u}}\|_{\boldsymbol{A}} \leq \sqrt{1 - \vartheta^2\kappa_{\boldsymbol{A}}^{-1}}\,\|\boldsymbol{u} - \boldsymbol{v}\|\ . \tag{3.9}$$

We note that, by construction, if $\vartheta > 0$, $\omega > 0$ and $\omega + \vartheta + \omega\vartheta \le 1$, then for all k, $\Xi^{(k+1)}$ in $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ is such that

$$\left\|(\boldsymbol{f} - \boldsymbol{A}\tilde{\boldsymbol{u}}^{(k)})|_{\Xi^{(k+1)}}\right\|_{\ell^2} \ge \bar{\vartheta}_k \left\|\boldsymbol{f} - \boldsymbol{A}\tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2} , \tag{3.10}$$

and $\bar{\vartheta}_k \ge \vartheta$. Thus Lemma 3.3 implies an error reduction of at least $\sqrt{1 - \vartheta^2\kappa_{\boldsymbol{A}}^{-1}}$ per step of $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$, up to the error τ_k in the approximation of the Galerkin projection.

Theorem 3.4. *If $\epsilon > 0$, $\chi > 0$, $\vartheta > 0$, $\omega > 0$, $\omega + \vartheta + \omega\vartheta \le 1$, $0 < \beta < 1$ and $0 < \sigma < 1 - \sqrt{1 - \vartheta^2\kappa_{\boldsymbol{A}}^{-1}}$, then $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}[\epsilon, \chi, \vartheta, \omega, \sigma, \beta]$ constructs a finitely supported $\boldsymbol{u}_\epsilon$ with*

$$\|\boldsymbol{u} - \boldsymbol{u}_\epsilon\|_{\boldsymbol{A}} \le \bar{\epsilon} \le \epsilon . \tag{3.11}$$

Moreover, for all $k \in \mathbb{N}_0$ reached by the iteration,

$$\kappa_{\boldsymbol{A}}^{-1/2}\frac{1-\omega}{1+\omega}\bar{\delta}_k \le \left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\boldsymbol{A}} \le \min(\delta_k, \bar{\delta}_k) . \tag{3.12}$$

We refer to [23, Theorem 3.4] for a proof of Theorem 3.4, see also [21, Theorem 2.7].

Remark 3.5. Due to (3.12), in each call of $\mathtt{Galerkin}_{\boldsymbol{A},\boldsymbol{f}}$, an error reduction of at most a fixed factor σ is required. Since the condition number of $\boldsymbol{A}$ restricted to any $\Xi \subset \mathbb{N}$ is at most $\kappa_{\boldsymbol{A}}$, a fixed number of steps of *e.g.* a conjugate gradient iteration suffice, with no need for preconditioning. ⌟

3.3. **Optimality properties.** For $\boldsymbol{v} \in \ell^2$ 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 ℓ^2 that minimizes $\|\boldsymbol{v} - \boldsymbol{v}_N\|_{\ell^2}$ over $\boldsymbol{v}_N \in \ell^2$ with $\#\operatorname{supp}\boldsymbol{v}_N \le N$. For $s \in (0, \infty)$, we define

$$\|\boldsymbol{v}\|_{\mathcal{A}^s} := \sup_{N\in\mathbb{N}_0} (N+1)^s \|\boldsymbol{v} - P_N(\boldsymbol{v})\|_{\ell^2} \tag{3.13}$$

and

$$\mathcal{A}^s := \left\{\boldsymbol{v} \in \ell^2 \,;\, \|\boldsymbol{v}\|_{\mathcal{A}^s} < \infty\right\} . \tag{3.14}$$

Setting $\epsilon = \|\boldsymbol{v} - P_N(\boldsymbol{v})\|_{\ell^2} - \eta$ with $\eta \ge 0$, it follows that

$$\|\boldsymbol{v}\|_{\mathcal{A}^s} = \sup_{\epsilon>0} \epsilon \left(\min\left\{N \in \mathbb{N}_0 \,;\, \|\boldsymbol{v} - P_N(\boldsymbol{v})\|_{\ell^2} \le \epsilon\right\}\right)^s , \tag{3.15}$$

so our definition is consistent with that in [19]. If the index set $\mathbb{N}$ is replaced by a countable set Ξ, we will write $\mathcal{A}^s(\Xi)$ for $\mathcal{A}^s$.

By definition, the space $\mathcal{A}^s$ contains all $\boldsymbol{v} \in \ell^2$ that can be approximated by finitely supported vectors with a rate s,

$$\|\boldsymbol{v} - P_N(\boldsymbol{v})\|_{\ell^2} \le \|\boldsymbol{v}\|_{\mathcal{A}^s} (N+1)^{-s} \qquad \forall N \in \mathbb{N}_0 . \tag{3.16}$$

The following theorem states that, under some conditions on the parameters of $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$, this method recovers the optimal rate s whenever $\boldsymbol{u} \in \mathcal{A}^s$, *i.e.* the approximate Galerkin projections $\tilde{\boldsymbol{u}}^{(k)}$ converge to $\boldsymbol{u}$ at a rate of s with respect to $\#\Xi^{(k)}$.

Theorem 3.6. *If the conditions of Theorem 3.4 are fulfilled,*

$$\hat{\vartheta} := \frac{\vartheta(1+\omega) + 2\omega}{1-\omega} < \kappa_{\boldsymbol{A}}^{-1/2} , \tag{3.17}$$

and $\boldsymbol{u} \in \mathcal{A}^s$ for an $s > 0$, then for all $k \in \mathbb{N}_0$ reached by $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$,

$$\left\|\boldsymbol{u} - \tilde{\boldsymbol{u}}^{(k)}\right\|_{\ell^2} \le 2^s \hat{c}^s \kappa_{\boldsymbol{A}} \tau^{-1} \varrho (1 - \varrho^{1/s})^{-s} \frac{1+\omega}{1-\omega} \|\boldsymbol{u}\|_{\mathcal{A}^s} (\#\Xi^{(k)})^{-s} \tag{3.18}$$

with $\varrho = \sigma + \sqrt{1 - \vartheta^2\kappa_{\boldsymbol{A}}^{-1}}$ and $\tau = \sqrt{1 - \hat{\vartheta}^2\kappa_{\boldsymbol{A}}}$.

The proof of Theorem 3.6 hinges on the following Lemma. We refer to [23, Theorem 4.2] and [21, 19] for details. For a proof of Lemma 3.7, we refer to [23, Lemma 4.1]. See also [21, Lemma 2.1] and [19, Lemma 4.1].

Lemma 3.7. *Let $\Xi^{(0)} \subset \mathbb{N}$ be a finite set and $\boldsymbol{v} \in \ell^2(\Xi^{(0)})$. If $0 < \hat{\vartheta} < \kappa_{\boldsymbol{A}}^{-1/2}$ and $\Xi^{(0)} \subset \Xi^{(1)} \subset \mathbb{N}$ with*

$$\#\Xi^{(1)} \leq c \min\left\{\#\Xi\,;\ \Xi^{(0)} \subset \Xi,\ \left\|(\boldsymbol{f}-\boldsymbol{A}\boldsymbol{v})|_{\Xi}\right\|_{\ell^2} \geq \hat{\vartheta}\left\|\boldsymbol{f}-\boldsymbol{A}\boldsymbol{v}\right\|_{\ell^2}\right\} \tag{3.19}$$

for a $c \geq 1$, then

$$\#(\Xi^{(1)} \setminus \Xi^{(0)}) \leq c \min\left\{\#\hat{\Xi}\,;\ \hat{\Xi} \subset \mathbb{N},\ \|\boldsymbol{u}-\hat{\boldsymbol{u}}\|_{\boldsymbol{A}} \leq \tau\,\|\boldsymbol{u}-\boldsymbol{v}\|_{\boldsymbol{A}}\right\} \tag{3.20}$$

for $\tau = \sqrt{1-\hat{\vartheta}^2\kappa_{\boldsymbol{A}}^{1/2}}$, where $\hat{\boldsymbol{u}}$ denotes the Galerkin projection of $\boldsymbol{u}$ onto $\ell^2(\hat{\Xi})$.

Theorem 3.6 implies that the algorithm $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ is stable in $\mathcal{A}^s$. If the conditions of the theorem are satisfied, then for all k reached in the iteration,

$$\left\|\tilde{\boldsymbol{u}}^{(k)}\right\|_{\mathcal{A}^s} \leq \left(1+\frac{2^{1+s}\hat{c}^s\kappa_{\boldsymbol{A}}\varrho(1+\omega)}{\tau(1-\varrho^{1/s})^s(1-\omega)}\right)\|\boldsymbol{u}\|_{\mathcal{A}^s}\ , \tag{3.21}$$

see *e.g.* [23, Lemma 4.6].

Remark 3.8. The sparsity of approximate solutions is of secondary importance compared to the computational cost of $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$. Under suitable assumptions, the number of operations used by a call of $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ is on the order of $\epsilon^{-1/s}\,\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s}$, which is optimal due to (3.15). Besides the conditions of Theorem 3.6, this presumes that a call of $\mathtt{Apply}_{\boldsymbol{A}}[\boldsymbol{v},\epsilon]$ has a computational cost on the order of

$$1+\#\operatorname{supp}\boldsymbol{v}+\epsilon^{-1/s}\,\|\boldsymbol{v}\|_{\mathcal{A}^s}^{1/s}\ , \tag{3.22}$$

and similarly the cost of $\mathtt{RHS}_{\boldsymbol{f}}[\epsilon]$ is $O(\epsilon^{-1/s}\,\|\boldsymbol{v}\|_{\mathcal{A}^s}^{1/s})$. Due to the geometric decrease of the tolerances ζ in $\mathtt{Residual}_{\boldsymbol{A},\boldsymbol{f}}$, the total cost of this routine is equivalent to that of the last iteration, which is $O(\zeta_k^{-1/s}\,\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s})$, using Theorem 3.6 and (3.21). This includes the cost of `Refine` if this is realized by an approximate sorting routine with linear complexity. Finally, since only a fixed number of steps of a linear iteration is required in $\mathtt{Galerkin}_{\boldsymbol{A},\boldsymbol{f}}$ by Remark 3.5, and each step can realistically be performed in at most the same complexity as $\mathtt{Apply}_{\boldsymbol{A}}$, the computational cost of the k-th iteration in $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ is $O(\zeta_k^{-1/s}\,\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s})$. Equation (3.5) implies that this is equivalent to $O(\bar{\delta}_k^{-1/s}\,\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s})$, and since the error estimates $\bar{\delta}_k$ decrease geometrically, the total cost of $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ is dominated by that of the last iteration of the loop, in which the error is on the order of ϵ. ⌟

3.4. **Extensions and alternatives.** The adaptive wavelet method suggested in [9] differs from $\mathtt{Solve}_{\boldsymbol{A},\boldsymbol{f}}$ in that an absolute tolerance is used in the approximation of the residual. In order to achieve optimality properties similar to those in Section 3.3, [9] requires a coarsening step, which truncates superfluous small entries of the approximate solution. Requiring a relative accuracy in the approximate residual overcame the need for explicit coarsening in [21].

A rather different approach is used in [10]. Instead of adaptively constructing and solving a sequence of finite problems, a linear iteration is applied directly to the full bi-infinite equation (3.1). Individual applications of $\boldsymbol{A}$ and occurrences of $\boldsymbol{f}$ are approximated by the routines $\mathtt{Apply}_{\boldsymbol{A}}$ and $\mathtt{RHS}_{\boldsymbol{f}}$ with tolerances that ensure convergence of the iteration. As in [9], a coarsening step ensures optimality properties of this algorithm. This method has a wider scope of applicability than [9], which includes indefinite linear systems.

As noted in [10], the assumptions of positivity and symmetry of $\boldsymbol{A}$ can be dropped with either approach if (3.1) is replaced by the normal equations $\boldsymbol{A}^*\boldsymbol{A}\boldsymbol{u} = \boldsymbol{A}^*\boldsymbol{f}$. The routines $\mathtt{Apply}_{\boldsymbol{A}}$ and $\mathtt{Apply}_{\boldsymbol{A}^*}$ combine to an adaptive multiplication routine for $\boldsymbol{A}^*\boldsymbol{A}$,

$$\mathtt{Apply}_{\boldsymbol{A}^*\boldsymbol{A}}[v,\epsilon] \mapsto \mathtt{Apply}_{\boldsymbol{A}^*}[\mathtt{Apply}_{\boldsymbol{A}}[v,\epsilon/(2\hat{\alpha})],\epsilon/2]\,, \tag{3.23}$$

see *e.g.* [31, Cor. 4.6]. Similarly, $\boldsymbol{A}^*\boldsymbol{f}$ can be approximated by

$$\mathtt{RHS}_{\boldsymbol{A}^*\boldsymbol{f}}[\epsilon] \mapsto \mathtt{Apply}_{\boldsymbol{A}^*}[\mathtt{RHS}_{\boldsymbol{f}}[\epsilon/(2\hat{\alpha})],\epsilon/2]\,. \tag{3.24}$$

This leads to a solver $\mathtt{Solve}_{\boldsymbol{A}^*\boldsymbol{A},\boldsymbol{A}^*\boldsymbol{f}}$ which does not require $\boldsymbol{A}$ to be a positive operator.

All of the above methods assume that the operator $\boldsymbol{A}$ is regular. However, if $\boldsymbol{A}$ arises from a frame discretization of a differential or integral operator, then $\boldsymbol{A}$ is generally singular. The method from [10] has been generalized to this setting in [32, 14, 15, 13]. We refer to [34] for a survey of adaptive wavelet methods for linear operator equations.

4. Adaptive application of s^*-compressible operators

4.1. **s^*-compressibility and s^*-computability.** A routine $\mathtt{Apply}_{\boldsymbol{A}}$ for approximately applying an operator $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ to a finitely supported vector constitutes an essential component of the adaptive solvers from Section 3. Such a routine can be constructed if $\boldsymbol{A}$ can be approximated by sparse operators, as in the following definition. Again, we interpret $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ also as a bi-infinite matrix, and restrict to the index set $\mathbb{N}$ only to simplify notation.

Definition 4.1. An operator $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ is *n-sparse* if each column contains at most n nonzero entries. It is *s^*-compressible* for an $s^* \in (0,\infty]$ if there exists a sequence $(\boldsymbol{A}_j)_{j\in\mathbb{N}}$ in $\mathcal{L}(\ell^2)$ such that $\boldsymbol{A}_j$ is n_j-sparse with $(n_j)_{j\in\mathbb{N}} \in \mathbb{N}^{\mathbb{N}}$ satisfying

$$c_{\boldsymbol{A}} := \sup_{j\in\mathbb{N}} \frac{n_{j+1}}{n_j} < \infty \tag{4.1}$$

and for every $s \in (0,s^*)$,

$$d_{\boldsymbol{A},s} := \sup_{j\in\mathbb{N}} n_j^s \left\|\boldsymbol{A}-\boldsymbol{A}_j\right\|_{\ell^2\to\ell^2} < \infty\,. \tag{4.2}$$

The operator $\boldsymbol{A}$ is *strictly s^*-compressible* if, in addition,

$$\sup_{s\in(0,s^*)} d_{\boldsymbol{A},s} < \infty\,. \tag{4.3}$$

⌟

Remark 4.2. Equation (4.2) states that for all $s \in (0,s^*)$, the approximation errors satisfy

$$e_{\boldsymbol{A},j} := \left\|\boldsymbol{A}-\boldsymbol{A}_j\right\|_{\ell^2\to\ell^2} \le d_{\boldsymbol{A},s} n_j^{-s}\,, \quad j\in\mathbb{N}\,. \tag{4.4}$$

If $s^* < \infty$, this is equivalent to the condition that $(n_j^{s^*} e_{\boldsymbol{A},j})_{j\in\mathbb{N}}$ grows subalgebraically in n_j, *i.e.*

$$n_j^{s^*} e_{\boldsymbol{A},j} \le \inf_{r>0} d_{\boldsymbol{A},s^*-r} n_j^r\,, \quad j\in\mathbb{N}\,. \tag{4.5}$$

Strict s^*-compressibility states that the right hand side of (4.5) is bounded in j, *i.e.*

$$d_{\boldsymbol{A},s^*} = \sup_{j\in\mathbb{N}} n_j^{s^*} e_{\boldsymbol{A},j} = \sup_{j\in\mathbb{N}} \sup_{s\in(0,s^*)} n_j^s e_{\boldsymbol{A},j} = \sup_{s\in(0,s^*)} d_{\boldsymbol{A},s} < \infty\,. \tag{4.6}$$

Of course, s^*-compressibility implies strict s-compressibility for all $s \in (0,s^*)$. ⌟

Proposition 4.3. *Let $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ be s^*-compressible with an approximating sequence $(\boldsymbol{A}_j)_{j\in\mathbb{N}}$ as in Definition 4.1, and set $\boldsymbol{A}_0 \coloneqq 0$. There is a map $j\colon [0,\infty) \to \mathbb{N}_0$ such that $\boldsymbol{A}_{j(r)}$ is r-sparse for all $r \in [0,\infty)$ and for all $s \in (0, s^*)$,*

$$e_{\boldsymbol{A},j(r)} = \left\|\boldsymbol{A} - \boldsymbol{A}_{j(r)}\right\|_{\ell^2\to\ell^2} \leq \max\left(c_{\boldsymbol{A}}^s d_{\boldsymbol{A},s}, n_1^s e_{\boldsymbol{A},0}\right) r^{-s} \tag{4.7}$$

for $r > 0$, where $e_{\boldsymbol{A},0} \coloneqq \|\boldsymbol{A}\|_{\ell^2\to\ell^2}$.

