Adaptive Methods for Boundary Integral Equations – Complexity and Convergence Estimates*

Wolfgang Dahmen, Helmut Harbrecht, Reinhold Schneider

March 28, 2005

Abstract

This paper is concerned with developing numerical techniques for the adaptive application of global operators of potential type in wavelet coordinates. This is a core ingredient for a new type of adaptive solvers that has so far been explored primarily for PDEs. We shall show how to realize asymptotically optimal complexity in the present context of global operators. "Asymptotically optimal" means here that any target accuracy can be achieved at a computational expense that stays proportional to the number of degrees of freedom (within the setting determined by an underlying wavelet basis) that would ideally be necessary for realizing that target accuracy if full knowledge about the unknown solution were given. The theoretical findings are supported and quantified by first numerical experiments.

1 Introduction

Recent developments of adaptive wavelet solvers rely in a crucial way on the ability of applying the relevant operator in wavelet coordinates to any finitely supported input vector within any desirable accuracy tolerance. So far this concept has been analyzed and to some extent realized numerically for local operators such as compositions or differential operators, see [8]. The objective of this paper is to explore the potential of this paradigm also for *global* operators such as boundary integral operators of potential type.

1.1 Problem Formulation and Background

Given a Hilbert space $\mathcal{H}$ and a linear operator $\mathcal{L}$ that takes $\mathcal{H}$ onto its normed dual $\mathcal{H}'$, we wish to find for any given $f \in \mathcal{H}'$ the solution $u \in \mathcal{H}$ of the operator equation $\mathcal{L}u = f$ in the sense that

$$\langle v, \mathcal{L}u \rangle = \langle v, f \rangle, \qquad v \in \mathcal{H} \tag{1.1}$$

where $\langle \cdot, \cdot \rangle$ is the duality pairing on $\mathcal{H} \times \mathcal{H}'$. It will always be assumed to agree with the standard L_2-inner product on the underlying domain or manifold whenever the arguments are square integrable.

*This research was supported in part by the EEC Human Potential Programme under contract HPRN-CT-2002-00286, "Breaking Complexity".

We are interested in this paper mainly in situations where $\mathcal{L}$ is a *global* operator. The following classical operators of potential type will serve as model cases throughout this paper.

$$\mathcal{L} = \mathcal{V} : \mathcal{H} := H^{-1/2}(\Gamma) \to H^{1/2}(\Gamma), \quad (\mathcal{V}v)(x) = \int_\Gamma \frac{v(y)}{4\pi\|x-y\|} d\sigma_y \tag{1.2}$$

$$\mathcal{L} = \mathcal{K} \pm \frac{1}{2} : \mathcal{H} := L_2(\Gamma) \to L_2(\Gamma), \quad (\mathcal{K}v)(x) = \int_\Gamma \frac{(x-y)^T n_y v(y)}{4\pi\|x-y\|^3} d\sigma_y \tag{1.3}$$

$$\mathcal{L} = \mathcal{W} : \mathcal{H} := H^{1/2}(\Gamma) \to H^{-1/2}(\Gamma), \quad (\mathcal{W}v)(x) = -\frac{\partial}{\partial n_x} \int_\Gamma \frac{(x-y)^T n_y v(y)}{4\pi\|x-y\|^3} d\sigma_y, \tag{1.4}$$

where $\|\cdot\|$ is the Euclidean norm in $\mathbb{R}^3$, n_y denotes the unit outer normal at the point y of the boundary manifold Γ and $H^s(\Gamma)$ are the usual Sobolev spaces of order s on Γ. It is always understood that for $s < 0$ one has $H^s(\Gamma) = (H^{-s}(\Gamma))'$. It is well-known that e.g. the Laplace equation $\Delta U = 0$ on some domain $\Omega \subset \mathbb{R}^3$ with either Dirichlet or Neumann boundary conditions can be transformed into a boundary integral equation (involving e.g. $\mathcal{V}, \mathcal{K}$ in the case of a Dirichlet Problem, or $\mathcal{W}$ for a Neumann problem). Moreover, in all these cases it is well-known that (1.1) is *well-posed* in the sense that there exist positive bounded constants c_L, C_L such that

$$c_L\|v\|_{\mathcal{H}} \le \|\mathcal{L}v\|_{\mathcal{H}'} \le C_L\|v\|_{\mathcal{H}}, \quad \forall \ v \in \mathcal{H}. \tag{1.5}$$

In fact, this can be derived in many cases of classical boundary integral equations from the validity of a Gårding inequality together with the injectivity of $\mathcal{L}$. For the case (1.3) on Lipschitz domains (1.5) was established by Verchota [36] although in this case the validity of a Gårding inequality is still an open problem. Moreover, we shall always assume that $\mathcal{L}$ has the form

$$(\mathcal{L}v)(x) = av(x) + \int_\Gamma K(x,y)v(y)d\sigma_y, \tag{1.6}$$

where for $\mathcal{H} = H^q(\Gamma)$

$$\left|\partial_x^\alpha \partial_y^\beta K(x,y)\right| \lesssim \operatorname{dist}(x,y)^{-(d+2q+|\alpha|+|\beta|)}. \tag{1.7}$$

A standard way of solving (1.1) numerically is to choose some finite dimensional trial space $S_h \subset \mathcal{H}$ and some test space T_h of equal dimension to determine $u_h \in S_h$ such that

$$\langle w_h, \mathcal{L}u_h\rangle = \langle w_h, f\rangle \quad \forall \ w_h \in T_h. \tag{1.8}$$

This covers Petrov Galerkin, in particular collocation schemes. For simplicity we confine the subsequent discussion to Galerkin schemes, i.e. $T_h = S_h$.

Once the linear system induced by (1.8) has been assembled one looks for possibly efficient ways of solving such systems numerically. The central goal is to come up with solution schemes that scale linearly in the size $N = \dim S_h$ of the problem with regard to storage and arithmetical operations (excluding for the moment storage access for simplicity). At any rate, a lower bound for the computational work is given by the number of degrees of freedom induced by the (a priori) choice of S_h. In the context of boundary integral equations one faces in addition the following well known obstructions to a linear scaling of the computational work:

(I) For operators $\mathcal{L}$ of nonzero order (such as $\mathcal{V}, \mathcal{W}$ above) the systems grow increasingly ill-conditioned when N becomes larger.

(II) The matrices are in general densely populated which is perhaps the most severe obstacle.

Several concepts have been developed to deal with these difficulties. They can be grouped into two main categories, namely *cluster methods* like *fast multipole methods* (FMM) or *panel clustering* and abstractions like *Hierarchical Matrices* (HM) [23, 22, 21] on one hand, and *wavelet compression* (WC) on the other hand see e.g. [2, 16, 28, 15, 24, 33]. (FMM) aims at efficiently applying the operator $\mathcal{A}$ to a given input thereby addressing mainly (II). Since it exploits the smoothness of the integral kernel in $\mathbb{R}^{d+1}$ when the boundary manifold has dimension d, it is very robust with respect to the geometry of the boundary surface. Except for the use of approximate inverses based on hierarchical matrices these methods do usually not incorporate any preconditioning which, however, is not an issue for second kind formulations. Once a trial space has been chosen, (WC) aims at assembling a compressed version of the wavelet representation of the operator. Meanwhile compression strategies and corresponding quadrature techniques have been developed which can be shown to yield *linearly scaling* solvers. This means the perturbed problems (due to compression) still realize *discretization error accuracy* while the compressed matrices exhibit only $O(N)$ nonzero entries [15, 24, 33]. It is important to note here that preconditioning is easily incorporated in this approach. Experience shows that the major part of the computational work is absorbed by computing the compressed matrix while each matrix/vector application of the compressed matrix is then much more efficient than in the (FMM) approach.

1.2 Objectives and Contents

Nevertheless, the following points remain unsatisfactory. Since the boundary manifolds are often only piecewise smooth the solutions typically exhibit only a low degree of global regularity. Thus the discretization error is still large compared to the invested number of degrees of freedom when working with quasi-uniform meshes. Since, especially for global operators, the size of the problem matters very much this calls for *adaptive solution* strategies. While such concepts have been extensively studied for PDEs it is fair to say that they are less understood for boundary integral equations, again a superficial reason being the global nature of the operator $\mathcal{L}$ in this case. Even though meanwhile efficient and reliable a posteriori error estimators are known [19] to our knowledge no *convergence* results for any adaptive refinement scheme based on such a posteriori bounds are available that guarantee a fixed error reduction per refinement step. Even less is known concerning corresponding *complexity estimates* or *error rates* relating the generated number of degrees of freedom to the achieved accuracy.

A central objective of this paper is to derive such estimates for a new adaptive wavelet scheme for boundary integral equations of the above potential type. Moreover, the asymptotical theoretical results are complemented by first numerical experiments of such type.

The remainder of the paper is organized as follows. Section 2 is devoted to the adaptive paradigm including the essentials of the complexity analysis. Section 3 contains the main part of this paper, namely the ingredients for the *numerical realization* of residuals in wavelet coordinates and the corresponding complexity analysis. In Section 4 we discuss several modifications respectively specifications of the prototype scheme that lead to significant quantitative improvements. This is supported quantitatively by several numerical experiments that to our knowledge are the first of this sort.

2 An Adaptive Paradigm

Adaptive strategies based on *wavelet concepts* appear particularly natural since boundary integral operators are nearly sparse in wavelet coordinates. This suggests applying the following general paradigm (whose concrete formulations have so far been mainly studied for variational problems involving *local operators* [7, 8]) to the present context. It essentially consists of the following steps:

(i) Given an operator equation establish its (infinite dimensional) well-posedness in the sense of (1.5), which amounts to identifying the right space $\mathcal{H}$. This is done for the above examples.

(ii) Transform the original problem (1.1) into an *equivalent* one $\mathbf{A}\mathbf{u} = \mathbf{f}$, that is now well-posed in the space $\ell_2(\mathcal{J})$ of square summable sequences (with respect to some infinite index set).

(iii) Use the well-posedness in $\ell_2(\mathcal{J})$ in order to contrive an (idealized) iteration scheme (for the full infinite dimensional problem) that exhibits a fixed error reduction rate

$$\|\mathbf{u}^{n+1} - \mathbf{u}\|_{\ell_2(\mathcal{J})} \leq \rho \|\mathbf{u}^n - \mathbf{u}\|_{\ell_2(\mathcal{J})}, \quad n = 0, 1, 2 \ldots, \tag{2.1}$$

where $\mathbf{u} \in \ell_2(\mathcal{J})$ is the coefficient array of the unknown solution of the transformed problem.

(iii) Realize the idealized iteration approximately by means of *adaptive* applications of $\mathcal{L}$ within suitable dynamically updated accuracy tolerances.

Obviously a core ingredient of such a strategy is the approximate application of (infinite dimensional) operators. By now an abstract criterion is known (referred to as s^*-sparsity as explained later below) in terms of conditions on the numerical approximation of the (infinite dimensional) *residuals* $\mathbf{f} - \mathbf{A}\mathbf{u}^n$ that ensures asymptotically optimal complexity in the following sense. If (in the given discretization framework) the unknown solution can be approximated with an optimal choice of N degrees of freedom at a rate N^{-s} (for some range of orders s), then the adaptive scheme matches this rate by producing for any target accuracy ε an approximate solution u_ε such that $\|u - u_\varepsilon\|_{\mathcal{H}} \leq \varepsilon$ at a computational expense that grows at most like $\varepsilon^{-1/s}$ when ε tends to zero, [7, 8]. However, the concrete realization of the announced criterion (s^*-sparsity) has been confined essentially to *local operators*. Our objective is to develop here adaptive schemes along such lines for global operators of the above type.

It should be mentioned that the above concepts applies also to noncoercive and nonlinear problems with optimal results for local operators [8, 9]. Here we shall confine the discussion for simplicity to linear positive symmetric problems, extensions being clear from the general treatment in [7, 8].

2.1 An Equivalent ℓ_2-Problem

As mentioned before, it is natural to base the transformation required in the above step (ii) on suitable *wavelet bases*. To this end, suppose that $\Psi = \{\psi_\lambda : \lambda \in \mathcal{J}\} \subset \mathcal{H}$ is a *Riesz basis* for $\mathcal{H}$, i.e. every $v \in \mathcal{H}$ has a unique expansion $v = \sum_{\lambda\in\mathcal{J}} v_\lambda \psi_\lambda$ with coefficient array $\mathbf{v} = (v_\lambda)_{\lambda\in\mathcal{J}}$ and there exist constants c_Ψ, C_Ψ such that

$$c_\Psi \|\mathbf{v}\|_{\ell_2(\mathcal{J})} \le \|v\|_{\mathcal{H}} \le C_\Psi \|\mathbf{v}\|_{\ell_2(\mathcal{J})}. \tag{2.2}$$

Setting

$$\mathbf{A} := (a(\psi_\lambda, \psi_\nu))_{\lambda,\nu\in\mathcal{J}}, \quad \mathbf{f} := (\langle \psi_\lambda, f\rangle)_{\lambda\in\mathcal{J}}, \tag{2.3}$$

it is not hard to establish the following facts.

Theorem 2.1 *The problem (1.1) is equivalent to*

$$\mathbf{A}\mathbf{u} = \mathbf{f}, \tag{2.4}$$

where $u = \sum_{\lambda\in\mathcal{J}} u_\lambda \psi_\lambda$ *is the solution to (1.1) with (unknown) coefficient array* $\mathbf{u} = (u_\lambda)_{\lambda\in\mathcal{J}}$. *Moreover, (2.4) is well-posed in* $\ell_2(\mathcal{J})$, *i.e.*

$$c_\Psi^2 c_L \|\mathbf{v}\|_{\ell_2(\mathcal{J})} \le \|\mathbf{A}\mathbf{v}\|_{\ell_2(\mathcal{J})} \le C_\Psi^2 C_L \|\mathbf{v}\|_{\ell_2(\mathcal{J})}, \quad \mathbf{v} \in \ell_2(\mathcal{J}). \tag{2.5}$$

In fact, (2.5) follows easily from combining (2.2) and (1.5) (see e.g. [16, 8]).

In the present context we shall employ certain spline wavelets based on parametric liftings of local bases on the unit cube satisfying (2.2) for the relevant choices of $\mathcal{H}$, see [4, 5, 11, 17] and [24] for implementation details and structural improvements. We shall address the specific properties when needed. For the time being it suffices to know that the index set $\mathcal{J}$ consists of a finite part $\mathcal{J}_\phi$ referring to coarse scale scaling function type basis functions and an infinite part $\mathcal{J}_\psi$ referring to the "true" wavelets having vanishing moments of a certain order. Moreover, each index $\lambda \in \mathcal{J}$ incorporates information about the scale $j = |\lambda|$, the spatial location $k = k(\lambda)$ and the type $e = e(\lambda)$ of the wavelet ψ_λ. Recall that for tensor product type wavelets in d dimensions one has $2^d - 1$ different types associated with the same location index. We work exclusively with compactly supported wavelets whose supports scales canonically as $\operatorname{diam}(\operatorname{supp}\psi_\lambda) \sim 2^{-|\lambda|}$. The infinite matrix $\mathbf{A}$ is called the *standard wavelet representation* of the operator $\mathcal{L}$.

