Fast Computation of Adaptive Wavelet Expansions *

A. Barinka†, W. Dahmen†, and R. Schneider‡

August 4, 2004

Abstract

In this paper we describe and analyze an algorithm for the fast computation of sparse wavelet coefficient arrays typically arising in adaptive wavelet solvers. The scheme improves on an earlier version from [17] in several respects motivated by recent developments of adaptive wavelet schemes. The new structure of the scheme is shown to enhance its performance while a completely different approach to the error analysis accommodates the needs put forward by the above mentioned context of adaptive solvers. The results are illustrated by numerical experiments for one and two dimensional examples.

AMS Subject Classification: 41A25, 41A55, 41A46, 65D99, 65M12
Key Words: Tree approximation, local multiscale transforms, adaptive wavelet schemes, quadrature, dual norms, error estimates

1 Introduction

During the past few years the design and analysis of adaptive wavelet techniques has undergone a rapid development. In particular, this concerns the theoretical understanding of convergence and complexity estimates for a new algorithmic paradigm [6, 7, 8, 9, 12]. These results bring now also new demands on computational tools into focus which try to exploit to the best possible extent the (near) sparseness of wavelet representations of functions and operators.

The central theme is the development of computational schemes that are capable of solving a given problem at a computational expense that stays proportional to the number of degrees of freedom that are intrinsically needed to achieve a desired target accuracy. In the context of the above mentioned adaptive schemes this boils down to a computational task that will be outlined next, see also [8].

1.1 Problem Formulation

The central problem treated in this paper will be explained now in a sufficiently general format in order to accommodate later several different cases of interest. In the following $\mathcal{H}$ will always denote a given Hilbert space which plays, for instance, the role of an energy space for a variational problem. One should think of $\mathcal{H}$, for instance, as $L_2(\Omega)$, Ω a domain or manifold, or as a (closed subspace of a) Sobolev space (defined e.g. by homogeneous boundary conditions) or a product of

*Work supported in part by the European Community's Human Potential Programme under contract HPRN-CT-2002-00286, [Breaking Complexity]

†Institut für Geometrie und Praktische Mathematik, RWTH Aachen, Templergraben 55, 52056 Aachen {barinka, dahmen}@igpm.rwth-aachen.de

‡Scientific Computing, Christian-Albrechts-Platz 4, 24118 Kiel, rs@numerik.uni-kiel.de

such spaces. The space of bounded linear functionals on $\mathcal{H}$, the normed dual of $\mathcal{H}$, is denoted by $\mathcal{H}'$. The dual pairing $\langle\cdot,\cdot\rangle$ on $\mathcal{H}\times\mathcal{H}'$ is always tacitly be assumed to be induced by the standard L_2-inner product on the underlying domain Ω. As usual we set $\|w\|_{\mathcal{H}'} := \sup_{\|v\|_{\mathcal{H}}\leq 1}\langle v, w\rangle$.

Suppose now that

$$\Psi = \{\psi_\lambda : \lambda \in \mathcal{J}\} \subset \mathcal{H}$$

is a *Riesz basis* for $\mathcal{H}$, which means that there exist constants c_Ψ, C_Ψ such that

$$c_\Psi\|\mathbf{v}\|_{\ell_2} \leq \|\sum_{\lambda\in\mathcal{J}} v_\lambda\psi_\lambda\|_{\mathcal{H}} \leq C_\Psi\|\mathbf{v}\|_{\ell_2}, \quad \forall \ \ \mathbf{v}\in\ell_2. \tag{1.1}$$

Here $\mathbf{v} = (v_\lambda)_{\lambda\in\mathcal{J}}$ is for any $v\in\mathcal{H}$ the array of coefficients in the unique expansion $v = \sum_{\lambda\in\mathcal{J}} v_\lambda\psi_\lambda$. Bases Ψ satisfying (1.1) for the above mentioned examples of $\mathcal{H}$ are available as *wavelet bases*, see e.g. [4, 5, 15, 16, 11].

The central problem treated in this paper can be formulated as the following *recovery* **Task R** which has two parts:

Task R1: Given $g\in\mathcal{H}'$ and some finite subset $\mathcal{T}\subset\mathcal{J}$ of indices, compute an array $\mathbf{w}$ supported in $\mathcal{T}$ that approximates $\mathbf{g}_\mathcal{T} := (g_\lambda)_{\lambda\in\mathcal{T}} := (\langle\psi_\lambda, g\rangle)_{\lambda\in\mathcal{T}}$.

Task R2: Moreover, whenever for a given target accuracy $\varepsilon>0$ the truncation error satisfies $\|\mathbf{g}-\mathbf{g}(\mathcal{T})\|_{\ell_2}\leq\varepsilon$, the approximation $\mathbf{w}$ should satisfy for some fixed constant C $\|\mathbf{g}(\mathcal{T})-\mathbf{w}\|_{\ell_2}\leq C\varepsilon$, i.e. the approximation error should be comparable to a given bound for the truncation error.

To understand this task, recall that the Riesz basis property of Ψ implies the existence of a *dual basis* $\tilde\Psi\subset\mathcal{H}'$ such that $\langle\psi_\lambda,\tilde\psi_\mu\rangle = \delta_{\lambda,\mu}$. Thus, the $g_\lambda = \langle\psi_\lambda, g\rangle$ are the expansion coefficients of $g\in\mathcal{H}'$ with respect to the dual basis

$$g = \sum_{\lambda\in\mathcal{J}}\langle\psi_\lambda, g\rangle\tilde\psi_\lambda.$$

Moreover, a duality argument yields (see e.g. [13])

$$C_\Psi^{-1}\|(\langle\psi_\lambda, w\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2} \leq \|w\|_{\mathcal{H}'} \leq c_\Psi^{-1}\|(\langle\psi_\lambda, w\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2}, \quad \forall \ \ w\in\mathcal{H}'. \tag{1.2}$$

Hence

$$\|\mathbf{w}-\mathbf{g}_\mathcal{T}\|_{\ell_2} \leq C_\Psi\|\sum_{\lambda\in\mathcal{T}} w_\lambda\tilde\psi_\lambda - \sum_{\lambda\in\mathcal{T}} g_\lambda\tilde\psi_\lambda\|_{\mathcal{H}'}, \tag{1.3}$$

i.e. the accuracy of $\mathbf{w}$ as an approximation to $\mathbf{g}_\mathcal{T}$ is controlled by an *approximation error* for the truncated expansion $g_\mathcal{T} := \sum_{\lambda\in\mathcal{T}} g_\lambda\tilde\psi_\lambda$ in $\mathcal{H}'$.

As will be seen later, the approximation of the finite array $\mathbf{g}_\mathcal{T}$ is often just a vehicle for fabricating an approximation to the *whole infinite* array $\mathbf{g}$. The rationale then is that, from some background information, the set $\mathcal{T}$ is known to contain the most significant terms $\langle\psi_\lambda, g\rangle$ of the whole expansion. Since, due to the norm equivalences, accuracy is measured in ℓ_2, "most significant" means here the largest in modulus. The relevant error is then controlled by

$$\|\mathbf{w}-\mathbf{g}\|_{\ell_2} \leq C_\Psi\|\sum_{\lambda\in\mathcal{T}} w_\lambda\tilde\psi_\lambda - \sum_{\lambda\in\mathcal{T}} g_\lambda\tilde\psi_\lambda\|_{\mathcal{H}'} + \|\mathbf{g}-\mathbf{g}_\mathcal{T}\|_{\ell_2}, \tag{1.4}$$

which explains **Task R2**. In fact, ideally the *approximation error* of the finite expansion in the first term should be at most of the same order as (a bound for) the *prediction error* in the second term.

1.2 Some Background and Motivation

One might think of many situations where the computation of wavelet coefficients is needed. In particular, the problem of computing sparse wavelet expansions (or nonlinear composition of such) has been addressed in earlier investigations, see e.g. [19, 20, 18, 3]. However, the results obtained there (with varying level of rigor) do not quite seem to meet the requirements mentioned above. The considerations in the present paper are guided primarily by the following two scenarios whose brief description may help identifying the particular computational demands and motivates **Task R**.

L_2-approximation: The first case concerns simply $\mathcal{H} = L_2(\Omega)$ (assuming that a wavelet basis is available). Thus $\mathcal{H}' = \mathcal{H} = L_2(\Omega)$ and the objective is to approximate g in $L_2(\Omega)$ by some finite expansion $\sum_{\lambda\in\mathcal{T}} w_\lambda\tilde{\psi}_\lambda$ (where the role of primal and dual basis does not matter). By (1.3), the accuracy of the coefficient array is comparable to the approximation error in L_2 which will later be seen to simplify matters and is essentially the situation considered in [17].