Proof. Set $n_0 \coloneqq 0$ and define

$$j(r) \coloneqq \max\left\{j \in \mathbb{N}_0 \,;\, n_j \leq r\right\}\,, \quad r \in [0,\infty)\,. \tag{4.8}$$

Then $\boldsymbol{A}_{j(r)}$ is r-sparse, and if $j(r) \geq 1$,

$$e_{\boldsymbol{A},j(r)} \leq d_{\boldsymbol{A},s} n_{j(r)}^{-s} \leq d_{\boldsymbol{A},s} c_{\boldsymbol{A}}^s n_{j(r)+1}^{-s} \leq d_{\boldsymbol{A},s} c_{\boldsymbol{A}}^s r^{-s}$$

by (4.4) and (4.1). If $j(r) = 0$, then $r < n_1$, and

$$e_{\boldsymbol{A},j(r)} = e_{\boldsymbol{A},0} \leq e_{\boldsymbol{A},0} n_1^s r^{-s}\,. \qquad \square$$

In particular, Proposition 4.3 implies that Definition 4.1 coincides with the notion of s^*-compressibility for example in [21, 31], *i.e.* one can assume $n_j = j$ in the definition of s^*-compressibility at the cost of increasing the constants (4.2) and obscuring the discrete structure of the sparse approximating sequence. We denote the resulting compressibility constants by

$$\tilde{d}_{\boldsymbol{A},s} \coloneqq \sup_{r\in(0,\infty)} r^s \left\|\boldsymbol{A} - \boldsymbol{A}_{j(r)}\right\|_{\ell^2\to\ell^2} \leq \max\left(c_{\boldsymbol{A}}^s d_{\boldsymbol{A},s}, n_1^s e_{\boldsymbol{A},0}\right) < \infty \tag{4.9}$$

for $s \in (0, s^*)$, where $j(r)$ is given by (4.8). Also, it follows using Proposition 4.3 that any s^*-compressible operator $\boldsymbol{A}$ for which $\boldsymbol{A}^*$ is also s^*-compressible, $\boldsymbol{A}$ is in the class $\mathcal{B}_s$ defined in [9] for all $s \in [0, s^*)$.

Although s^*-compressibility is a precise mathematical property, it is only useful for applications if the sparse approximations to the bi-infinite matrix can be computed efficiently. This is the context of the following, more restrictive definition.

Definition 4.4. An operator $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ is *s^*-computable* for an $s^* \in (0,\infty]$ if it is s^*-compressible with an approximating sequence $(\boldsymbol{A}_j)_{j\in\mathbb{N}}$ as in Definition 4.1 such that $\boldsymbol{A}_j$ is n_j-sparse and there exists a routine

$$\texttt{Build}_{\boldsymbol{A}}[j,k] \mapsto \left[(l_i)_{i=1}^{n_j}, (a_i)_{i=1}^{n_j}\right] \tag{4.10}$$

such that the k-th column of $\boldsymbol{A}_j$ is equal to

$$\sum_{i=1}^{n_j} a_i \epsilon_{l_i}\,, \tag{4.11}$$

where ϵ_{l_i} is the Kronecker sequence that is 1 at l_i and 0 elsewhere, and there is a constant $b_{\boldsymbol{A}}$ such that the number of arithmetic operations and storage locations used by a call of $\texttt{Build}_{\boldsymbol{A}}[j,k]$ is less than $b_{\boldsymbol{A}} n_j$ for any $j \in \mathbb{N}$ and $k \in \mathbb{N}$. ⌟

Note that the indices l_i in (4.10) are not assumed to be distinct, so a single entry of $\boldsymbol{A}_j$ may be given by a sum of values a_i. However, the total number of a_i computed by $\texttt{Build}_{\boldsymbol{A}}[j,k]$ is at most n_j.

4.2. **An adaptive approximate multiplication routine.** It was shown in [9, 10] that s^*-computable operators can be applied efficiently to finitely supported vectors. A routine with computational advantages was presented in [19]. We extend this method by using a greedy algorithm to solve the optimization problem at the heart of the routine.

Let $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ and for all $k \in \mathbb{N}_0$, let $\boldsymbol{A}_k$ be n_k-sparse with $n_0 = 0$ and

$$\|\boldsymbol{A} - \boldsymbol{A}_k\|_{\ell^2 \to \ell^2} \leq \bar{e}_{\boldsymbol{A},k} \,. \tag{4.12}$$

We consider a partitioning of a vector $v \in \ell^2$ into $v_{[p]} := v|_{\Xi_p}$, $p = 1, \dots, P$, for disjoint index sets $\Xi_p \subset \mathbb{N}$. This can be approximate in that $v_{[1]} + \cdots + v_{[P]}$ only approximates v in ℓ^2. We think of $v_{[1]}$ as containing the largest elements of v, $v_{[2]}$ the next largest, and so on.

Such a partitioning can be constructed by the approximate sorting algorithm

$$\texttt{BucketSort}[v, \epsilon] \mapsto \left[(v_{[p]})_{p=1}^P, (\Xi_p)_{p=1}^P\right] , \tag{4.13}$$

which, given a finitely supported $v \in \ell^2$ and a threshold $\epsilon > 0$, returns index sets

$$\Xi_p := \left\{\mu \in \mathbb{N} \,;\, |v_\mu| \in (2^{-p/2}\,\|v\|_{\ell^\infty}\,, 2^{-(p-1)/2}\,\|v\|_{\ell^\infty}]\right\} \tag{4.14}$$

and $v_{[p]} := v|_{\Xi_p}$, see [28, 2, 21, 19]. The integer P is minimal with

$$2^{-P/2}\,\|v\|_{\ell^\infty}\,\sqrt{\#\operatorname{supp} v} \leq \epsilon \,. \tag{4.15}$$

By [21, Rem. 2.3] or [19, Prop. 4.4], the number of operations and storage locations required by a call of $\texttt{BucketSort}[v, \epsilon]$ is bounded by

$$\#\operatorname{supp} v + \max(1, \lceil \log(\|v\|_{\ell^\infty}\,\sqrt{\#\operatorname{supp} v}/\epsilon)\rceil) \,. \tag{4.16}$$

This analysis uses that every v_μ, $\mu \in \mathbb{N}$, can be mapped to p with $\mu \in \Xi_p$ in constant time by evaluating

$$p := \left\lceil 1 + 2\log_2\left(\frac{\|v\|_{\ell^\infty}}{|v_\mu|}\right)\right\rceil \,. \tag{4.17}$$

Alternatively, any exact comparison-based sorting algorithm can be used to construct the partitioning of v, albeit with an additional logarithmic factor in the complexity.

For any $\boldsymbol{k} = (k_p)_{p=1}^\ell \in \mathbb{N}_0^\ell$, with $\ell \in \mathbb{N}_0$ determined as in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$, define

$$\zeta_{\boldsymbol{k}} := \sum_{p=1}^{\ell} \bar{e}_{\boldsymbol{A},k_p}\left\|v_{[p]}\right\|_{\ell^2(\Xi_p)} \qquad \text{and} \qquad \sigma_{\boldsymbol{k}} := \sum_{p=1}^{\ell} n_{k_p}(\#\operatorname{supp} v_{[p]}) \,. \tag{4.18}$$

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

$(v_{[p]})_{p=1}^P \longleftarrow \texttt{BucketSort}\left[v, \dfrac{\epsilon}{2\bar{e}_{\boldsymbol{A},0}}\right]$

compute the minimal $\ell \in \{0, 1, \dots, P\}$ s.t. $\delta := \bar{e}_{\boldsymbol{A},0}\left\|v - \sum_{p=1}^{\ell} v_{[p]}\right\|_{\ell^2} \leq \dfrac{\epsilon}{2}$

$\boldsymbol{k} = (k_p)_{p=1}^\ell \longleftarrow (0)_{p=1}^\ell$

while $\zeta_{\boldsymbol{k}} > \epsilon - \delta$ **do**

  $\boldsymbol{k} \longleftarrow \texttt{NextOpt}[\boldsymbol{k}]$ with objective $-\zeta_{\boldsymbol{k}}$ and cost $\sigma_{\boldsymbol{k}}$

$z \longleftarrow \sum_{p=1}^{\ell} \boldsymbol{A}_{k_p} v_{[p]}$

The algorithm $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ has three distinct parts. First, the elements of v are grouped according to their magnitude. Elements smaller than a certain tolerance are neglected. This truncation of the vector v produces an error of at most $\delta \leq \epsilon/2$.

Next, a greedy algorithm is used to assign to each segment $v_{[p]}$ of v a sparse approximation $\boldsymbol{A}_{k_p}$ of $\boldsymbol{A}$, see Appendix A. Starting with $\boldsymbol{A}_{k_p} = \boldsymbol{0}$ for all $p = 1, \dots, \ell$, these approximations are refined iteratively until an estimate for the error resulting from the approximation of $\boldsymbol{A}$ by $\boldsymbol{A}_{k_p}$ for all $p = 1, \dots, \ell$ is bounded by $\zeta_{\boldsymbol{k}} \leq \epsilon - \delta$.

Finally, the multiplications determined by the previous two steps are performed. A few elementary properties of this method are summarized in the following proposition.

Proposition 4.5. *For any finitely supported $v \in \ell^2$ and any $\epsilon > 0$, if $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ terminates, its output is a finitely supported $z \in \ell^2$ with*

$$\#\operatorname{supp} z \leq \sum_{p=1}^{\ell} n_{k_p} (\#\operatorname{supp} v_{[p]}) \tag{4.19}$$

and

$$\|\boldsymbol{A}v - z\|_{\ell^2} \leq \delta + \zeta_{\boldsymbol{k}} \leq \epsilon \,, \tag{4.20}$$

where $\boldsymbol{k} = (k_p)_{p=1}^{\ell}$ is the vector constructed by the greedy algorithm in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$. Furthermore, the number of arithmetic operations required by the final step of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ is bounded by

$$\sum_{p=1}^{\ell} n_{k_p} (\#\operatorname{supp} v_{[p]}) \tag{4.21}$$

if the relevant entries of $\boldsymbol{A}_{k_p}$ are precomputed.

Proof. We show (4.20). Since $\|\boldsymbol{A}\|_{\ell^2 \to \ell^2} \leq \bar{e}_{\boldsymbol{A},0}$,

$$\left\| \boldsymbol{A}v - \boldsymbol{A} \sum_{p=1}^{\ell} v_{[p]} \right\|_{\ell^2} \leq \bar{e}_{\boldsymbol{A},0} \left\| v - \sum_{p=1}^{\ell} v_{[p]} \right\|_{\ell^2} = \delta \leq \frac{\epsilon}{2} \,.$$

By (4.12), if $\boldsymbol{k} = (k_p)_{p=1}^{\ell}$ is the final value of $\boldsymbol{k}$,

$$\sum_{p=1}^{\ell} \left\| \boldsymbol{A} v_{[p]} - \boldsymbol{A}_{k_p} v_{[p]} \right\|_{\ell^2} \leq \sum_{p=1}^{\ell} \bar{e}_{\boldsymbol{A},k_p} \left\| v_{[p]} \right\|_{\ell^2(\Xi_p)} = \zeta_{\boldsymbol{k}} \leq \epsilon - \delta \,.$$

□

Let $v \in \ell^2$ be finitely supported and $\epsilon > 0$. Note that by (4.14) and (4.15),

$$\left\| v - \sum_{p=1}^{P} v_{[p]} \right\|_{\ell^2} \leq 2^{-P/2} \|v\|_{\ell^\infty} \sqrt{\#\operatorname{supp} v} \leq \frac{\epsilon}{2\bar{e}_{\boldsymbol{A},0}} \,,$$

so ℓ is well-defined. It is not immediately clear, however, that the greedy algorithm in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ terminates. This requires a few additional mild assumptions. For all $k \in \mathbb{N}_0$, define

$$\eta_k \coloneqq \frac{\bar{e}_{\boldsymbol{A},k} - \bar{e}_{\boldsymbol{A},k+1}}{n_{k+1} - n_k} \,. \tag{4.22}$$

Assumption 4.A. *$(\bar{e}_{\boldsymbol{A},k})_{k \in \mathbb{N}_0}$ is nonincreasing and converges to 0; $n_0 = 0$ and $(n_k)_{k \in \mathbb{N}_0}$ is strictly increasing. Furthermore, the sequence $(\eta_k)_{k \in \mathbb{N}_0}$ is nonincreasing.*

Note that Assumption 4.A implies Assumption A.A from Appendix A. Let $\mathcal{M}$ denote the set of $p \in \{0, \dots, P\}$ for which $\operatorname{supp} v_{[p]} \neq \emptyset$. For all $p \in \mathcal{M}$, the sequences of costs and values from Appendix A are given by

$$c_k^p := n_k(\#\operatorname{supp} v_{[p]}) \qquad \text{and} \qquad \omega_k^p := -\bar{e}_{\boldsymbol{A},k} \left\| v_{[p]} \right\|_{\ell^2} . \tag{4.23}$$

By Assumption 4.A, $c_0^p = 0$, $(c_k^p)_{k\in\mathbb{N}_0}$ is strictly increasing and $(\omega_k^p)_{k\in\mathbb{N}_0}$ is nondecreasing for all $p \in \mathcal{M}$. Also,

$$q_k^p = \frac{\Delta\omega_k^p}{\Delta c_k^p} = \eta_k \frac{\left\| v_{[p]} \right\|_{\ell^2(\Xi_p)}}{\#\operatorname{supp} v_{[p]}} \tag{4.24}$$

is nonincreasing in k for all $p \in \mathcal{M}$.

Proposition 4.6. *For any $\boldsymbol{k}$ generated in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$, if $\boldsymbol{j} \in \mathbb{N}_0^\ell$ with $\sigma_{\boldsymbol{j}} \leq \sigma_{\boldsymbol{k}}$, then $\zeta_{\boldsymbol{j}} \geq \zeta_{\boldsymbol{k}}$. If $\boldsymbol{j} \in \mathbb{N}_0^\ell$ with $\zeta_{\boldsymbol{j}} \leq \zeta_{\boldsymbol{k}}$, then $\sigma_{\boldsymbol{j}} \geq \sigma_{\boldsymbol{k}}$.*

Proof. The assertion follows from Theorem A.5 with (4.23) and using Assumption 4.A. Note that $\sigma_{\boldsymbol{j}} \geq 0$ for all $\boldsymbol{j} \in \mathbb{N}_0^\ell$, and if $\sigma_{\boldsymbol{k}} > 0$, the second statement in Theorem A.5 applies. □

Let $(\boldsymbol{k}_i)_{i\in\mathbb{N}_0}$ denote the sequence of $\boldsymbol{k}$ generated in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ if the loop is not terminated. We abbreviate $\zeta_i := \zeta_{\boldsymbol{k}_i}$ and $\sigma_i := \sigma_{\boldsymbol{k}_i}$.

Remark 4.7. In particular, Proposition 4.6 implies convergence of the greedy subroutine in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$. Since $n_{k+1} \geq n_k + 1$ for all $k \in \mathbb{N}_0$ and $k_{i,p} = 0$ for all $i \in \mathbb{N}_0$ if $\#\operatorname{supp} v_{[p]} = 0$, σ_i goes to infinity as $i \to \infty$. Since $\zeta_{\boldsymbol{j}}$ can be made arbitrarily small for suitable $\boldsymbol{j} \in \mathbb{N}_0^\ell$, it follows that $\zeta_i \to 0$. ⌟

5. Analysis of the adaptive application routine

5.1. **Convergence analysis.** For the analysis of $\texttt{Apply}_{\boldsymbol{A}}$, we assume that the values $\bar{e}_{\boldsymbol{A},k}$ are spaced sufficiently regularly, with at most geometric convergence to 0.

Assumption 5.A. $\bar{r}_{\boldsymbol{A}} := \sup_{k\in\mathbb{N}_0} \dfrac{\bar{e}_{\boldsymbol{A},k}}{\bar{e}_{\boldsymbol{A},k+1}} < \infty$.

In particular, $\bar{e}_{\boldsymbol{A},k} > 0$ for all $k \in \mathbb{N}_0$, *i.e.* if $\boldsymbol{A}$ is sparse, this is not reflected in the bounds $\bar{e}_{\boldsymbol{A},k}$. An admissible value is $\bar{e}_{\boldsymbol{A},k} = d_{\boldsymbol{A},s} n_k^{-s}$ since for all $k \in \mathbb{N}_0$,

$$\frac{\bar{e}_{\boldsymbol{A},k}}{\bar{e}_{\boldsymbol{A},k+1}} = \left(\frac{n_{k+1}}{n_k}\right)^s \leq c_{\boldsymbol{A}}^s < \infty .$$

Lemma 5.1. *For all $i \in \mathbb{N}_0$, $\zeta_i \leq \bar{r}_{\boldsymbol{A}} \zeta_{i+1}$.*

Proof. Let $i \in \mathbb{N}_0$. Note that $\zeta_i - \zeta_{i+1} = (\bar{e}_{\boldsymbol{A},k_{q_i}} - \bar{e}_{\boldsymbol{A},k_{q_i}+1}) \left\| v_{[q_i]} \right\|_{\ell^2}$ and $\zeta_{i+1} \geq \bar{e}_{\boldsymbol{A},k_{q_i}+1} \left\| v_{[q_i]} \right\|_{\ell^2}$. Therefore,

$$\frac{\zeta_i}{\zeta_{i+1}} = 1 + \frac{\zeta_i - \zeta_{i+1}}{\zeta_{i+1}} \leq 1 + \frac{\bar{e}_{\boldsymbol{A},k_{q_i}} - \bar{e}_{\boldsymbol{A},k_{q_i}+1}}{\bar{e}_{\boldsymbol{A},k_{q_i}+1}} = \frac{\bar{e}_{\boldsymbol{A},k_{q_i}}}{\bar{e}_{\boldsymbol{A},k_{q_i}+1}} \leq \bar{r}_{\boldsymbol{A}} .$$

□

The following is adapted from [19, Thm. 4.6]. We emphasize in advance that knowledge of s and s^* is not required in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$. The algorithm satisfies Theorem 5.2 with any s^* for which $\boldsymbol{A}$ is s^*-compressible, provided that the bounds $\bar{e}_{\boldsymbol{A},k}$ from (4.12) decay at the rate implied by s^*-compressibility. We note that the constant in (5.2) may degenerate as $s \to s^*$.