We close this section with a slight specification of the norm equivalence (2.2). A typical approach taken in all the quoted references is to construct an *anchor pair* of biorthogonal Riesz bases $\Psi^\circ, \tilde{\Psi}^\circ$ in $L_2(\Gamma)$. It then turns out that when $\Psi^\circ \subset H^s(\Gamma)$ for $s \in [0, \gamma)$ and

$\tilde{\Psi}^\circ \subset H^s(\Gamma)$ for $s \in [0, \tilde{\gamma})$ then properly scaled versions of Ψ° and $\tilde{\Psi}^\circ$ form Riesz bases for a whole range of Sobolev spaces which for Ψ° reads as follows. For any $s \in (-\tilde{\gamma}, \gamma)$ there exist positive constants c_s, C_s such that

$$c_s \|(v_\lambda)_{\lambda\in\mathcal{J}}\|_{\ell_2(\mathcal{J})} \leq \|\sum_{\lambda\in\mathcal{J}} v_\lambda 2^{-s|\lambda|}\psi_\lambda^\circ\|_{H^s(\Gamma)} \leq C_s \|(v_\lambda)_{\lambda\in\mathcal{J}}\|_{\ell_2(\mathcal{J})}, \quad (v_\lambda)_{\lambda\in\mathcal{J}} \in \ell_2(\mathcal{J}). \quad (2.6)$$

In the present context one should have $q, -q \in (-\tilde{\gamma}, \gamma)$ when $\mathcal{H} = H^q(\Gamma)$. The particular basis appearing in (2.2) is then obtained with $\psi_\lambda := 2^{-q|\lambda|}\psi_\lambda^\circ$, $\lambda \in \mathcal{J}$, and we shall from now on always assume this to be the case. It will be convenient to view Ψ as an infinite vector whose components are the basis functions ψ_λ (in some fixed but unspecified order) in order to express this scaling relation in the following compact form

$$\Psi = \mathbf{D}\Psi^\circ, \qquad \mathbf{D} := (2^{-q|\lambda|}\delta_{\lambda,\lambda'})_{\lambda,\lambda'\in\mathcal{J}}. \quad (2.7)$$

2.2 An Idealized Iteration

Suppose first that $\mathbf{A}$ is symmetric positive definite. Clearly, (2.5) implies that there is some $\alpha > 0$ such that $(\mathbf{I} - \alpha\mathbf{A})$ is a contraction in $\ell_2(\mathcal{J})$. Hence the iteration

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \alpha(\mathbf{f} - \mathbf{A}\mathbf{u}^n), \quad n = 0, 1, 2, \ldots, \quad (2.8)$$

of Richardson type satisfies (2.1). Of course, a smaller ρ can obtained by choosing a perhaps stage dependent step size $\alpha = \alpha_n$. Moreover, as long as (2.5) holds (2.1) can be realized by finding suitable preconditioners $\mathbf{C}_n$ in

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \mathbf{C}_n(\mathbf{f} - \mathbf{A}\mathbf{u}^n), \quad n = 0, 1, 2, \ldots, \quad (2.9)$$

even when $\mathbf{A}$ is not symmetric. For instance, in principle, $\mathbf{C}_n = \alpha_n\mathbf{A}^T$ always works although the quantitative performance may suffer from the resulting squared (although bounded) condition number. Now in addition to applying $\mathbf{A}$ one also has to apply $\mathbf{A}^T$ which, however, is a problem of the same type. In general one should think of $\mathbf{C}_n$ as the application of some sort of approximate inverse in order to improve the quantitative performance of the iteration. This will actually be done in our numerical examples, see Section 4.

2.3 A Perturbed Iteration Scheme

So far we have made only use of the norm equivalence (2.2) in the relevant energy space to arrive at step (iii) of the paradigm. The envisaged numerical realization of (2.8) now requires for any given finitely supported input array $\mathbf{v}$ the approximate realization of $\mathbf{A}\mathbf{v}$ within any desired accuracy tolerance.

In order to outline first the essential structure of the adaptive scheme we postpone for the moment describing concrete ways of performing such approximate realizations but *assume* at this point that a routine with the following properties is available:

$\textsc{Apply}\,[\eta, \mathbf{A}, \mathbf{v}] \to \mathbf{w}$:
DETERMINES FOR ANY FINITELY SUPPORTED INPUT $\mathbf{v}$ A FINITELY SUPPORTED OUTPUT $\mathbf{w}_\eta$ SUCH THAT

$$\|\mathbf{A}\mathbf{v} - \mathbf{w}_\eta\|_{\ell_2(\mathcal{J})} \le \eta. \tag{2.10}$$

We need one further ingredient which is known to be important for controlling the complexity of the perturbed realizations of (2.8), namely a *coarsening* routine [6, 8]:

$\textsc{Coarse}\,[\eta, \mathbf{v}] \to \mathbf{v}_\eta$:
DETERMINES FOR ANY FINITELY SUPPORTED INPUT $\mathbf{v}$ AN OUTPUT $\mathbf{v}_\eta$ SUCH THAT

$$\|\mathbf{v} - \mathbf{v}_\eta\|_{\ell_2(\mathcal{J})} \le \eta \tag{2.11}$$

SUCH THAT THE SUPPORT OF $\mathbf{v}_\eta$ IS (NEARLY) MINIMAL (SUBJECT TO POSSIBLE CONSTRAINTS ON THE CORRESPONDING INDEX DISTRIBUTIONS TO BE DETAILED LATER).

Possible constraints on the distribution of coefficients mentioned in COARSE concern the natural *tree-structure* of the typical index sets $\mathcal{J}$. Roughly speaking each index λ of level $j = |\lambda|$ can be viewed as a *child* of some index λ' of level $|\lambda'| = j - 1$ where typically $\operatorname{supp}\psi_\lambda \subset \operatorname{supp}\psi_{\lambda'}$, see [8, 9]. From a theoretical point of view such tree structures are, according to [7], *not necessary* for linear operator equations. In fact, they make coarsening and its analysis more delicate. But tree structures do play an essential role for the realization of efficient routines APPLY and are therefore considered here.

The realization of the routine COARSE that scales linearly in the size of the input array while respecting tree structure is not obvious. It can be based though on the tree approximation scheme from [3], see [8].

The realization of an adaptive scheme always hinges on having complete access to the data as they are given by the user. We shall therefore always assume that the array $\mathbf{f}$ is already finite and ordered by size (either because it is initially given that way or because in some preprocessing step it has been arranged to match some "ideal data" with in some accuracy tolerance which suffices for the final target accuracy $\bar{\varepsilon}$ for the solution, recall (2.5)). Therefore the complexity estimates derived here refer strictly speaking to the approximate inversion of the operator, given full access to the data.

The approximate realization of (2.9) requires approximating the *weighted residual* $\mathbf{C}(\mathbf{f} - \mathbf{A}\mathbf{v})$ for any given finite array $\mathbf{v}$. In the present context such a residual approximation is required to perform the following task:

$\textsc{Res}\,[\eta, \mathbf{C}, \mathbf{A}, \mathbf{v}] \to \mathbf{r}$:
DETERMINES FOR ANY GIVEN TARGET ACCURACY $\eta \ge \bar{\varepsilon} > 0$ A FINITELY SUPPORTED ARRAY $\mathbf{r}$ SUCH THAT

$$\|\mathbf{C}(\mathbf{f} - \mathbf{A}\mathbf{v}) - \mathbf{r}\|_{\ell_2(\mathcal{J})} \le \eta \tag{2.12}$$

For instance, when $\mathbf{A}$ is symmetric positive definite one can take $\mathbf{C} = \alpha\mathbf{I}$, for a suitable damping parameter in (2.8). A possible realization has the form

$$\alpha(\textsc{Coarse}\,[\eta/2\alpha, \mathbf{f}] - \textsc{Apply}\,[\eta/2\alpha, \mathbf{A}, \mathbf{v}]) \to \mathbf{r}, \tag{2.13}$$

provided that the scheme Apply is available. More generally, as long as (2.5) holds even though $\mathcal{L}$ is unsymmetric or indefinite, recall from Section 2.2 that one can take, in principle, $\mathbf{C} = \alpha\mathbf{A}^T$ again for a suitable damping parameter, [7]. The corresponding Res would then be given e.g. as follows:

Let $C_A := C_\Psi^2 C_L$ (see (2.5)) and compute

$$\begin{array}{l} \text{Coarse}\,[\eta/2C_A\alpha, \mathbf{f}] - \text{Apply}\,[\eta/2C_A\alpha, \mathbf{A}, \mathbf{v}]) \to \bar{\mathbf{r}}; \\ \alpha\text{Coarse}\,[\eta/2\alpha, \mathbf{A}^T, \bar{\mathbf{r}}] \to \mathbf{r}. \end{array} \tag{2.14}$$

As already mentioned a realization of Apply will be given later. Moreover, we shall discuss later ways of realizing more efficient preconditioners $\mathbf{C}$.

With these ingredients at hand, we can formulate now the following *prototype* solution scheme.

Solve$[\varepsilon, \mathbf{C}, \mathbf{A}, \mathbf{f}, \mathbf{u}^0, \beta, C^*] \to \bar{\mathbf{u}}(\varepsilon)$

(i) Choose some $\bar{\rho} \in (0,1)$. Set $\bar{\mathbf{u}}^0 = \mathbf{u}^0$ which is a given initial guess for which an initial bound ε_0 satisfying $\|\mathbf{u} - \mathbf{u}^0\|_{\ell_2(\mathcal{J})} \le \varepsilon_0$ is known (if no particular guess is given, one can take $\mathbf{u}^0 = 0$ and $\varepsilon_0 = c_L^{-1}\|\mathbf{f}\|_{\ell_2(\mathcal{J})}$). Set $j = 0$;

(ii) If $\varepsilon_j \le \varepsilon$ stop and output $\bar{\mathbf{u}}(\varepsilon) := \bar{\mathbf{u}}^j$; else set $\mathbf{v}^0 := \bar{\mathbf{u}}^j$ and $k = 0$

(ii.1) Set $\eta_k := \bar{\rho}^k \varepsilon_j$ and compute

$$\mathbf{r}^k = \text{Res}[\eta_k, \mathbf{C}, \mathbf{A}, \mathbf{f}, \mathbf{v}^k], \qquad \mathbf{v}^{k+1} = \mathbf{v}^k - \mathbf{r}^k.$$

(ii.2) If

$$\beta(\eta_k + \|\mathbf{r}^k\|_{\ell_2(\mathcal{J})}) \le \varepsilon_j/(2(1+2C^*)), \tag{2.15}$$

set $\tilde{\mathbf{v}} := \mathbf{v}^k$ and go to (iii). Else set $k+1 \to k$ and go to (ii.1).

(iii) Coarse$[\frac{2C^*\varepsilon_j}{2(1+2C^*)}, \tilde{\mathbf{v}}] \to \bar{\mathbf{u}}^{j+1}$, $\varepsilon_{j+1} = \varepsilon_j/2$, $j+1 \to j$, go to (ii).

The constant C^* depends on the coarsening scheme. If no constraints are imposed on the index distribution any fixed $C^* > 1/2$ will do. In the case of tree approximation C^* is the constant from [3]. The constant β depends only on the constants c_L, C_L, c_ψ, C_ψ in (2.2) and (1.5). With these choices the following has been shown in [8].

Proposition 2.2 *Given routines* Coarse, Res, *satisfying (2.11), (2.12), respectively, the scheme* Solve *terminates for any target accuracy $\varepsilon \ge \bar{\varepsilon}$ after finitely many steps and*

$$\|\mathbf{u} - \bar{\mathbf{u}}(\varepsilon)\|_{\ell_2(\mathcal{J})} \le \varepsilon.$$

Note that (2.2) implies that one has for the corresponding expansions the energy error estimate $\|u - \bar{u}(\varepsilon)\|_{\mathcal{H}} \le C_\psi\varepsilon$.

For a wide range of problems Solve can be shown to exhibit asymptotically optimal complexity in a sense to be made precise in the next section. In order to guarantee also a good quantitative performance the scheme should be specified so as to take up the specific nature of a given concrete application.

2.4 Some Ingredients of the Complexity Analysis

A lower bound for the computational complexity of SOLVE is, of course, the number of nonzero entries in the output $\bar{\mathbf{u}}(\varepsilon)$. Thus as a natural bench mark for measuring the performance of SOLVE the following *size/accuracy* rate suggests itself. Given all information about the (infinite) array $\mathbf{u}$ one can ask how small one can make the approximation error in $\ell_2(\mathcal{J})$ when using at most N (optimally) selected terms from the wavelet basis Ψ. This is usually referred to as *best N-term approximation*. When emphasizing that such a selection $\Lambda \subset \mathcal{J}$ of terms is completely arbitrary, i.e. $\Lambda \in \mathcal{J}^{\mathcal{J}}$ is *any* subset of $\mathcal{J}$, we speak of *unconstrained* N-term approximation. In contrast, depending on the context it is often useful or even necessary to impose some mild constraints on the distribution of *active* indices, i.e. active sets Λ should be taken from some strict subset $\mathcal{P} \subset \mathcal{J}^{\mathcal{J}}$ of the set of all subsets. Specifically, recall from [9] that $\mathcal{J}$ itself may be viewed as an *infinite tree* where the roots are the elements of $\mathcal{J}_\phi$. Moreover, denoting in the following $S_\lambda := \operatorname{supp} \psi_\lambda$, when $\hat{\lambda} \in \mathcal{J}$ is a *parent* of some $\lambda \in \mathcal{J}$ we have $S_\lambda \subset S_{\hat{\lambda}}$, see [9] for more details. We shall use the set $\mathbb{T}$ of all finite subtrees of $\mathcal{J}$ whose roots belong to $\mathcal{J}_\phi$. In this case we speak of *best tree N-term approximation*.

Obviously, a numerical scheme deserves to be termed (asymptotically) optimal if its *work/accuracy* rate stays proportional to the *size/accuracy* rate of the (underlying version of) best N-term approximation. To make this precise, consider the error of best $\mathcal{P}$-constrained N-term approximation in some Banach space X with respect to a given unconditional basis $\Theta = \{\theta_\lambda : \lambda \in \mathcal{J}\}$

$$\sigma^{\mathcal{P}}_{X,N}(v) := \inf \{\|v - \sum_{\lambda\in\Lambda} w_\lambda \theta_\lambda\|_X : w_\lambda \in \mathbb{R},\ \lambda \in \Lambda,\ \Lambda \in \mathcal{P},\ \#\Lambda \leq N\} \tag{2.16}$$

along with the corresponding *approximation spaces*

$$\mathcal{A}^s_X(\mathcal{P}) := \{v \in X : \sup_{N\in\mathbb{N}} N^s \sigma^{\mathcal{P}}_{X,N}(v) := |v|_{\mathcal{A}^s_X(\mathcal{P})} < \infty\}. \tag{2.17}$$

It is easy to see that $\|\cdot\|_{\mathcal{A}^s_X(\mathcal{P})} := \|\cdot\|_X + |\cdot|_{\mathcal{A}^s_X(\mathcal{P})}$ is a quasi-norm.

Obviously *any finite* expansion belongs to $\mathcal{A}^s_X(\mathcal{P})$ for *all* $s > 0$. Note that for the present choice of $\Theta = \Psi$ when $X = \mathcal{H}$ and $\Theta = \{(\delta_{\lambda,\nu})_{\nu\in\mathcal{J}} : \lambda \in \mathcal{J}\}$ the set of coordinate vectors when $X = \ell_2(\mathcal{J})$, the Riesz basis property says that

$$c_\psi \sigma^{\mathcal{P}}_{\ell_2(\mathcal{J}),N}(\mathbf{v}) \leq \sigma^{\mathcal{P}}_{\mathcal{H},N}(v) \leq C_\psi \psi \sigma^{\mathcal{P}}_{\ell_2(\mathcal{J}),N}(\mathbf{v}), \quad v = \sum_{\lambda\in\mathcal{J}} v_\lambda \psi_\lambda \in \mathcal{H}. \tag{2.18}$$

Thus, by definition, $\mathcal{A}^s_X(\mathcal{P})$ gathers all elements in X that can be approximated by expansions containing at most N terms subject to the constraints given by $\mathcal{P}$ with an approximation rate N^{-s}. Moreover, in the present setting approximation in $\mathcal{H}$ corresponds *directly* to approximation in $\ell_2(\mathcal{J})$.

In these terms we may call SOLVE *$\mathcal{P}$-optimal* (in some range $s < s^*$) if, whenever the exact solution $\mathbf{u}$ belongs to $\mathcal{A}^s_{\ell_2(\mathcal{J})}(\mathcal{P})$ (viz. $u \in \mathcal{A}^s_{\mathcal{H}}(\mathcal{P})$) (for $s < s^*$) then the computational work needed in SOLVE to produce $\bar{\mathbf{u}}(\varepsilon)$ (see Proposition 2.2) (and hence also the number of nonzero terms in $\bar{\mathbf{u}}(\varepsilon)$) grows at most as $\varepsilon^{-1/s}$.