Dual norms: The second scenario typically involves topologies where $\mathcal{H}$ is compactly embedded in $L_2(\Omega)$ such as a Sobolev space of positive order. A corresponding model problem may be formulated as follows, see [8] for more details. Given $\Omega \subset \mathbb{R}^d$, consider

$$-\mathrm{div}(a\nabla u) + G(u) = f \quad \text{in} \quad \Omega, \quad u = 0 \quad \text{on} \quad \partial\Omega, \tag{1.5}$$

where $G : v \mapsto G\circ v = G(v)$ is a possibly nonlinear composition map and a is a (possibly variable) uniformly positive definite matrix. For this problem to be well-posed it helps looking at the weak formulation

$$\langle\nabla v, a\nabla u\rangle + \langle v, G(u)\rangle = \langle v, f\rangle, \quad v \in \mathcal{H}, \tag{1.6}$$

where now $\mathcal{H}$ is to be chosen suitably. Clearly the leading second order term suggests $\mathcal{H} = H^1_0(\Omega)$ (the space of L_2-functions on Ω whose first order derivatives are also in L_2 and whose trace on $\partial\Omega$ vanishes). Thus we require that G maps $\mathcal{H} = H^1_0(\Omega)$ into $\mathcal{H}' = H^{-1}(\Omega)$ so that (1.6) makes sense whenever the data f belong to $\mathcal{H}' = H^{-1}(\Omega)$ as well. The general format of such problems can be stated as follows. Given $F : \mathcal{H} \to \mathcal{H}'$ and any $f \in \mathcal{H}'$ we wish to find a $u \in \mathcal{H}$ such that

$$\langle v, F(u)\rangle = \langle v, f\rangle \qquad \forall \quad v \in \mathcal{H}, \tag{1.7}$$

where this problem is assumed to be *well-posed* in the sense that the local linearization of F defines in a neighborhood of a locally unique solution u an isomorphism from $\mathcal{H}$ onto $\mathcal{H}'$, [8, 9] (see also [10]). In the above example it is not hard to confirm well-posedness when G is monotone so that (1.6) is the Euler-Lagrange equation of a strictly convex minimization problem. For instance,

$$G(v) := v^3 \quad \Longrightarrow \quad G : v \mapsto G(v),\ G : H^1_0(\Omega) \to H^{-1}(\Omega), \quad \text{for} \quad d \le 3. \tag{1.8}$$

To explain how **Task R** fits into this context we briefly recall the main ideas from [8], see also [10]. The strategy proposed there deviates from conventional approaches in that the problem is first transformed with the aid of the basis Ψ into an equivalent one that lives now on ℓ_2. In fact, testing both sides of $F(u) = f$ with all basis functions $\psi_\lambda, \lambda \in \mathcal{J}$ generates for $u = \sum_{\lambda\in\mathcal{J}} u_\lambda\psi_\lambda$ the arrays $\mathbf{F}(\mathbf{u}) = (\langle\psi_\lambda, F(u)\rangle)_{\lambda\in\mathcal{J}}$, $\mathbf{f} = (\langle\psi_\lambda, f\rangle)_{\lambda\in\mathcal{J}}$ so that (1.7) is equivalent to

$$\mathbf{F}(\mathbf{u}) = \mathbf{f}, \tag{1.9}$$

where $\mathbf{u}$ is the array of wavelet coefficients of u with respect to the Riesz basis Ψ for $\mathcal{H}$. Moreover, one derives from the Riesz basis property and the above mentioned well-posedness that (1.9) is

now well-posed in ℓ_2. This means that its local linearizations are boundedly invertible on ℓ_2. This in turn, can be used to contrive an iteration

$$\mathbf{u}^{n+1} = \mathbf{u}^n - \mathbf{C}_n(\mathbf{F}(\mathbf{u}^n) - \mathbf{f}), \quad n = 0, 1, 2, \ldots, \tag{1.10}$$

where the (infinite) matrix $\mathbf{C}_n$ is a preconditioner, so that (for a suitable initial guess $\mathbf{u}_0$ with u_0 in some neighborhood $\mathcal{U}$ of $\mathbf{u}$) the error is reduced in each step by at least a fixed factor $\rho < 1$, i.e.

$$\|\mathbf{u} - \mathbf{u}^{n+1}\|_{\ell_2} \leq \rho \|\mathbf{u} - \mathbf{u}^n\|_{\ell_2}, \quad n = 0, 1, 2, \ldots. \tag{1.11}$$

Only then one addresses a numerical realization by carrying out such an iteration approximately. The key task is then to evaluate *adaptively* for a given finitely supported input array $\mathbf{v}$ the image $\mathbf{F}(\mathbf{v})$ within some dynamically updated accuracy tolerance. Such an evaluation hinges on two pillars, namely first the *a-posteriori* information on the given input v (i.e. the knowledge of the significant coefficients in $\mathbf{v}$) and, second, the *a-priori* information on the mapping F, viz. $\mathbf{F}$. Aside from the norm equivalences (1.1), at this point a second key feature of wavelet bases namely the *cancellation properties* often expressed in terms of *vanishing moments*, come into play see [9]. This will allow one then to *predict* the indices hosting the significant coefficients of the image $\mathbf{F}(\mathbf{v})$. (The prediction set $\mathcal{T}$ will actually in general not be completely arbitrary but will have some tree structure as will be explained later, see [9]). The structure of such an evaluation scheme looks therefore as follows:

(I) Given an η-significant set of indices $\mathcal{T}_\eta(\mathbf{v})$ for $\mathbf{v}$, i.e. $\|\mathbf{v} - \mathbf{v}|_{\mathcal{T}_\eta(\mathbf{v})}\|_{\ell_2} \leq \eta$, *predict* a possibly small set of indices $\mathcal{T} = \mathcal{T}_\eta(\mathbf{F}, \mathbf{v})$ such that for a fixed constant C

$$\|\mathbf{F}(\mathbf{v}) - \mathbf{F}(\mathbf{v})|_{\mathcal{T}}\|_{\ell_2} \leq C\eta. \tag{1.12}$$

(II) Compute an approximation $\mathbf{w}_\eta$ such that

$$\|\mathbf{F}(\mathbf{v})|_{\mathcal{T}} - \mathbf{w}_\eta\|_{\ell_2} \leq C\eta. \tag{1.13}$$

Thus here we have $g = F(v) \in \mathcal{H}'$ and the set of significant coefficients identified by the index set $\mathcal{T}$ is *known* here from the prediction in step (I). For the concrete construction of *asymptotically optimal* prediction sets for a certain class of linear and nonlinear operators F (of at most polynomial growth at infinity) we refer to [9]. In brief it means that, given a target accuracy $\varepsilon > 0$, one can predict an ε-significant tree $\mathcal{T} = \mathcal{T}_\varepsilon$ such that $\|\mathbf{g} - \mathbf{g}(\mathcal{T})\|_{\ell_2} \leq \varepsilon$ while the cardinality of $\mathcal{T}_\varepsilon$ grows with decreasing ε at some optimal rate. In [8] the precise requirements on the approximation of $\mathbf{F}(\mathbf{v})$ have been identified under which the overall adaptive scheme has asymptotically optimal complexity, see the notion of s^*-sparsity in [8]. Together with the above mentioned prediction results these requirements are met whenever **Task R** is fulfilled.

Therefore we focus in this paper on step (II) which is exactly **Task R**. It will be important that the accuracy in the ℓ_2-error (1.12) is here related to the accuracy of the corresponding functions (or distributions) in $\mathcal{H}'$, due to the above mentioned mapping properties of F.

Of course, in order to benefit from the ability of the above schemes to optimally track the significant coefficients of the unknown solution, one would like to compute the entries of $\mathbf{w}$ in a possibly efficient way which means at a computational cost (in terms of floating point operations and storage manipulations) that ideally stays proportional to $\#\mathcal{T}$. Moreover, one has to assert the accuracy of such computations e.g. via estimates like (1.4).

A first natural idea would be to compute each individual quantity $\langle \psi_\lambda, g \rangle$ by quadrature. However, a quick thought confirms that this would never allow one to keep the above desirable

computational budget. In fact, the coarse scale wavelets give rise to integration domains that are comparable to the size of Ω and each of these entries would require a growing quadrature effort with increasing accuracy. In addition wavelets from different scales still interact.