Theorem 5.2. *Let $v \in \ell^2$ be finitely supported and $\epsilon > 0$. A call of* $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ *produces a finitely supported $z \in \ell^2$ with*

$$\|\boldsymbol{A}v - z\|_{\ell^2} \leq \delta + \zeta_{\boldsymbol{k}} \leq \epsilon \,. \tag{5.1}$$

If $\boldsymbol{A}$ is s^*-compressible for an $s^* \in (0, \infty]$ and $\sup_{k\in\mathbb{N}} \bar{e}_{\boldsymbol{A},k} n_k^s < \infty$ for all $s \in (0, s^*)$, then for any $s \in (0, s^*)$,

$$\#\operatorname{supp} z \leq \sigma_{\boldsymbol{k}} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \tag{5.2}$$

with a constant depending only on s, $\bar{e}_{\boldsymbol{A},0}$, $c_{\boldsymbol{A}}$, n_1, $(d_{\boldsymbol{A},\bar{s}})_{\bar{s}\in(s,s^*)}$ and $\bar{r}_{\boldsymbol{A}}$.

Proof. Convergence of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ follows from Proposition 4.6, see Remark 4.7. Then (5.1) is shown in Proposition 4.5.

Let $\boldsymbol{k} = (k_p)_{p=1}^{\ell}$ be the final value of $\boldsymbol{k}$ in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$, and $s \in (0, s^*)$. By Proposition 4.5, to prove (5.2) it suffices to show that there is a $\boldsymbol{j} \in \mathbb{N}_0^{\ell}$ with $\zeta_{\boldsymbol{j}} \leq \zeta_{\boldsymbol{k}} =: \zeta$ and $\sigma_{\boldsymbol{j}} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s}$. Then Proposition 4.6 implies

$$\#\operatorname{supp} z \leq \sigma_{\boldsymbol{k}} \leq \sigma_{\boldsymbol{j}} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \,.$$

The construction of such a $\boldsymbol{j}$ is analogous to the proof of [19, Thm. 4.6] with ζ in place of $\epsilon - \delta$. We provide it here for completeness.

Let $\tau \in (0, 2)$ be defined by $\tau^{-1} = s + \frac{1}{2}$, and let $s < \bar{s}_1 < \bar{s}_2 < s^*$. Then

$$\#\operatorname{supp} v_{[p]} \leq \#\left\{\mu \in \Xi \,;\, \left|v_\mu\right| > 2^{-p/2} \|v\|_{\ell^\infty}\right\} \lesssim 2^{p\tau/2} \|v\|_{\ell^\infty}^{-\tau} \|v\|_{\mathcal{A}^s}^{\tau} \,,$$

see *e.g.* [18]. In particular,

$$\left\|v_{[p]}\right\|_{\ell^2} \leq 2^{-p/2} \|v\|_{\ell^\infty} \sqrt{\#\operatorname{supp} v_{[p]}} \lesssim 2^{-ps\tau/2} \|v\|_{\ell^\infty}^{1-\tau/2} \|v\|_{\mathcal{A}^s}^{\tau/2} \,.$$

Let $J \geq \ell$ be the smallest integer with $\sum_{p=1}^{\ell} 2^{-(J-p)\bar{s}_1\tau/2} \left\|v_{[p]}\right\|_{\ell^2} \leq \zeta$ and let $\boldsymbol{j} = (j_p)_{p=1}^{\ell} \in \mathbb{N}_0^{\ell}$ with j_p minimal such that $\bar{e}_{\boldsymbol{A},j_p} \leq 2^{-(J-p)\bar{s}_1\tau/2}$. Then

$$\zeta_{\boldsymbol{j}} = \sum_{p=1}^{\ell} \bar{e}_{\boldsymbol{A},j_p} \left\|v_{[p]}\right\|_{\ell^2} \leq \sum_{p=1}^{\ell} 2^{-(J-p)\bar{s}_1\tau/2} \left\|v_{[p]}\right\|_{\ell^2} \leq \zeta \,.$$

It remains to be shown that $\sigma_{\boldsymbol{j}} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s}$.

If $j_p \geq 2$, since $\bar{e}_{\boldsymbol{A},j_p-1} n_{j_p-1}^{\bar{s}_2} \lesssim 1$,

$$n_{j_p} \lesssim n_{j_p-1} \lesssim \bar{e}_{\boldsymbol{A},j_p-1}^{-1/\bar{s}_2} \leq 2^{(J-p)(\bar{s}_1/\bar{s}_2)\tau/2} \,.$$

This estimate extends to $j_p \in \{0, 1\}$ since $p \leq J$. Therefore, using $\bar{s}_1 < \bar{s}_2$,

$$\begin{aligned}\sigma_{\boldsymbol{j}} = \sum_{p=1}^{\ell} n_{j_p}(\#\operatorname{supp} v_{[p]}) &\lesssim \sum_{p=1}^{\ell} 2^{(J-p)(\bar{s}_1/\bar{s}_2)\tau/2} 2^{-ps\tau/2} \|v\|_{\ell^\infty}^{-\tau} \|v\|_{\mathcal{A}^s}^{\tau} \\ &\lesssim 2^{(J-\ell)(\bar{s}_1/\bar{s}_2)\tau/2} 2^{-\ell s\tau/2} \|v\|_{\ell^\infty}^{-\tau} \|v\|_{\mathcal{A}^s}^{\tau} \leq 2^{J\tau/2} \|v\|_{\ell^\infty}^{-\tau} \|v\|_{\mathcal{A}^s}^{\tau} \,.\end{aligned}$$

Thus, the assertion reduces to $2^{J\tau/2} \|v\|_{\ell^\infty}^{-\tau} \|v\|_{\mathcal{A}^s}^{\tau} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s}$.

If $J = \ell$, by minimality of ℓ,

$$\frac{\epsilon}{2} < \bar{e}_{\boldsymbol{A},0} \left\|v - \sum_{p=1}^{\ell-1} v_{[p]}\right\|_{\ell^2} = \bar{e}_{\boldsymbol{A},0} \sqrt{\sum_{p=\ell}^{\infty} \left\|v_{[p]}\right\|_{\ell^2}^2} \lesssim \bar{e}_{\boldsymbol{A},0} 2^{-\ell s\tau/2} \|v\|_{\ell^\infty}^{1-\tau/2} \|v\|_{\mathcal{A}^s}^{\tau/2} \,.$$

If $J > \ell$, then by minimality of J, using $s < \bar{s}_1$,

$$\begin{aligned}\zeta &< \sum_{p=1}^{\ell} 2^{-(J-1-p)\bar{s}_1\tau/2} \left\|v_{[p]}\right\|_{\ell^2} \lesssim \sum_{p=1}^{\ell} 2^{-(J-1-p)\bar{s}_1\tau/2} 2^{-ps\tau/2} \|v\|_{\ell^\infty}^{1-\tau/2} \|v\|_{\mathcal{A}^s}^{\tau/2} \\ &\lesssim 2^{-(J-1-\ell)\bar{s}_1\tau/2} 2^{-\ell s\tau/2} \|v\|_{\ell^\infty}^{1-\tau/2} \|v\|_{\mathcal{A}^s}^{\tau/2} \le 2^{-(J-1)s\tau/2} \|v\|_{\ell^\infty}^{1-\tau/2} \|v\|_{\mathcal{A}^s}^{\tau/2} \ .\end{aligned}$$

Lemma 5.1 implies $\epsilon \le \bar{r}_{\boldsymbol{A}} \zeta$. Therefore, in both cases,

$$\epsilon \lesssim 2^{-Js\tau/2} \|v\|_{\ell^\infty}^{1-\tau/2} \|v\|_{\mathcal{A}^s}^{\tau/2} \ ,$$

or equivalently,

$$2^{J\tau/2} \|v\|_{\ell^\infty}^{-\tau} \|v\|_{\mathcal{A}^s}^{\tau} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \ ,$$

which completes the proof. □

It is known that s^*-compressible operators $\boldsymbol{A}$ map $\mathcal{A}^s$ boundedly into $\mathcal{A}^s$ for all $s \in (0, s^*)$, see [9, Proposition 3.8]. Theorem 5.2 implies that this carries over to the approximate multiplication routine $\texttt{Apply}_{\boldsymbol{A}}$.

Corollary 5.3. *Let $\boldsymbol{A}$ be s^*-compressible for some $s^* \in (0, \infty]$, and assume that for all $s \in (0, s^*)$, $\sup_{k\in\mathbb{N}} \bar{e}_{\boldsymbol{A},k} n_k^s < \infty$. Then for any $s \in (0, s^*)$ there is a constant C depending only on s, $\bar{e}_{\boldsymbol{A},0}$, $c_{\boldsymbol{A}}$, n_1, $(d_{\boldsymbol{A},\bar{s}})_{\bar{s}\in(s,s^*)}$ and $\bar{r}_{\boldsymbol{A}}$ such that for all $v \in \mathcal{A}^s$ and all $\epsilon > 0$, the output z of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ satisfies*

$$\|z\|_{\mathcal{A}^s} \le C \|v\|_{\mathcal{A}^s} \ . \tag{5.3}$$

Proof. Let z be the output of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ for some $v \in \mathcal{A}^s$ and some $\epsilon > 0$, and define $w := \boldsymbol{A}v$. By [9, Proposition 3.8], $w \in \mathcal{A}^s$, and $\|w\|_{\mathcal{A}^s} \lesssim \|v\|_{\mathcal{A}^s}$. Therefore, it suffices to show $\|z\|_{\mathcal{A}^s} \lesssim \|w\|_{\mathcal{A}^s}$. Since z is finitely supported, $z \in \mathcal{A}^s$. Let $N := \#\operatorname{supp} z$. Theorem 5.2 implies

$$\|w - z\|_{\ell^2} \lesssim \|w\|_{\mathcal{A}^s} N^{-s} \ .$$

For any $n \ge N$, $P_n(z) = z$, and thus $(n+1)^s \|z - P_n(z)\|_{\ell^2} = 0$. Let $n \le N - 1$ and $z_n \in \ell^2$ with $\#\operatorname{supp} z_n \le n$. Then

$$(n+1)^s \|z - z_n\|_{\ell^2} \le (n+1)^s \|w - z\|_{\ell^2} + (n+1)^s \|w - z_n\|_{\ell^2} \ .$$

The first term is bounded by

$$(n+1)^s \|w - z\|_{\ell^2} \lesssim (n+1)^s N^{-s} \|w\|_{\mathcal{A}^s} \lesssim \|w\|_{\mathcal{A}^s} \ .$$

Taking the infimum over z_n with $\#\operatorname{supp} z_n \le n$, we have

$$(n+1)^s \|z - P_n(z)\|_{\ell^2} \lesssim \|w\|_{\mathcal{A}^s} + (n+1)^s \inf_{z_n} \|w - z_n\|_{\ell^2} \lesssim \|w\|_{\mathcal{A}^s} \ .$$

The assertion follows by taking the supremum over $n \in \mathbb{N}_0$. □

5.2. **Complexity analysis.** By (4.16), the number of operations and storage locations required by $\texttt{BucketSort}$ in a call of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ is bounded by

$$\begin{aligned}&\#\operatorname{supp} v + \max(1, \lceil \log(2\bar{e}_{\boldsymbol{A},0} \|v\|_{\ell^\infty} \sqrt{\#\operatorname{supp} v}/\epsilon)\rceil) \\ &\qquad \lesssim 1 + \#\operatorname{supp} v + \log(\epsilon^{-1} \|v\|_{\ell^\infty}) \ . \end{aligned} \tag{5.4}$$

The value of ℓ can be determined with at most $\#\operatorname{supp} v$ operations. We assume that the values of $\left\|v_{[p]}\right\|_{\ell^2(\Xi_p)}$ are known from the computation of ℓ. Then by Proposition A.6, initialization of the greedy subroutine requires $O(\ell \log \ell)$ operations, and each iteration requires $O(1 + \log \ell)$ operations *e.g.* if a tree data structure is used

for $\mathcal{N}$ from Section A.3. As $\|\boldsymbol{k}\|_{\ell^1}$ iterations are performed if $\boldsymbol{k} = (k_p)_{p=1}^{\ell}$ is the final value of $\boldsymbol{k}$ in $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$, the total cost of determining ℓ and $\boldsymbol{k}$ is on the order of

$$\#\operatorname{supp} v + \ell \log^+ \ell + (1 + \log^+ \ell) \sum_{p=1}^{\ell} k_p \;, \tag{5.5}$$

where $\log^+ x \coloneqq \log(\max(x, 1))$. Since $\ell \le P$, (4.15) implies

$$\ell \lesssim 1 + \log^+(\#\operatorname{supp} v) + \log^+(\epsilon^{-1} \|v\|_{\ell^\infty}) \;. \tag{5.6}$$

Finally, the number of arithmetic operations required by the last step of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ is bounded by

$$\sigma_{\boldsymbol{k}} = \sum_{p=1}^{\ell} n_{k_p}(\#\operatorname{supp} v_{[p]}) \;, \tag{5.7}$$

and this value is optimal in the sense of Proposition 4.6. If $\boldsymbol{A}$ is s^*-computable for any $s^* \in (0, \infty]$, then (5.7) includes the assembly costs of $\boldsymbol{A}_{k_p}$.

Theorem 5.4. *Let $v \in \ell^2$ be finitely supported and $\epsilon > 0$. If $\boldsymbol{A}$ is s^*-computable for an $s^* \in (0, \infty]$ and $\sup_{k \in \mathbb{N}} \bar{e}_{\boldsymbol{A},k} n_k^s < \infty$ for all $s \in (0, s^*)$, then for any $s \in (0, s^*)$, the number of operations and storage locations required by $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ is less than a multiple of*

$$1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \left(1 + \log^+ \log^+\left(\#\operatorname{supp} v + \epsilon^{-1} \|v\|_{\ell^\infty}\right)\right) \tag{5.8}$$

with a constant depending only on s, $\bar{e}_{\boldsymbol{A},0}$, $c_{\boldsymbol{A}}$, n_1, $(d_{\boldsymbol{A},\bar{s}})_{\bar{s} \in (s,s^*)}$, $\bar{r}_{\boldsymbol{A}}$ and $b_{\boldsymbol{A}}$. The double logarithmic term in (5.8) *is due only to the greedy subroutine and does not apply to the storage requirements.*[1]

Proof. We first note that

$$\log(\epsilon^{-1} \|v\|_{\ell^\infty}) \lesssim \epsilon^{-1/s} \|v\|_{\ell^\infty}^{1/s} \le \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \;.$$

Therefore and by (5.4), the cost of `BucketSort` is less than

$$1 + \#\operatorname{supp} v + \log(\epsilon^{-1} \|v\|_{\ell^\infty}) \lesssim 1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \;.$$

The cost of the last step of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ is $\sigma_{\boldsymbol{k}}$, which in Theorem 5.2 is bounded by

$$\sigma_{\boldsymbol{k}} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \;.$$

The cost of the rest of $\texttt{Apply}_{\boldsymbol{A}}[v, \epsilon]$ is given in (5.5). By (5.6), for $\chi > 1$,

$$\begin{aligned} \ell \log \ell \lesssim \ell^\chi &\lesssim 1 + \log(\#\operatorname{supp} v)^\chi + \log(\epsilon^{-1} \|v\|_{\ell^\infty})^\chi \\ &\lesssim 1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\ell^\infty}^{1/s} \le 1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \;. \end{aligned}$$

Since

$$\ell \lesssim 1 + \log(\#\operatorname{supp} v) + \log(\epsilon^{-1} \|v\|_{\ell^\infty}) \lesssim 1 + \log(\#\operatorname{supp} v + \epsilon^{-1} \|v\|_{\ell^\infty}) \;,$$

we have

$$\log \ell \le C + \log(1 + \log(\#\operatorname{supp} v + \epsilon^{-1} \|v\|_{\ell^\infty})) \lesssim 1 + \log\log(\#\operatorname{supp} v + \epsilon^{-1} \|v\|_{\ell^\infty}) \;.$$

Finally, since $k \le n_k$ for all $k \in \mathbb{N}_0$ and $k_p = 0$ if $\#\operatorname{supp} v_{[p]} = 0$,

$$\sum_{p=1}^{\ell} k_p \le \sum_{p=1}^{\ell} n_{k_p}(\#\operatorname{supp} v_{[p]}) = \sigma_{\boldsymbol{k}} \lesssim \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \;.$$

□

[1] As above, $\log^+ x \coloneqq \log(\max(x, 1))$.

Remark 5.5. The double logarithmic term in (5.8) can be dropped under mild conditions. If $n_k \gtrsim k^\alpha$ for an $\alpha > 1$, then by Hölder's inequality,

$$\sum_{p=1}^{\ell} k_p \lesssim \sum_{p=1}^{\ell} n_{k_p}^{1/\alpha} \le \left(\sum_{p=1}^{\ell} n_{k_p}\right)^{1/\alpha} \ell^{\frac{\alpha-1}{\alpha}} .$$

Furthermore, for a $\chi > 1$, as in the proof of Theorem 5.4,

$$\ell^{\frac{\alpha-1}{\alpha}} \log \ell \lesssim (\ell^\chi)^{\frac{\alpha-1}{\alpha}} \lesssim \left(1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s}\right)^{\frac{\alpha-1}{\alpha}} .$$

It follows that

$$\log \ell \sum_{p=1}^{\ell} k_p \lesssim \sigma_{\boldsymbol{k}}^{1/\alpha} \left(1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s}\right)^{\frac{\alpha-1}{\alpha}} \lesssim 1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} ,$$

and (5.8) can be replaced by

$$1 + \#\operatorname{supp} v + \epsilon^{-1/s} \|v\|_{\mathcal{A}^s}^{1/s} \tag{5.9}$$

in Theorem 5.4, with a constant that also depends on α. The assumption $n_k \gtrsim k^\alpha$ is generally not restrictive, since by (4.1), n_k may grow exponentially for an s^*-compressible operator. ⌟

6. Computation of spectral norms by the power method

6.1. **Estimation of errors in sparse approximations of s^*-compressible operators.** The routine $\texttt{Apply}_{\boldsymbol{A}}$ in Section 4.2 makes explicit use of bounds $\bar{e}_{\boldsymbol{A},k}$ on the errors $\|A - A_k\|_{\ell^2\to\ell^2}$, where A_k is an n_k-sparse approximation of an operator $A \in \mathcal{L}(\ell^2)$, see (4.12). Such bounds are derived *e.g.* in [33, 3] for a large class of operators in wavelet bases. However, these estimates only hold up to an unspecified constant.

We suggest a power method for numerically approximating $\|A - A_k\|_{\ell^2\to\ell^2}$, which is equal to the square root of the spectral radius of the bounded positive symmetric operator $(A - A_k)^*(A - A_k)$ on ℓ^2.