Clearly, whenever $\mathcal{P}$ is a strict subset of $\mathcal{J}^{\mathcal{J}}$ one has $\mathcal{A}_X^s(\mathcal{P}) \subset \mathcal{A}_X^s(\mathcal{J}^{\mathcal{J}})$. Hence, $\mathcal{J}^{\mathcal{J}}$-optimality would be the strongest notion as it ensures optimal performance for the largest class. To appraise the meaning of such optimality notions one has to understand the class $\mathcal{A}_{\ell_2(\mathcal{J})}^s(\mathcal{P})$ a little better. In absence of any constraints $\mathcal{P} = \mathcal{J}^{\mathcal{J}}$ it is known that (see e.g. [18, 6])

$$\mathcal{A}_{\ell_2(\mathcal{J})}^s(\mathcal{J}^{\mathcal{J}}) = \ell_\tau^w := \{\mathbf{v} \in \ell_2(\mathcal{J}) : \sup_{n\in\mathbb{N}} n^{1/\tau}|v_n^*| := |\mathbf{v}|_{\ell_\tau^w} < \infty\}, \quad \text{where } \frac{1}{\tau} = s + \frac{1}{2}, \tag{2.19}$$

where $(v_n^*)_{n\in\mathbb{N}}$ is a rearrangement of $\mathbf{v}$ such that $v_{n+1}^* \leq v_n^*$, $n \in \mathbb{N}$. The *weak* space ℓ_τ^w of sequences whose decreasing rearrangement decays at least like $n^{-1/\tau}$ is very close to ℓ_τ, i.e. we have the embeddings $\ell_\tau \subset \ell_\tau^w \subset \ell_{\tau'}$ for any $\tau' > \tau$. When $\mathcal{H} = H^q(\Gamma)$ (and as before Ψ is a Riesz basis for $\mathcal{H}$) ℓ_τ in turn is *isomorphic* to the *Besov space* $B_\tau^{q+sd}(L_\tau)$, i.e. (see e.g. [12])

$$|\sum_{\lambda\in\mathcal{J}} v_\lambda \psi_\lambda|_{B_\tau^{q+sd}(L_\tau)} \sim \|\mathbf{v}\|_{\ell_\tau}. \tag{2.20}$$

This space is in some sense the *largest space* with smoothness $q + sd$ that is still embedded in H^q. One can picture this as a point in the $(1/p, \alpha)$ plane where the point $(1/p, \alpha)$ is associated with the space $B_p^\alpha(L_p)$ consisting of functions with regularity α measured in L_p. Note that when p decreases, i.e., one moves to the right in the $(1/p, \alpha)$-plane, the spaces $B_p^\alpha(L_p)$ become larger for fixed α and a more and more singular behavior is admitted. In order to have $B_p^\alpha(L_p)$ still embedded in H^q, say, one can move to the right up to the *Sobolev embedding line* where for the spatial dimension d the smoothness α and the metric $\bar{p}$ are coupled through the relation $\frac{1}{\bar{p}} = \frac{\alpha-q}{d} + \frac{1}{2}$. Thus for $\alpha = q + sd$ the critical relation is $1/\bar{p} = s + 1/2$.

Admitting only approximations whose active components are grouped in a tree might, of course, affect the approximation quality in that a given accuracy might require more terms since unnecessary small coefficients may be involved just to have the set of active coefficients form a tree. However, the constraints induced by $\mathcal{P} = \mathcal{T}$ are known to cause only a small difference in the following sense. It has been shown in [10] that, whenever

$$\frac{1}{\tau} < s + \frac{1}{2} \tag{2.21}$$

one has

$$B_\tau^{q+sd}(L_\tau) \subset \mathcal{A}_{H^q}^s(\mathcal{T}) \quad \text{and} \quad \sup_{\|v\|_{B_\tau^{q+sd}(L_\tau)}\leq 1} \sigma_{H^q,N}^{\mathcal{J}^{\mathcal{J}}}(v) \gtrsim N^{-s}, \tag{2.22}$$

i.e., in the class $B_\tau^{q+sd}(L_\tau)$ (in the worst case sense) the rate of best tree N-term approximation agrees with the unconstrained rate provided that (2.21) holds. This means that $B_\tau^{q+sd}(L_\tau)$ has to be just left off the embedding line through the anchor space H^q which in turn means that the embedding $B_\tau^{q+sd}(L_\tau) \hookrightarrow H^q$ is compact.

In principle, we could strive for realizing a scheme that is optimal with regard to the strongest notion of *unconstrained best N-term approximation*. In fact, the problem class considered here is covered by the corresponding results in [6] except for a concrete realization of certain compressed matrices in a corresponding operator application scheme. It will be seen, however, that (as in image compression/encoding) tree structures are extremely important for (quantitatively) efficient numerical realizations of the routine Apply. On the other hand, the analysis of tree approximation schemes is a little more delicate. Therefore and in view of the above mentioned small difference in these optimality notions, most attention will be paid to the case of *tree N-term approximation* ($\mathcal{P} = \mathcal{T}$). Recall that tree structures also appeared to be helpful when dealing with nonlinear problems [8] as well as for entropy-optimal encoding schemes for image compression [10].

In order to simplify notation we shall henceforth briefly write $\mathcal{A}^s := \mathcal{A}^s_{\ell_2(\mathcal{J})}(\mathcal{P})$ and also, in view (2.18), $\mathcal{A}^s := \mathcal{A}^s_{\mathcal{H}}(\mathcal{P})$ for the corresponding function space. Which one is meant and whether this refers to tree approximation ($\mathcal{P} = \mathcal{T}$) or *unconstrained N-term approximation* ($\mathcal{P} = \mathcal{J}^{\mathcal{J}}$) will always be clear from the context.

The following result from [8] gives the announced criterion – s^*-*sparsity* – for $\mathcal{P}$-optimality for both $\mathcal{P} \in \{\mathcal{T}, \mathcal{J}^{\mathcal{J}}\}$:

The residual scheme Res *is called s^*-sparse for some $s^* > 0$ if the following holds. Whenever the exact solution u of (1.1) belongs to $\mathcal{A}^s$ for some $s < s^*$, then one has for any finitely supported input $\mathbf{v}$ and any tolerance $\eta > 0$ that the output $\mathbf{r}_\eta :=$ Res $[\eta, \mathbf{C}, \mathbf{A}, \mathbf{f}, \mathbf{v}]$ satisfies*

$$\begin{array}{lcl} \#\mathrm{supp}\, \mathbf{r}_\eta & \leq & C\eta^{-1/s} \left(\|\mathbf{v}\|_{\mathcal{A}^s}^{1/s} + \|\mathbf{u}\|_{\mathcal{A}^s}^{1/s} \right), \\ \|\mathbf{r}_\eta\|_{\mathcal{A}^s} & \leq & C \left(\|\mathbf{v}\|_{\mathcal{A}^s} + \|\mathbf{u}\|_{\mathcal{A}^s} \right), \end{array} \tag{2.23}$$

where (in addition to the dependence on the constants in (1.5) and (2.2)) C depends only on s when $s \to s^*$. Moreover, the number of operations needed to compute $\mathbf{r}_\eta$ stays proportional to $\#\mathrm{supp}\, \mathbf{r}_\eta$.

Note that the condition differs slightly from those in [8] in that additional constant terms on the right hand side of (2.23) are missing. This is justified since we are dealing here only with linear problems [8, 9].

Theorem 2.3 *If the exact solution $u = \sum_{\lambda \in \mathcal{J}} u_\lambda \psi_\lambda$ belongs to $\mathcal{A}^s$, for any $s < s^* = s^*(\mathbf{A}, \Psi)$, then, for any target accuracy $\varepsilon > 0$, the approximations $\bar{\mathbf{u}}(\varepsilon)$ produced by* Solve *satisfy*

$$\|u - \sum_\lambda \bar{u}(\varepsilon)_\lambda \psi_\lambda\|_{\mathcal{H}} \leq C_\Psi \varepsilon,$$

and

$$\#\mathrm{supp}\, \bar{\mathbf{u}}(\varepsilon),\ \text{comp. work} \lesssim \varepsilon^{-1/s} \|u\|_{\mathcal{A}^s}^{1/s}.$$

Thus it remains to construct a concrete s^*-sparse residual scheme Res. To this end, recall that Res consists essentially of (a) approximating the data $\mathbf{f}$ within some tolerance and (b) of an approximate application of $\mathbf{A}$ to a finitely supported input vector.

We shall address first (a). Viewing $\mathbf{f}$, as pointed out before, as some preprocessed finite array, this latter approximation may be realized by an application of Coarse. In the unconstrained case coarsening of any finitely supported input array $\mathbf{v}$ is just *thresholding* the entries of $\mathbf{v}$ which, in turn, requires sorting its entries [6]. However, the strict sorting can be relaxed to quasi-sorting based on binary binning, so as to ensure the linear scaling of the computational work needed to comply with s^*-sparsity. In the following we shall use the notation

$$\mathcal{S}(\eta, \mathbf{v}) := \operatorname{supp} \mathbf{v}_\eta \quad \text{for} \quad \mathbf{v}_\eta = \textsc{Coarse}\,[\eta, \mathbf{v}], \tag{2.24}$$

when we are working in the setting of unconstrained best N-term approximation. Thus, $\#\mathcal{S}(\eta, \mathbf{v})$ is bounded by a fixed constant multiple of the smallest number of terms needed to approximate $\mathbf{v}$ within tolerance η.

Moreover, the Coarsening Lemma from [6] ensures stability in the corresponding space $\mathcal{A}^s$ which means that, whenever for some finitely supported $\mathbf{w}$ such that $\|\mathbf{v}-\mathbf{w}\|_{\ell_2(\mathcal{J})} \leq \eta$, one has for any fixed $a > 1$ that the output $\mathbf{w}_\eta = \textsc{Coarse}\,[a\eta, \mathbf{w}]$ satisfies

$$\|\mathbf{w}_\eta\|_{\mathcal{A}^s} \lesssim \|\mathbf{v}\|_{\mathcal{A}^s}, \quad \#\mathcal{S}(a\eta, \mathbf{w}) = \#\operatorname{supp} \mathbf{w}_\eta \lesssim \eta^{-1/s} |\mathbf{v}|_{\mathcal{A}^s}^{1/s}. \tag{2.25}$$

When $\mathcal{P} = \mathbb{T}$ the coarsening procedure has to respect tree structure. This would require solving the following task: Given $\mathbf{w}$, a tree $\mathcal{T} = \mathcal{T}^*(\eta, \mathbf{w})$ is called η-best for $\mathbf{w}$ if $\sigma^{\mathbb{T}}_{N,\ell_2(\mathcal{J})}(\mathbf{w}) = \|\mathbf{w} - \mathbf{w}|_{\mathcal{T}}\|_{\ell_2(\mathcal{J})}$. It is not obvious how to realize efficiently such a *best* N-term tree approximation. In fact, one cannot expect to have a scheme whose computational work scales linearly. Therefore, one has to relax again the notion of *best* N-term tree approximation to *near best* N-term tree approximation defined as follows: A tree $\mathcal{T} = \mathcal{T}(\eta, \mathbf{w})$ is called (η, C)-*near best* (or briefly near best when the parameters are clear from the context) if

$$\|\mathbf{w} - \mathbf{w}|_{\mathcal{T}}\|_{\ell_2(\mathcal{J})} \leq \eta \quad \text{and} \quad \#\mathcal{T} \leq C \#\mathcal{T}^*(\eta/C, \mathbf{w}).$$

From now on it will always be understood that $\mathbf{v}_\eta$ in (2.11) is given by $\mathbf{v}_\eta = \mathbf{v}|_{\mathcal{T}(\eta,\mathbf{v})}$ with $\mathcal{T}(\eta, \mathbf{w})$ the near best tree defined above.

As pointed out in [8] a near best tree can be determined at linear cost by a specification of the Binev/DeVore scheme from [3] which is always used here as well. Thus, in the context of tree approximation we have

$$\mathcal{T}(\eta, \mathbf{v}) = \operatorname{supp} \mathbf{v}_\eta, \quad \text{where} \quad \mathbf{v}_\eta = \textsc{Coarse}\,[\eta, \mathbf{v}]. \tag{2.26}$$

The key properties of this coarsening scheme may be formulated as follows.

Proposition 2.4 *For any given finitely supported input* $\mathbf{v}$ *the computational cost of the output* $\mathbf{v}_\eta$ *produced by* $\textsc{Coarse}\,[\eta, \mathbf{v}]$ *remains proportional to* $\#\mathcal{T}(\operatorname{supp} \mathbf{v})$*, ($\mathcal{T}(\mathcal{I})$ denoting the smallest tree containing the index set $\mathcal{I}$). The underlying tree $\mathcal{T}(\eta, \mathbf{v})$ is (η, C^*)-near best, where C^* is the constant appearing in the estimate (5.8) in Theorem 5.2 of [3].*

The following counterpart to (2.25) for the tree version of $\textsc{Coarse}$ has also been established in [8].

Proposition 2.5 *If* $\mathbf{v} \in \mathcal{A}^s$ *and* $\|\mathbf{v} - \mathbf{w}\|_{\ell_2(\mathcal{J})} \leq \eta$ *with* $\#\operatorname{supp} \mathbf{w} < \infty$*, then* $\bar{\mathbf{w}}_\eta := \textsc{Coarse}\,[2C^*\eta, \mathbf{w}]$ *satisfies*

$$\#\operatorname{supp} \bar{\mathbf{w}}_\eta \lesssim \|\mathbf{v}\|_{\mathcal{A}^s}^{1/s} \eta^{-1/s}, \quad \|\mathbf{v} - \bar{\mathbf{w}}_\eta\|_{\ell_2(\mathcal{J})} \leq (1 + 2C^*)\eta, \tag{2.27}$$

and

$$\|\bar{\mathbf{w}}_\eta\|_{\mathcal{A}^s} \lesssim \|\mathbf{v}\|_{\mathcal{A}^s}, \tag{2.28}$$

where the constants depend only on s when $s \to 0$ and on C^* in $\textsc{Coarse}$.

According to (2.25), Proposition 2.5 applies to the unconstrained case for any fixed $C^* > 1/2$.

As a consequence of Proposition 2.5 and the preceding remarks, the approximation $\mathbf{f}_\eta = \textsc{Coarse}\,[\eta/2\alpha, \mathbf{f}]$ in (2.13) in a typical realization of $\textsc{Res}$ is stable in $\mathcal{A}^s$, i.e. $\|\mathbf{f}_\eta\|_{\mathcal{A}^s} \lesssim \|\mathbf{f}\|_{\mathcal{A}^s}$ for either choice of $\mathcal{P} \in \{\mathcal{J}^{\mathcal{J}}, \mathbb{T}\}$. Moreover, whenever we have that also $\mathbf{A}$ is bounded on $\mathcal{A}^s$ we would have $\|\mathbf{f}\|_{\mathcal{A}^s} = \|\mathbf{A}\mathbf{u}\|_{\mathcal{A}^s} \lesssim \|\mathbf{u}\|_{\mathcal{A}^s}$. This provides the following fact.

Remark 2.6 *Any routine* $\textsc{Res}$ *of the form (2.13) is s^*-sparse whenever the following properties hold:*

(i) $\mathbf{A}$ *is bounded on* $\mathcal{A}^s$ *for* $s < s^*$, *i.e.*

$$\|\mathbf{A}\mathbf{v}\|_{\mathcal{A}^s} \lesssim \|\mathbf{v}\|_{\mathcal{A}^s}. \tag{2.29}$$

(ii) The output $\mathbf{w}_\eta$ *of* $\textsc{Apply}\,[\eta, \mathbf{A}, \mathbf{v}]$ *satisfies for any finitely supported input* $\mathbf{v}$

$$\begin{array}{lcr} \#\mathrm{supp}\,\mathbf{w}_\eta & \leq & C\eta^{-1/s}\|\mathbf{v}\|_{\mathcal{A}^s}^{1/s}, \\ \|\mathbf{w}_\eta\|_{\mathcal{A}^s} & \leq & C\|\mathbf{v}\|_{\mathcal{A}^s}, \end{array} \tag{2.30}$$

and the computational work required to produce $\mathbf{w}_\eta$ *is bounded by a fixed constant multiple of* $\#(\mathrm{supp}\,\mathbf{v}) + \eta^{-1/s}\|\mathbf{v}\|_{\mathcal{A}^s}^{1/s}$ *when* $\mathcal{P} = \mathcal{J}^{\mathcal{J}}$ *and of* $\#\mathcal{T}(\mathrm{supp}\,\mathbf{v}) + \eta^{-1/s}\|\mathbf{v}\|_{\mathcal{A}^s}^{1/s}$ *when* $\mathcal{P} = \mathbb{T}$. *(in the latter case we shall usually have that the support of* $\mathbf{v}$ *already has tree structure so that* $\mathcal{T}(\mathrm{supp}\,\mathbf{v}) = \mathrm{supp}\,\mathbf{v}$*).*

3 Realization of Res

According to the above remarks we shall address in this section the construction of application schemes $\textsc{Apply}$ that satisfy the conditions in Remark 2.6.