Instead we shall pursue a different strategy and take up an idea suggested already in [17]. There an analytic framework has been proposed that facilitates complexity estimates also for those situations where the sparseness of wavelet expansions is significant and prohibits transformations to more local single scale representations.

The work in this paper improves on these earlier findings in several respects. First, on the algorithmic side, we shall develop and analyze here a new scheme which differs from that in [17] in important points. It has a much simpler structure and offers a significant quantitative improvement of performance and storage demands. The main point is that in [17] at some point a full usually pessimistically large prediction index set had to be assembled and allocated. Moreover, approximate coefficients were generated through a coarse-to-fine-to-coarse sweep. This is avoided in the new scheme which generates the significant inner products in a single sweep from fine to coarse. This results in an overall more simply structured algorithm with less subroutines and reduced storage demands.

Second, we present here a different approach to the complexity analysis which addresses the demands put forward by the above mentioned recent developments of adaptive wavelet methods. The major deficiencies of the approach in [17] can be summarized as follows. The error analysis relied on relating the accuracy of the computed entries to an approximation error of a function approximation in L_2. As pointed out above, in the context of adaptive schemes the role of L_2 is played by the dual $\mathcal{H}'$ of an energy space, typically a Sobolev space of negative order. We shall explain later in Section 3 in more detail why, in spite of norm equivalences induced by scaling wavelet bases, this poses a serious obstruction. Moreover, the error analysis in [17] worked under the assumption that local polynomial errors (caused by quadrature) are essentially equibalanced, a fact that can generally not be guaranteed in the context of interest. Finally, the regularity assumptions which the error analysis in [17] was based upon are not quite compatible with the demands of the above mentioned developments.

1.3 The Layout

The layout of the paper is as follows. At first, we collect a few wavelet prerequisites and introduce the notion of *index-trees* in Section 2, which will play a pivotal role for the recovery scheme.

Section 3 is devoted to an outline of the basic strategy of our approach as well as to the introduction of our main computational scheme RECOVER.

Section 4 is devoted to accuracy considerations for the two scenarios indicated above. The main focus is on the balance between the truncation and quadrature error. After indicating briefly a range of cases where the quadrature error can (in principle) be made zero, we turn to the general situation where quadrature effects need to be accounted for. We address briefly the first scenario of L_2-approximation and relate the results to previous work in [17]. The second scenario involving dual norms deserves more care. We shall first explain the principal difficulties arising in connection with an error analysis which is compatible with **Task R**. The subsequent discussion in Section 5 is then to identify suitable quadrature strategies satisfying **Task R**. The main objective here is to understand the essential ingredients which would have to be quantified and made more precise in any concrete application.

These asymptotic considerations are complemented by some numerical one- and two-dimensional experiments in Section 6 that are to shed more light on the quantitative performance of the scheme.

For realization details concerning some core steps of RECOVER in the case of cardinal B-spline

wavelets, we refer to an extended version of this paper [2].

2 Some Wavelet Prerequisites

Meanwhile several constructions of wavelet bases for nontrivial domain geometries are available, see e.g. [4, 5, 15, 16, 11]. We only recall briefly the main features that are relevant in the present context and refer to the original literature for further details.

2.1 Norm Equivalences

We shall make heavy use of the fact that rescaling a single pair of *anchor bases* for $L_2(\Omega)$ provide Riesz bases for a whole range of smoothness spaces. Thus, corresponding constructions aim at providing first a *dual pair* of biorthogonal wavelet bases

$$\Theta = \{\theta_\lambda : \lambda \in \mathcal{J}\}, \quad \tilde{\Theta} = \{\tilde{\theta}_\lambda : \lambda \in \mathcal{J}\},$$

which form Riesz bases in $L_2(\Omega)$. Recall that this means that there exist constants c_Θ, C_Θ such that for every $v \in L_2(\Omega)$

$$\begin{aligned} c_\Theta \|(\langle \tilde{\theta}_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2} &\leq \|v\|_{L_2(\Omega)} \leq C_\Theta \|(\langle \tilde{\theta}_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2}, \\ C_\Theta^{-1} \|(\langle \theta_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2} &\leq \|v\|_{L_2(\Omega)} \leq c_\Theta^{-1} \|(\langle \tilde{\theta}_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2}. \end{aligned} \tag{2.1}$$

Here $\mathcal{J}$ is the set of indices λ which usually encode the scale $j = j(\lambda) = |\lambda|$, the location $k = k(\lambda)$ and type of the wavelet. Clearly, one has $\Theta = \tilde{\Theta}$ in the case of orthonormal bases. Such wavelet bases are usually constructed with the aid of a *multiresolution analysis* which means (in the case of biorthogonal wavelets) two sequences $\mathcal{S} := \{S_j\}_{j\in\mathbb{N}_0}$ and $\tilde{\mathcal{S}} = \{\tilde{S}_j\}_{j\in\mathbb{N}_0}$ of nested spaces $S_j \subset S_{j+1}$, $\tilde{S}_j \subset S_{j+1}$. These spaces are usually defined as the span of corresponding *generator bases* $\Phi_j = \{\phi_\lambda : \lambda \in \mathcal{I}_j\}$, i.e., denoting for any finite collection Θ of functions by $S(\Theta)$ their linear span, $S_j = S(\Phi_j)$ and likewise $\tilde{S}_j = S(\tilde{\Phi}_j)$.

Nestedness of the spaces can then be expressed by a two-scale or *refinement relation* which says that any coarse scale basis function can be written as a linear combination of the fine scale basis function. Viewing the basis Φ_j as a *vector*, this can be written as

$$\Phi_j^T = \Phi_{j+1}^T \mathbf{M}_{j,\Phi}, \quad \tilde{\Phi}_j^T = \tilde{\Phi}_{j+1}^T \mathbf{M}_{j,\tilde{\Phi}}, \tag{2.2}$$

where the λ-th column of $\mathbf{M}_{j,\Phi}$ consists of the *mask* of ϕ_λ. Biorthogonality of the generators, i.e. $\langle \phi_\lambda, \tilde{\phi}_\mu\rangle = \delta_{\lambda,\mu}$, $\lambda, \mu \in \mathcal{I}_j$, implies

$$\mathbf{M}_{j,\Phi}{}^T \mathbf{M}_{j,\tilde{\Phi}} = \mathbf{I}. \tag{2.3}$$

One then looks for complement spaces $W_j, \tilde{W}_j$ such that $S_{j+1} = S_j \oplus W_j$ and $\tilde{S}_{j+1} = \tilde{S}_j \oplus \tilde{W}_j$ with

$$\tilde{W}_j \perp S_j, \quad W_j \perp \tilde{S}_j. \tag{2.4}$$

Wavelets are now obtained as bases for these complements $W_j = S(\Theta_j)$ and $\tilde{W}_j = S(\tilde{\Theta}_j)$, where $\Theta_j = \{\theta_\lambda : \lambda \in \mathcal{J}_j\}$, $\tilde{\Theta}_j = \{\tilde{\theta}_\lambda : \lambda \in \mathcal{J}_j\}$ respectively. Again since $\Theta_j \subset S_{j+1}$ and $\tilde{\Theta}_j \subset \tilde{S}_{j+1}$ there must exist $(\#\mathcal{I}_{j+1}) \times (\#\mathcal{J}_j)$-matrices $\mathbf{M}_{j,\Theta}$, and $\mathbf{M}_{j,\tilde{\Theta}}$ such that

$$\Theta_j^T = \Phi_{j+1}^T \mathbf{M}_{j,\Theta}, \quad \tilde{\Theta}_j^T = \tilde{\Phi}_{j+1}^T \mathbf{M}_{j,\tilde{\Theta}}. \tag{2.5}$$

Again biorthogonality (2.4) gives

$$\mathbf{M}_{j,\Theta}{}^T \mathbf{M}_{j,\tilde{\Theta}} = \mathbf{I}. \tag{2.6}$$

Now we can set

$$\Theta = \Phi_0 \bigcup_{j\in\mathbb{N}_0} \Theta_j, \quad \tilde{\Theta} = \tilde{\Phi}_0 \bigcup_{j\in\mathbb{N}_0} \tilde{\Theta}_j,$$

to obtain a pair of biorthogonal multiscale bases.