Remark 6.1. If A is s^*-compressible with a sequence $(A_j)_{j\in\mathbb{N}}$ of n_j-sparse approximations, then $A - A_k$ is also s^*-compressible with approximations $(A_{k+j} - A_k)_{j\in\mathbb{N}}$. We have

$$\big\|(A - A_k) - (A_{k+j} - A_k)\big\|_{\ell^2\to\ell^2} = \big\|A - A_{k+j}\big\|_{\ell^2\to\ell^2} = e_{\boldsymbol{A},k+j} \le d_{\boldsymbol{A},s} n_{k+j}^{-s} . \tag{6.1}$$

Furthermore, $A_{k+j} - A_k$ is at most $(n_{k+j} + n_k)$-sparse, which implies $d_{\boldsymbol{A}-\boldsymbol{A}_k,s} \le 2^s d_{\boldsymbol{A},s}$. If the nonzero entries of A_k are also nonzero for A_{k+j}, then $A_{k+j} - A_k$ is n_{k+j}-sparse, or even $(n_{k+j} - n_k)$-sparse if the values of these entries coincide. In either case, $d_{\boldsymbol{A}-\boldsymbol{A}_k,s} \le d_{\boldsymbol{A},s}$. Similar considerations lead to $c_{\boldsymbol{A}-\boldsymbol{A}_k} \le 2c_{\boldsymbol{A}}^2/(c_{\boldsymbol{A}} - 1)$. ⌟

6.2. **Analysis of an idealized iteration.** Let $A \in \mathcal{L}(\ell^2)$ be a positive symmetric operator. The power method successively approximates the spectral radius $r_{\boldsymbol{A}}$ of A by Rayleigh quotients

$$R_n := \frac{\left(A^{n+1}v, A^n v\right)_{\ell^2}}{\|A^n v\|_{\ell^2}^2} = \frac{\left(A^{2n+1}v, v\right)_{\ell^2}}{(A^{2n}v, v)_{\ell^2}} , \quad n \in \mathbb{N} , \tag{6.2}$$

for some starting value $v \in \ell^2$.

Remark 6.2. The classical analysis of the power method in a finite dimensional setting makes use of the gap between the two largest eigenvalues. For real symmetric matrices, the convergence to the largest eigenvalue is quadratic in the quotient of the two largest eigenvalues, and the convergence to a corresponding eigenvector is linear. In our infinite dimensional setting, the spectrum does not have to consist

of discrete points, and there can be points in the spectrum with no corresponding eigenvector. Also, it no longer makes sense to consider the difference between the two largest eigenvalues since the maximum of the spectrum may also be a limit point of the spectrum. These fundamental differences call for an equally different analysis. ⌟

Theorem 6.3. *For appropriate starting values $v \in \ell^2$ and any $\vartheta \in (0,1)$, there is a constant $c_{v,\vartheta} > 0$ such that*

$$r_A \geq R_n \geq \vartheta r_A (1 - c_{v,\vartheta} n^{-1}) \qquad \forall n \in \mathbb{N} \,. \tag{6.3}$$

In particular, $R_n \to r_A$.

Proof. We note that $R_n \leq r_A$ for all $n \in \mathbb{N}$ by definition. Due to the spectral theorem for bounded symmetric operators, there is a σ-finite measure μ on some domain S and a unitary map $U\colon L^2_\mu(S) \to \ell^2$ such that

$$U^* A U \varphi = f\varphi \qquad \forall \varphi \in L^2_\mu(S) \;,$$

where $f \in L^\infty_\mu(S)$ with $f \geq 0$ and $r_A = \|f\|_{L^\infty_\mu(S)}$. We assume without loss of generality that $\|v\|_{\ell^2} = 1$ and define $\varphi := U^* v$. Then the Rayleigh quotients (6.2) are

$$R_n = \frac{\int_S f^{2n+1} |\varphi|^2 \,\mathrm{d}\mu}{\int_S f^{2n} |\varphi|^2 \,\mathrm{d}\mu} = \frac{\int_S f^{2n+1} \,\mathrm{d}\mu_\varphi}{\int_S f^{2n} \,\mathrm{d}\mu_\varphi}$$

for the probability measure $\mathrm{d}\mu_\varphi := |\varphi|^2 \,\mathrm{d}\mu$. By Jensen's inequality, $\|f\|_{L^{2n+1}_{\mu_\varphi}(S)} \geq \|f\|_{L^{2n}_{\mu_\varphi}(S)}$, and thus

$$R_n \geq \frac{\left(\int_S f^{2n} \,\mathrm{d}\mu_\varphi \right)^{\frac{2n+1}{2n}}}{\int_S f^{2n} \,\mathrm{d}\mu_\varphi} = \left(\int_S f^{2n} \,\mathrm{d}\mu_\varphi \right)^{\frac{1}{2n}} = \|f\|_{L^{2n}_{\mu_\varphi}(S)} \;.$$

Since $\|f\|_{L^p_{\mu_\varphi}(S)} \to \|f\|_{L^\infty_{\mu_\varphi}(S)}$ as $p \to \infty$, convergence of R_n to r_A follows, provided that

$$\operatorname*{ess\,sup}_{x \in \operatorname{supp} \varphi} f(x) = \operatorname*{ess\,sup}_{x \in S} f(x) \,. \tag{6.4}$$

We estimate $\|f\|_{L^{2n}_{\mu_\varphi}(S)}$ from below in order to get a convergence rate. Let $\vartheta \in (0,1)$. Then Markov's inequality implies

$$\|f\|_{L^{2n}_{\mu_\varphi}(S)} \geq \vartheta \|f\|_{L^\infty_{\mu_\varphi}(S)} \kappa^{1/2n} \quad \text{for} \quad \kappa := \mu_\varphi\Big(\Big\{ x \in S \,;\, f(x) \geq \vartheta \|f\|_{L^\infty_{\mu_\varphi}(S)} \Big\} \Big) \in (0,1] \,.$$

Furthermore, by the fundamental theorem of calculus,

$$\kappa^{1/2n} \geq 1 - (1-\kappa) \frac{1}{2n} \kappa^{\frac{1}{2n}-1} \geq 1 - \frac{1-\kappa}{2\kappa} \frac{1}{n} \,. \qquad \square$$

The proof of Theorem 6.3 clarifies the conditions on the starting value v: It must satisfy (6.4) for $\varphi = U^* v$ and f as in the proof. This condition is analogous to the assumption that the starting vector in a finite dimensional power method is not orthogonal to the eigenspace associated to the largest eigenvalue. We expect round-off errors to make this condition irrelevant for numerical computations.

6.3. **A practical algorithm.** The Rayleigh quotients (6.2) cannot be computed exactly since the operator $\boldsymbol{A}$ cannot by applied exactly. We suggest an approximate adaptive procedure for evaluating $\boldsymbol{Av}$ similar to the routine $\mathtt{Apply}_{\boldsymbol{A}}$ from Section 4.2. To this end, we assume that for all $k \in \mathbb{N}_0$, $\boldsymbol{A}_k$ is an n_k-sparse approximation of $\boldsymbol{A}$, with $n_0 = 0$, $n_{k+1} \geq n_k + 1$ for all $k \in \mathbb{N}_0$ and

$$\|\boldsymbol{A} - \boldsymbol{A}_k\|_{\ell^2 \to \ell^2} \leq C\tilde{e}_{\boldsymbol{A},k} \tag{6.5}$$

for a constant C. We emphasize that this assumption is weaker than (4.12) since the constant C need not be known, and our algorithm does not depend on this constant. If it is known that $\boldsymbol{A}$ is s^*-compressible, then we may set $\tilde{e}_{\boldsymbol{A},k} \coloneqq n_k^{-s}$ for any $s \in (0, s^*)$.

Let $\boldsymbol{v} = (v_\mu)_{\mu\in\mathbb{N}}$ be a finitely supported sequence. We consider a sorting routine

$$\mathtt{Sort}[\boldsymbol{v}] \mapsto (\mu_i)_{i=1}^M \tag{6.6}$$

with $M \coloneqq \#\operatorname{supp} \boldsymbol{v}$ and such that $(|v_{\mu_i}|)_{i=1}^M$ is a decreasing rearrangement of $(|v_\mu|)_{\mu\in\mathbb{N}}$. To approximate $\boldsymbol{Av}$, we apply either $\boldsymbol{A}_k$ or a better approximation of $\boldsymbol{A}$ to the first m_k terms of this decreasing rearrangement, *i.e.* we apply $\boldsymbol{A}_k$ to $\boldsymbol{v}$ restricted to the set $\{\mu_i\,;\, m_{k+1} + 1 \leq i \leq m_k\}$. For any nonincreasing sequence $\boldsymbol{m} = (m_k)_{k=1}^\infty$, the number of multiplications performed in this approximate application of $\boldsymbol{A}$ is at most

$$\sigma_{\boldsymbol{m}} \coloneqq \sum_{k=1}^\infty n_k(m_k - m_{k+1}) = \sum_{k=1}^\infty (n_k - n_{k-1})m_k\,, \tag{6.7}$$

and the error is bounded by

$$\chi_{\boldsymbol{m}} \coloneqq \sum_{k=1}^\infty \tilde{e}_{\boldsymbol{A},k}\Bigg(\sum_{i=m_{k+1}+1}^{m_k} \left|v_{\mu_i}\right|^2\Bigg)^{1/2}. \tag{6.8}$$

Even though $\chi_{\boldsymbol{m}}$ is not of the form (A.2), and thus Appendix A does not apply, we use a greedy algorithm to adaptively select a sequence $\boldsymbol{m}$. The routine $\mathtt{NextOptInf}$ from Appendix A.3 easily extends to the present setting, and its output $\boldsymbol{m}$ is assured to be nonincreasing.

$\mathtt{NApply}_{\boldsymbol{A}}[\boldsymbol{v}, N] \mapsto \boldsymbol{z}$

$(\mu_i)_{i=1}^M \longleftarrow \mathtt{Sort}[\boldsymbol{v}]$
$\boldsymbol{m} = (m_k)_{k=1}^\infty \longleftarrow (0)_{k=1}^\infty$
$\hat{\boldsymbol{m}} = (\hat{m}_k)_{k=1}^\infty \longleftarrow (0)_{k=1}^\infty$
while $\sigma_{\hat{\boldsymbol{m}}} \leq N$ **do**
  $\boldsymbol{m} \longleftarrow \hat{\boldsymbol{m}}$
  $\hat{\boldsymbol{m}} \longleftarrow \mathtt{NextOptInf}[\boldsymbol{m}]$ with objective $-\chi_{\boldsymbol{m}}$ and cost $\sigma_{\boldsymbol{m}}$
forall $k \in \mathbb{N}$ **do** $\Xi_k \longleftarrow \{\mu_i\,;\, m_{k+1} + 1 \leq i \leq m_k\}$
$\boldsymbol{z} \longleftarrow \displaystyle\sum_{k=1}^\infty \boldsymbol{A}_k\boldsymbol{v}|_{\Xi_k}$

The routine $\mathtt{NApply}_{\boldsymbol{A}}$ does not ensure a fixed error, contrary to $\mathtt{Apply}_{\boldsymbol{A}}$. This would not be possible due to the unknown constant in the estimate (6.5). Instead, $\mathtt{NApply}_{\boldsymbol{A}}$ limits the computational cost of the approximate multiplication. It can be thought of as an adaptively constructed matrix representation of $\boldsymbol{A}$ of size $N \times M$.

Remark 6.4. By construction, $\sigma_{\boldsymbol{m}} \leq N$ for the final value of $\boldsymbol{m}$ in $\mathtt{NApply}_{\boldsymbol{A}}$. This implies that no more than N multiplications are performed in the computation of $\boldsymbol{z}$ in the final step of $\mathtt{NApply}_{\boldsymbol{A}}$, and thus $\#\operatorname{supp} \boldsymbol{z} \leq N$. ⌟

Remark 6.5. The exact sorting in the first step of $\texttt{NApply}_{\boldsymbol{A}}$ uses $O(M \log M)$ operations. If n_k increases exponentially in k and $\tilde{e}_{\boldsymbol{A},k}$ decreases exponentially in k, then at most $O(N \log N)$ steps are required in the subsequent greedy algorithm. By Proposition A.7, these can be realized at a computational cost of $O(N(\log N)^2)$. Finally, as noted in Remark 6.4, the actual computation of z uses $O(N)$ operations. ⌟

Starting from an arbitrary finitely supported nonzero $v \in \ell^2$, $\texttt{SpecRad}_{\boldsymbol{A}}$ iteratively uses $\texttt{NApply}_{\boldsymbol{A}}$ to approximate multiplications by $\boldsymbol{A}$ in the Rayleigh quotients (6.2). As a termination criterion, lacking alternatives, we simply compare two consecutive approximations of the spectral radius of $\boldsymbol{A}$.

$\texttt{SpecRad}_{\boldsymbol{A}}[v, N, \epsilon] \mapsto \varrho$

$\varrho \longleftarrow \infty$
$v \longleftarrow v / \|v\|_{\ell^2}$
repeat
  $\varrho_0 \longleftarrow \varrho$
  $w \longleftarrow \texttt{NApply}_{\boldsymbol{A}}[v, N]$
  $\varrho \longleftarrow w \cdot v$
  $v \longleftarrow w / \|w\|_{\ell^2}$
until $\left|\varrho - \varrho_0\right| \leq \epsilon \varrho$

Remark 6.6. Since N is held constant throughout $\texttt{SpecRad}_{\boldsymbol{A}}$, assuming $\#\operatorname{supp} v \leq N$ for the starting value of v, each step of $\texttt{SpecRad}_{\boldsymbol{A}}$ has a computational cost of $O(N(\log N)^2)$ due to Remark 6.5. Consequently, the choice of v is not particularly important—a poor choice is likely to be compensated by a few steps of the iteration, and the cost of subsequent steps is not affected. Note that the situation would be different if $\texttt{Apply}_{\boldsymbol{A}}$ were used in place of $\texttt{NApply}_{\boldsymbol{A}}$. ⌟

Remark 6.7. In order to compute the spectral radius of $\boldsymbol{A}^*\boldsymbol{A}$ for an operator $\boldsymbol{A} \in \mathcal{L}(\ell^2)$ that is not positive, instead of constructing sparse approximations of $\boldsymbol{A}^*\boldsymbol{A}$, the algorithm $\texttt{SpecRad}_{\boldsymbol{A}^*\boldsymbol{A}}$ can be used with $\texttt{NApply}_{\boldsymbol{A}^*\boldsymbol{A}}[v, N]$ replaced by

$$\texttt{NApply}_{\boldsymbol{A}^*}[\texttt{NApply}_{\boldsymbol{A}}[v, N], N] \,. \tag{6.9}$$

All vectors appearing in the iteration are still ensured to have at most N nonzero entries, and Remark 6.6 still holds. This can be used in the setting of Section 6.1, with $\boldsymbol{A} - \boldsymbol{A}_k$ in place of $\boldsymbol{A}$. ⌟

7. Sparse approximations of discrete random operators

7.1. **Definition of approximations.** We return to the discrete random operator $\boldsymbol{A}$ defined in (2.12). Let $(\boldsymbol{D}_j)_{j \in \mathbb{N}_0}$ and $(\boldsymbol{R}_{m,j})_{j \in \mathbb{N}_0}$ be approximating sequences of $\boldsymbol{D}$ and $\boldsymbol{R}_m$, respectively, such that $\boldsymbol{D}_j$ is $n_{0,j}$-sparse and $\boldsymbol{R}_{m,j}$ is $n_{m,j}$-sparse, $m \in \mathbb{N}$. We assume $n_{m,0} = 0$ and $n_{m,j}$ is strictly increasing in j for all $m \in \mathbb{N}_0$. Furthermore, let

$$\left\|\boldsymbol{D} - \boldsymbol{D}_j\right\|_{\ell^2(\Xi) \to \ell^2(\Theta)} \leq \bar{e}_{0,j} \quad \text{and} \quad \left\|\boldsymbol{R}_m - \boldsymbol{R}_{m,j}\right\|_{\ell^2(\Xi) \to \ell^2(\Theta)} \leq \bar{e}_{m,j} \tag{7.1}$$

for all $m \in \mathbb{N}$. Such bounds can be computed numerically by a power method, see Section 6.

For all finitely supported sequences $\boldsymbol{j} \coloneqq (j_m)_{m \in \mathbb{N}_0}$ in $\mathbb{N}_0$, define the operator

$$\boldsymbol{A}_{\boldsymbol{j}} \coloneqq \boldsymbol{I} \otimes \boldsymbol{D}_{j_0} + \sum_{m=1}^{\infty} \boldsymbol{K}_m \otimes \boldsymbol{R}_{m,j_m} \,. \tag{7.2}$$

Let $\sigma_m \coloneqq 2$ if the distribution π_m is symmetric, and $\sigma_m \coloneqq 3$ otherwise. We set $\sigma_0 \coloneqq 1$ and define $\bar{n}_{m,j} \coloneqq \sigma_m n_{m,j}$ for $m \in \mathbb{N}_0$. Then for all $j \in \mathbb{N}_0$, $\boldsymbol{I} \otimes \boldsymbol{D}_j$ is $\bar{n}_{0,j}$-sparse and $\boldsymbol{K}_m \otimes \boldsymbol{R}_{m,j}$ is $\bar{n}_{m,j}$-sparse, $m \in \mathbb{N}$.