3.1 Compressible Matrices

This is where a second main feature of wavelets comes into play namely the *cancellation property* which can be formulated as follows:

$$|\langle v, \psi_\lambda\rangle| \leq C2^{-|\lambda|\tilde{m}}|v|_{W^{\tilde{m}}(L_p(S_\lambda))}\|\psi_\lambda\|_{L_{p'}}, \quad \frac{1}{p} + \frac{1}{p'} = 1, \tag{3.1}$$

where we recall that $S_\lambda = \mathrm{supp}\,\psi_\lambda$ Here the order $\tilde{m}$ of cancellation properties agrees with the order of polynomial exactness of the *dual* wavelet basis when dealing with Euclidean domains.

This has been used in matrix compression to quantify the quasi-sparseness of principal sections of $\mathbf{A}$ corresponding to trial spaces spanned by *all* wavelets up to some highest level J of resolution.

A convenient way of formulating this quasi-sparseness for the full operator $\mathbf{A}$ is based on the notion of s^*-compressibility, see [6]. We say that $\mathbf{A}$ is s^*-*compressible* if for every

$s < s^*$ there exists a summable sequence $(\alpha_j)_{j\in\mathbb{N}_0}$ of positive numbers and for each j there exists a matrix $\mathbf{A}_j$ whose rows and columns have at most the order of $\alpha_j 2^j$ nonzero entries while

$$\|\mathbf{A} - \mathbf{A}_j\| \lesssim \alpha_j 2^{-sj}, \quad j \in \mathbb{N}. \tag{3.2}$$

Here and in the sequel $\|\mathbf{A}\| := \sup_{\|\mathbf{v}\|_{\ell_2(\mathcal{J})}=1} \|\mathbf{A}\mathbf{v}\|_{\ell_2(\mathcal{J})}$ denotes the spectral norm. This will later be crucial with regard to storage economy and efficiency of (approximate) matrix/vector multiplication. In what follows we shall employ the weights $\alpha_j := (1+j)^{-(M+M'+2)}$, i.e. $\mathbf{A}$ is called s^*-compressible if there exist submatrices $\mathbf{A}_j$ s.t. for any fixed $\bar{s} < s^*$

$$\|\mathbf{A} - \mathbf{A}_j\|_{\ell_2(\mathcal{J})} \le (1+j)^{-(2+M+M')} 2^{-j\bar{s}}, \qquad \#(\operatorname{supp} \mathbf{A}_j^{\mu}) \lesssim (1+j)^{-(2+M+M')} 2^j, \tag{3.3}$$

where M, M' are specified later in connection with quadrature.

We shall always assume in what follows that the operator $\mathcal{L}$ (aside from being an isomorphism from $\mathcal{H} = H^q(\Gamma) \to \mathcal{H}' = H^{-q}(\Gamma)$) is still bounded from $H^{q+\sigma} \to H^{-q+\sigma}$ for $\sigma \in [0, \hat{\sigma}]$ and some (sufficiently large) $\hat{\sigma} > 0$. This is the case when Γ is smooth but would otherwise need to be confirmed in each particular case. Under such hypothesis and when using spline wavelets of the type constructed in [4, 5, 11, 17, 24] one can derive from estimates of the form (1.7) decay estimates for the entries of $\mathbf{A}$ of the following form [16, 15, 26, 27, 33, 34]

$$|\langle \psi_\lambda, \mathcal{L}\psi_{\lambda'} \rangle| \lesssim \left(\frac{2^{-||\lambda|-|\lambda'||/2}}{1 + d(\lambda, \lambda')} \right)^{d+2\tilde{m}+2q}, \tag{3.4}$$

where

$$d(\lambda, \lambda') := 2^{\min(|\lambda|,|\lambda'|)} \operatorname{dist}(\operatorname{supp}(\psi_\lambda), \operatorname{supp}(\psi_{\lambda'})). \tag{3.5}$$

Using decay estimates of the form (3.4) compressibility ranges were identified already in [6]. These estimates made no further assumptions on the wavelets other than locality, a certain number of vanishing moments and a certain order of global regularity so that the resulting compressibility ranges turned out to be far from sharp. Much more can be said when using wavelet bases that are obtained by parametric liftings of tensor product spline wavelets whose construction can be found e.g. in [4, 17, 24]. An important additional ingredient is the so called *second compression* from [33]. Denoting by $\operatorname{sing} S_\lambda$ the singular support of the wavelet ψ_λ, i.e. the union of interfaces separating the smooth parts of the wavelet, let

$$\bar{d}(\lambda, \lambda') := 2^{|\lambda'|} \operatorname{dist}(S_\lambda, \operatorname{sing} S_{\lambda'}) \quad \text{whenever} \quad |\lambda| > |\lambda'|.$$

One can then show that [15, 33, 34]

$$\left\{ \begin{array}{c} |\langle \psi_\lambda, \mathcal{L}\psi_{\lambda'} \rangle| \\ |\langle \mathcal{L}^*\psi_{\lambda'}, \psi_\lambda \rangle| \end{array} \right\} \lesssim \frac{2^{(|\lambda'|-|\lambda|)(\tilde{m}+q+d/2)}}{\bar{d}(\lambda, \lambda')^{\tilde{m}+2q}}. \tag{3.6}$$

It was recently shown in [34] that whenever (3.4), (3.6) are valid the wavelet representation is s^*-compressible of *optimal order*.

Theorem 3.1 ([34]) *Suppose that for $\mathcal{H} = H^q(\Gamma)$ the operator $\mathcal{L}$ is still bounded as a mapping form $H^{q+\sigma}(\Gamma)$ to $H^{-q+\sigma}(\Gamma)$ for a sufficiently large range of positive σ. Moreover, suppose that the wavelet bases $\Psi, \tilde{\Psi}$ have (exactness) order $m, \tilde{m}$, respectively and that*

$$\tilde{m} > \gamma - 2q, \qquad \tilde{m} > m - 2q, \tag{3.7}$$

where γ is the constant from (2.6). Then $\mathbf{A}$ is s^*-compressible with

$$s^* > (m-q)/d \tag{3.8}$$

where d is the dimension of Γ.

Note that $N^{-(m-q)/d}$ is the highest possible convergence rate attainable for approximations in the energy norm $\|\cdot\|_{H^q(\Gamma)}$ from the multiresolution spaces generated by wavelets of exactness order m.

Moreover, setting $\hat{j} := j + \log_2(\alpha_j)$, the corresponding compression pattern has also been identified in [34] and reads as follows. To obtain $\mathbf{A}_j$ from $\mathbf{A}$ Replace $\langle \psi_\lambda, \mathcal{L}\psi_{\lambda'} \rangle = a_{\lambda,\lambda'}$ by zero when:

$$\begin{aligned} & ||\lambda| - |\lambda'|| > \hat{j}/(d-1) \ \text{ if } \ d > 1, \ \text{ else } > 2^{\hat{j}}, \quad \text{or} \\ & d(\lambda, \lambda') \geq a \max\{1, 2^{b(\frac{\hat{j}}{d} - ||\lambda|-|\lambda'||)}\} \quad \text{or} \\ & ||\lambda| - |\lambda'|| > \hat{j}/d \ \text{ and } \ \bar{d}(\lambda, \lambda') \geq a' \max\{\gamma_{||\lambda|-|\lambda'||-\frac{\hat{j}}{d}} 2^{\hat{j}-d||\lambda|-|\lambda'||}, 2^{-||\lambda|-|\lambda'||}\}, \end{aligned} \tag{3.9}$$

where a, a' and $b \in [1/2, 1)$ are suitable fixed parameters, and where $(\gamma_n)_{n\in\mathbb{N}_0}$ is again a polynomially decreasing sequence such that $\sum_n \gamma_n < \infty$. Specifically, we take $\gamma_n := (1+n)^{-2}$.

3.2 The Significance of Compressibility

To explain the relevance of compressibility for both settings, unconstrained and tree N-term approximation, let

$$\mathcal{Y}(\eta, \mathbf{v}) := \begin{cases} \mathcal{S}(\eta, \mathbf{v}) & \text{when} \quad \mathcal{P} = \mathcal{J}^{\mathcal{J}}; \\ \mathcal{T}(\eta, \mathbf{v}) & \text{when} \quad \mathcal{P} = \mathbb{T}. \end{cases}$$

Now recall (3.8) and fix some $\bar{s}$ with $(m-t)/d \leq \bar{s} < s^*$. Clearly (3.3) holds then for all $s \leq \bar{s}$. Moreover, for a wavelet basis of order m we can expect an N-term approximation rate (in either setting) of at most $O(N^{-s})$ for $s \leq (m-t)/d \leq \bar{s}$.

Given any finitely supported $\mathbf{v}$ (whose support is assumed to be a tree when dealing with tree constraints) and any positive target accuracy η we wish to construct an η-accurate approximation to $\mathbf{Av}$. To this end, consider the output supports of the respective coarsening schemes for a sequence of growing tolerances

$$\mathcal{Y}_j := \mathcal{Y}(2^{j\bar{s}}\eta, \mathbf{v}). \tag{3.10}$$

Thus, we have

$$\mathcal{Y}_0 = \mathcal{Y}(\eta, \mathbf{v}), \quad \mathcal{Y}_{j+1} \subset \mathcal{Y}_j, \quad \mathcal{Y}_j = \emptyset \quad \text{for} \quad j > J = J(\eta, \mathbf{v}) := \frac{\log_2(\|\mathbf{v}\|_{\ell_2(\mathcal{J})}/\eta)}{\bar{s}}, \tag{3.11}$$

so that by (2.11), we have

$$\|\mathbf{v} - \mathbf{v}|_{\mathcal{Y}_j}\|_{\ell_2(\mathcal{J})} \leq 2^{\bar{s}j}\eta. \tag{3.12}$$

Finally, we define the *difference sets* and the corresponding portions of $\mathbf{v}$

$$\Delta_j := \mathcal{Y}_j \setminus \mathcal{Y}_{j+1}, \quad \mathbf{v}_{[j]} := \mathbf{v}|_{\Delta_j}. \tag{3.13}$$

Now let $\mathbf{A}_j$ be the compressed matrix from (3.3) and let $\mathbf{A}_j^\mu$ its μth column. Following essentially [6], a candidate for a finitely supported approximation to $\mathbf{Av}$ is given by $\mathbf{w}_\eta := \sum_{j=0}^J \mathbf{A}_j\mathbf{v}_{[j]}$. In order to describe the support of $\mathbf{w}_\eta$ let

$$S_j^\mu := \operatorname{supp} \mathbf{A}_j^\mu. \tag{3.14}$$

and set

$$\mathcal{Y}_\eta := \bigcup_{j=0}^J \bigcup_{\mu\in\Delta_j} S_j^\mu. \tag{3.15}$$

It is clear that

$$\operatorname{supp}\Big(\sum_{j=0}^J \mathbf{A}_j\mathbf{v}_{[j]}\Big) \subseteq \mathcal{Y}_\eta. \tag{3.16}$$

Theorem 3.2 *Let $\mathbf{v}$ be any finitely supported array and $\bar{s}$ as above. When dealing with tree approximation the support of $\mathbf{v}$ is assumed to have tree structure. Then there exists a constant C, depending on $\bar{s} < s^*$, on the constants in (1.5), (2.2) and on the tree algorithm from [3] when $\mathcal{P} = \mathbb{T}$ such that*

$$\Big\|\mathbf{Av} - \sum_{j=0}^J \mathbf{A}_j\mathbf{v}|_{[j]}\Big\|_{\ell_2(\mathcal{J})} \leq C\eta. \tag{3.17}$$

Moreover, the computational work W_η needed to determine $\mathcal{Y}_\eta$, i.e, the work required by the application of the coarsening scheme for determining the $\mathcal{Y}_j$, can be bounded as

$$W_\eta \leq C\#\operatorname{supp}\mathbf{v} \tag{3.18}$$

and for any $s \leq \bar{s}$ one has

$$\#\mathrm{supp}\mathcal{Y}_\eta \leq C\eta^{-1/s}|\mathbf{v}|_{\mathcal{A}^s}^{1/s}, \tag{3.19}$$

where the spaces $\mathcal{A}^s$ are the approximation spaces for best unconstrained respectively best tree N-term approximation.

Proof: First note that by (2.11)

$$\|\mathbf{v}_{[j]}\|_{\ell_2(\mathcal{J})} = \|\mathbf{v} - \mathbf{v}|_{\mathcal{Y}_{j+1}} - (\mathbf{v} - \mathbf{v}|_{\mathcal{Y}_j})\|_{\ell_2(\mathcal{J})} \leq (2^{\bar{s}(j+1)} + 2^{\bar{s}j})\eta \leq (2^{\bar{s}} + 1)2^{\bar{s}j}\eta. \tag{3.20}$$

Then, since $\mathbf{v}|_{\mathcal{Y}(\eta,\mathbf{v})} = \sum_{j=0}^J \mathbf{v}_{[j]}$, we obtain

$$\begin{aligned}
\Big\|\mathbf{A}\mathbf{v} - \sum_{j=0}^J \mathbf{A}_j \mathbf{v}_{[j]}\Big\|_{\ell_2(\mathcal{J})} &= \Big\|\sum_{j=0}^J (\mathbf{A} - \mathbf{A}_j)\mathbf{v}_{[j]} + \mathbf{A}(\mathbf{v} - \mathbf{v}|_{\mathcal{Y}_0})\Big\|_{\ell_2(\mathcal{J})} \\
&\le \sum_{j=0}^J \|\mathbf{A} - \mathbf{A}_j\| \, \|\mathbf{v}_{[j]}\|_{\ell_2(\mathcal{J})} + \|\mathbf{A}\| \|\mathbf{v} - \mathbf{v}|_{\mathcal{Y}_0}\|_{\ell_2(\mathcal{J})} \\
&\le \eta(2^{\bar{s}} + 1) \sum_{j=0}^J \alpha_j 2^{-\bar{s}j} 2^{\bar{s}j} + C_L C_\Psi^2 \eta \le C\eta, \qquad (3.21)
\end{aligned}$$

where we have used (2.5) and (2.11). This confirms (3.17).

The work count (3.18) follows from the results in [3]. As for (3.19), note that by (3.3),

$$\begin{aligned}
\# \mathcal{Y}_\eta &\le \sum_{j=0}^J \sum_{\mu \in \Delta_j} \# S_j^\mu \le C_0 \sum_{j=0}^J \sum_{\mu \in \Delta_j} \alpha_j 2^j \\
&= C_0 \sum_{j=0}^J \# \Delta_j \alpha_j 2^j. \qquad (3.22)
\end{aligned}$$

Since by the near optimality of the Binev-DeVore scheme and the definition of $|\cdot|_{\mathcal{A}^s}$, (recalling that $\mathbf{v}$ has finite support) one has

$$\# \Delta_j \le \# \mathcal{Y}_j \le C^* (2^{j\bar{s}} \eta / C^*)^{-1/s} |\mathbf{v}|_{\mathcal{A}^s}^{1/s}, \qquad (3.23)$$

we conclude from (3.22) and $s \le \bar{s}$ that

$$\begin{aligned}
\# \mathcal{T}_\eta &\le C'' \sum_{j=0}^J \alpha_j \eta^{-1/s} |\mathbf{v}|_{\mathcal{A}^s}^{1/s} \\
&= \bar{C} \eta^{-1/s} |\mathbf{v}|_{\mathcal{A}^s}^{1/s}, \qquad (3.24)
\end{aligned}$$

which completes the proof. □

According to Remark 2.6, we have to confirm the boundedness of $\mathbf{A}$ on the respective version of $\mathcal{A}^s$ for $s < s^*$. When dealing with unconstrained N-term approximation this has been already done in [6]. The argument will be implicitly contained in the subsequent reasoning for the case of tree approximation $\mathcal{P} = \mathbb{T}$ which requires a little more care. The key is the following observation.