To describe the decomposition and synthesis operations, we need the inverses of the complemented matrices $\mathbf{M}_j := (\mathbf{M}_{j,\Phi}, \mathbf{M}_{j,\Theta})$, $\tilde{\mathbf{M}}_j := (\mathbf{M}_{j,\tilde{\Phi}}, \mathbf{M}_{j,\tilde{\Theta}})$, respectively,

$$\mathbf{G}_j := \begin{pmatrix} \mathbf{G}_{j,\Phi} \\ \mathbf{G}_{j,\Theta} \end{pmatrix} := {\mathbf{M}_j}^{-1}, \quad \tilde{\mathbf{G}}_j := \begin{pmatrix} \mathbf{G}_{j,\tilde{\Phi}} \\ \mathbf{G}_{j,\tilde{\Theta}} \end{pmatrix} := {\tilde{\mathbf{M}}_j}^{-1}. \tag{2.7}$$

Note that, by (2.3), (2.6),

$$\mathbf{G}_{j,\Phi} := {\mathbf{M}_{j,\tilde{\Phi}}}^T, \quad \mathbf{G}_{j,\Theta} := {\mathbf{M}_{j,\tilde{\Theta}}}^T, \quad \mathbf{G}_{j,\tilde{\Phi}} := {\mathbf{M}_{j,\Phi}}^T, \quad \mathbf{G}_{j,\tilde{\Theta}} := {\mathbf{M}_{j,\Theta}}^T. \tag{2.8}$$

In particular, (2.7) implies

$$\tilde{\Phi}_{j+1}^T = \tilde{\Phi}_j^T \mathbf{G}_{j,\tilde{\Phi}} + \tilde{\Theta}_j^T \mathbf{G}_{j,\tilde{\Theta}}, \tag{2.9}$$

expressing fine scale generator basis function as linear combinations of coarse scale basis functions and wavelets. In the above mentioned constructions all two scale matrices are uniformly sparse in the sense that all rows and columns of the $\mathbf{M}_j, \mathbf{G}_j, \tilde{\mathbf{M}}_j, \tilde{\mathbf{G}}_j$ have a finite uniformly bounded number of non-vanishing entries, see [2]for details in the special case of cardinal B-spline wavelets.

Given such a dual pair of L_2-anchor bases one can generate a whole scale of bases for other function spaces reflecting for instance different scales of smoothness. The most prominent example are Sobolev spaces H^s for some range $s \in (-\tilde{\gamma}, \gamma)$ (where for $s < 0$ we define $H^s = (H^{-s})'$ by duality) but other variants of energy spaces such as $\|v\|_{\mathcal{H}}^2 = \nu\|\nabla v\|_{L_2(\Omega)}^2 + \|v\|_{L_2(\Omega)}^2$ are covered as well [13]. Here H^s stands for $H^s(\Omega)$ or for a closed subspace of $H^s(\Omega)$ defined e.g. by homogeneous boundary conditions. In this case a suitable diagonal scaling of an L_2 basis yields one for a Sobolev space. Therefore we shall assume in the sequel that the Riesz basis Ψ for the relevant Hilbert space $\mathcal{H}$ is obtained through scaling an L_2-basis, i.e.

$$\psi_\lambda := \omega_\lambda \theta_\lambda, \qquad \tilde{\psi}_\lambda := \omega_\lambda^{-1} \tilde{\theta}_\lambda, \quad \lambda \in \mathcal{J}, \tag{2.10}$$

and there exist constants c_Ψ, C_Ψ such that

$$\begin{array}{rcccl} c_\Psi \|(\langle \tilde{\psi}_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2} & \leq & \|v\|_{\mathcal{H}} & \leq & C_\Psi \|(\langle \tilde{\psi}_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2}, \quad v \in \mathcal{H} \\ C_\Psi^{-1} \|(\langle \psi_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2} & \leq & \|v\|_{\mathcal{H}'} & \leq & c_\Psi^{-1} \|(\langle \tilde{\psi}_\lambda, v\rangle)_{\lambda\in\mathcal{J}}\|_{\ell_2}, \quad v \in \mathcal{H}'. \end{array} \tag{2.11}$$

Note that the scaling can be easily incorporated directly in the two scale relations. Let $\mathbf{D}_j := \mathrm{diag}\,(\omega_\lambda : \lambda \in \mathcal{J}_j)$. We immediately infer from (2.5) that

$$\Psi_j^T = \Phi_{j+1}^T \mathbf{M}_{j,\Psi}, \quad \mathbf{M}_{j,\Psi} := \mathbf{M}_{j,\Theta} \mathbf{D}_j. \tag{2.12}$$

Since by (2.10) $\tilde{\Psi}_j^T = \tilde{\Theta}_j^T \mathbf{D}_j^{-1}$ we infer from (2.9)

$$\tilde{\Phi}_{j+1}^T = \tilde{\Phi}_j^T \mathbf{G}_{j,\tilde{\Phi}} + \tilde{\Psi}_j^T \mathbf{G}_{j,\tilde{\Psi}}, \quad \text{where} \quad \mathbf{G}_{j,\tilde{\Psi}} := \mathbf{D}_j \mathbf{G}_{j,\tilde{\Theta}}. \tag{2.13}$$

2.2 Tree-Structures

What is important in our context is the fact that one can in all presently known cases associate with $\mathcal{J}$ a *tree structure*. In general one can (not necessarily in a unique way) assign a collection of a uniformly bounded number of wavelets on level $|\lambda| = j+1$ to a parent wavelet on the preceeding level $|\lambda| = j$. This can be made more specific for the above quoted constructions based on patching together parametric liftings of tensor product bases on the unit cube. For each parametric patch the indices have a structure similar to that for wavelets on the full Euclidean space, i.e. $\lambda = (|\lambda|, \mathbf{k}, \mathbf{e})$ where $\mathbf{k} = \mathbf{k}(\lambda)$ is a multi-index encoding the spatial location and $\mathbf{e} = \mathbf{e}(\lambda) \in \{0,1\}^d \setminus \mathbf{0}$ refers to the *type* of the wavelet. The index sets $\mathcal{J}_j$ identify the true wavelets ψ_λ, $|\lambda| = j, \mathbf{e}(\lambda) \neq \mathbf{0}$, on level j. The type $\mathbf{e} = \mathbf{0}$ is always reserved to the scaling functions and we will therefore identify $(j, \mathbf{k}, \mathbf{0})$ with $\lambda^\circ = (j, \mathbf{k})$, $|\lambda^\circ| = j, \mathbf{k}(\lambda^\circ) = \mathbf{k}$. The set of scaling function indices $(j, \mathbf{k})$ on level j is denoted by $\mathcal{I}_j$ while $\mathcal{I}$ denotes the union of these sets over all levels.

For all the constructions mentioned earlier one can associate with $\lambda^\circ = (j, \mathbf{k})$ a *support cell* $\square_{\lambda^\circ}$ with the property $\square_{\lambda^\circ} \subseteq \operatorname{supp} \psi_\lambda, \quad |\lambda| = j, \quad \mathbf{k}(\lambda) = \mathbf{k}$. Moreover the $\mathcal{P}_j := \{\square_{\lambda^\circ} : \lambda^\circ \in \mathcal{I}_j\}$ shall form a hierarchy of nested partitions of Ω, i.e. $\bar{\Omega} = \bigcup_{\lambda^\circ \in \mathcal{I}_j} \square_{\lambda^\circ}$ and each $\square_{\lambda^\circ}$ is an essentially disjoint union of support cells of level $|\lambda| + 1$. In the simplest case of translation invariant scaling functions the support cells have the form $\square_{j,\mathbf{k}} := 2^{-j}(\mathbf{k} + [0,1]^d)$.

In what follows, we will often identify $\lambda \in \mathcal{J}$ with the support cell $\square_{\lambda^\circ}$ and by Λ° we will denote the index-set of support cells associated with the set of indices in $\Lambda \subset \mathcal{J}$. We will make use of the fact, that $\Lambda^\circ \subset \mathcal{I}$. Note, that except for the case of Haar-wavelets, $\square_{\lambda^\circ} \subsetneq \operatorname{supp} \psi_\lambda$ and that the difference grows with the order of the wavelets.

The hierarchy of nested partitions $\mathcal{P}_j$ induces a natural tree structure with respect to set inclusion. Whenever $\square_{\mu^\circ} \subset \square_{\lambda^\circ}$ we say that μ° is a descendant of λ° which we express by $\mu^\circ \succ \lambda^\circ$. When equality is permitted we write $\mu^\circ \succeq \lambda^\circ$. Conversely λ° is called an *ancestor* of μ°, which will be denoted by $\lambda^\circ \prec \mu^\circ$ (or $\lambda^\circ \preceq \mu^\circ$). When $|\lambda^\circ| = |\mu^\circ| - 1$, μ° is called a *child* of λ° and λ° is referred to as the *parent* of μ°.