Lemma 7.1. *For any finitely supported sequence $\boldsymbol{j} = (j_m)_{m\in\mathbb{N}_0}$ in $\mathbb{N}_0$, $\boldsymbol{A}_{\boldsymbol{j}}$ is $N_{\boldsymbol{j}}$-sparse for*

$$N_{\boldsymbol{j}} \coloneqq \sum_{m=0}^{\infty} \bar{n}_{m,j_m} \,, \tag{7.3}$$

and

$$\left\|\boldsymbol{A} - \boldsymbol{A}_{\boldsymbol{j}}\right\|_{\ell^2(\Lambda\times\Xi)\to\ell^2(\Lambda\times\Theta)} \le \sum_{m=0}^{\infty} \bar{e}_{m,j_m} =: \bar{e}_{\boldsymbol{A},\boldsymbol{j}} \,. \tag{7.4}$$

Proof. The first part of the assertion follows by construction since $\boldsymbol{I}$ is 1-sparse and $\boldsymbol{K}_m$ is σ_m-sparse for all $m \in \mathbb{N}$. Equation (7.4) is a consequence of Lemma 2.2 and Proposition 2.3. □

We use the greedy algorithm from Appendix A to select specific $\boldsymbol{j}$ in (7.2). The cost $c_{\boldsymbol{j}}$ and objective $\omega_{\boldsymbol{j}}$ are given by

$$c_{\boldsymbol{j}} \coloneqq N_{\boldsymbol{j}} = \sum_{m=0}^{\infty} \bar{n}_{m,j_m} \qquad \text{and} \qquad \omega_{\boldsymbol{j}} \coloneqq -\bar{e}_{\boldsymbol{A},\boldsymbol{j}} = \sum_{m=0}^{\infty} -\bar{e}_{m,j_m} \,. \tag{7.5}$$

We initialize $\boldsymbol{j}_0 \coloneqq \boldsymbol{0} \in \mathbb{N}_0^{\mathbb{N}_0}$ and construct $(\boldsymbol{j}_k)_{k\in\mathbb{N}_0}$ recursively by

$$\boldsymbol{j}_{k+1} \coloneqq \texttt{NextOptInf}[\boldsymbol{j}_k] \,, \quad k \in \mathbb{N}_0 \,, \tag{7.6}$$

using (7.5). Then

$$\boldsymbol{A}_k \coloneqq \boldsymbol{A}_{\boldsymbol{j}_k} \,, \quad k \in \mathbb{N}_0 \,, \tag{7.7}$$

defines a sequence of approximations of $\boldsymbol{A}$. By Lemma 7.1, $\boldsymbol{A}_k$ is $N_k \coloneqq N_{\boldsymbol{j}_k}$-sparse and its distance to $\boldsymbol{A}$ is bounded by $\bar{e}_{\boldsymbol{A},k} \coloneqq \bar{e}_{\boldsymbol{A},\boldsymbol{j}_k}$.

Under mild assumptions, (7.7) defines the optimal N_k-sparse approximation of $\boldsymbol{A}$ given the bounds (7.1) and the estimates in Lemma 7.1.

Assumption 7.A. *For all $m \in \mathbb{N}$, $n_{m,0} = 0$ and the $(n_{m,j})_{j\in\mathbb{N}_0}$ is strictly increasing. The sequence $(\bar{e}_{m,0})_{m\in\mathbb{N}}$ is in ℓ^1, and $(\bar{e}_{m,j})_{j\in\mathbb{N}_0}$ is nonincreasing. Furthermore, if $i \ge j$, then*

$$\frac{-(\bar{e}_{m,i+1} - \bar{e}_{m,i})}{\bar{n}_{m,i+1} - \bar{n}_{m,i}} \le \frac{-(\bar{e}_{m,j+1} - \bar{e}_{m,j})}{\bar{n}_{m,j+1} - \bar{n}_{m,j}} \,, \tag{7.8}$$

and $\bar{n}_{m,1}^{-1}(\bar{e}_{m,1} - \bar{e}_{m,0})$ is nonincreasing in m.

Corollary 7.2. *For all $k \in \mathbb{N}_0$, $\boldsymbol{j}_k$ minimizes the error bound $\bar{e}_{\boldsymbol{A},\boldsymbol{j}}$ among all finitely supported sequences $\boldsymbol{j}$ in $\mathbb{N}_0$ with sparsity bound $N_{\boldsymbol{j}} \le N_k$. Furthermore, if $\bar{e}_{\boldsymbol{A},k} \ne 0$, then $\boldsymbol{j}_k$ minimizes $N_{\boldsymbol{j}}$ among all $\boldsymbol{j}$ with $\bar{e}_{\boldsymbol{A},\boldsymbol{j}} \le \bar{e}_{\boldsymbol{A},k}$.*

Proof. The assertion follows from Theorem A.5, see Remark A.1, since Assumption 7.A implies Assumption A.A for (7.5). □

7.2. **Numerical computation.** We consider the complexity of a routine $\texttt{Build}_{\boldsymbol{A}}$ as in Def. 4.4 for constructing columns of $\boldsymbol{A}_k$, interpreted as bi-infinite matrices. To this end, we assume that such assembly routines are available for $\boldsymbol{D}$ and $\boldsymbol{R}_m$, $m \in \mathbb{N}$. More specifically, the routines

$$\begin{aligned}
\texttt{Build}_0[j,\iota] &\mapsto \left[(\lambda_i)_{i=1}^{n_{0,j}}, (d_i)_{i=1}^{n_{0,j}}\right] \,, \\
\texttt{Build}_m[j,\iota] &\mapsto \left[(\lambda_i)_{i=1}^{n_{m,j}}, (r_i^m)_{i=1}^{n_{m,j}}\right] \,, \quad m \in \mathbb{N} \,,
\end{aligned}$$

construct all nonzero elements of the ι-th column of $\boldsymbol{D}_j$ and $\boldsymbol{R}_{m,j}$, respectively, using no more than $b_m n_{m,j}$ arithmetic operations and storage locations for a constant b_m independent of j and ι.

$\texttt{Build}_{\boldsymbol{A}}[k, (\mu, \iota)] \mapsto \left[((\nu_i, \lambda_i))_{i=1}^{N_k}, (a_i)_{i=1}^{N_k}\right]$

$\left[(\lambda_i)_{i=1}^{n_{0,j_{k,0}}}, (d_i)_{i=1}^{n_{0,j_{k,0}}}\right] \longleftarrow \texttt{Build}_0[j_{k,0}, \iota]$
for $i = 1, \ldots, n_{0,j_{k,0}}$ **do** $[(\nu_i, \lambda_i), a_i] \longleftarrow [(\mu, \lambda_i), d_i]$
$n \longleftarrow n_{0,j_{k,0}}$
for $m \in \mathbb{N}$; $j_{k,m} \geq 1$ **do**
  $\left[(\lambda_i)_{i=1}^{n_{m,j_{k,m}}}, (r_i^m)_{i=1}^{n_{m,j_{k,m}}}\right] \longleftarrow \texttt{Build}_m[j_{k,m}, \iota]$
  $t \longleftarrow 0$
  for $i = 1, \ldots, n_{m,j_{k,m}}$ **do**
    $(\nu_{n+t+1}, \lambda_{n+t+1}) \longleftarrow (\mu + \epsilon_m, \lambda_i)$
    $a_{n+t+1} \longleftarrow \beta^m_{\mu_m+1} r_i^m$
    if $\mu_m \geq 1$ **then**
      $(\nu_{n+t+2}, \lambda_{n+t+2}) \longleftarrow (\mu - \epsilon_m, \lambda_i)$
      $a_{n+t+2} \longleftarrow \beta^m_{\mu_m} r_i^m$
    if $\sigma_m = 3$ **then**
      $(\nu_{n+t+3}, \lambda_{n+t+3}) \longleftarrow (\mu, \lambda_i)$
      $a_{n+t+3} \longleftarrow \alpha^m_{\mu_m} r_i^m$
    $t \longleftarrow t + \sigma_m$
  $n \longleftarrow n + \sigma_m n_{m,j_{k,m}}$

Lemma 7.3. *The number of arithmetic operations and storage locations required by a call of* $\texttt{Build}_{\boldsymbol{A}}[k, (\mu, \iota)]$ *is bounded uniformly in* k *by*

$$N_k + \sum_{m=0}^{\infty} b_m n_{m,j_{k,m}} \, .$$

Proof. This is a direct consequence of the assumptions on $\texttt{Build}_m$, $m \in \mathbb{N}_0$. □

Remark 7.4. It is often necessary to construct $\boldsymbol{j}_k$ before calling $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$, for example to determine N_k and $\bar{e}_{\boldsymbol{A},k}$. In this case, we can assume $\boldsymbol{j}_k$ to be readily available in $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$. Otherwise, `NextOptInf` from Appendix A can be used to compute $\boldsymbol{j}_k$ in the first call of $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$. If this is done directly for an arbitrary $k \in \mathbb{N}_0$, it adds $O(k \log(k))$ to the complexity of $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$ even if $\mathcal{N}$ is realized by a tree data structure, which may dominate *e.g.* if $N_k \lesssim k$. However, if $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$ is called successively for $k \in \mathbb{N}$ and the values $\boldsymbol{j}_k$, $\mathcal{N}$ and M are cached, then the cost of `NextOptInf` is negligible even if $\mathcal{N}$ is realized by a simple linked list. ⌟

7.3. **Adaptive application of discrete random operators.** In this section, we analyze the structure of the adaptive multiplication routine $\texttt{Apply}_{\boldsymbol{A}}$ from Section 4.2 for a discretized parametric operator $\boldsymbol{A}$ and the approximating sequence $(\boldsymbol{A}_k)$ from Section 7.1.

By Assumption 7.A and Lemma 7.3, $(N_k)_{k \in \mathbb{N}}$ is strictly increasing, and $N_0 = 0$ since $\boldsymbol{j}_0 = \boldsymbol{0}$. By definition, $(j_{k,m})_{k \in \mathbb{N}_0}$ is nondecreasing for all $m \in \mathbb{N}_0$. Therefore, Assumption 7.A implies that $(\bar{e}_{\boldsymbol{A},k})_{k \in \mathbb{N}_0}$ is nondecreasing. If $\bar{e}_{m,j} \to 0$ as $j \to \infty$ for all $m \in \mathbb{N}_0$, since $(\bar{e}_{m,0})_{m \in \mathbb{N}_0} \in \ell^1$ by Assumption 7.A, Corollary 7.2 implies that $\bar{e}_{\boldsymbol{A},k} \to 0$ as $k \to \infty$. We note that

$$\eta_k = \frac{\bar{e}_{\boldsymbol{A},k} - \bar{e}_{\boldsymbol{A},k+1}}{N_{k+1} - N_k} = \frac{\bar{e}_{m_k, j_{k,m_k}} - \bar{e}_{m_k, j_{k,m_k}+1}}{\bar{n}_{m_k, j_{k,m_k}+1} - \bar{n}_{m_k, j_{k,m_k}}} \, , \tag{7.9}$$

which is nonincreasing in k by construction of $(\boldsymbol{j}_k)_{k \in \mathbb{N}_0}$, see Lemma A.4. Consequently, Assumption 4.A is satisfied under the sole additional requirement that $\bar{e}_{m,j} \to 0$ as $j \to \infty$ for all $m \in \mathbb{N}_0$.

Also, since

$$\frac{\bar{e}_{\boldsymbol{A},k}}{\bar{e}_{\boldsymbol{A},k+1}} = \frac{\bar{e}_{\boldsymbol{A},k}}{\bar{e}_{\boldsymbol{A},k} + \bar{e}_{m_k,j_{k,m_k}+1} - \bar{e}_{m_k,j_{k,m_k}}} \leq \frac{\bar{e}_{m_k,j_{k,m_k}}}{\bar{e}_{m_k,j_{k,m_k}+1}} ,$$

Assumption 5.A is satisfied if

$$\sup_{m\in\mathbb{N}_0} \sup_{j\in\mathbb{N}_0} \frac{\bar{e}_{m,j}}{\bar{e}_{m,j+1}} < \infty . \tag{7.10}$$

Assuming the sequences (j_k) and (m_k) are known, the first two parts of $\texttt{Apply}_{\boldsymbol{A}}[v,\epsilon]$ can be used to partition the vector v into $(v_{[p]})_{p=1}^{\ell}$ and a negligible remainder term, and to assign to each of these a $k_p \in \mathbb{N}_0$.

The final step of $\texttt{Apply}_{\boldsymbol{A}}[v,\epsilon]$ performs the multiplications

$$z \coloneqq \sum_{p=1}^{\ell} \boldsymbol{A}_{k_p} v_{[p]} . \tag{7.11}$$

Using the tensor product structure from Proposition 2.3, (7.11) can be decomposed into multiplications with the coefficient operators $\boldsymbol{D}_j$ and $\boldsymbol{R}_{m,j}$, $m \in \mathbb{N}$.

Let $v_{[p],\mu}$ denote the μ-th coefficient of $v_{[p]}$, *i.e.* $v_{[p],\mu} = (v_{\mu\iota})_{\iota}$ for $\iota \in \Xi$ such that $(\mu,\iota) \in \Xi_p$. Then assuming π_m is symmetric for all $m \in \mathbb{N}$, $z = (z_\mu)_{\mu\in\Lambda}$ with

$$z_\mu = \sum_{p=1}^{\ell} \left(\boldsymbol{D}_{j_{k_p,0}} v_{[p],\mu} + \sum_{m=1}^{M_p} \beta^m_{\mu_m+1} \boldsymbol{R}_{m,j_{k_p,m}} v_{[p],\mu+\epsilon_m} + \beta^m_{\mu_m} \boldsymbol{R}_{m,j_{k_p,m}} v_{[p],\mu-\epsilon_m} \right) , \tag{7.12}$$

where $M_p \coloneqq \max\{m \in \mathbb{N}_0 \,;\, j_{k_p,m} \neq 0\}$. This does not, however, represent an efficient way to construct z. It is not clear which z_μ are nonzero, and many multiplications with $\boldsymbol{R}_{m,j}$ are done twice. The routine $\texttt{Multiply}_{\boldsymbol{A}}$ does the same computation efficiently, for arbitrary π_m, by looping over p and the support of $v_{[p]}$.

$\texttt{Multiply}_{\boldsymbol{A}}[(v_{[p]})_{p=1}^{\ell}, (k_p)_{p=1}^{\ell}] \mapsto z$

$z \longleftarrow \mathbf{0}$
for $p = 1,\dots,\ell$ **do**
  forall $\mu \in \Lambda$ *with* $v_{[p],\mu} \neq 0$ **do**
    $z_\mu \longleftarrow z_\mu + \boldsymbol{D}_{j_{k_p,0}} v_{[p],\mu}$
    for $m = 1,\dots,M_p$ **do**
      $w \longleftarrow \boldsymbol{R}_{m,j_{k_p,m}} v_{[p],\mu}$
      $z_{\mu+\epsilon_m} \longleftarrow z_{\mu+\epsilon_m} + \beta^m_{\mu_m+1} w$
      if $\mu_m \geq 1$ **then** $z_{\mu-\epsilon_m} \longleftarrow z_{\mu-\epsilon_m} + \beta^m_{\mu_m} w$
      if $\sigma_m = 3$ **then** $z_\mu \longleftarrow z_\mu + \alpha^m_{\mu_m} w$

Remark 7.5. In $\texttt{Multiply}_{\boldsymbol{A}}[(v_{[p]})_{p=1}^{\ell}, (k_p)_{p=1}^{\ell}]$, each multiplication with $\boldsymbol{R}_{m,j}$ is performed only once, and copied to σ_m components of z. This suggests defining $\bar{n}_m \coloneqq n_{\boldsymbol{R}_m}$ for $m \in \mathbb{N}$, without the factor of σ_m from the original definition. ⌟

Remark 7.6. By Proposition 2.6, the discrete adjoint $\boldsymbol{A}^*$ of a discretized parametric operator $\boldsymbol{A}$ has the same tensor product structure as $\boldsymbol{A}$. Therefore, sparse approximations of $\boldsymbol{A}^*$ can be constructed as in Section 7.1, with $\boldsymbol{D}$ and $\boldsymbol{R}_m$, $m \in \mathbb{N}$, replaced by their adjoints. Theorem 8.4 below carries over to show s^*-compressibility of $\boldsymbol{A}^*$ under suitable assumptions, and s^*-computability follows as a corollary. In particular, $\texttt{Apply}_{\boldsymbol{A}^*}$ has the same structure as described above. An adaptive multiplication routine for $\boldsymbol{A}^*\boldsymbol{A}$ can be constructed as in (3.23). ⌟

8. s^*-compressibility of discrete random operators

8.1. **Preliminary estimates.** For an $s > 0$, assume for the moment that $\boldsymbol{D}$ and $\boldsymbol{R}_m$, $m \in \mathbb{N}$, are strictly s-compressible. By Proposition 4.3, there is a map $j_0 \colon [0,\infty) \to \mathbb{N}_0$ such that the sparse approximation $\boldsymbol{D}_{j_0(r)}$ is r-sparse and

$$\left\|\boldsymbol{D} - \boldsymbol{D}_{j_0(r)}\right\|_{\ell^2(\Xi)\to\ell^2(\Theta)} \le \bar{e}_{0,j_0(r)} \le \tilde{d}_{0,s} r^{-s} , \quad r > 0 , \tag{8.1}$$

with $\tilde{d}_{0,s} \coloneqq \tilde{d}_{\boldsymbol{D},s}$.[2] Similarly, for all $m \in \mathbb{N}$ there is a map $j_m \colon [0,\infty) \to \mathbb{N}_0$ such that the sparse approximation $\boldsymbol{R}_{m,j_m(r)}$ is $r\sigma_m^{-1}$-sparse and

$$\left\|\boldsymbol{R}_m - \boldsymbol{R}_{m,j_m(r)}\right\|_{\ell^2(\Xi)\to\ell^2(\Theta)} \le \bar{e}_{m,j_m(r)} \le \tilde{d}_{m,s} r^{-s} , \quad r > 0 , \tag{8.2}$$

with $\tilde{d}_{m,s} \coloneqq \sigma_m^s \tilde{d}_{\boldsymbol{R}_m,s}$.