Proposition 3.3 *When $\mathcal{Y}_j = \mathcal{T}(2^{\bar{s}j}\eta, \mathbf{v})$, i.e., $\mathcal{P} = \mathbb{T}$, the index set $\mathcal{Y}_\eta := \mathcal{T}_\eta$ defined by (3.15) is a tree.*

Proof: We have to show that whenever $\nu \in \mathcal{T}_\eta$ then either ν is a root or there exists a parent $\hat{\nu} \in \mathcal{T}_\eta$, i.e., $|\hat{\nu}| = |\nu| - 1$ and $S_\nu \subset S_{\hat{\nu}}$. By definition, $\nu \in \mathcal{T}_\eta$ implies the existence of some $\lambda \in \Delta_j$, $j \le J$, such that $\nu \in S_{\hat{j}}^\lambda$. Recall that $\hat{j} = j + \log_2(\alpha_j)$. We shall distinguish two cases:

(a) $|\nu| > |\lambda|$: In this case ν cannot be a root. Let $\hat{\nu}$ be a parent of ν such that $S_\nu \subset S_{\hat{\nu}}$ and $|\hat{\nu}| = |\nu| - 1 \geq |\lambda|$. We shall show that $\hat{\nu} \in S_{\hat{j}}^{\lambda}$ as well. We know from (3.9) that $||\lambda| - |\nu|| \leq \hat{j}/(d-1)$ and hence $||\lambda| - |\hat{\nu}|| \leq \hat{j}/(d-1)$. Since $\frac{\hat{j}}{d} - ||\nu| - |\lambda|| < \frac{\hat{j}}{d} - ||\hat{\nu}| - |\lambda||$, we obtain by (3.9)

$$\begin{aligned} d(\hat{\nu}, \lambda) &= 2^{|\lambda|}\mathrm{dist}\,(S_{\hat{\nu}}, S_\lambda) \leq 2^{|\lambda|}\mathrm{dist}\,(S_\nu, S_\lambda) = d(\nu, \lambda) \\ &< a \max\{1, 2^{b\left(\frac{\hat{j}}{d} - ||\nu| - |\lambda||\right)}\} \\ &\leq a \max\{1, 2^{b\left(\frac{\hat{j}}{d} - ||\hat{\nu}| - |\lambda||\right)}\}. \end{aligned}$$

Now suppose that $||\hat{\nu}| - |\lambda|| > \hat{j}/d$. Then, since $\nu \in S_{\hat{j}}^{\lambda}$, it follows

$$\begin{aligned} \bar{d}(\hat{\nu}, \lambda) &= 2^{|\lambda|}\mathrm{dist}\,(S_{\hat{\nu}}, \mathrm{sing}\, S_\lambda) \leq 2^{|\lambda|}\mathrm{dist}\,(S_\nu, \mathrm{sing}\, S_\lambda) = \bar{d}(\nu, \lambda) \\ &\leq a' \max\,\{\gamma_{||\nu| - |\lambda|| - \frac{\hat{j}}{d}} 2^{\hat{j} - d||\nu| - |\lambda||}, 2^{-||\nu| - |\lambda||}\} \\ &\leq a' \max\,\{\gamma_{||\hat{\nu}| - |\lambda|| - \frac{\hat{j}}{d}} 2^{\hat{j} - d||\hat{\nu}| - |\lambda||}, 2^{-||\hat{\nu}| - |\lambda||}\}. \end{aligned}$$

By (3.9), we therefore conclude that $\hat{\nu} \in S_{\hat{j}}^{\lambda} \subset \mathcal{T}_\eta$.

(b) $|\nu| \leq |\lambda|$: If ν is a root there is nothing to show. Thus $|\nu| > 0$ and therefore λ cannot be a root either. We shall show now that there exists an ancestor $\bar{\lambda}$ of λ and a $j' \geq j$ such that the parent $\hat{\nu}$ belongs to $S_{\hat{j}'}^{\bar{\lambda}}$ and thus belongs also to $\mathcal{T}_\eta$. To this end, note that since $\lambda \in \Delta_j \subset \mathcal{T}_j$ and $\mathcal{T}_j$ is a tree it must have a parent $\hat{\lambda}$ in $\mathcal{T}_j$. Thus there must exist a $j' \geq j$ such that $\hat{\lambda} \in \Delta_{j'}$. Now note that $||\hat{\nu}| - |\hat{\lambda}|| = ||\nu| - |\lambda||$ and $||\lambda| - |\nu|| \leq \hat{j}/(d-1)$ implies $||\hat{\lambda}| - |\hat{\nu}|| \leq \hat{j}'/(d-1)$ so that

$$\begin{aligned} d(\hat{\nu}, \hat{\lambda}) &= 2^{|\hat{\nu}|}\mathrm{dist}\,(S_{\hat{\nu}}, S_{\hat{\lambda}}) \leq 2^{|\nu|}\mathrm{dist}\,(S_\nu, S_\lambda) = d(\nu, \lambda) \\ &< a \max\{1, 2^{b\left(\frac{\hat{j}}{d} - ||\nu| - |\lambda||\right)}\} \\ &\leq a \max\{1, 2^{b\left(\frac{\hat{j}'}{d} - ||\hat{\nu}| - |\hat{\lambda}||\right)}\}. \end{aligned}$$

Thus, when $|\lambda| - |\nu| \leq \hat{j}/d \geq 0$ it follows $\hat{\nu} \in S_{\hat{j}'}^{\hat{\lambda}} \subset \mathcal{T}_\eta$ and we have found the asserted $\bar{\lambda} = \hat{\lambda}$. On the other hand, in the case $\hat{j}/(d-1) \geq |\lambda| - |\nu| > \hat{j}/d \geq 0$ we have to take the second compression into account. To this end, note that when $0 \leq |\hat{\lambda}| - |\nu| \leq \hat{j}'/d$ we conclude from (3.9) that $\nu \in S_{\hat{j}'}^{\hat{\lambda}}$. Likewise, in the case $|\hat{\lambda}| - |\nu| > \hat{j}'/d$ we conclude from

$$\begin{aligned} \bar{d}(\nu, \hat{\lambda}) &= 2^{|\nu|}\mathrm{dist}\,(\mathrm{sing} S_\nu,\ S_{\hat{\lambda}}) \leq 2^{|\nu|}\mathrm{dist}\,(\mathrm{sing} S_\nu,\ S_\lambda) = \bar{d}(\nu, \lambda) \\ &\leq a' \max\,\{\gamma_{||\nu| - |\lambda|| - \frac{\hat{j}}{d}} 2^{\hat{j} - d||\nu| - |\lambda||}, 2^{-||\nu| - |\lambda||}\} \\ &\leq a' \max\,\{\gamma_{||\nu| - |\hat{\lambda}|| - \frac{\hat{j}'}{d}} 2^{\hat{j}' - d||\nu| - |\hat{\lambda}||}, 2^{-||\nu| - |\hat{\lambda}||}\}, \end{aligned}$$

which means that $\nu \in S_{\hat{j}'}^{\hat{\lambda}}$. Hence we can replace λ by $\hat{\lambda}$ and repeat the argument. Since each time the level is lowered by one we eventually find an ancestor $\bar{\lambda}$ of λ such that

$0 \leq |\bar{\lambda}| - |\nu| \leq \hat{j}'/d$ and $\nu \in S_{j'}^{\bar{\lambda}}$. This finishes the proof by the first part of the above argument in the case (b). □

Corollary 3.4 *When* $\mathbf{A}$ *is* s^**-compressible then* $\mathbf{A}$ *is bounded on* $\mathcal{A}^s$ *for any* $s < s^*$ *for either setting of unconstrained* N*-term approximation,* $\mathcal{P} = \mathcal{J}^{\mathcal{J}}$*, and tree approximation,* $\mathcal{P} = \mathbb{T}$*.*

Proof: We have to show that for any $\eta > 0$ and any $\mathbf{v} \in \mathcal{A}^s$, $s < s^*$, we can find a $\mathbf{w}_\eta$ such that $\|\mathbf{A}\mathbf{v} - \mathbf{w}_\eta\|_{\ell_2(\mathcal{J})} \leq \eta$ and $\mathcal{S} := \operatorname{supp} \mathbf{w}_\eta$ satisfies $(\#\mathcal{S})^s \eta \leq \bar{C}$ where $\bar{C}$ does not depend on η. Moreover, when working in the setting of tree approximation, $\mathcal{P} = \mathbb{T}$, we must have $\mathcal{S} = \mathcal{T}$ is a tree. By Theorem 3.2, (3.19), and Proposition 3.3, $\mathbf{w}_\eta := \sum_{j=0}^{J} \mathbf{A}_j \mathbf{v}_{[j]}$ satisfies these requirements, which completes the proof. □

For later use we record the following simple fact.

Remark 3.5 *Incidentally we have shown that there exists a constant* C*, depending only on* $\bar{s} < s^*$*, on the constants in (1.5), (2.2) and on the constant* C^* *from the tree algorithm from [3] in case* $\mathcal{P} = \mathbb{T}$*, such that for* $\mathcal{Y}_\eta$ *given by (3.15)*

$$\|\mathbf{A}\mathbf{v} - (\mathbf{A}\mathbf{v})|_{\mathcal{Y}_\eta}\|_{\ell_2(\mathcal{J})} \leq C\eta, \tag{3.25}$$

and hence also

$$\|(\mathbf{A}\mathbf{v})|_{\mathcal{Y}_\eta} - \sum_{j=0}^{J} \mathbf{A}_j \mathbf{v}_{[j]}\|_{\ell_2(\mathcal{J})} \leq C\eta, \tag{3.26}$$

with $\mathbf{v}_{[j]}$ *defined in (3.13).*

Proof: We have

$$\begin{aligned} \mathbf{A}\mathbf{v} - (\mathbf{A}\mathbf{v})|_{\mathcal{Y}_\eta} &= \mathbf{A}\mathbf{v} - \sum_{j=0}^{J} \mathbf{A}_j \mathbf{v}_{[j]} + \left(\sum_{j=0}^{J} (\mathbf{A} - \mathbf{A}_j) \mathbf{v}_{[j]} \right) \Big|_{\mathcal{Y}_\eta} \\ &\quad + \left(\mathbf{A}(\mathbf{v}_\eta - \mathbf{v}) \right) |_{\mathcal{Y}_\eta}. \end{aligned} \tag{3.27}$$

The first sum on the right hand side of (3.27) has been already estimated in (3.17). The second and third sum are estimated in (3.17). □

So far we have confirmed part (i) in Remark 2.6 as well as the asymptotically optimal growth rate of the prediction sets $\mathcal{Y}_\eta$. These findings suggest the following principle strategies for a concrete realization of Apply.

(I): Based on the a-posteriori information from the input $\mathbf{v}$ and the a-priori knowledge about the operator $\mathcal{L}$, given in terms of the compression rules (3.9) and the resulting supports S_j^μ, one determines in a first step a *prediction set* $\mathcal{Y}_\eta$ according to (3.15). Note that this does *not* require the knowledge of the entries of $\mathbf{A}$. By Theorem 3.2, the computational work for this step scales linearly in the input size. By (3.25) in Remark

3.5, this set contains all the coefficients of the image $\mathbf{Av}$ needed to have an order η-accurate approximation to $\mathbf{Av}$.

In a second step one has to determine an order η-accurate approximation to the *finite* array $(\mathbf{Av})|_{\mathcal{Y}_\eta}$, i.e., some $\tilde{\mathbf{w}}_\eta$ with $\operatorname{supp}\tilde{\mathbf{w}}_\eta \subseteq \mathcal{Y}_\eta$, such that also

$$\|(\mathbf{Av})|_{\mathcal{Y}_\eta} - \tilde{\mathbf{w}}_\eta\|_{\ell_2(\mathcal{J})} \le C\eta. \tag{3.28}$$

In principle, one can now try to realize such a $\tilde{\mathbf{w}}_\eta$ through *clustering techniques* like panel clustering or multipole expansions *without* ever computing the significant entries of $\mathbf{A}$ individually within some suitable accuracy. A similar idea in the context of linear solvers has been proposed in [32].

(II): The second option is to use Theorem 3.2, (3.17) directly, i.e., to seek a $\tilde{\mathbf{w}}_\eta$ such that

$$\|\sum_{j=0}^{J} \mathbf{A}_j \mathbf{v}_{[j]} - \tilde{\mathbf{w}}_\eta\|_{\ell_2(\mathcal{J})} \le C\eta, \tag{3.29}$$

where as before the $\mathbf{v}_{[j]}$ are defined in (3.13). This can be done by using evaluation techniques like clustering methods to approximate the summands $\mathbf{A}_j\mathbf{v}_{[j]}$ or $\mathbf{A}\mathbf{v}_{[j]}$ within a suitable tolerance. Alternatively, one can exploit the expertise on *matrix compression* gained in the context of linear solvers [25] to approximate the compressed matrices $\mathbf{A}_j$ in the spectral norm by computing the entries of $\mathbf{A}_j$ within a suitable accuracy tolerance by *adaptive quadrature.*

Option (I) will be addressed in more detail in a forthcoming paper. In the remainder of this paper we focus on option (II).

Remark 3.6 *Suppose that the entries of $\mathbf{A}_j$ were given. Then there exists a constant C, depending only on $\bar{s}$ and the constants in (2.5) but not on s and $\mathbf{v}$, such that the number $W(\eta, \mathbf{A}, \mathbf{v})$ of operations needed to compute $\sum_{j=0}^{J} \mathbf{A}_j\mathbf{v}_{[j]}$ for $J = J(\eta, \mathbf{v})$ as above, remains bounded by*

$$W(\eta, \mathbf{A}, \mathbf{v}) \le C(\eta^{-1/s}|\mathbf{v}|_{\mathcal{A}^s}^{1/s} + \#\mathcal{T}(\operatorname{supp}\mathbf{v})), \tag{3.30}$$

where in the setting $\mathcal{P} = \mathbb{T}$, $\mathcal{T}(\operatorname{supp}\mathbf{v})$ is again the smallest tree containing $\operatorname{supp}\mathbf{v}$.

Proof: The term $\#\mathcal{T}(\operatorname{supp}\mathbf{v})$ stems from the Binev-DeVore scheme. Now the number of operations needed to compute $\mathbf{A}_j\mathbf{v}_{[j]}$ is, by definition of $\mathbf{A}_j$, bounded by a constant multiple of

$$\alpha_j 2^j \#\Delta_j \lesssim \alpha_j 2^{j(1-\frac{\bar{s}}{s})}\eta^{-1/s}|\mathbf{v}|_{\mathcal{A}^s}^{1/s}. \tag{3.31}$$

Summing over $j \le J$ yields the assertion. □

It therefore remains to approximate the terms $\mathbf{A}_j\mathbf{v}_{[j]}$.

3.3 Approximation of Significant Entries

According to Theorem 3.2, an η-accurate approximation to $\mathbf{Av}$ is given by $\sum_{j=0}^{J}\mathbf{A}_j\mathbf{v}_{[j]}$, where $J = J(\eta,\mathbf{v})$ and $\mathbf{v}_{[j]}$ are given in (3.11), (3.13). Thus we have to approximate the compressed matrices $\mathbf{A}_j^{\Delta_j} := ((\mathbf{A}_j)_{\lambda,\lambda'})_{\lambda\in\mathcal{J},\lambda'\in\Delta_j}$ by computing the nonzero entries of $\mathbf{A}_j^{\Delta_j}$ through suitable quadrature techniques. This will be done by *adaptive hp-quadrature* based on Gauß-Legendre formulas along the lines already developed in the matrix compression framework, see e.g. [25] and also [24, 33, 28, 31] for further references. The resulting perturbed matrices are denoted by $\tilde{\mathbf{A}}_j^{\Delta_j}$. The following observation is an immediate consequence of the proof of Theorem 3.2.

Remark 3.7 *Setting* $\mathbf{w}_\eta := \sum_{j=0}^{J}\tilde{\mathbf{A}}_j(\mathbf{v}|_{\Delta_j})$ *we have*

$$\|\mathbf{w}_\eta - \mathbf{Av}\|_{\ell_2(\mathcal{J})} \le C\eta$$

provided that

$$\|\mathbf{A}_j^{\Delta_j} - \tilde{\mathbf{A}}_j^{\Delta_j}\|_{\ell_2(\mathcal{J})\to\ell_2(\mathcal{J})} \le C\alpha_j 2^{-\bar{s}j}, \quad j \le J(\eta,\mathbf{v}). \tag{3.32}$$

We shall proceed in two steps. First we shall determine the accuracy for each entry which ensures the validity of (3.32). Then as a second step we shall estimate the corresponding computational work. Here we can resort to known results from [15, 25, 24].