A subset $\Lambda \subset \mathcal{I}$ is called a *tree* if for each $\lambda^\circ \in \Lambda$ with $|\lambda^\circ| > j_0$ also its parent belongs to Λ. It will greatly simplify data structures when a tree is *complete* in the sense that whenever $\lambda^\circ \in \Lambda$ implies that *all* the siblings of λ° also belong to Λ. We shall work in what follows exclusively with complete trees.

The tree structure of $\mathcal{I}$ induces a *tree-like structure* in $\mathcal{J}$ by saying that $\mathcal{T} \subset \mathcal{J}$ has tree structure if $\mathcal{T}^\circ$ is a tree in the above sense. Note that therefore $\lambda \in \mathcal{T}$ implies that *all* $\mu \in \mathcal{T}$ with $|\mu| = |\lambda|$ and $\mathbf{k}(\mu) = \mathbf{k}(\lambda)$. We shall again refer to this property as *completeness* and call also the set $\mathcal{T}$ (in slight abuse of terminology) a *(complete) tree*.

Recall that the set of *leaves* of a tree $\mathcal{T}^\circ \subset \mathcal{I}$ consists of those $\lambda^\circ \in \mathcal{T}^\circ$ which belong to $\mathcal{T}^\circ$ but none of their children are in $\mathcal{T}^\circ$. We shall have to deal also with the set $\partial\mathcal{T}^-$ of *outer leaves* of $\mathcal{T}^\circ$ which consists of those $\lambda^\circ \notin \mathcal{T}^\circ$ whose parent is in $\mathcal{T}^\circ$. It is easy to see that the set of outer leaves $\partial\mathcal{T}^-$ forms a partition of Ω consisting of support cubes. Hence the span of wavelets from a complete tree $\mathcal{T} \subset \mathcal{J}$ are associated in a natural way with a locally refined mesh, namely $\partial\mathcal{T}^-$. Thus the adaptation potential offered by spans of wavelets whose indices form a complete tree is comparable to trial spaces on locally refined meshes, cf. [8, 9, 10].

3 The Recovery Scheme

We address now **Task R1** from Section 1.1. Recall that $\mathcal{T}$ denotes always a complete finite tree. We wish to compute a sequence $\mathbf{w} = (w_\lambda)_{\lambda\in\mathcal{T}}$ with $\operatorname{supp}\mathbf{w} \subseteq \mathcal{T}$ that approximates the array

$$\mathbf{g}(\mathcal{T}) := (\langle\psi_\lambda, g\rangle)_{\lambda\in\mathcal{T}}.$$

Recall that the entries $\langle\psi_\lambda, g\rangle$ are the wavelet coefficients of $g = \sum_{\lambda\in\mathcal{J}}\langle\psi_\lambda, g\rangle\tilde{\psi}_\lambda$ with respect to the *dual basis* $\tilde{\Psi}$ whose projection to the span of $\tilde{\Psi}_\mathcal{T} := \{\tilde{\psi}_\lambda : \lambda \in \mathcal{T}\}$ is denoted by $g_\mathcal{T} = \sum_{\lambda\in\mathcal{T}}\langle\psi_\lambda, g\rangle\tilde{\psi}_\lambda$. Since $\langle\psi_\lambda, g\rangle\tilde{\psi}_\lambda = \langle\theta_\lambda, g\rangle\tilde{\theta}_\lambda$ it suffices to compute the array

$$\mathbf{d}(\mathcal{T}) = (\langle\theta_\lambda, g\rangle)_{\lambda\in\mathcal{T}} = (\langle\theta_\lambda, g_\mathcal{T}\rangle)_{\lambda\in\mathcal{J}},$$

to obtain

$$\mathbf{g}(\mathcal{T}) = \mathbf{D}_\mathcal{T}\mathbf{d}(\mathcal{T}), \quad \mathbf{D}_\mathcal{T} := \operatorname{diag}(\omega_\lambda : \lambda \in \mathcal{T}). \tag{3.1}$$

3.1 Motivation and Main Idea

The scheme we are going to develop next is based on the coefficients $\langle\psi_\lambda, g\rangle$, $\lambda \notin \mathcal{T}$, being presumably small in a sense that will be made more precise later. Specifically, we shall at this point only make use of the fact (to be established later) that quadrature errors are small on $\operatorname{supp}\tilde{\phi}_{j,\mathbf{k}}$ whenever $\square_{j,\mathbf{k}}$ is a support cell whose index belongs to $\partial\mathcal{T}^-$, the set of outer leaves of $\mathcal{T}^\circ$. Recall that the support cells associated with $\partial\mathcal{T}^\circ$, form a partition of Ω. On account of these accuracy considerations we shall speak of *safe approximations/quadrature* whenever it applies to the quantities $\langle\phi_{j,\mathbf{k}}, g\rangle$ for $(j,\mathbf{k}) \notin \mathcal{T}^\circ$.

We shall explain next how to use this in order to compute correspondingly accurate approximations of all $\langle\Theta_\lambda, g\rangle$ (resp. $\langle\Psi_\lambda, g\rangle$), $\lambda \in \mathcal{T}$. We deliberately postpone a detailed discussion of accuracy issues and focus first on the computational ingredients.

We shall abbreviate in the following $c_{j,\mathbf{k}} := \langle\phi_{j,\mathbf{k}}, g\rangle$ for $(j,\mathbf{k}) \in \mathcal{I}_j$, and $d_\lambda := \langle\Theta_\lambda, g\rangle, \lambda \in \mathcal{J}$, and write for any subset $\mathcal{G}$ of $\mathcal{I}_j$ or of $\mathcal{J}$ briefly

$$\mathbf{c}_j(\mathcal{G}) := (c_{j,\mathbf{k}} : (j,\mathbf{k}) \in \mathcal{G}), \quad \mathbf{d}(\mathcal{G}) := (d_\lambda : \lambda \in \mathcal{G}),$$

where we simply set $\mathbf{c}_j = \mathbf{c}_j(\mathcal{I}_j)$, $\mathbf{d}_j := \mathbf{d}(\mathcal{J}_j)$. In terms of these coefficients (2.5) takes, upon using the two-scale relations (2.2), (2.5), the form

$$\tilde{\Theta}_j^T\mathbf{d}_j = \tilde{\Phi}_{j+1}^T\mathbf{M}_{j,\tilde{\Theta}}\mathbf{d}_j = \tilde{\Phi}_{j+1}^T(\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}}\mathbf{c}_j), \tag{3.2}$$

whence we conclude that

$$\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}}\mathbf{c}_j = \mathbf{M}_{j,\tilde{\Theta}}\mathbf{d}_j. \tag{3.3}$$

We shall employ the canonical projectors

$$P_j w := \sum_{\mathbf{k}\in\mathcal{I}_j}\langle\phi_{j,\mathbf{k}}, w\rangle\tilde{\phi}_{j,\mathbf{k}} = \sum_{\mathbf{k}\in\mathcal{I}_j} c_{j,\mathbf{k}}\tilde{\phi}_{j,\mathbf{k}} \tag{3.4}$$

and exploit the fact that $(P_{j+1} - P_j)w = \sum_{\lambda\in\mathcal{J}_j}\langle\Theta_\lambda, w\rangle\tilde{\Theta}_\lambda$.

To point out the significance of (3.3), it will be convenient to introduce some further notation. Let $\mathcal{T}_j := \mathcal{T} \cap \mathcal{J}_j$ denote the set of elements in $\mathcal{T}$ of level j and let $J = \max\{j \geq j_0 : \mathcal{T}_j \neq \emptyset\}$ the highest level appearing in $\mathcal{T}$. If $\mathbf{d}_j$ is supported on $\mathcal{T}_j$, (3.3) says that the array $\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}}\mathbf{c}_j$ vanishes outside the union of the columns of $\mathbf{M}_{j,\tilde{\Theta}}$ selected by $\mathcal{T}_j$.