Lemma 8.1. *If $(\tilde{d}_{m,s})_m \in \ell^{\frac{1}{s+1}}(\mathbb{N}_0)$, then for all $r > 0$ there is a finitely supported sequence $\boldsymbol{j}(r)$ in $\mathbb{N}_0$ such that $N_{\boldsymbol{j}(r)} \le r$ and*

$$\bar{e}_{\boldsymbol{A},\boldsymbol{j}(r)} \le \left(\sum_{m=0}^{\infty} \tilde{d}_{m,s}^{\frac{1}{s+1}}\right)^{s+1} r^{-s} . \tag{8.3}$$

Proof. Let $t > 0$ and define $r_m \coloneqq \tilde{d}_{m,s}^{\frac{1}{s+1}} t$ for all $m \in \mathbb{N}_0$. Set $\boldsymbol{j} \coloneqq (j_m(r_m))_{m\in\mathbb{N}_0}$. This sequence is finitely supported since $r_m < 1$ for all but finitely many $m \in \mathbb{N}_0$. By Lemma 7.1,

$$N_{\boldsymbol{j}} = \sum_{m=0}^{\infty} \bar{n}_{m,j_m(r_m)} \le \sum_{m=0}^{\infty} r_m = \sum_{m=0}^{\infty} \tilde{d}_{m,s}^{\frac{1}{s+1}} t =: r$$

and

$$\bar{e}_{\boldsymbol{A},\boldsymbol{j}} = \sum_{m=0}^{\infty} \bar{e}_{m,j_m(r_m)} \le \sum_{m=0}^{\infty} \tilde{d}_{m,s} r_m^{-s} = \sum_{m=0}^{\infty} \tilde{d}_{m,s}^{\frac{1}{s+1}} t^{-s} = \left(\sum_{m=0}^{\infty} \tilde{d}_{m,s}^{\frac{1}{s+1}}\right)^{s+1} r^{-s} .$$

□

If $(\tilde{d}_{m,s})_m$ is not in $\ell^{\frac{1}{s+1}}(\mathbb{N}_0)$, a similar property still holds if we replace the infinite sum by a partial sum. We define the operators

$$\boldsymbol{A}_{[M]} \coloneqq \boldsymbol{I} \otimes \boldsymbol{D} + \sum_{m=1}^{M} \boldsymbol{K}_m \otimes \boldsymbol{R}_m \in \mathcal{L}(\ell^2(\Lambda\times\Xi), \ell^2(\Lambda\times\Theta)) . \tag{8.4}$$

Let

$$\|\boldsymbol{D}\|_{\ell^2(\Xi)\to\ell^2(\Theta)} \le \bar{e}_{0,0} \quad \text{and} \quad \|\boldsymbol{R}_m\|_{\ell^2(\Xi)\to\ell^2(\Theta)} \le \bar{e}_{m,0} , \quad m \in \mathbb{N} . \tag{8.5}$$

Then by Lemma 2.2 and Proposition 2.3,

$$\left\|\boldsymbol{A} - \boldsymbol{A}_{[M]}\right\|_{\ell^2(\Lambda\times\Xi)\to\ell^2(\Lambda\times\Theta)} \le \sum_{m=M+1}^{\infty} \bar{e}_{m,0} . \tag{8.6}$$

For any $s > 0$, if either

$$\bar{e}_{m,0} \le s\delta_{\boldsymbol{A},s}(m+1)^{-s-1} \qquad \forall m \in \mathbb{N} \tag{8.7}$$

or

$$\left(\sum_{m=1}^{\infty} \bar{e}_{m,0}^{\frac{1}{s+1}}\right)^{s+1} \le \delta_{\boldsymbol{A},s} , \tag{8.8}$$

[2] Proposition 4.3 initially only implies that the first term in (8.1) is bounded by the third. However, if (8.1) does not hold, we can replace $\bar{e}_{0,j_0(r)}$ by $\tilde{d}_{0,s} r^{-s}$ in (7.1).

then it follows as in [23, Prop. 4.4] that

$$\sum_{m=M+1}^{\infty} \bar{e}_{m,0} \le \delta_{\boldsymbol{A},s}(M+1)^{-s} \qquad \forall M \in \mathbb{N}_0 \,. \tag{8.9}$$

We define

$$\bar{e}_{\boldsymbol{A}_{[M]},\boldsymbol{j}} \coloneqq \sum_{m=0}^{M} \bar{e}_{m,j_m} \,. \tag{8.10}$$

Then for all sequences $\boldsymbol{j}$ in $\mathbb{N}_0$ with support in $\{0,1,\dots,M\}$,

$$\bar{e}_{\boldsymbol{A},\boldsymbol{j}} = \bar{e}_{\boldsymbol{A}_{[M]},\boldsymbol{j}} + \sum_{m=M+1}^{\infty} \bar{e}_{m,0} \,. \tag{8.11}$$

Lemma 8.2. *For all $M \in \mathbb{N}_0$ and all $r > 0$, there is a sequence $\boldsymbol{j}(r)$ in $\mathbb{N}_0$ with support in $\{0,1,\dots,M\}$ such that $N_{\boldsymbol{j}(r)} \le r$ and*

$$\bar{e}_{\boldsymbol{A}_{[M]},\boldsymbol{j}(r)} \le \left(\sum_{m=0}^{M} \tilde{d}_{m,s}^{\frac{1}{s+1}}\right)^{s+1} r^{-s} \,. \tag{8.12}$$

Proof. The proof is analogous to the proof of Lemma 8.1. □

Proposition 8.3. *Let* (8.7) *or* (8.8) *be satisfied for an $s_\sigma > 0$ and*

$$\left(\sum_{m=0}^{M} \tilde{d}_{m,s}^{\frac{1}{s+1}}\right)^{s+1} \le \hat{d}_s M^{t_s} \,, \quad M \in \mathbb{N} \,, \tag{8.13}$$

with $\hat{d}_s > 0$ and $t_s \ge 0$. Then for all $r \in [1,\infty)$ there is a finitely supported sequence $\boldsymbol{j}(r)$ in $\mathbb{N}_0$ such that $N_{\boldsymbol{j}(r)} \le r$ and

$$\bar{e}_{\boldsymbol{A},\boldsymbol{j}(r)} \le \left(\hat{d}_s + \delta_{\boldsymbol{A},s_\sigma}\right) r^{\frac{-s}{1+t_s/s_\sigma}} \,. \tag{8.14}$$

Proof. Let $r \in [1,\infty)$ and set $M \coloneqq \lfloor r^{\frac{s}{s_\sigma+t_s}} \rfloor$. Then for the sequence $\boldsymbol{j}(r)$ from Lemma 8.2,

$$\bar{e}_{\boldsymbol{A}_{[M]},\boldsymbol{j}(r)} \le \hat{d}_s M^{t_s} r^{-s} \le \hat{d}_s r^{\frac{-ss_\sigma}{s_\sigma+t_s}} \,.$$

Equation (8.9) implies

$$\sum_{m=M+1}^{\infty} \bar{e}_{m,0} \le \delta_{\boldsymbol{A},s_\sigma}(M+1)^{-s_\sigma} \le \delta_{\boldsymbol{A},s_\sigma} r^{\frac{-ss_\sigma}{s_\sigma+t_s}} \,.$$

Then the assertion follows using (8.11). □

8.2. **s^*-compressibility.** The above estimates combine with Corollary 7.2 to show s^*-compressibility of $\boldsymbol{A}$ with the approximating sequence $(\boldsymbol{A}_k)_{k\in\mathbb{N}}$ from Section 7.1. Define the constants

$$\tilde{c}_m \coloneqq \max\left(\bar{n}_{m,1}, \sup_{j\in\mathbb{N}} \frac{\bar{n}_{m,j+1}}{\bar{n}_{m,j}}\right) < \infty \,, \quad m \in \mathbb{N}_0 \,. \tag{8.15}$$

Note that $c_{\boldsymbol{D}} \le \tilde{c}_0$ and $c_{\boldsymbol{R}_m} \le \sigma_m \tilde{c}_m$ for $m \in \mathbb{N}$.

Theorem 8.4. *Let $s_\delta^*, s_\sigma^* \in (0,\infty]$ and assume*

$$\tilde{c} \coloneqq \sup_{m\in\mathbb{N}_0} \tilde{c}_m < \infty \,. \tag{8.16}$$

(1) *If $(\tilde{d}_{m,s})_m \in \ell^{\frac{1}{s+1}}(\mathbb{N}_0)$ for all $s \in (0, s_\delta^*)$, then $\boldsymbol{A}$ is s^*-compressible for $s^* = s_\delta^*$.*

(2) *If* (8.7) *or* (8.8) *holds for all* $s \in (0, s_\sigma^*)$ *and* (8.13) *holds for all* $s \in (0, s_\delta^*)$ *with* $t_s \le \hat{t} < \infty$, *then* $\boldsymbol{A}$ *is* s^**-compressible for*

$$s^* = \frac{s_\delta^*}{1 + \hat{t}/s_\sigma^*} \,. \tag{8.17}$$

In both cases, $(\boldsymbol{A}_k)_{k\in\mathbb{N}}$ *is a valid approximating sequence with* $c_{\boldsymbol{A}} \le \tilde{c}$,

$$d_{\boldsymbol{A},s} \le \left\|(\tilde{d}_{m,s})_m\right\|_{\ell^{\frac{1}{s+1}}(\mathbb{N}_0)} \,, \quad s \in (0, s^*) \tag{8.18}$$

in the first case and

$$d_{\boldsymbol{A},s} \le \inf_{\frac{s\hat{t}}{s_\delta^* - s} < s_\sigma < s_\sigma^*} \left(\hat{d}_{s(1+\hat{t}/s_\sigma)} + \delta_{\boldsymbol{A},s_\sigma}\right) \,, \quad s \in (0, s^*) \tag{8.19}$$

in the second case.

Proof. Condition (8.16) ensures (4.1) for $(\boldsymbol{A}_k)_{k\in\mathbb{N}}$ since for $k \in \mathbb{N}$ and $j \coloneqq j_{k,m_k}$, if $j \ge 1$,

$$\frac{N_{k+1}}{N_k} = \frac{N_k + \bar{n}_{m_k,j+1} - \bar{n}_{m_k,j}}{N_k} = \frac{n + \bar{n}_{m_k,j+1}}{n + \bar{n}_{m_k,j}} \le \frac{\bar{n}_{m_k,j+1}}{\bar{n}_{m_k,j}} \le \tilde{c}_{m_k} \,,$$

where $n = N_k - \bar{n}_{m_k,j} \ge 0$, and if $j = 0$,

$$\frac{N_{k+1}}{N_k} = \frac{N_k + \bar{n}_{m_k,1}}{N_k} \le \bar{n}_{m_k,1} \le \tilde{c}_{m_k} \,.$$

Let $s \in (0, s^*)$. In case 1, Corollary 7.2 and Lemma 8.1 with $r = N_k$ imply

$$\bar{e}_{\boldsymbol{A},k} \le \bar{e}_{\boldsymbol{A},\boldsymbol{j}(N_k)} \le \left(\sum_{m=0}^{\infty} \tilde{d}_{m,s}^{\frac{1}{s+1}}\right)^{s+1} N_k^{-s} \,.$$

In case 2, select $s_\delta \in (0, s_\delta^*)$ and $s_\sigma \in (0, s_\sigma^*)$ such that

$$s = \frac{s_\delta}{1 + \hat{t}/s_\sigma} \,.$$

This is possible since the right hand side is increasing in s_δ and s_σ. By monotonicity, (8.13) holds with $t_s = \hat{t}$. Then Corollary 7.2 and Proposition 8.3 with $r = N_k$ imply

$$\bar{e}_{\boldsymbol{A},k} \le \bar{e}_{\boldsymbol{A},\boldsymbol{j}(N_k)} \le \left(\hat{d}_{s_\delta} + \delta_{\boldsymbol{A},s_\sigma}\right) N_k^{-s} \,.$$

Equation (8.19) follows since $s_\delta = s(1 + \hat{t}/s_\sigma)$. □

8.3. s^***-computability.** Under the assumption that the sequence $(\boldsymbol{j}_k)_{k\in\mathbb{N}_0}$ is available, s^*-computability of $\boldsymbol{A}$ follows from Theorem 8.4 as a corollary.

Corollary 8.5. *In the setting of Theorem 8.4, if*

$$\sup_{m\in\mathbb{N}_0} b_m < \infty \tag{8.20}$$

for b_m *from Section 7.2 and the sequences* $\boldsymbol{j}_k$ *are given as in Remark A.9, then* $\boldsymbol{A}$ *is* s^**-computable and* $\texttt{Build}_{\boldsymbol{A}}$ *is a valid assembly routine.*

Proof. s^*-compressibility follows from Theorem 8.4. By Lemma 7.3, (8.20) and Remark A.9, the number of arithmetic operations and storage locations required by a call of $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$ is $O(N_k)$. □

If $\boldsymbol{j}_k$ are not readily available, Proposition A.7 implies that recursive application of `NextOptInf` from Appendix A can construct $\boldsymbol{j}_k$ in $O(k \log(k))$ time. Thus $\boldsymbol{A}$ is still s^*-computable if $k \log(k) \lesssim N_k$. As discussed in Remark 7.4, the cost of computing $\boldsymbol{j}_k$ from $\boldsymbol{j}_{k-1}$ using `NextOptInf` is only $O(\log(k))$. Therefore, if `NextOptInf` is used to construct $\boldsymbol{j}_k$ in the first call of $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$, then $\texttt{Build}_{\boldsymbol{A}}[k, \cdot]$ requires

$O(N_k)$ operations provided that $\boldsymbol{j}_{k-1}$ is known, for example from a previous call of $\mathtt{Build}_A[k-1,\cdot]$.

9. An illustrative example

9.1. **An elliptic boundary value problem.** As a model problem, 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

$$\begin{aligned} -\nabla\cdot(a(x)\nabla u(x)) &= f(x)\;, \quad x \in G\;,\\ u(x) &= 0\;, \quad x \in \partial G\;. \end{aligned} \tag{9.1}$$

We view f as deterministic, but model the coefficient a as a series

$$a(y,x) \coloneqq \bar{a}(x) + \sum_{m=1}^{\infty} y_m a_m(x)\;, \tag{9.2}$$

with $y_m \in [-1,1]$ for all $m \in \mathbb{N}$. Hence a depends on a parameter $y = (y_m)_{m=1}^\infty$ in $\Gamma = [-1,1]^\infty$.

We define the parametric operator

$$A(y)\colon H^1_0(G) \to H^{-1}(G)\;, \quad v \mapsto -\nabla\cdot(a(y)\nabla v)\;, \tag{9.3}$$

for $y \in \Gamma$. Due to the linear dependence of A on a,

$$A(y) = D + R(y)\;, \quad R(y) \coloneqq \sum_{m=1}^{\infty} y_m R_m \qquad \forall y \in \Gamma \tag{9.4}$$

with convergence in $\mathcal{L}(H^1_0(G), H^{-1}(G))$, as assumed in (1.3) and (1.8), for

$$\begin{aligned} D&\colon H^1_0(G) \to H^{-1}(G)\;, \quad v \mapsto -\nabla\cdot(\bar{a}\nabla v)\;,\\ R_m&\colon H^1_0(G) \to H^{-1}(G)\;, \quad v \mapsto -\nabla\cdot(a_m\nabla v)\;, \quad m \in \mathbb{N}\;. \end{aligned}$$

To ensure bounded invertibility of D, we assume there is a constant $\delta > 0$ such that

$$\operatorname*{ess\,inf}_{x\in G} \bar{a}(x) \geq \delta^{-1}\;. \tag{9.5}$$

Since $\|R_m\|_{H^1_0(G)\to H^{-1}(G)} \leq \|a_m\|_{L^\infty(G)}$, (1.9) follows from

$$\delta \sum_{m=1}^{\infty} \|a_m\|_{L^\infty(G)} \leq \gamma < 1\;. \tag{9.6}$$

This condition can be loosened by defining $\langle D\cdot,\cdot\rangle$ as the inner product of $H^1_0(G)$, in which case the factor δ in (9.6) vanishes, and $\|a_m\|_{L^\infty(G)}$ is replaced by $\|a_m/\bar{a}\|_{L^\infty(G)}$. We refer to *e.g.* [25, 22, 30] for further extensions that still ensure (1.5).

9.2. **Optimal finite element discretization.** Approximation results for the solution u of (9.1) have been shown in [11] for the case that y_m are uniformly distributed. In this setting, the orthogonal polynomials P^m_n from Section 2.1 are Legendre polynomials, normalized with respect to the uniform probability measure on $[-1,1]$.

Let $(V_j)_{j=0}^\infty$ be a sequence of finite element spaces in $H^1_0(G)$ with geometrically increasing dimensions $M_j \coloneqq \dim V_j$, satisfying

$$\inf_{v_j\in V_j} \left\|v - v_j\right\|_{H^1_0(G)} \leq C M_j^{-t}\,|v|_Z \qquad \forall v \in Z\;, \tag{9.7}$$

where $Z \subset H^1_0(G)$ with norm $(\|\cdot\|^2_{H^1_0(G)} + |\cdot|^2_Z)^{1/2}$. We consider approximations to u in which, for some finite set $\Xi \subset \Lambda$, each coefficient u_μ for $\mu \in \Xi$ is approximated in some finite element space $V_\mu \coloneqq V_{j(\mu)}$, and the remaining u_μ are set to zero.

If $\boldsymbol{u} \in \ell^p(\Lambda; H^1_0(G))$ for some $p \in (0,2)$, then Stechkin's lemma implies that if Ξ_N contains the first $N-1$ indices μ in a decreasing rearrangement of $\|u_\mu\|_{H^1_0(G)}$, the truncation error satisfies

$$\left(\sum_{\mu \in \Lambda \setminus \Xi_N} \left\| u_\mu \right\|^2_{H^1_0(G)} \right)^{1/2} \leq \|\boldsymbol{u}\|_{\ell^p(\Lambda; H^1_0(G))} N^{-s} , \quad s = \frac{1}{p} - \frac{1}{2} . \tag{9.8}$$

Following [11], we select spaces V_μ, $\mu \in \Xi_N$, to match this rate. To this end, suppose $\boldsymbol{u} \in \ell^q(\Lambda; Z)$ for a $q \in [p, \infty]$. Using a Lagrange multiplier to minimize the total dimension $N_{\mathrm{dof}} \coloneqq \sum_{\mu \in \Xi_N} M_\mu$, with $M_\mu = \dim V_\mu$, under the condition that the total error is equivalent to N^{-s}, leads to a choice of M_μ proportional to $|u_\mu|_Z^{\frac{2}{2t+1}}$. This approximation has a convergence rate of t with respect to N_{dof} if $t \leq \frac{1}{q} - \frac{1}{2}$, which coincides with the rate for a single finite element approximation, see (9.7). If $t \geq \frac{1}{q} - \frac{1}{2}$, the resulting approximation rate is

$$s \, \frac{t}{t + \frac{1}{p} - \frac{1}{q}} . \tag{9.9}$$

This is generally less than the semidiscrete approximation rate s, with equality if $q = p$; this last case is considered in [11, Theorem 5.5].