Proposition 3.8 *If the quadrature realizes an accuracy*

$$e^j_{\lambda,\lambda'} = e^j := |(\mathbf{A}_j^{\Delta_j})_{\lambda,\lambda'} - (\tilde{\mathbf{A}}_j^{\Delta_j})_{\lambda,\lambda'}| \lesssim \frac{2^{-\bar{s}j}}{\sqrt{2^j\min\{2^j, \#\Delta_j/\alpha_j\}}}, \tag{3.33}$$

then the estimate (3.32) required in Remark 3.7 holds for some constant independent of $\mathbf{v}$ *and* $s \le \bar{s}$.

Proof: Let $\mathbf{E}_j := \mathbf{A}_j^{\Delta_j} - \tilde{\mathbf{A}}_j^{\Delta_j}$ denote the error matrix. By interpolation we know that

$$\|\mathbf{E}_j\|^2_{\ell_2(\mathcal{J})\to\ell_2(\mathcal{J})} \le \|\mathbf{E}_j\|_{\ell_\infty\to\ell_\infty}\|\mathbf{E}_j\|_{\ell_1\to\ell_1}. \tag{3.34}$$

Let us assume first that $2^j\alpha_j < \#\Delta_j$. By definition we have at most $\alpha_j 2^j$ nonzero entries per row and column in $\mathbf{A}_j$. On account of (3.33) we therefore have

$$\|\mathbf{E}_j\|_{\ell_\infty\to\ell_\infty} \quad \lesssim \quad \alpha_j 2^j e^j \lesssim \alpha_j 2^j\frac{2^{-\bar{s}j}}{2^j} = \alpha_j 2^{-\bar{s}j}. \tag{3.35}$$

Similarly we obtain $\|\mathbf{E}_j\|_{\ell_1\to\ell_1} \lesssim \alpha_j 2^{-\bar{s}j}$. Thus, when $2^j\alpha_j < \#\Delta_j$ the assertion follows from these estimates together with (3.34).

Suppose now that $2^j\alpha_j \ge \#\Delta_j$, i.e. $e^j \sim 2^{-(\bar{s}+\frac{1}{2})j}\sqrt{\alpha_j/\#\Delta_j}$. Now, by definition of $\mathbf{E}_j$ we have at most $\#\Delta_j$ entries per row. Thus

$$\|\mathbf{E}_j\|_{\ell_\infty\to\ell_\infty} \lesssim \#\Delta_j e^j \lesssim 2^{-(\bar{s}+\frac{1}{2})j}\sqrt{\alpha_j\#\Delta_j}. \tag{3.36}$$

Likewise we obtain

$$\|\mathbf{E}_j\|_{\ell_1\to\ell_1} \lesssim \alpha_j 2^j e^j \lesssim \alpha_j 2^{-(\bar{s}-\frac{1}{2})j}\sqrt{\alpha_j/\#\Delta_j} \tag{3.37}$$

Inserting (3.36) and (3.37) into (3.34) completes the proof. □

We can now turn to the work count. The following result is in the spirit of earlier work on matrix compression, see e.g. [25, Theorem 9.9] and [31]. However, the estimates presented there do not apply directly to the present situation as required. Therefore we sketch a derivation trying to make as much use as possible of related published material. In particular, we refer to [25] for the precise formulation of the quadrature rules along with corresponding error estimates. For the sake of simplicity we do not insist here on the sharpest possible bounds.

Theorem 3.9 *For any boundary integral operator $\mathcal{L}$ from the class considered in Section 1.1 any entry $\langle\psi_\lambda, \mathcal{L}\psi_{\lambda'}\rangle$ can be computed within an accuracy tolerance ε at a computational expense of $O(||\lambda|-|\lambda'||^{2d}\,|\log\varepsilon|^{2d})$ operations if $d>1$ and $O(||\lambda|-|\lambda'||^{3}\,|\log\varepsilon|^{2})$ if $d=1$.*

Proof: Without loss of generality we can always assume $\varepsilon \le 1/2$. In what follows we suppose that $l := |\lambda| \ge l' := |\lambda'|$. For the computation of $a_{\lambda',\lambda} = \langle\psi_{\lambda'}, \mathcal{L}\psi_\lambda\rangle$ the domain of integration is devided into dyadic boxes

$$[S_{\lambda'} \times S_\lambda] = \bigcup_{j=1}^{K}\bigcup_{i=1}^{L} \square_{\lambda',j} \times \square_{\lambda,i},$$

where $\psi_\lambda|_{\square_{\lambda,i}}, \psi_\lambda|_{\square_{\lambda',j}}$ are lifted polynomials of degree $m-1$, i.e. the $\square_{\lambda,i}$ are the largest regions where the wavelet ψ_λ is a smooth function. Here the cells $\square_{\lambda,i}$ will typically have the same size $\sim 2^{-l}$. Therefore L is uniformly bounded, $L \lesssim 1$. On the other hand, the $\square_{\lambda',j}$ are cells in the support $S_{\lambda'}$ of $\psi_{\lambda'}$ corresponding to *all* levels l'' between l' and l and have to be chosen so as to satisfy the admissibility condition

$$l'' < l \quad\Longrightarrow\quad \mathrm{dist}(\square_{\lambda,i}, \square_{\lambda',j}) \ge 2^{-l''} r, \tag{3.38}$$

where r is fixed constant to be specified below. Hence their number scales like the difference in levels with a constant that depends on d, i.e.,

$$K \lesssim ||\lambda|-|\lambda'|| = |l-l'|. \tag{3.39}$$

Clearly, the diameters $\mathrm{diam}\,\square_{\lambda,i}, \mathrm{diam}\,\square_{\lambda',j}$ are then bounded by 2^{-l} respectively, $2^{-l''}$, where $l' \le l'' \le l$.

Let $\square := [0,1]^d$ be the unit d-cube and suppose that $\gamma_{\lambda,i} : \square \to \square_{\lambda,i}$ and $\gamma_{\lambda',j} : \square \to \square_{\lambda',j}$ are analytic parametrizations. Denoting by $\kappa_{\lambda,i}$ and $\kappa_{\lambda',j'}$ the corresponding surface measures, the transported kernel functions

$$\hat{k}(s,t) := k(\gamma_{\lambda,i}(s), \gamma_{\lambda',j}(t))\kappa_{\lambda,i}(s)\kappa_{\lambda',j}(t)$$

are generally analytic standard kernels, that is, there exists a $\rho > 0$ such that

$$|\partial_s^\alpha \partial_t^\beta \hat{k}(s,t)| \lesssim 2^{-(|\alpha|+d)l} 2^{-(|\beta|+d)l''} \frac{(|\alpha|+|\beta|)!}{\rho^{|\alpha|+|\beta|}} \|\gamma_{\lambda,i}(s) - \gamma_{\lambda',j}(t)\|^{-(d+2q+|\alpha|+|\beta|)}, \tag{3.40}$$

see [25] for details. Now recall that, as mentioned above $\psi_\lambda(\gamma_{\lambda,i}(s))$ is a polynomial in $s \in \square$ of order m. Note that the normalization of $\psi_\lambda, \psi_{\lambda'}$ implies then for $0 \le n < m$

$$|\partial_{s_i}^n \psi_\lambda(\gamma_{\lambda,i}(s))| \lesssim 2^{(\frac{d}{2}-q)l}, \qquad |\partial_{t_i}^n \psi_{\lambda'}(\gamma_{\lambda',j}(t))| \lesssim 2^{(\frac{d}{2}-q)l'} 2^{n(l'-l'')}, \tag{3.41}$$

while these derivatives vanish for $n \ge m$. Of course, the constants depend on the (fixed) order m of the polynomials. The factor $2^{n(l'-l'')}$ in the second inequality results from the scaling induced by $\gamma_{\lambda',j}$ and an inverse inequality since $\square_{\lambda',j}$ is possibly only a portion of the domain where $\psi_{\lambda'}(\gamma_{\lambda',j})$ is a polynomial. Combining the estimates (3.40) and (3.41) and taking, in view of (3.38), $\|\gamma_{\lambda,i}(s) - \gamma_{\lambda',j}(t)\| \ge 2^{-l''} r$ into account, we find by straightforward estimates that for $n \ge m$

$$\begin{aligned}
E_{\lambda,i;\lambda',j}^{s_k,n} &:= |\partial_{s_k}^n [\hat{k}(s,t)\psi_\lambda(\gamma_{\lambda,i}(s))\psi_{\lambda'}(\gamma_{\lambda',j}(t))]| &\lesssim\ & 2^{\frac{d}{2}(l'-l)} 2^{-q(l+l'-2l'')} \frac{n!}{(r\rho)^n} 2^{(n+1-m)(l''-l)}, \\
E_{\lambda,i;\lambda',j}^{t_k,n} &:= |\partial_{t_k}^n [\hat{k}(s,t)\psi_\lambda(\gamma_{\lambda,i}(s))\psi_{\lambda'}(\gamma_{\lambda',j}(t))]| &\lesssim\ & 2^{\frac{d}{2}(l'-l)} 2^{-q(l+l'-2l'')} \frac{n!}{(r\rho)^n},
\end{aligned} \tag{3.42}$$

holds for all $s, t \in \square$. Now we fix r so that $r\rho = 1$.

We shall approximate the integral

$$I_{\lambda,i;\lambda',j} := \int_\square \int_\square \hat{k}(s,t)\psi_\lambda(\gamma_{\lambda,i}(s))\psi_{\lambda'}(\gamma_{\lambda',j}(t))dt\, ds$$

by a tensor product Gauss quadrature formula $Q_{\lambda,i;\lambda',j}^{Q_s,Q_t}$ as defined in [25]. Following Lemma 9.1 in [25], we have to estimate then

$$E_{\lambda,i;\lambda',j}^{Q_s,Q_t} := |I_{\lambda,i;\lambda',j} - Q_{\lambda,i;\lambda',j}^{Q_s,Q_t}| \lesssim \max_k \left\{ \frac{4^{-2Q_s}}{(2Q_s)!} E_{\lambda,i;\lambda',j}^{s_k,2Q_s} \right\} + \max_k \left\{ \frac{4^{-2Q_t}}{(2Q_t)!} E_{\lambda,i;\lambda',j}^{t_k,2Q_t} \right\},$$

where Q_s, Q_t are the (coordinate) degrees of the tensor product Gauss quadrature rules on $\square_{\lambda,i}, \square_{\lambda',j}$, respectively, i.e. these rules are exact of coordinate degrees $2Q_s, 2Q_t$, respectively. Consider first the case $\operatorname{dist}(\square_{\lambda,i}, \square_{\lambda',j}) \ge 2^{-l} r$ which implies $l \ge l''$. Then, invoking (3.42), we obtain the error bound

$$E_{\lambda,i;\lambda',j}^{Q_s,Q_t} \lesssim 2^{\frac{d}{2}(l'-l)} 2^{-t(l+l'-2l'')} (4^{-2Q_t} + 4^{-2Q_s} 2^{(l''-l)(2Q_s+1-m)}). \tag{3.43}$$

Choosing

$$Q_s := \frac{2Q_t + (l-l'')\frac{m-1}{2}}{2+l-l''}, \tag{3.44}$$

we obtain $4^{-2Q_s} 2^{(l''-l)(2Q_s+1-m)} \sim 4^{-2Q_t}$ and thus

$$E_{\lambda,i;\lambda',j}^{Q_s,Q_t} \lesssim 2^{\frac{d}{2}(l'-l)} 2^{-q(l'+l-2l'')} 4^{-2Q_t}. \tag{3.45}$$

There will generally remain $\mathcal{O}(1)$ nearly singular and singular integrals, i.e. dist $(\square_{\lambda,i}, \square_{\lambda',j}) < 2^{-l}r$ and $l = l''$. However, in case of the nearly singular integrals we still have dist $(\square_{\lambda,i}, \square_{\lambda',j}) > 2^{-l}c_\Gamma$, where c_Γ depends only on Γ. Hence, it suffices to choose $Q_s = Q_t$ as above for the case $l = l''$ and to apply an M-fold dyadic subdivision to both domains of integration, where $M = -\lfloor \log_2(c_\Gamma \rho) \rfloor$, in order to realize the error estimate (3.45). In fact, without such a subdivision we would obtain factors of the form $(4c_\Gamma\rho)^{-2Q}$ in (3.45), while the subdivision gives rise to factors of the form $(4c_\Gamma\rho 2^M)^{-2Q} \leq 4^{-2Q}$ as needed above. The singular integrals are treated by transformation techniques [29, 30] and then like the nearly singular integrals.

We are now prepared to assess the requirements on the quadrature that ensure a given target accuracy. On account of (3.39) and $L \lesssim 1$, to obtain now the desired accuracy ε for $a_{\lambda',\lambda}$ the individual quadrature contributions $Q_{\lambda,i;\lambda',j}$ have to be accurate within a tolerance of order $\varepsilon/|l-l'|$. To achieve this one has to choose

$$Q_t \geq \frac{1}{4}\left[|\log_2 \varepsilon| + \log_2(l-l') + \frac{d}{2}(l'-l) - q(l+l'-2l'')\right]. \tag{3.46}$$

This fixes Q_s by (3.44). Note that (3.46) can be realized by some Q_t which is bounded from above like $C(|\log_2 \varepsilon| + |l-l'|)$ for a suitable C. Therefore the number of quadrature points $\sum_{i,j=1}^{L,K} N_{\lambda,i;\lambda',j}$ can be estimated by $|\log_2 \varepsilon|^2 |l-l'|^3$ if $d=1$. If $d>1$ we can exploit the fact that Q_s is considerable smaller than Q_t when l'' is smaller than l, namely, $Q_s < 2Q_t/(2+l-l'') + (m-1)/2$. Summing over i, j can, in view of (3.39), be bounded by summing over l'' which leads to

$$\begin{aligned}\sum_{i,j=1}^{L,K} N_{\lambda,i;\lambda',j} &\lesssim \sum_{l''=l'}^{l} [C(|\log_2 \varepsilon| + |l-l'|)]^d \left[\frac{2C(|\log_2 \varepsilon| + |l-l'|)}{2+l-l''} + \frac{m-1}{2}\right]^d \\ &\lesssim (|\log_2 \varepsilon| + |l-l'|)^{2d},\end{aligned}$$

since $\sum_{l=1}^{\infty} l^{-d} \lesssim 1$ if $d > 1$. This finishes the proof. $\square$

We can now formulate the main result of this section.

Theorem 3.10 *There exists a constant C independent of $\mathbf{v}$ and $s \leq \bar{s}$ such that, employing the adaptive Gauß-Legendre quadrature techniques from [25] the work count in Theorem 3.9 is based upon, the number of operations $W(\eta, \mathbf{A})$ needed to compute the matrices $\tilde{\mathbf{A}}_j^{\Delta_j}$, $j = 0, \ldots, J = J(\eta, \mathbf{v})$ is bounded for every $s \leq \bar{s}$ by*

$$W(\eta, \mathbf{A}) \leq C\eta^{-1/s} |\mathbf{v}|_{\mathcal{A}^s}^{1/s}. \tag{3.47}$$

Moreover, the work $W(\eta, \mathbf{A}, \mathbf{v})$ needed to compute $\tilde{\mathbf{w}}_\eta = \sum_{j=0}^{J} \tilde{\mathbf{A}}_j \mathbf{v}_{[j]})$, with $\mathbf{v}_{[j]} = \mathbf{v}|_{\Delta_j}$, can be estimated by

$$W(\eta, \mathbf{A}, \mathbf{v}) \leq C(\eta^{-1/s} |\mathbf{v}|_{\mathcal{A}^s}^{1/s} + \#\mathcal{T}(\operatorname{supp} \mathbf{v})). \tag{3.48}$$

Proof: The following argument refers to a work count of the form

$$||\lambda| - |\lambda'||^{M'} |\log_2 \varepsilon|^M. \tag{3.49}$$

Above we have derived such bounds for $M' = 2d+1$, respectively $M' = 2d$ and $M = 2d$. Let us denote by W_j for $0 \le j \le J(\eta, \mathbf{v})$ the work required for the computation of $\tilde{\mathbf{A}}_j^{\Delta_j}$. First it is easy to show that the turnover level j_t for which $2^{j_t}\alpha_{j_t} \sim \#\Delta_{j_t} \lesssim (2^{\bar{s}j_t}\eta)^{-1/s}$ behaves like

$$j_t \ge \frac{J}{2} - c_{\mathbf{v}}, \tag{3.50}$$

where the constant $c_{\mathbf{v}}$ depends on $|\log_2 \|\mathbf{v}\|_{\ell_2(\mathcal{J})}|$.