Remark 3.1 *Denoting for $\lambda \in \mathcal{I} \cup \mathcal{J}_j$ by $\tilde{\mathbf{M}}_j{}^{|\lambda}$, $\overline{\tilde{\mathbf{G}}_j{}^{\lambda}}$ the λ-th column of $\tilde{\mathbf{M}}_j$, respectively the λ-th row of $\tilde{\mathbf{G}}_j$, and defining $\mathcal{G}_{j+1} \subset \mathcal{I}_{j+1}$ by*

$$\mathcal{G}_{j+1} := \bigcup_{\lambda \in \mathcal{T}_j^\circ} \operatorname{supp} \tilde{\mathbf{M}}_j{}^{|\lambda} \cup \bigcup_{\lambda \in \mathcal{T}_j^\circ} \operatorname{supp} \overline{\tilde{\mathbf{G}}_j{}^{\lambda}}, \tag{3.5}$$

one concludes from (3.3) that, whenever $\operatorname{supp} \mathbf{d}_j \subseteq \mathcal{T}_j$*, one has*

$$(\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j)(\mathcal{I}_{j+1} \setminus \mathcal{G}_{j+1}) = \mathbf{0}. \tag{3.6}$$

Relation (3.6) means that if we approximate the values of $\mathbf{c}_{j+1}$ and $\mathbf{c}_j$ only for a subset of indices, such truncated arrays combined via (3.3) cause no error outside a certain finite set of indices. The following considerations aim at exploiting this fact systematically. To this end, one derives from (2.9) that

$$\tilde{\Phi}_{j+1}^T \mathbf{c}_{j+1} = \tilde{\Phi}_j^T \mathbf{G}_{j,\tilde{\Phi}} \mathbf{c}_{j+1} + \tilde{\Theta}_j^T \mathbf{G}_{j,\tilde{\Theta}} \mathbf{c}_{j+1} = \tilde{\Phi}_j^T \mathbf{c}_j + \tilde{\Theta}_j^T \mathbf{d}_j, \tag{3.7}$$

so that

$$\mathbf{c}_j = \mathbf{G}_{j,\tilde{\Phi}} \mathbf{c}_{j+1}, \quad \mathbf{d}_j = \mathbf{G}_{j,\tilde{\Theta}} \mathbf{c}_{j+1}. \tag{3.8}$$

We wish to compute now as few scaling function coefficients as possible in order to generate in a reliable way the arrays $\mathbf{d}(\mathcal{T}_j)$ of non-vanishing wavelet coefficients. This requires taking the truncation effects in the two-scale relations into careful account. First, note that, by the second relation in (3.8), we only need to know $\mathbf{c}_{j+1}(\mathcal{G}_{j+1})$ to determine $\mathbf{d}_j(\mathcal{T}_j)$ because only the rows $\overline{\tilde{\mathbf{G}}_j{}^{\lambda}}$ with $\lambda \in \mathcal{T}_j$ are required and they are all supported in $\mathcal{G}_{j+1}$. Likewise, the first relation in (3.8) says by the same reasoning that $\mathbf{c}_j(\mathcal{T}_j^\circ)$ is accurately determined from $\mathbf{c}_{j+1}(\mathcal{G}_{j+1})$. So we can think of $\mathbf{c}_j(\mathcal{T}_j^\circ)$ being obtained from accurate data on level $j+1$ through (3.8). The remaining coefficients $\mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ)$ need to be computed independently. So, splitting $\mathbf{c}_j$ into those two parts in (3.7), provides, upon making use of (2.2),

$$\tilde{\Phi}_{j+1}^T \Big(\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ) \Big) = \tilde{\Phi}_j^T (\mathbf{G}_{j,\tilde{\Phi}} \mathbf{c}_{j+1})(\mathcal{T}_j^\circ) + \tilde{\Theta}_j^T \mathbf{d}_j(\mathcal{T}_j). \tag{3.9}$$

As immediate consequence of (3.6), this tells us, that Relation (3.9) remains valid when the full array $\mathbf{c}_{j+1}$ and the array $\mathbf{M}_{j,\tilde{\phi}} \mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ)$ are restricted to $\mathcal{G}_{j+1}$, i.e.

$$\tilde{\Phi}_{j+1}^T \Big(\mathbf{c}_{j+1}(\mathcal{G}_{j+1}) - (\mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ))(\mathcal{G}_{j+1}) \Big) = \tilde{\Phi}_j^T (\mathbf{G}_{j,\tilde{\Phi}} \mathbf{c}_{j+1})(\mathcal{T}_j^\circ) + \tilde{\Theta}^T \mathbf{d}_j(\mathcal{T}_j). \tag{3.10}$$

Applying now $\mathbf{G}_{j,\tilde{\Phi}}$ to the array $\mathbf{c}_{j+1}(\mathcal{G}_{j+1}) - (\mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ))(\mathcal{G}_{j+1})$ yields by (3.8) the safe coefficients $(\mathbf{G}_{j,\tilde{\Phi}} \mathbf{c}_{j+1})(\mathcal{T}_j^\circ)$ and applying $\mathbf{G}_{j,\tilde{\Theta}}$ yields $\mathbf{d}_j(\mathcal{T}_j)$, provided the partial array $\mathbf{c}_{j+1}(\mathcal{G}_{j+1})$ is safe. Moreover, we do not need all of $\mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ)$ but only $\mathbf{c}_j(\mathcal{G}_j \setminus \mathcal{T}_j^\circ)$ because, by the above reasoning these are the only relevant coefficients for the next lower level. The main point conveyed by these observations is that, computing only the entries $\mathbf{c}_{j+1}(\mathcal{G}_{j+1})$ and also a suitable truncation of $\mathbf{c}_j$ on the next lower level, these truncated arrays suffice to represent $\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j$ exactly on all of $\mathcal{I}_{j+1}$ whenever $\mathbf{d}_j$ is supported in $\mathcal{T}_j$. This motivates us to express expansions in terms of quantities like $\mathbf{c}_{j+1} - \mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j$. In fact, we can write in view of (3.9),

$$P_{\mathcal{T}} g := P_{j_0} g + \sum_{j=j_0}^{J} (P_{j+1} - P_j) g = \sum_{j=j_0+1}^{J+1} \tilde{\Phi}_j^T \Big(\mathbf{c}_j(\mathcal{I}_j \setminus \mathcal{T}_j^\circ) - \mathbf{M}_{j-1,\tilde{\Phi}} \mathbf{c}_{j-1}(\mathcal{I}_{j-1} \setminus \mathcal{T}_{j-1}^\circ) \Big). \tag{3.11}$$

Hence, in order to derive all necessary information within suitable accuracy tolerances, we only have to take into account $\mathcal{G}_j^- := \mathcal{G}_j \setminus \mathcal{T}_j^\circ,\ j \le J$, and $\mathcal{G}_{J+1}^- = \mathcal{G}_{J+1}$.
Main idea: The idea now is to compute the arrays $\mathbf{c}_j(\mathcal{G}_j^-)$ by *quadrature*. Since quadrature is therefore applied only at places where local truncation errors are small one can hope that a fixed quadrature order, independent of the scale, suffices. This gives rise to the scheme described in the next section.

Note that the above reasoning remains unchanged when the sets $\mathcal{G}_j$ are replaced by somewhat larger sets. We shall point out in [2] that such supersets are efficiently and easily obtained for an important class of wavelet constructions. So employing for the purpose of analysis the exact format of the sets $\mathcal{G}$ from (3.5) does not interfere with practical realizations using slightly larger sets.

3.2 A Top–To–Bottom Scheme

We shall compute an approximation to $\mathbf{d}(\mathcal{T})$ working from top to bottom, guided by the considerations of the previous section. One main point will be that the vanishing of the coefficients d_λ for $\lambda \notin \mathcal{T}$ implies that the scaling function coefficients $c_{j,\mathbf{k}} := \langle \phi_{j,\mathbf{k}}, g_\mathcal{T} \rangle$, $(j,\mathbf{k}) \notin \mathcal{T}^\circ$ can be computed by quadrature with high accuracy. At this point it does not matter yet which quadrature rule is used. Since its choice will be intertwined with a subsequent error analysis we postpone its specification.

As mentioned before, quadrature will only be used to approximate the arrays $\mathbf{c}_j(\mathcal{G}_j^-)$ and these approximations will be denoted by $\mathbf{q}_j = \mathbf{q}_j(\mathcal{G}_j^-)$. The key is then relation (3.10) which is used to successively decompose the representation (3.11) from top to bottom. The scaling function coefficients produced on the next lower level are then those obtained by quadrature on the sets $\mathcal{G}_j^-$ complemented by those living on $\mathcal{T}_j^\circ$ obtained from transforming safe coefficients from higher levels. This leads to the scheme RECOVER that determines for a given tree $\mathcal{T}$ an approximation $\mathbf{d}^{\mathrm{R}}(\mathcal{T})$ supported on $\mathcal{T}$ to the wavelet coefficients of $g_\mathcal{T}$ with respect to the dual basis $\tilde{\Theta}$ as follows:

RECOVER — $[g, \mathcal{T}] \to \mathbf{d}^{\mathrm{R}}(\mathcal{T})$

Find minimal $J \in \mathbb{N}$ such that $\mathcal{T}_j = \emptyset$, $j > J$

Determine $\mathcal{G}_{J+1}^- = \mathcal{G}_{J+1}$, compute $\mathbf{q}_{J+1}(\mathcal{G}_{J+1}^-)$ and set $\check{\mathbf{c}}_{J+1} := \mathbf{0}$.