The above summability assumptions are proven in [11] for the case that $|v|_Z = \|\Delta v\|_{L^2(G)}$. Then $\boldsymbol{u} \in \ell^p(\Lambda; H^1_0(G))$ if $(a_m) \in \ell^p(\mathbb{N}; L^\infty(G))$, and $\boldsymbol{u} \in \ell^q(\Lambda; Z)$ holds under the condition $(a_m) \in \ell^q(\mathbb{N}; W^{1,\infty}(G))$. In this setting, t has a maximal value of $1/d$.

Remark 9.1. A similar analysis can be performed if, instead of choosing M_μ by a continuous optimization problem, the finite element spaces are selected to equidistribute the error among all coefficients u_μ, as in the heuristic from [23, 24].[3] Due to (9.7), this is achieved for $M_\mu^t \sim |u_\mu|_Z$. The resulting convergence rate with respect to N_{dof} is

$$\frac{2s}{2s+1} \, t \tag{9.10}$$

if $t \leq 1/q$, and coincides with (9.9) if $t \geq 1/q$. In the former case, the approximation rate is slightly less than the optimal value t; the rate in the second case is optimal, but it only sets in for $t \geq 1/q$ instead of $t \geq \frac{1}{q} - \frac{1}{2}$. ⌟

9.3. **Application of the adaptive stochastic Galerkin method.** In Section 2.3, D and R_m are discretized by a wavelet basis or frame of $H^1_0(G)$, leading to operators $\boldsymbol{D}$ and $\boldsymbol{R}_m$ on ℓ^2, which can be interpreted as bi-infinite matrices. Although these matrices are generally not sparse, they can be approximated by sparse matrices, and these approximations are pivotal in the efficient adaptive application of the discrete random operator $\boldsymbol{A}$. We refer to [29] and references therein for constructions of wavelet bases.

It is shown in [33] that for wavelets of order n, *i.e.* if the dual wavelets have n vanishing moments, $\boldsymbol{D}$ and $\boldsymbol{R}_m$ can be s^*_δ-compressible with $s^*_\delta = (n-1)/d$. This is the highest rate of compressibility that adaptive wavelet methods can take advantage of since the order of the wavelets limits the solution of a generic discrete deterministic problem to the space $\mathcal{A}^s$ for $s < s^*_\delta$, see [18, 8]. For higher compressibility, the sparsity of the exact solution becomes the limiting factor in the convergence of adaptive wavelet algorithms.

We consider the example $G \coloneqq (0,1)$ and

$$a_m(x) \coloneqq C m^{-k} \sin(m\pi x) , \quad m \in \mathbb{N} , \tag{9.11}$$

[3] This heuristic is actually used to distribute tolerances for a subproblem in [23, 24]; it is not clear whether the resulting error in the approximation of u is distributed evenly among all active coefficients.

with C sufficiently small such that (9.6) holds. Since trigonometric functions often appear in Karhunen–Loève expansions of random fields, this academic example is quite representative. We note that $(a_m) \in \ell^p(\mathbb{N}; L^\infty(G))$ and $(a_m) \in \ell^q(\mathbb{N}; W^{1,\infty}(G))$ for any $p > 1/k$ and $q > 1/(k-1)$. Thus $\boldsymbol{u} \in \ell^p(\Lambda; H^1_0(G))$ and $\boldsymbol{u} \in \ell^q(\Lambda; H^2(G))$ for the same ranges of p and q by [11]. The resulting approximation rates from Section 9.2 are 1 for $k \geq 5/2$ and $\frac{1}{2}(k - \frac{1}{2}) \leq 1$ for $k \leq 5/2$.

As mentioned above, it is realistic to assume that the operators $\boldsymbol{D}$ and $\boldsymbol{R}_m$, $m \in \mathbb{N}$, are s^*_δ-compressible with $s^*_\delta \geq 1$. In order to derive s^*-compressibility of the discrete stochastic operator $\boldsymbol{A}$, Theorem 8.4 requires a degree of summability of the compressibility constants of these operators. Entries in the matrix representations of these operators are zero for basis functions with disjoint supports, and they generally also become insignificant if the supports overlap, but the wavelets have sufficiently different length scales. In this example, the latter effect only sets in once the smaller length scale is below $1/m$. Consequently, we are left with $O(m)$ significant entries in columns of $\boldsymbol{R}_m$ corresponding to coarse-scale basis functions.

For any $r > 0$, let $e_{m,r}$ denote the error in an r-sparse approximation of $\boldsymbol{R}_m$. Then the sparsity required to achieve an error of $e_{m,r} \sim m^{-k} e_{1,\varrho}$ in the approximation of $\boldsymbol{R}_m$ is $r \sim \varrho m$. This implies

$$\tilde{d}_{m,s} \sim \sup_{r>0} r^s e_{m,r} \sim \sup_{\varrho>0} \varrho^s m^s m^{-k} e_{1,\varrho} = m^{-(k-s)} \tilde{d}_{1,s} \,. \tag{9.12}$$

In this setting, the condition $(\tilde{d}_{m,s})_m \in \ell^{\frac{1}{s+1}}(\mathbb{N}_0)$ of Theorem 8.4 is equivalent to $k - s > s + 1$, *i.e.* $s < (k-1)/2$. Hence we can realistically expect s^*-compressibility of $\boldsymbol{A}$ for $s^* = (k-1)/2$, provided $s^*_\delta \geq s^*$.

For $k \leq 3$, the compression rate s^* is less than or equal to the approximation rate, and thus s^*-compressibility is the limiting factor in the complexity of adaptive wavelet methods for our model problem. For $k \geq 3$, the limited spatial regularity shown in [11] becomes the main obstacle, and the compression rate is larger than the approximation rate given here.

Despite the slightly suboptimal complexity of adaptive wavelet methods due to the compression rate s^* being smaller than the approximation rate, the direct application of these methods to the fully discrete problem improves on the heuristic used in [23, 24]. For example, if $k = 3$, then $\boldsymbol{A}$ is s^*-compressible for $s^* = 1$, and $\boldsymbol{u} \in \mathcal{A}^s(\Lambda \times \Xi)$ for all $s < 1$. However, if u is approximated by finite elements with the same approximation error in each active coefficient, then the optimal approximation rate is only 5/6, see Remark 9.1. A similar property holds for any $k \geq (3 + \sqrt{5})/2$ since the approximation rate with equidistributed errors is essentially $1 - \frac{1}{2k}$ for $k \geq 2$.

Appendix A. Greedy algorithms

A.1. **A generalized knapsack problem.** We consider a discrete optimization problem in which both the objective and the constraints are given by sums over an arbitrary set $\mathcal{M} \subset \mathbb{N}_0$. For each $m \in \mathcal{M}$, we have two increasing sequences $(c^m_j)_{j\in\mathbb{N}_0}$ and $(\omega^m_j)_{j\in\mathbb{N}_0}$, which we interpret as costs and values. We define the total cost of a $\boldsymbol{j} = (j_m)_{m\in\mathcal{M}} \in \mathbb{N}_0^{\mathcal{M}}$ as

$$c_{\boldsymbol{j}} \coloneqq \sum_{m\in\mathcal{M}} c^m_{j_m} \tag{A.1}$$

and the total value of $\boldsymbol{j}$ as

$$\omega_{\boldsymbol{j}} \coloneqq \sum_{m\in\mathcal{M}} \omega^m_{j_m} \,. \tag{A.2}$$

Our goal is to maximize $\omega_{\boldsymbol{j}}$ under a constraint on $c_{\boldsymbol{j}}$, or to minimize $c_{\boldsymbol{j}}$ under a constraint on $\omega_{\boldsymbol{j}}$.

Remark A.1. The above two goals are essentially equivalent. If $\boldsymbol{j} \in \mathbb{N}_0^{\mathcal{M}}$ such that for all $\boldsymbol{i} \in \mathbb{N}_0^{\mathcal{M}}$, $c_{\boldsymbol{i}} \leq c_{\boldsymbol{j}}$ implies $\omega_{\boldsymbol{i}} \leq \omega_{\boldsymbol{j}}$, then by contraposition, $\omega_{\boldsymbol{i}} > \omega_{\boldsymbol{j}}$ implies $c_{\boldsymbol{i}} > c_{\boldsymbol{j}}$. Similarly, if $c_{\boldsymbol{i}} < c_{\boldsymbol{j}}$ implies $\omega_{\boldsymbol{i}} < \omega_{\boldsymbol{j}}$, then also $\omega_{\boldsymbol{i}} \geq \omega_{\boldsymbol{j}}$ implies $c_{\boldsymbol{i}} \geq c_{\boldsymbol{j}}$. In both cases, the two statements are equivalent. ⌟

Remark A.2. The classical knapsack problem is equivalent to the above optimization problem in the case that $\mathcal{M}$ is finite, and for all $m \in \mathcal{M}$, $\omega_0^m = 0$ and $\omega_j^m = \omega_1^m$ for all $j \geq 1$. Then without loss of generality, we can set $c_0^m \coloneqq 0$ for all $m \in \mathcal{M}$, and the values c_j^m for $j \geq 2$ are irrelevant due to the assumption that $(c_j^m)_{j \in \mathbb{N}_0}$ is increasing. Optimal sequences $\boldsymbol{j} \in \mathbb{N}_0^{\mathcal{M}}$ will only take the values 0 and 1, and can thus be interpreted as subsets of $\mathcal{M}$.

We note that greedy methods only construct a sequence of optimal solutions. They do not maximize $\omega_{\boldsymbol{j}}$ under an arbitrary constraint on $c_{\boldsymbol{j}}$, and thus do not solve an NP-hard problem. ⌟

Remark A.3. We are particularly interested in minimizing an error under constraints on the computational cost of an approximation with this error tolerance. Given sequences $(e_j^m)_{j \in \mathbb{N}_0}$ and $(c_j^m)_{j \in \mathbb{N}_0}$ of errors and corresponding costs, we define a sequence of values by $\omega_j^m \coloneqq -e_j^m$. If $(e_j^m)_{j \in \mathbb{N}_0}$ is decreasing, then $(\omega_j^m)_{j \in \mathbb{N}_0}$ is increasing. Typically, as $j \to \infty$, we have $e_j^m \to 0$ and $c_j^m \to \infty$. Then, although it is increasing, $(\omega_j^m)_{j \in \mathbb{N}_0}$ remains bounded. In particular, it is reasonable to assume that $(\omega_j^m)_{j \in \mathbb{N}_0}$ increases more slowly than $(c_j^m)_{j \in \mathbb{N}_0}$, in a sense that is made precise below. ⌟

A.2. **A sequence of optimal solutions.** We iteratively construct a sequence $(\boldsymbol{j}^k)_{k \in \mathbb{N}_0}$ in $\mathbb{N}_0^{\mathcal{M}}$ such that, under some assumptions, each $\boldsymbol{j}^k$ is optimal in the sense of Remark A.1. For all $m \in \mathcal{M}$ and all $j \in \mathbb{N}_0$, let

$$\Delta c_j^m \coloneqq c_{j+1}^m - c_j^m \qquad \text{and} \qquad \Delta \omega_j^m \coloneqq \omega_{j+1}^m - \omega_j^m \,. \tag{A.3}$$

Furthermore, let q_j^m denote the quotient of these two increments,

$$q_j^m \coloneqq \frac{\Delta \omega_j^m}{\Delta c_j^m} \,, \quad j \in \mathbb{N}_0 \,, \tag{A.4}$$

which can be interpreted as the value to cost ratio of passing from j to $j+1$ in the index $m \in \mathcal{M}$.

Let $\boldsymbol{j}^0 \coloneqq \boldsymbol{0} \in \mathbb{N}_0^{\mathcal{M}}$. For all $k \in \mathbb{N}_0$, we construct $\boldsymbol{j}^{k+1}$ from $\boldsymbol{j}^k$ as follows. Let $m_k = m \in \mathbb{N}_0$ maximize $q_{j_m^k}^m$. Existence of such maxima is ensured by the last statement in Assumption A.A. If the maximum is not unique, select m_k to be minimal among all maxima. Then define $j_{m_k}^{k+1} \coloneqq j_{m_k}^k + 1$, and set $j_m^{k+1} \coloneqq j_m^k$ for all $m \neq m_k$. For this sequence, we abbreviate $c_k \coloneqq c_{\boldsymbol{j}^k}$ and $\omega_k \coloneqq \omega_{\boldsymbol{j}^k}$.

Assumption A.A. *For all $m \in \mathcal{M}$,*

$$c_0^m = 0 \qquad \textit{and} \quad \Delta c_j^m > 0 \qquad \forall\, j \in \mathbb{N}_0 \,, \tag{A.5}$$

i.e. $(c_j^m)_{j \in \mathbb{N}_0}$ is strictly increasing. Also, $(\omega_0^m)_{m \in \mathcal{M}} \in \ell^1(\mathcal{M})$ and $(\omega_j^m)_{j \in \mathbb{N}_0}$ is nondecreasing for all $m \in \mathcal{M}$, i.e. $\Delta \omega_j^m \geq 0$ for all $j \in \mathbb{N}_0$. Furthermore, for each $m \in \mathcal{M}$, the sequence $(q_j^m)_{j \in \mathbb{N}_0}$ is nonincreasing, i.e. if $i \geq j$, then $q_i^m \leq q_j^m$. Finally, for any $\epsilon > 0$, there are only finitely many $m \in \mathcal{M}$ for which $q_0^m \geq \epsilon$.

The assumption that $(q_j^m)_{j\in\mathbb{N}_0}$ is nonincreasing is equivalent to

$$\frac{\Delta\omega_i^m}{\Delta\omega_j^m} \le \frac{\Delta c_i^m}{\Delta c_j^m} \qquad \text{if} \quad i \ge j \tag{A.6}$$

if $\Delta\omega_j^m > 0$. In this sense, $(\omega_j^m)_{j\in\mathbb{N}_0}$ increases more slowly than $(c_j^m)_{j\in\mathbb{N}_0}$. Also, this assumption implies that if $\Delta\omega_j^m = 0$, then $\omega_i^m = \omega_j^m$ for all $i \ge j$.

We define a total order on $\mathcal{M}\times\mathbb{N}_0$ by

$$(m,j) \prec (n,i) \quad \text{if} \quad \begin{cases} q_j^m > q_i^n & \text{or} \\ q_j^m = q_i^n & \text{and} \quad m < n \quad \text{or} \\ q_j^m = q_i^n & \text{and} \quad m = n \quad \text{and} \quad j < i\,. \end{cases} \tag{A.7}$$

To any sequence $\boldsymbol{j} = (j_m)_{m\in\mathcal{M}}$ in $\mathbb{N}_0$, we associate the set

$$\{\!\{\boldsymbol{j}\}\!\} := \{(m,j) \in \mathcal{M}\times\mathbb{N}_0\,;\; j < j_m\}\,. \tag{A.8}$$

Lemma A.4. *For all $k \in \mathbb{N}_0$, $\{\!\{k\}\!\} := \{\!\{\boldsymbol{j}^k\}\!\}$ consists of the first k terms of $\mathcal{M}\times\mathbb{N}_0$ with respect to the order $\prec$.*

Proof. The assertion is trivial for $k = 0$. Assume it holds for some $k \in \mathbb{N}_0$. By definition,

$$\{\!\{k+1\}\!\} = \{\!\{k\}\!\} \cup \{(m_k, j_{m_k}^k)\}\,,$$

and $(m_k, j_{m_k}^k)$ is the $\prec$-minimal element of the set $\{(m, j_m^k)\,;\, m \in \mathcal{M}\}$. For each $m \in \mathcal{M}$, Assumption A.A implies $q_i^m \le q_{j_m^k}^m$ for all $i \ge j_m^k + 1$. Therefore, $(m, j_m^k) \prec (m,i)$ for all $i \ge j_m^k + 1$, and consequently $(m_k, j_{m_k}^k)$ is the $\prec$-minimal element of $(\mathcal{M}\times\mathbb{N}_0)\setminus\{\!\{k\}\!\}$. □

Theorem A.5. *For all $k \in \mathbb{N}_0$, the sequence $\boldsymbol{j}^k$ maximizes $\omega_{\boldsymbol{j}}$ among all finitely supported sequence $\boldsymbol{j} = (j_m)_{m\in\mathcal{M}}$ in $\mathbb{N}_0$ with $c_{\boldsymbol{j}} \le c_k$. Furthermore, if $c_{\boldsymbol{j}} < c_k$ and there exist k pairs $(m,i) \in \mathcal{M}\times\mathbb{N}_0$ with $\Delta\omega_i^m > 0$, then $\omega_{\boldsymbol{j}} < \omega_k$.*

Proof. Let $k \in \mathbb{N}$ and let $\boldsymbol{j} = (j_m)_{m\in\mathcal{M}}$ be a finitely supported sequence in $\mathbb{N}_0$ with $c_{\boldsymbol{j}} \le c_k$. By definition,

$$\omega_{\boldsymbol{j}} = \sum_{m\in\mathcal{M}} \omega_0^m + \sum_{m\in\mathcal{M}} \sum_{i=0}^{j_m-1} q_i^m \Delta c_i^m = \omega_{\boldsymbol{j}^0} + \sum_{(m,i)\in\{\!\{\boldsymbol{j}\}\!\}} q_i^m \Delta c_i^m\,.$$

Therefore, the assertion reduces to

$$\sum_{(m,i)\in\{\!\{\boldsymbol{j}\}\!\}\setminus\{\!\{k\}\!\}} q_i^m \Delta c_i^m \le \sum_{(m,i)\in\{\!\{k\}\!\}\setminus\{\!\{\boldsymbol{j}\}\!\}} q_i^m \Delta c_i^m\,.$$

Note that by (A.1) and (A.3),

$$\sum_{(m,i)\in\{\!\{\boldsymbol{j}\}\!\}\setminus\{\!\{k\}\!\}} \Delta c_i^m = c_{\boldsymbol{j}} - c' \qquad \text{for} \qquad c' := \sum_{(m,i)\in\{\!\{\boldsymbol{j}\}\!\}\cap\{\!\{k\}\!\}} \Delta c_i^m\,.$$

By Lemma A.4 and (A.7), $q := q_{j_{m_{k-1}}^{k-1}}^{m_{k-1}}$ satisfies $q \le q_i^m$ for all $(m,i) \in \{\!\{k\}\!\}$, and $q_i^m \le q$ for all $(m,i) \in (\mathcal{M}\times\mathbb{N}_0)\setminus\{\!\{k\}\!\}$. In particular, $q > 0$ if there exist k pairs $(m,i) \in \mathcal{M}\times\mathbb{N}_0$ with $q_i^m > 0$ since $\#\{\!\{k\}\!\} = k$. Consequently,

$$\begin{aligned} \sum_{(m,i)\in\{\!\{\boldsymbol{j}\}\!\}\setminus\{\!\{k\}\!\}} q_i^m \Delta c_i^m &\le q \sum_{(m,i)\in\{\!\{\boldsymbol{j}\}\!\}\setminus\{\!\{k\}\!\}} \Delta c_i^m = q(c_{\boldsymbol{j}} - c') \\ &\le q(c_k - c') \le \sum_{(m,i)\in\{\!\{k\}\!\}\setminus\{\!\{\boldsymbol{j}\}\!\}} q_i^m \Delta c_i^m\,, \end{aligned}$$

and this inequality is strict if $q > 0$ and $c_k > c_{\boldsymbol{j}}$. □

The optimality property in Theorem A.5 can be reinterpreted as in Remark A.1, *i.e.* $\boldsymbol{j}^k$ also minimizes $c_{\boldsymbol{j}}$ among $\boldsymbol{j}$ with $\omega_{\boldsymbol{j}} \geq \omega_k$.