We consider first the case $2^j\alpha_j \ge \#\Delta_j$, i.e. $j \ge j_t$. Recall from Proposition 3.8 the accuracy tolerance (3.33) needed for the computation of the entries of $\tilde{\mathbf{A}}_j^{\#\Delta}$. Due to Theorem 3.9 and the compression strategy (3.9), the work for computing a matrix entry in $\tilde{\mathbf{A}}_j^{\#\Delta}$ is bounded by a constant multiple of

$$\left(\frac{\hat{j}}{d-1}\right)^{M'} \left[\log_2\left(\frac{2^{-\bar{s}j}}{2^{j/2}\sqrt{\#\Delta_j/\alpha_j}}\right)\right]^M \le C\hat{j}^{M'}(j^M + |\log\eta|^M + |\log_2\alpha_j|^M + c'_{\mathbf{v}}),$$

where we have used (3.23) and $c'_{\mathbf{v}}$ depends on $|\log_2|\mathbf{v}|_{\mathcal{A}^s}|$. In order to estimate the work W_j one has to sum over $\alpha_j 2^j$ entries in each column of $\tilde{\mathbf{A}}_j^{\Delta_j}$ for $\#\Delta_j$ columns. Recalling that $|\log_2\eta| \le \bar{s}J$, we obtain

$$\begin{aligned} W_j &\lesssim \#\Delta_j\alpha_j 2^j\hat{j}^{M'}(j^M + |\log\eta|^M + |\log_2\alpha_j|^M + c_{\mathbf{v}}) \\ &\lesssim |\mathbf{v}|_{\mathcal{A}^s}^{1/s}\eta^{-1/s}2^{-j(\frac{\bar{s}}{s}-1)}\alpha_j j^{M'}(j^M + J^M + c_{\mathbf{v}}). \end{aligned}$$

Summing over $j_t \le j \le J$ gives, in view of (3.23) (see also (3.31)),

$$\sum_{j=j_t}^{J} W_j \lesssim |\mathbf{v}|_{\mathcal{A}^s}^{1/s}\eta^{-1/s}\sum_{j=J/2}^{J} 2^{-j(\frac{\bar{s}}{s}-1)}\alpha_j\hat{j}^{M'}(j^M + J^M + |\log_2\alpha_j|^M + c_{\mathbf{v}}) \lesssim |\mathbf{v}|_{\mathcal{A}^s}^{1/s}\eta^{-1/s},$$

where we have used (3.50) and the fact that $\sum_{j=J/2}^{J}\alpha_j\hat{j}^{M'}J^M \lesssim 1$.

In the the remaining case $2^j\alpha_j \le \#\Delta_j$, i.e. $j < j_t$, the work for a single nonzero matrix entry in $\tilde{\mathbf{A}}_j^{\#\Delta_j}$ is, again in view of the corresponding target accuracy from (3.33), bounded by

$$\left(\frac{\hat{j}}{d-1}\right)^{M'} (j(\bar{s}+1))^M \le C\hat{j}^{M+M'},$$

and we have to compute at most $\alpha_j 2^j\#\Delta_j$ nonzero matrix coefficients in $\mathbf{A}_j^{\Delta_j}$. Summation over $0 \le j < j_t \le J$ yields on account of (3.31)

$$\sum_{j=0}^{j_t} W_j \lesssim \sum_{j=0}^{j_t}\alpha_j\hat{j}^{M+M'}2^j\#\Delta_j \lesssim |\mathbf{v}|_{\mathcal{A}^s}^{1/s}\eta^{-1/s}\sum_{j\ge 0}\hat{j}^{M+M'}\alpha_j 2^{j(1-\bar{s}/s)} \lesssim |\mathbf{v}|_{\mathcal{A}^s}^{1/s}\eta^{-1/s},$$

where we have used again that α_j is chosen to decay so fast that $\sum_{j\in\mathbb{N}}\hat{j}^{M+M'}\alpha_j < \infty$. This completes the proof of (3.47). Since $\mathbf{A}_j^{\Delta_j}$ and $\tilde{\mathbf{A}}_j^{\Delta_j}$ have the same pattern of nonzero entries, (3.48) follows immediately from Remark 3.6. This completes the proof. □

The above considerations give rise to the following evaluation scheme:

APPLY $[\eta, \mathbf{A}, \mathbf{v}] \to \mathbf{w}_\eta$
GIVEN ANY FINITELY SUPPORTED $\mathbf{v}$ AND ANY TARGET ACCURACY $\eta > 0$ PERFORM THE FOLLOWING STEPS:

- DETERMINE THE LAYERS Δ_j, $j = 0, \dots, J = J(\eta, \mathbf{v})$ ACCORDING TO (3.13). USING THE BINEV-DEVORE TO GENERATE THE $\mathcal{T}_j$ AS INDICATED (SEE [8]);

- DETERMINE THE COMPRESSION PATTERN IN TERMS OF THE COLUMNS S_j^μ, $\mu \in \Delta_j$, $j = 0, \dots, J$ OF THE MATRICES $\mathbf{A}_j$.

- COMPUTE THE COMPRESSED BLOCKS $\tilde{\mathbf{A}}_j^{\Delta_j}$, $j = 0, \dots, J$ BY THE ADAPTIVE QUADRATURE TECHNIQUES.

- COMPUTE
$$\mathbf{w}_\eta := \sum_{j=0}^{J} \tilde{\mathbf{A}}_j(\mathbf{v}|_{\Delta_j}). \tag{3.51}$$

When $\mathcal{L}$ is symmetric positive definite (as in the case of the single layer potential) the preconditioner $\mathbf{C}$ appearing in the residual (2.12) can be taken as a damped identity $\mathbf{C} = \alpha\mathbf{I}$, where the choice of α depends on some estimates for the constants appearing in (2.2) and (1.5). Thus RES is given by (2.13). When $\mathcal{L}$ is not symmetric or indefinite one can, in principle, take $\mathbf{C} = \alpha\mathbf{A}^T$ again with a suitable damping parameter $\alpha > 0$. An application of $\mathbf{A}^T$ can be done by the same scheme and RES takes the form (2.14). Alternatively, for saddle point problems everything can be reduced to an Uzawa-based realization of operators. In all these cases the scheme APPLY given above in conjunction with COARSE (bin-thresholding for $\mathcal{P} = \mathcal{J}^{\mathcal{J}}$ or tree coarsening for $\mathcal{P} = \mathcal{T}$, see (2.13)) gives rise to a realization of RES which is s^*-*sparse* for some $s^* > (m-t)/d$. In particular, it follows now from Theorem 3.2, Proposition 3.3, Corollary 3.4, Proposition 3.8 and Theorem 3.9 that the requirements of Remark 3.7 are satisfied. The following result is then an immediate consequence of Remark 3.7 and Theorem 2.3.

Theorem 3.11 *Let* APPLY *be defined as above and* $\mathbf{C}$ *be either one of the above choices depending on the properties of the operator* $\mathcal{L}$ *of potential type. The realization of* SOLVE *based on the residual approximation* RES *of the form (2.13) or (2.14) is asymptotically optimal in the sense of Theorem 2.3.*

We have deliberately kept the structure of the scheme SOLVE simple for two reasons. First, in combination with a suitable choice of $\mathbf{C}$ it gives a scheme with asymptotically optimal complexity for a wide range of problems that need neither be symmetric nor coercive. Second we wish to bring out the essential mechanisms reflected by the notion of s^*-sparsity or by the resulting key requirements on the scheme APPLY given in Remark 2.6. These latter facts identify the essential features of APPLY that are needed in all variants or specifications of SOLVE. Those can be realized with the aid of the quadrature techniques outlined above.

We stress, however, that in any concrete application the prototype scheme SOLVE needs to by specified in order to ensure a good *quantitative* performance. We shall address this issue in the next section.

4 Implementation and Numerical Results

Theorem 3.11 so far gives only a complexity estimate in the *asymptotic* sense. A poor quantitative performance could arise from several sources. (i) For instance, COARSE will always reduce the size of a current approximation by a constant factor but this factor may be very large which means that many previously computed degrees of freedom are thrown away. Moreover, the coarsening parameter depends on an *upper bound* of the current error obtained by successive halving (see step (III) of SOLVE). The current error and the left hand side of (2.15) could actually happen to be much smaller than the right hand side. Therefore one could estimate the current error bound through the computed approximate residual, namely the left hand side of (2.15). (ii) Using the simple preconditioner $\mathbf{C} = \alpha\mathbf{I}$, one may be forced, based on the available estimates for the constants in (1.5) and (2.2), to choose α rather small which results in a (fixed but) bad error reduction in (2.1). (iii) A corresponding frequent recomputation of the compressed matrices would take its toll in computing time, although the asymptotics remain the same.

It is therefore important to *complement* the asymptotic analysis by numerical tests in combination with further specifications of the algorithmic ingredients. Here we shall demonstrate this for the boundary integral formulation of Laplace's equation based on the single layer potential (1.2) which gives rise to a symmetric positive definite problem. On can then take additional advantage of this additional structural information with regard to all the above points (i) - (iii).

Residual Coarsening: Somewhat in spirit of an earlier version of SOLVE from [6] the issue of coarsening (i) can be addressed as follows. It is shown in [20] that, when dealing with symmetric positive definite ($\mathcal{H}$-elliptic) problems, one can get rid of coarsening the current approximation of $\mathbf{u}$ provided the residual is coarsened properly. More precisely, only that part of the residual having components outside the support of the current input vector $\mathbf{v}$ need to be coarsened so that the thresholding is not diluted by inaccurate interior residual components. We refer to [20] for a detailed analysis and corresponding estimates.

Why Trees? The above analysis covers the case of unconstrained N-term approximation as optimality bench mark. However, the efficiency of implementations benefits essentially from tree structured index sets regarding the determination of the prediction set $\mathcal{T}_\eta$ from (3.15) as well as the computation of the compressed (sub)matrices in linear time. This due to the fact that to a large extent the techniques developed for matrix compression in the nonadaptive case in [15, 25] can still be employed in essentially the same way with properly chosen threshold parameters, see Theorem 3.10 above. In fact, in this setting the index sets contained always all wavelets up to some highest level J which are special cases of trees. The computation of entries $\langle\psi_\lambda, \mathcal{L}\psi_{\lambda'}\rangle = a_{\lambda,\lambda'}$ for (up to parametric lifting) piecewise polynomial wavelets was reduced to an elementwise computation of integrals based on an elementwise representation of the wavelets in terms of polynomial shape functions. Moreover, when using tree structured index sets one can still employ the "recycling" techniques from the non-adaptive setting. We refer to [24, 25] for corresponding details. In particular, when determining the compression pattern we can in the present case still take advantage of the fact that when $\lambda \in \Delta_j$, say, then the father $\hat{\lambda}$ belongs to some $\Delta_{j'}$ for $j' \geq j$. Moreover, we still have that whenever for $|\hat{\lambda}| \geq |\nu|$ an entry $\tilde{a}_{\hat{\lambda},\nu}$ of the compressed matrix vanishes, then also $\tilde{a}_{\lambda,\nu} = 0$ for the son λ of $\hat{\lambda}$. Especially when $|\hat{\lambda}| = |\hat{\nu}|$ then $\tilde{a}_{\hat{\lambda},\hat{\nu}} = 0$ implies $\tilde{a}_{\lambda,\nu} = 0$ for the sons λ, ν.

Moreover, one can still take advantage of *symmetrization* as described in [25] for the nonadaptive case. It suffices to compute only the columns of the compressed matrices in the lower half of the compressed matrix and complete the matrix by symmetry. Recall from Theorem 3.2 the approximation $\mathbf{w}_\eta = \sum_{j=0}^{J} \mathbf{A}_j \mathbf{v}|_{\Delta_j}$ can be schematically written as

$$(\mathbf{A}\mathbf{v})_{\text{approx}} = \left[\begin{array}{c|c|c|c|c|c} \mathbf{A}_J & \mathbf{A}_{J-1} & \mathbf{A}_{J-2} & \cdots & \mathbf{A}_0 & \mathbf{A}_{-1} \end{array}\right] \cdot \left[\begin{array}{c} \mathbf{v}|_{\Delta_J} \\ \hline \mathbf{v}|_{\Delta_{J-1}} \\ \hline \mathbf{v}|_{\Delta_{J-2}} \\ \hline \vdots \\ \hline \mathbf{v}|_{\Delta_0} \\ \hline \mathbf{0}|_{\Delta_{-1}} \end{array}\right]$$

where $\mathcal{T}_{-1} := \mathcal{T}_\eta$, i.e., $\Delta_{-1} := \mathcal{T}_\eta \setminus \mathcal{T}_0$. This involves (formally) a $(\#\mathcal{T}_\eta) \times (\#\mathcal{T}_\eta)$ matrix. Suppose now that only the significant entries of the column $\mathbf{A}_j^{\Delta_j}$, identified by (3.9), below and up to the diagonal have been computed approximately by the quadrature techniques described above. Note that, whenever $\lambda' \in S_j^\lambda$ belongs to the support of the column $\mathbf{A}_j^\lambda$ with $|\lambda'| > |\lambda|$. By symmetry also $\lambda \in S_j^{\lambda'}$, i.e., $a_{\lambda',\lambda} = a_{\lambda,\lambda'}$ needs to be computed. Thus, reflecting the lower part of the columns of $\tilde{\mathbf{A}}_j^{\Delta_j}$ across the diagonal creates a matrix whose entry pattern contains the one of $\tilde{\mathbf{A}}_j^{\Delta_j}$. Therefore,

$$(\mathbf{A}\mathbf{v})_{\text{approx}} = \left[\begin{array}{c|c|c|c} A_{J,J} & A_{J,J-1} & \cdots & A_{J,0} \\ \hline A_{J-1,J} & A_{J-1,J-1} & \cdots & A_{J-1,-1} \\ \hline \vdots & \vdots & & \vdots \\ \hline A_{0,J} & A_{0,J-1} & \cdots & A_{-1,-1} \end{array}\right] \cdot \left[\begin{array}{c} \mathbf{v}|_{\Delta_J} \\ \hline \mathbf{v}|_{\Delta_{J-1}} \\ \hline \vdots \\ \hline \mathbf{0}_{\Delta_{-1}} \end{array}\right]. \tag{4.1}$$

gives rise to an even more accurate approximation, where the involved symmetric matrix can be generated along the lines of [25].

The Preconditioner: In spite of the linear work count, the computation of the compressed matrices is the most time consuming part. So it would be reasonable to exploit the invested work as much as possible. Upgrading accuracy after every application of RES would degrade the quantitative performance significantly. This can be remedied by freezing the space associated with the current prediction set $\mathcal{T}_\lambda$, computing the residual at a somewhat higher accuracy and then use any preconditioner to solve the corresponding fixed Galerkin problem within a suitable accuracy. Formally this process could be viewed as the application of $\mathbf{C}$. In the subsequent numerical tests this is combined with the residual coarsening version mentioned above.

Numerical Results: We shall present next our numerical results. We solve an interior Dirichlet problem for the Laplacian by the indirect approach using the single layer potential operator. This gives rise to a Fredholm integral equation of the first kind for an unknown density u. As domain Ω we consider the Fichera vertex $[0,1]^3 \setminus [0,1/2]^3$ as a first example and a crankshaft as a second one. The surface of the geometries are represented via 12 and 142 patches, respectively, as depicted in Figure 4.1.