For $j = J, J-1, \dots, j_0$ do

- Determine $\mathcal{G}_j^- = \mathcal{G}_j \backslash \mathcal{T}_j$ and $\mathbf{q}_j(\mathcal{G}_j^-)$
- Set
$$\bar{\mathbf{c}}_{j+1} := \check{\mathbf{c}}_{j+1} + \mathbf{q}_{j+1}(\mathcal{G}_{j+1}^-) - \left(\mathbf{M}_{j,\tilde{\Phi}} \mathbf{q}_j(\mathcal{G}_j^-)\right)\left(\mathcal{G}_{j+1}^-\right) \tag{3.12}$$
- Compute
$$\check{\mathbf{c}}_j := \mathbf{G}_{j,\tilde{\Phi}} \bar{\mathbf{c}}_{j+1}, \quad \text{and} \quad \mathbf{d}_j^{\mathrm{R}} := \mathbf{G}_{j,\tilde{\Theta}} \bar{\mathbf{c}}_{j+1} \tag{3.13}$$

Set $\mathbf{d}_{j_0-1}^{\mathrm{R}} := \mathbf{c}_{j_0}^{\mathrm{R}} := \check{\mathbf{c}}_{j_0}$

From now on we define for the sake of convenience $\mathbf{d}_{j_0-1} := \mathbf{c}_{j_0}$ and $\mathbf{d}_{j_0-1}^{\mathrm{R}} := \mathbf{c}_{j_0}^{\mathrm{R}}$.

Remark 3.2 *Note that the support of* $\mathbf{d}_j^R$ *is always contained in* $\mathcal{T}_j$. *The scheme requires computing the quantities* $q_{j,\mathbf{k}}$ *as approximations to* $c_{j,\mathbf{k}} = \langle \phi_{j,\mathbf{k}}, g\rangle$ *only on the set*

$$\mathcal{G}^- := \mathcal{G}^-(\mathcal{T}) = \bigcup_{j=j_0}^{J+1} \mathcal{G}_j^-. \tag{3.14}$$

Moreover, one has for some constants C, C'

$$\#\mathcal{G}^- \leq C\#\partial\mathcal{T} \leq C'\#\mathcal{T}. \tag{3.15}$$

Hence, whenever the quadrature requires at most a constant cost per entry, the number of flops needed in RECOVER *remains uniformly proportional to* $\#\mathcal{T}$.

Thus from the computational complexity point of view the above scheme satisfies all requirements put forward in the adaptive solution context described earlier in Section 1.2.

Recall that in the context of **Task R2** the objective of RECOVER is to produce an array $\mathbf{w}$, that satisfies $\|\mathbf{g}(\mathcal{T}) - \mathbf{w}\|_{\ell_2} \leq C\varepsilon$ whenever $\|\mathbf{g} - \mathbf{g}(\mathcal{T})\|_{\ell_2} < \varepsilon$. However, when $\mathbf{g} = \mathbf{F}(\mathbf{v})$ is a composition, the prediction set $\mathcal{T}$ depends on a-posteriori information about $\mathbf{v}$ and priori knowledge of $\mathbf{F}$. The latter typically refers to a whole class of nonlinearities and may lead in concrete cases to rather pessimistic predictions rendering $\mathcal{T}$ unnecessarily large. The top-to-bottom structure of RECOVER easily accommodates individual adaptive adjustments on the fly: As soon as the array $\mathbf{d}^{\mathrm{R}}(\mathcal{T}_j)$ is computed in (3.13) we can replace it by an array $\mathbf{d}_j^{\mathrm{R}}$ with possibly small support in $\mathcal{T}_j$ such that $\|\mathbf{d}^{\mathrm{R}}(\mathcal{T}_j) - \mathbf{d}_j^{\mathrm{R}}\|_{\ell_2(\mathcal{T}_j)} \leq \varepsilon_j$, where ε_j is such that $\sum_{j=j_0}^{J} \varepsilon_j = \varepsilon$. This is a typical coarsening step as described in [8] based on quasi-sorting and thresholding coefficients with smallest modulus. In this way the complete tree $\mathcal{T}$ may never have to be completely assembled which is a crucial distinction from earlier versions in [17]. We refer to [2] for more details relevant for practical relizations of RECOVER. In particular the sets $\mathcal{G}_j$ can be replaced by somewhat larger, symmetric supersets that can be constructed very easily.

4 Error Estimates

The scheme RECOVER introduced above produces for a tree $\mathcal{T}$ an array $\mathbf{d}^{\mathrm{R}}(\mathcal{T})$ supported on $\mathcal{T}$ to approximate the exact sequence $\mathbf{d}$ of wavelet coefficients of g with respect to $\tilde{\Theta}$. Likewise the scaled array $\mathbf{g}^{\mathrm{R}}(\mathcal{T}) = \mathbf{D}_{\mathcal{T}}\mathbf{d}^{\mathrm{R}}(\mathcal{T})$ should approximate the sequence $\mathbf{g}$ of wavelet coefficients of g with respect to $\tilde{\Psi}$.

Recall from Section 1.1 that in either case the total error of such an approximation has two sources. First, there is the *truncation error* $\|\mathbf{d} - \mathbf{d}(\mathcal{T})\|_{\ell_2}$, respectively $\|\mathbf{g} - \mathbf{g}(\mathcal{T})\|_{\ell_2}$, which depends on the prediction set $\mathcal{T}$. Second, there is the error incurred by quadrature in the scheme RECOVER when recovering the truncated array $\mathbf{d}(\mathcal{T})$.

4.1 Exact Quadrature

In a number of relevant cases the goal in **Task R2**, namely that the quadrature error is dominated by (a bound for) the truncation error, can be achieved trivially because, in principle, the quadrature can be made exact. To explain this, recall the example in (1.5) with $F(v) = -\mathrm{div}\, a\nabla v + G(v)$ and suppose first that $G \equiv 0$, i.e. the problem is linear. In this case one has to approximate the entries $c_{j,\mathbf{k}} = \langle \phi_{j,\mathbf{k}}, g\rangle$ with $g = F(v)$. In the context of adaptive schemes v is a finite wavelet expansion $v = \sum_{\lambda\in\mathcal{T}_v} v_\lambda \psi_\lambda$, with $\mathcal{T}_v$ the tree of significant coefficients of v. Just as done in (3.11)

we can rewrite v in local scaling function representation $v = \sum_{(j,\mathbf{k})\in\mathcal{G}_v^-} \bar{c}_{j,\mathbf{k}}\phi_{j,\mathbf{k}}$. Assuming that $\mathcal{T}_v$ is well-graded, which means that the transition between scales is sufficiently gradual (see (4.8) below for a precise definition), only a uniformly finite number of scaling functions in this expansion overlap a given point. Thus

$$\begin{aligned} c_{j,\mathbf{k}} &= \langle \phi_{j,\mathbf{k}}, F(v)\rangle = \sum_{(j,\mathbf{k}')\in\mathcal{G}_v^-} \bar{c}_{j,\mathbf{k}'}\langle \phi_{j,\mathbf{k}}, -\mathrm{div}(a\nabla\phi_{j,\mathbf{k}'})\rangle \\ &= \sum_{\substack{(j,\mathbf{k}')\in\mathcal{G}_v^-,\\ \mathrm{supp}\,\phi_{j,\mathbf{k}'}\cap\mathrm{supp}\,\phi_{j,\mathbf{k}}\neq\emptyset}} \bar{c}_{j,\mathbf{k}'}\langle \nabla\phi_{j,\mathbf{k}}, a\nabla\phi_{j,\mathbf{k}'}\rangle. \end{aligned} \tag{4.1}$$

By the previous remarks, the sets $\{(j,\mathbf{k}') : \mathrm{supp}\,\phi_{j,\mathbf{k}'}\cap\mathrm{supp}\,\phi_{j,\mathbf{k}}\neq\emptyset\}$ have uniformly bounded cardinality. Thus, when using piecewise polynomial scaling functions and whenever the diffusion matrix a is piecewise polynomial, the quantities $\langle\nabla\phi_{j,\mathbf{k}}, a\nabla\phi_{j,\mathbf{k}'}\rangle$ can be computed exactly, so that in this case $q_{j,\mathbf{k}} = c_{j,\mathbf{k}}, (j,\mathbf{k})\in\mathcal{G}^-$, and no quadrature error occurs. By the above reasoning, the computational cost stays proportional to $\#\mathcal{G}^-$ so that Remark 3.2 applies.