A.3. **Numerical construction.** We consider numerical methods for constructing the sequence $(\boldsymbol{j}^k)_{k\in\mathbb{N}_0}$ from Section A.2. To this end, we assume that, for each $m \in \mathcal{M}$, the sequences $(c_j^m)_{j\in\mathbb{N}_0}$ and $(\omega_j^m)_{j\in\mathbb{N}_0}$ are stored as linked lists.

Initially, we consider the case that $\mathcal{M}$ is finite with $\#\mathcal{M} =: M$. To construct $(\boldsymbol{j}^k)_{k\in\mathbb{N}_0}$, we use a list $\mathcal{N}$ of the triples $(m, j_m^k, q_{j_m^k}^m)$, sorted in ascending order with respect to $\prec$. This list may be realized as a linked list or as a tree. The data structure must provide functions `PopMin` for removing the minimal element from the list, and `Insert` for inserting a new element into the list.

`NextOpt`$[\boldsymbol{j}, \mathcal{N}] \mapsto [\boldsymbol{j}, m, \mathcal{N}]$
$m \longleftarrow$ `PopMin`$(\mathcal{N})$
$j_m \longleftarrow j_m + 1$
$q \longleftarrow (\omega_{j_m+1}^m - \omega_{j_m}^m)/(c_{j_m+1}^m - c_{j_m}^m)$
$\mathcal{N} \longleftarrow$ `Insert`$(\mathcal{N}, (m, j_m, q))$

Proposition A.6. *Let $\mathcal{N}_0$ be initialized as $\{(m, 0, q_0^m)\,;\, m \in \mathcal{M}\}$ and $\boldsymbol{j}^0 := \mathbf{0} \in \mathbb{N}_0^{\mathcal{M}}$. Then the recursive application of*

$$\texttt{NextOpt}[\boldsymbol{j}^k, \mathcal{N}_k] \mapsto [\boldsymbol{j}^{k+1}, m_k, \mathcal{N}_{k+1}] \tag{A.9}$$

constructs the sequence $(\boldsymbol{j}^k)_{k\in\mathbb{N}_0}$ as defined above. Initialization of the data structure $\mathcal{N}_0$ requires $O(M \log M)$ operations and $O(M)$ memory. One step of (A.9) *requires $O(M)$ operations if $\mathcal{N}$ is realized as a linked list, and $O(\log M)$ operations if $\mathcal{N}$ is realized as a tree. The total number of operations required by the first k steps is $O(kM)$ in the former case and $O(k \log M)$ in the latter. In both cases, the total memory requirement for the first k steps is $O(M + k)$.*

Proof. Recursive application of `NextOpt` as in (A.9) constructs the sequence $(\boldsymbol{j}^k)_{k\in\mathbb{N}_0}$ by Lemma A.4 and the definition of $\prec$. In the k-th step, the element m_k is removed from $\mathcal{N}$ and reinserted in a new position. Therefore, the size of $\mathcal{N}$ remains constant at M. The computational cost of (A.9) is dominated by the insert operation on $\mathcal{N}$, which has the complexity stated above. □

We turn to the case that $\mathcal{M}$ is countably infinite. By enumerating the elements of $\mathcal{M}$, it suffices to consider $\mathcal{M} = \mathbb{N}$. We assume in this case that the sequence $(q_0^m)_{m\in\mathcal{M}}$ is nonincreasing.

As above, we use a list $\mathcal{N}$ of triples $(m, j_m^k, q_{j_m^k}^m)$ to construct the sequence $(\boldsymbol{j}^k)_{k\in\mathbb{N}_0}$. However, $\mathcal{N}$ should only store triples for which m is a candidate for the next value of m_k, *i.e.* all m with $j_m^k \neq 0$ and the smallest m with $j_m^k = 0$. As in the finite case, $\mathcal{N}$ can be realized as a linked list or a tree. The data structure should provide functions for removing the smallest element with respect to the ordering $\prec$, and for inserting a new element.

Proposition A.7. *Let $\mathcal{N}_0$ be initialized as $\{(1, 0, q^1)\}$, $M_0 := 1$ and $\boldsymbol{j}^0 := \mathbf{0} \in \mathbb{N}_0^{\mathcal{M}}$. Then the recursion*

$$\texttt{NextOptInf}[\boldsymbol{j}^k, \mathcal{N}_k, M_k] \mapsto [\boldsymbol{j}^{k+1}, m_k, \mathcal{N}_{k+1}, M_{k+1}] \tag{A.10}$$

constructs the sequence $(\boldsymbol{j}^k)_{k\in\mathbb{N}_0}$ as defined above. For all $k \in \mathbb{N}_0$, the ordered set $\mathcal{N}_k$ contains exactly M_k elements, and $M_k \leq k$. The k-th step of (A.10) *requires $O(k)$ operations if $\mathcal{N}$ is realized as a linked list, and $O(\log k)$ operations if $\mathcal{N}$ is realized as a tree. The total number of operations required by the first k steps is $O(k^2)$ in the former case and $O(k \log k)$ in the latter. In both cases, the total memory requirement for the first k steps is $O(k)$.*

`NextOptInf`$[j, \mathcal{N}, M] \mapsto [j, m, \mathcal{N}, M]$

$m \longleftarrow$ `PopMin`$(\mathcal{N})$
$j_m \longleftarrow j_m + 1$
$q \longleftarrow (\omega^m_{j_m+1} - \omega^m_{j_m})/(c^m_{j_m+1} - c^m_{j_m})$
$\mathcal{N} \longleftarrow$ `Insert`$(\mathcal{N}, (m, j_m, q))$
if $m = M$ **then**
  $M \longleftarrow M + 1$
  $q \longleftarrow (\omega^M_1 - \omega^M_0)/c^M_1$
  $\mathcal{N} \longleftarrow$ `Insert`$(\mathcal{N}, (M, 1, q))$

Proof. It follows from the definitions that recursive application of `NextOptInf` as in (A.10) constructs the sequence $(j^k)_{k\in\mathbb{N}_0}$. In the k-th step, the element m_k is removed from $\mathcal{N}$ and reinserted in a new position. If $m_k = M$, an additional element is inserted, and M is incremented. Therefore, the number of elements in $\mathcal{N}$ is M, and $M \le k$. The computational cost of (A.10) is dominated by the insert operation on $\mathcal{N}$, which has the complexity stated above, see *e.g.* [12]. □

Remark A.8. As mentioned above, $(c^m_j)_{j\in\mathbb{N}_0}$ and $(\omega^m_j)_{j\in\mathbb{N}_0}$ are assumed to be stored in a linked list for each $m \in \mathcal{M}$. By removing the first element from the $\mathcal{M}_k$-th list in the k-th step of (A.9) or (A.10), `NextOpt` and `NextOptInf` only ever access the first two elements of one of these lists, which takes $O(1)$ time. The memory locations of the lists can be stored in a hash table for efficient access. ⌟

Remark A.9. An appropriate way to store $(j^k)_{k\in\mathbb{N}_0}$ is to collect $(m_k)_{k\in\mathbb{N}_0}$ in a linked list. Then j^k can be reconstructed by reading the first k elements of this list, which takes $O(k)$ time independently of the size of the list. Also, the total memory requirement is $O(\bar{k})$ if the first $\bar{k}$ elements are stored. ⌟

References

[1] I. M. Babuška, R. Tempone, and G. E. Zouraris, *Galerkin finite element approximations of stochastic elliptic partial differential equations*, SIAM J. Numer. Anal., 42 (2004), pp. 800–825 (electronic).

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

[3] A. Barinka, S. Dahlke, and W. Dahmen, *Adaptive application of operators in standard representation*, Adv. Comput. Math., 24 (2006), pp. 5–34.

[4] H. Bauer, *Wahrscheinlichkeitstheorie*, de Gruyter Lehrbuch. [de Gruyter Textbook], Walter de Gruyter & Co., Berlin, fifth ed., 2002.

[5] M. Bieri, R. Andreev, and C. Schwab, *Sparse tensor discretization of elliptic SPDEs*, SIAM J. Sci. Comput., 31 (2009/10), pp. 4281–4304.

[6] M. Bieri and C. Schwab, *Sparse high order FEM for elliptic sPDEs*, Comput. Methods Appl. Mech. Engrg., 198 (2009), pp. 1149–1170.

[7] P. A. Cioica, S. Dahlke, N. Döhring, S. Kinzel, F. Lindner, T. Raasch, K. Ritter, and R. L. Schilling, *Adaptive wavelet methods for elliptic stochastic partial differential equations*, Tech. Rep. 77, DFG Schwerpunktprogramm 1324, 2010.

[8] A. Cohen, *Numerical analysis of wavelet methods*, vol. 32 of Studies in Mathematics and its Applications, North-Holland Publishing Co., Amsterdam, 2003.

[9] A. Cohen, W. Dahmen, and R. A. DeVore, *Adaptive wavelet methods for elliptic operator equations: convergence rates*, Math. Comp., 70 (2001), pp. 27–75 (electronic).

[10] ———, *Adaptive wavelet methods. II. Beyond the elliptic case*, Found. Comput. Math., 2 (2002), pp. 203–245.

[11] A. Cohen, R. A. DeVore, and C. Schwab, *Analytic regularity and polynomial approximation of parametric stochastic elliptic PDEs*, Tech. Rep. 2010-3, Seminar for Applied Mathematics, ETH Zürich, 2010. In review.

[12] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to algorithms*, MIT Press, Cambridge, MA, third ed., 2009.

[13] S. Dahlke, M. Fornasier, M. Primbs, T. Raasch, and M. Werner, *Nonlinear and adaptive frame approximation schemes for elliptic PDEs: theory and numerical experiments*, Numer. Methods Partial Differential Equations, 25 (2009), pp. 1366–1401.
[14] S. Dahlke, M. Fornasier, and T. Raasch, *Adaptive frame methods for elliptic operator equations*, Adv. Comput. Math., 27 (2007), pp. 27–63.
[15] S. Dahlke, T. Raasch, M. Werner, M. Fornasier, and R. Stevenson, *Adaptive frame methods for elliptic operator equations: the steepest descent approach*, IMA J. Numer. Anal., 27 (2007), pp. 717–740.
[16] W. Dahmen, T. Rohwedder, R. Schneider, and A. Zeiser, *Adaptive eigenvalue computation: complexity estimates*, Numer. Math., 110 (2008), pp. 277–312.
[17] M. K. Deb, I. M. Babuška, and J. T. Oden, *Solution of stochastic partial differential equations using Galerkin finite element techniques*, Comput. Methods Appl. Mech. Engrg., 190 (2001), pp. 6359–6372.
[18] R. A. DeVore, *Nonlinear approximation*, in Acta numerica, 1998, vol. 7 of Acta Numer., Cambridge Univ. Press, Cambridge, 1998, pp. 51–150.
[19] T. J. Dijkema, C. Schwab, and R. Stevenson, *An adaptive wavelet method for solving high-dimensional elliptic PDEs*, Constr. Approx., 30 (2009), pp. 423–455.
[20] P. Frauenfelder, C. Schwab, and R. A. Todor, *Finite elements for elliptic problems with stochastic coefficients*, Comput. Methods Appl. Mech. Engrg., 194 (2005), pp. 205–228.
[21] T. Gantumur, H. Harbrecht, and R. Stevenson, *An optimal adaptive wavelet method without coarsening of the iterands*, Math. Comp., 76 (2007), pp. 615–629 (electronic).
[22] C. J. Gittelson, *Adaptive Galerkin Methods for Parametric and Stochastic Operator Equations*, PhD thesis, ETH Zürich, 2011. ETH Dissertation No. 19533.
[23] ———, *An adaptive stochastic Galerkin method*, Tech. Rep. 2011-11, Seminar for Applied Mathematics, ETH Zürich, 2011.
[24] ———, *Adaptive stochastic Galerkin methods: Beyond the elliptic case*, Tech. Rep. 2011-12, Seminar for Applied Mathematics, ETH Zürich, 2011.
[25] ———, *Stochastic Galerkin approximation of operator equations with infinite dimensional noise*, Tech. Rep. 2011-10, Seminar for Applied Mathematics, ETH Zürich, 2011.
[26] ———, *Uniformly convergent adaptive methods for parametric operator equations*, Tech. Rep. 2011-19, Seminar for Applied Mathematics, ETH Zürich, 2011.
[27] H. G. Matthies and A. Keese, *Galerkin methods for linear and nonlinear elliptic stochastic partial differential equations*, Comput. Methods Appl. Mech. Engrg., 194 (2005), pp. 1295–1331.
[28] A. Metselaar, *Handling Wavelet Expansions in Numerical Methods*, PhD thesis, University of Twente, 2002.
[29] H. Nguyen and R. Stevenson, *Finite element wavelets with improved quantitative properties*, J. Comput. Appl. Math., 230 (2009), pp. 706–727.
[30] C. Schwab and C. J. Gittelson, *Sparse tensor discretization of high-dimensional parametric and stochastic PDEs*, in Acta Numerica, vol. 20 of Acta Numer., Cambridge Univ. Press, Cambridge, 2011, pp. 291–467.
[31] C. Schwab and R. Stevenson, *Space-time adaptive wavelet methods for parabolic evolution problems*, Math. Comp., 78 (2009), pp. 1293–1318.
[32] R. Stevenson, *Adaptive solution of operator equations using wavelet frames*, SIAM J. Numer. Anal., 41 (2003), pp. 1074–1100 (electronic).
[33] ———, *On the compressibility of operators in wavelet coordinates*, SIAM J. Math. Anal., 35 (2004), pp. 1110–1132 (electronic).
[34] ———, *Adaptive wavelet methods for solving operator equations: an overview*, in Multiscale, nonlinear and adaptive approximation, Springer, Berlin, 2009, pp. 543–597.
[35] R. A. Todor and C. Schwab, *Convergence rates for sparse chaos approximations of elliptic problems with stochastic coefficients*, IMA J. Numer. Anal., 27 (2007), pp. 232–261.
[36] X. Wan and G. E. Karniadakis, *An adaptive multi-element generalized polynomial chaos method for stochastic differential equations*, J. Comput. Phys., 209 (2005), pp. 617–642.
[37] ———, *Multi-element generalized polynomial chaos for arbitrary probability measures*, SIAM J. Sci. Comput., 28 (2006), pp. 901–928 (electronic).
[38] D. Xiu, *Fast numerical methods for stochastic computations: a review*, Commun. Comput. Phys., 5 (2009), pp. 242–272.
[39] D. Xiu and G. E. Karniadakis, *The Wiener-Askey polynomial chaos for stochastic differential equations*, SIAM J. Sci. Comput., 24 (2002), pp. 619–644 (electronic).

Seminar für Angewandte Mathematik, Eidgenössische Technische Hochschule Zürich, Rämistrasse 101, CH-8092 Zurich, Switzerland
E-mail address: claude.gittelson@sam.math.ethz.ch

Research Reports

No.	Authors/Title
11-37	*C.J. Gittelson* Adaptive wavelet methods for elliptic partial differential equations with random operators
11-36	*A. Barth and A. Lang* Milstein approximation for advection–diffusion equations driven by multiplicative noncontinuous martingale noises
11-35	*A. Lang* Almost sure convergence of a Galerkin approximation for SPDEs of Zakai type driven by square integrable martingales
11-34	*F. Müller, D.W. Meyer and P. Jenny* Probabilistic collocation and Lagrangian sampling for tracer transport in randomly heterogeneous porous media
11-33	*R. Bourquin, V. Gradinaru and G.A. Hagedorn* Non-adiabatic transitions near avoided crossings: theory and numerics
11-32	*J. Šukys, S. Mishra and Ch. Schwab* Static load balancing for multi-level Monte Carlo finite volume solvers
11-31	*C.J. Gittelson, J. Könnö, Ch. Schwab and R. Stenberg* The multi-level Monte Carlo Finite Element Method for a stochastic Brinkman problem
11-30	*A. Barth, A. Lang and Ch. Schwab* Multi-level Monte Carlo Finite Element method for parabolic stochastic partial differential equations
11-29	*M. Hansen and Ch. Schwab* Analytic regularity and nonlinear approximation of a class of parametric semilinear elliptic PDEs
11-28	*R. Hiptmair and S. Mao* Stable multilevel splittings of boundary edge element spaces
11-27	*Ph. Grohs* Shearlets and microlocal analysis
11-26	*H. Kumar* Implicit-explicit Runge-Kutta methods for the two-fluid MHD equations
11-25	*H. Papasaika, E. Kokiopoulou, E. Baltsavias, K. Schindler and D. Kressner* Sparsity-seeking fusion of digital elevation models
11-24	*H. Harbrecht and J. Li* A fast deterministic method for stochastic elliptic interface problems based on low-rank approximation