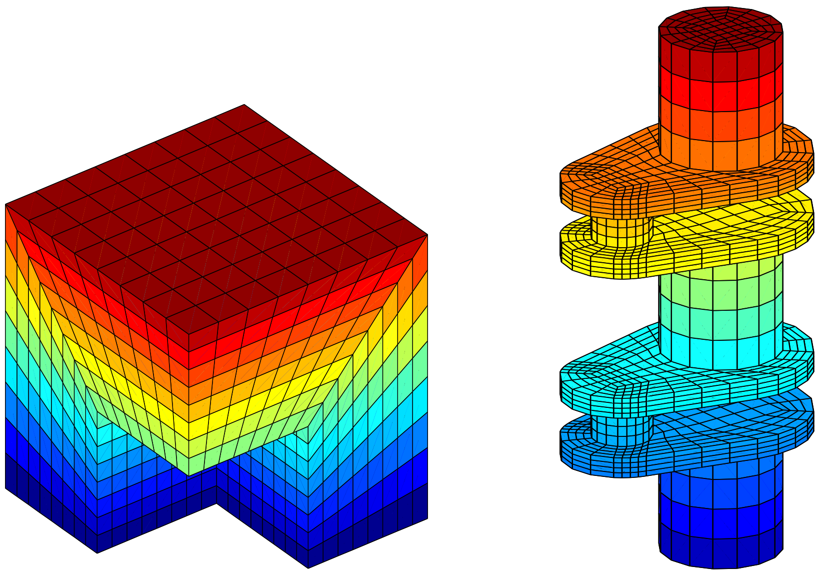

Figure 4.1: The Fichera vertex and the crankshaft.

As Dirichlet data we choose the restriction of the harmonic function

$$U(x) = \frac{(a, x-b)}{\|x-b\|^3}, \quad a = (1,2,4), \quad b = (1.5,0,0) \notin \Omega$$

to Γ. Then, U is the unique solution of the Dirichlet problem. We discretize the given boundary integral equation by piecewise constant wavelets with three vanishing moments. We emphasize that we do not know the density function u in the indirect method. However, since the density u exhibits the singularities of both, the interior and exterior Laplace equation, it is in fact not in $H^1(\Gamma)$.

In order to measure the error produced by the method, we consider point evaluations $U_N = Au_N$ inside the domain, which yields the following estimate in case of the best-N-term approximation

$$|U(x) - U_N(x)| = |A(u - u_N)(x)| \lesssim \|u - u_N\|_{H^{-1/2}(\Gamma)} \lesssim N^{-0.75}.$$

To remain on save grounds, we evaluate the approximate solution in several points inside the domain. Then, denoting the discrete potentials by

$$\mathbf{U} := [U(x_i)], \qquad \mathbf{U}_N := [(Au_N)(x_i)],$$

we measure the error $\|\mathbf{U} - \mathbf{U}_N\|_\infty$.

n	$N^{(n)}$	nnz $(\mathbf{res}^{(n-1)})$	$\|\mathbf{res}^{(n-1)}\|$	$\|\mathbf{U} - \mathbf{U}_{N^{(n)}}\|_\infty$	%	cpu-time
1	45	48	2.4	4.6e-1	100.00	0
2	126	185	1.7	1.7e-1	87.25	0
3	349	504	6.3e-1	1.0e-1	51.96	1
4	916	1496	3.2e-1	6.5e-2	29.66	2
5	2353	4210	1.5e-1	1.7e-3	13.81	13
6	5200	10701	7.9e-2	1.1e-3	6.50	36
7	11531	24408	4.9e-2	1.0e-3	2.98	95
8	25011	58261	3.0e-2	4.7e-4	1.34	243
9	54044	129223	2.0e-2	2.9e-4	0.62	609
10	119966	293707	1.2e-2	1.7e-4	0.27	1558
11	246318	664292	8.5e-3	4.9e-5	0.13	3818

Table 1: Numerical results with respect to the Fichera vertex.

First, we list in Table 1 the results produced by the adaptive wavelet Galerkin scheme in case of the Fichera vertex. We start the iteration with the single scale basis, that is the number of active components is $N^{(0)} = 12$. The first column shows the number n of the iteration step, the second one the actual number of unknowns $N^{(n)}$, which are derived from the coarsening of the residuum $\mathbf{res}^{(n-1)}$ of the previous step. Its support length and norm is presented in the third and fourth column. The column entitled by "$\|\mathbf{U} - \mathbf{U}_{N^{(n)}}\|_\infty$" refers to the error of the potential. According to Figure 4.2 both, the norm of the residuum and the potential, behave like the best possible rate $N^{-0.75}$. The column entitled by "%" refers to the number of nonzero coefficients (measured in percent) of the Galerkin matrix. It verifies that the Galerkin matrix becomes also quite sparse in the adaptive case. The last column refers to the cpu-time required to compute and coarsen the residual arising from the previous iteration and to compute and solve the Galerkin system of the actual iteration. The cpu-time seems in fact to scale linearly with the number of unknowns.

Next, we consider the more complicated crankshaft as domain Ω. As for the Fichera vertex, we listed the observed results in Table 2. We start again the iteration with the single scale basis, that is $N^{(0)} = 142$. Also in this example both, the norm of the residual and the potential, behave like $N^{-0.75}$, see also Figure 4.3. Likewise to above, the Galerkin matrix is quite sparse while the cpu-time seems to scale linearly with the number of unknowns.

References

[1] A. Barinka, Fast computation tools for adaptive wavelet methods, Doctoral Dissertation, RWTH Aachen, 2005.

[2] G. Beylkin, R. R. Coifman, V. Rokhlin, Fast wavelet transforms and numerical algorithms I, Comm. Pure and Appl. Math., 44 (1991), 141–183.

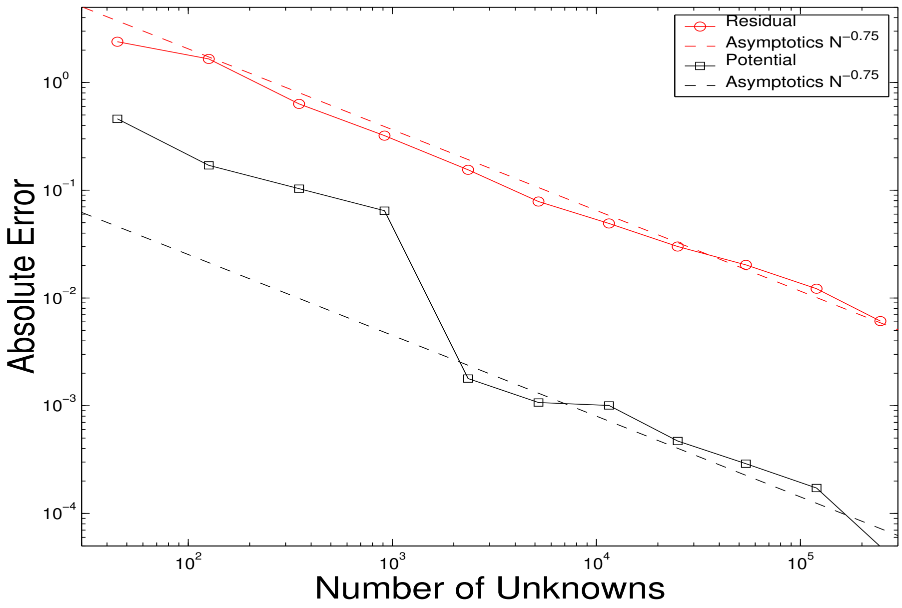

Figure 4.2: Norm of the residual and potential error versus the number of unknowns.

[3] P. Binev and R. DeVore (2004) *Fast computation in adaptive tree approximation*, Num. Math. **97**, 193–217.

[4] A. Canuto, A. Tabacco, K. Urban, The wavelet element method, part I: Construction and analysis, Appl. Comp. Harm. Anal., 6(1999), 1–52.

[5] A. Canuto, A. Tabacco, K. Urban, The wavelet element method, part II: Realization and additional features, Appl. Comp. Harm. Anal. 8 (2000), 123–165.

[6] A. Cohen, W. Dahmen, R. DeVore, Adaptive wavelet methods for elliptic operator equations – Convergence rates, Math. Comp. 70 (2001), 27–75.

n	$N^{(n)}$	$\mathrm{nnz}\,(\mathbf{res}^{(n-1)})$	$\\|\mathbf{res}^{(n-1)}\\|$	$\\|\mathbf{U}-\mathbf{U}_{N^{(n)}}\\|_\infty$	%	cpu-time
1	467	568	5.3	2.6	81.59	1
2	515	1868	2.9	7.6e-1	52.23	1
3	891	2503	1.2	1.3e-1	31.54	4
4	2358	3887	3.6e-1	8.7e-2	15.31	10
5	7590	11541	1.5e-1	4.8e-2	7.10	58
6	20228	37942	6.9e-2	8.8e-3	3.12	257
7	49234	93175	3.1e-2	4.9e-3	1.28	696
8	104267	254736	2.3e-2	2.9e-3	0.64	2463
9	210745	569833	1.3e-2	1.3e-3	0.29	5425

Table 2: Numerical results for the crankshaft.

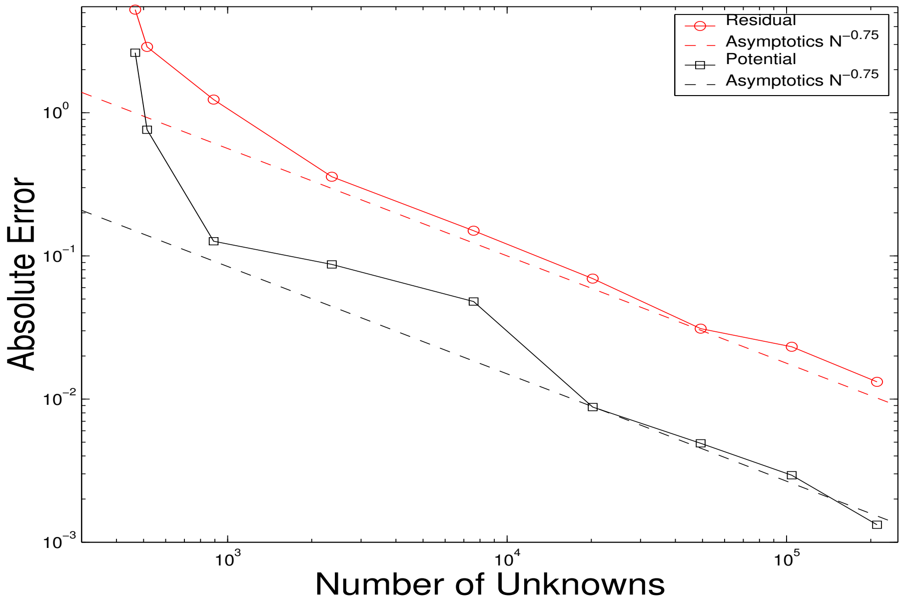

Figure 4.3: Norm of the residual and potential error versus the number of unknowns.

[7] A. Cohen, W. Dahmen, R. DeVore, Adaptive wavelet methods II - Beyond the elliptic case, Foundations of Computational Mathematics, 2 (2002), 203–245.

[8] A. Cohen, W. Dahmen, R. DeVore, Adaptive Wavelet Schemes for Nonlinear Variational Problems, SIAM J. Numer. Anal., (5)41(2003), 1785–1823.

[9] A. Cohen, W. Dahmen, R. DeVore, Sparse evaluation of compositions of functions using multiscale expansions, SIAM J. Math. Anal., 35 (2003), 279–303.

[10] Cohen, A., W. Dahmen, I. Daubechies and R. DeVore (2001) *Tree-structured approximation and optimal encoding*, App. Comp. Harm. Anal. **11**, 192–226.

[11] A. Cohen, R. Masson, Wavelet adaptive methods for second order elliptic problems, boundary conditions and domain decomposition, Numer. Math., 8 (1997), 21–47.

[12] S. Dahlke, W. Dahmen, R. DeVore, Nonlinear approximation and adaptive techniques for solving elliptic operator equations, in: *Multiscale Wavelet Methods for PDEs*, W. Dahmen, A. Kurdila, P. Oswald (eds.), Academic Press, London, 237–283, 1997.

[13] S. Dahlke, W. Dahmen, K. Urban, Adaptive wavelet methods for saddle point problems – Convergence rates, SIAM J. Numer. Anal., 40 (No. 4) (2002), 1230–1262.

[14] W. Dahmen, Wavelet and Multiscale Methods for Operator Equations, (invited contribution) Acta Numerica, Cambridge University Press, 6(1997), 55–228.

[15] W. Dahmen, H. Harbrecht, R. Schneider, Compression Techniques for Boundary Integral Equations – Optimal Complexity Estimates, IGPM Report # 218, RWTH Aachen, June 2002.

[16] W. Dahmen, S. Prößdorf, R. Schneider, Multiscale methods for pseudo-differential equations on smooth manifolds, in: *Proceedings of the International Conference on Wavelets: Theory, Algorithms, and Applications*, C.K. Chui, L. Montefusco, L. Puccio (eds.), Academic Press, 1994, 385-424.

[17] W. Dahmen and R. Schneider, Composite Wavelet Bases for Operator Equations, Math. Comp., 68 (1999), 1533–1567.

[18] DeVore, R. (1998) *Nonlinear approximation*, Acta Numerica **7**, 51-150.

[19] B. Faermann, Localization of the Aronszajn-Slobodeckij norm and application to adaptive boundary element methods. Part II. The three-dimensional case. Numer. Math., 92 (2002), 467–499.

[20] T. Gantumur, H. Harbrecht, R. Stevenson, An Optimal Adaptive Wavelet Method for Elliptic Equations Without Coarsening. Preprint No. 1325, Department of Mathematics, University of Utrecht, 2005.

[21] L. Greengard and V. Rokhlin, A fast algorithm for particle simulation, J. Comput. Phys., 73 (1987), 325–348.

[22] W. Hackbusch, A sparse matrix arithmetic based on $\mathcal{H}$-matrices. Part I: Introduction to $\mathcal{H}$-matrices, Computing, 64 (1999), 89–108.

[23] W. Hackbusch and Z.P. Nowak, On the fast matrix multiplication in the boundary element method by panel clustering, Numer. Math., 54 (1989), 463–491.

[24] H. Harbrecht, Wavelet Galerkin schemes for the boundary element method in three dimensions, *PHD Thesis, Technische Universität Chemnitz, Germany*, 2001.

[25] H. Harbrecht, R. Schneider, Wavelet Galerkin Schemes for Boundary Integral Equations – Implementation and Quadrature, *Preprint SFB 393/02-21, Technische Universität Chemnitz, Germany*, 2002, submitted to SISC.

[26] T. von Petersdorff, C. Schwab, Wavelet approximation for first kind integral equations on polygons, Numer. Math., 74 (1996), 479–519.

[27] T. von Petersdorff, C. Schwab, Fully discrete multiscale Galerkin BEM, in: *Multiscale Wavelet Methods for PDEs*, W. Dahmen, A. Kurdila, P. Oswald (eds.), Academic Press, San Diego, 1997, 287–346.

[28] T. von Petersdorff, C. Schwab, Fully discrete multiscale Galerkin BEM, in: *Multiscale Wavelet Methods for PDEs*, W. Dahmen, A. Kurdila, P. Oswald (eds.), Academic Press.

[29] S. Sauter, Über die effiziente Verwendung des Galerkin Verfahrens zur Lösung Fredholmscher Integralgleichungen. *PHD Thesis, Christian–Albrecht–Universität Kiel, Germany*, 1992.

[30] S. Sauter and C. Schwab. Quadrature for the *hp*-Galerkin BEM in $\mathbb{R}^3$. *Numer. Math.*, 78:211–258, 1997.

[31] S. Sauter, C. Schwab, Randelementmethoden: Analyse, Numerik und Implementierung schneller Algorithmen, B.G. Teubner, Stuttgart, 2005.

[32] G. Schmidlin, C. Lage, C. Schwab, Rapid solution of first kind boundary integral equations in $\mathbb{R}^3$, Research Report No. 2002-07, Seminar für Angewandte Mathematik, ETH Zürich.

[33] R. Schneider, *Multiskalen- und Wavelet-Matrixkompression: Analysisbasierte Methoden zur Lösung großer vollbesetzter Gleichungssysteme*, B.G. Teubner, Stuttgart, 1998.

[34] R. Stevenson, On the compressibility of operators in wavelet coordinates, *SIAM J. Math. Anal.* 35(5), 1110–1132 (2004).

[35] P. Tchamitchian, Wavelets, Functions, and Operators, in: *Wavelets: Theory and Applications*, G. Erlebacher, M.Y. Hussaini, and L. Jameson (eds.), ICASE/LaRC Series in Computational Science and Engineering, Oxford University Press, 1996, 83–181.

[36] G. Verchota, Layer potentials and regularity for the Dirichlet problem for the Laplace's equation in Lipschitz domains, J. Funct. Anal., 59, 572–611 (1984).