This extends to simple nonlinearities. For instance, in the case (1.8) we have $g = G(v) = v^3$ so that

$$c_{j,\mathbf{k}} = \sum_{\substack{(j_i\mathbf{k}_i)\in\mathcal{G}_v^-, i=1,2,3,\\ \mathrm{supp}\,\phi_{(j_i,\mathbf{k}_i)}\cap\mathrm{supp}\,\phi_{j,k}\neq\emptyset}} \left\langle \phi_{j,\mathbf{k}}, \prod_{i=1}^{3}\phi_{j_i,\mathbf{k}_i}\right\rangle.$$

Again for each $(j,\mathbf{k})$ there is a uniformly bounded number of integrals involving still only piecewise polynomials that can be calculated exactly.

Thus in such cases one can, in principle, make the scheme RECOVER exact so that, in view of the relation $\|\mathbf{d}^{\mathrm{R}}(\mathcal{T}) - \mathbf{d}(\mathcal{T})\|_{\ell_2} \lesssim \|\mathbf{d} - \mathbf{d}(\mathcal{T})\|_{\ell_2}$, and likewise $\|\mathbf{D}_{\mathcal{T}}(\mathbf{d}^{\mathrm{R}}(\mathcal{T}) - \mathbf{d}(\mathcal{T}))\|_{\ell_2} \lesssim \|\mathbf{g} - \mathbf{g}(\mathcal{T})\|_{\ell_2}$, **Task R2** is trivially fulfilled.

In general, however, one may have to accept a quadrature error or prefers to accept a quadrature error, because even in the above cases the necessary exactness order of quadrature may be fixed but rather high. Therefore we need to estimate this error part and devise ways of decreasing this error if necessary. The subsequent sections are devoted to both issues. We present ways of estimating this error, out of necessity in an essentially different way from [17], and derive strategies for controlling it relative to the truncation or prediction error.

4.2 The L_2-Case

We shall address first the case $\mathcal{H} = L_2(\Omega)$ which corresponds to the first scenario indicated in Section 1.2. Let for $(j,\mathbf{k})\in\mathcal{G}^-$

$$q_{j,\mathbf{k}} - c_{j,\mathbf{k}} = e_{j,\mathbf{k}} \tag{4.2}$$

denote the deviation of the computed value $q_{j,\mathbf{k}}$ from the true one $c_{j,\mathbf{k}} = \langle\phi_{j,\mathbf{k}}, g\rangle$. The corresponding arrays $\mathbf{e}_j = (e_{j,\mathbf{k}})_{(j,\mathbf{k})\in\mathcal{G}_j^-}$ are always supported on $\mathcal{G}_j^-$ without further mentioning. Recall that $\mathcal{G}_{J+1}^- = \mathcal{G}_{J+1}$. Since (3.11) involves only coefficients that will be approximated by quadrature we can think of RECOVER to produce the approximation

$$P_{\mathcal{T}}^{\mathrm{R}} g = \sum_{j=j_0}^{J+1} \tilde{\Phi}_j^T\Big(\mathbf{q}_j - \mathbf{M}_{j-1,\tilde{\Phi}}\mathbf{q}_{j-1}(\mathcal{G}_{j-1}^-)\Big)(\mathcal{G}_j^-) \tag{4.3}$$

to $P_{\mathcal{T}}g$, which, by (4.2), immediately yields the error representation

$$E_{\mathcal{T}} := P^{\mathrm{R}}_{\mathcal{T}}g - P_{\mathcal{T}}g = \sum_{j=j_0}^{J+1} \tilde{\Phi}_j^T \Big(\mathbf{e}_j - \mathbf{M}_{j-1,\tilde{\Phi}}\mathbf{e}_{j-1}(\mathcal{G}_{j-1}^-)\Big)(\mathcal{G}_j^-). \tag{4.4}$$

Of course, the approximate coefficients in $\mathbf{d}^{\mathrm{R}}(\mathcal{T})$ produced by RECOVER arise by decomposing $P^{\mathrm{R}}_{\mathcal{T}}g$ which generates as byproducts the auxiliary arrays $\check{\mathbf{c}}_j, \bar{\mathbf{c}}_j$, cf. (3.12), (3.13). It will be therefore important to understand how the quadrature errors related to $\mathcal{G}^-$ effects those arrays and thereby $\mathbf{d}^{\mathrm{R}}(\mathcal{T})$. Inserting (2.13) into (4.4), proceeding from top to bottom, provides $E_{\mathcal{T}} = \tilde{\Phi}_{j_0}^T \mathbf{E}_{j_0-1} + \sum_{j=j_0}^{J} \tilde{\Theta}_j^T \Big(\mathbf{G}_{j,\tilde{\Theta}}\mathbf{E}_{j+1}\Big)(\mathcal{T}_j)$, where for $j = J, J-1, \ldots, j_0$

$$\begin{aligned} \mathbf{E}_{j+1} \;=\; & \Big[\mathbf{G}_{j+1,\tilde{\Phi}}(\ldots \mathbf{G}_{J-2,\tilde{\Phi}}([\mathbf{G}_{J-1,\tilde{\Phi}}([\mathbf{G}_{J,\tilde{\Phi}}\mathbf{R}_{J+1}](\mathcal{T}_J^\circ) + \mathbf{R}_J)](\mathcal{T}_{J-1}^\circ) \\ & + \mathbf{R}_{J-1})(\mathcal{T}_{J-2}^\circ) + \ldots + \mathbf{R}_{j+1})\Big](\mathcal{G}_{j+1}), \end{aligned} \tag{4.5}$$

and

$$\mathbf{R}_j := \mathbf{R}_j(\mathcal{G}_j^-) := \Big(\mathbf{e}_j - \mathbf{M}_{j-1,\tilde{\Phi}}\mathbf{e}_{j-1}(\mathcal{G}_{j-1}^-)\Big)(\mathcal{G}_j^-). \tag{4.6}$$

Thus the error on level $j+1$ prior to decomposing is obtained by successively decomposing error contributions from higher levels. Here and in what follows we always assume that the multiscale decompositions are applied exactly. Alternatively, one can describe $\mathbf{E}_j$ recursively as follows

$$\mathbf{E}_j = \Big[\mathbf{G}_{j,\tilde{\Phi}}(\mathbf{E}_{j+1})\Big](\mathcal{T}_j^\circ) + \mathbf{R}_j. \tag{4.7}$$

Thus the error on level j consists of a part living inside the tree obtained from decomposing higher level contributions plus a remainder situated on $\mathcal{G}_j^-$ that is newly introduced on that level through quadrature.

The overall ℓ_2-error in $\mathbf{d}^{\mathrm{R}}(\mathcal{T})$ caused by the quadrature in RECOVER can be easily estimated by exploiting the norm equivalence (2.1). To simplify matters technically we shall specify next somewhat our requirements on the tree $\mathcal{T}$. The theoretical foundation of good predictions for significant trees $\mathcal{T}$ developed in [9] required the trees to satisfy the so called *expansion property* which, roughly speaking means that the leaves of subsequent levels do not overlap too much, which is familiar in this context, see also [17]. The tree $\mathcal{T}$ is called *well-graded* if one has

$$\Big(\bigcup_{\lambda:\lambda^\circ\in\mathcal{G}_j^-} \mathbf{M}_{j,\tilde{\Phi}}{}^{|\lambda}\Big) \cap \mathcal{T}_{j+1}^\circ = \emptyset. \tag{4.8}$$

In other words, as soon as one leaves the tree on level j those scaling functions will not overlap the safe area on level $j+1$. It has been shown in [9] that the expansion property does not affect the asymptotic growth of the ε-significant tree $\mathcal{T} = \mathcal{T}_\varepsilon$ when $\varepsilon \to 0$. Since condition (4.8) can be realized by at most a finite uniformly bounded number of subdivisions of the leaves of a tree with expansion property, the same is true for well-graded trees. We shall always assume this property in what follows. The main result of this section can now be stated as follows.

Theorem 4.1 *There exists a constant C depending only on the wavelet bases $\Theta, \tilde{\Theta}$ and the well-gradedness parameters so that the computed array $\mathbf{d}^R(\mathcal{T})$ satisfies*

$$\|\mathbf{c}_{j_0} - \mathbf{d}^R_{j_0-1}\|_{\ell_2}^2 + \|\mathbf{d}(\mathcal{T}) - \mathbf{d}^R(\mathcal{T})\|_{\ell_2}^2 \le C \sum_{j=j_0+1}^{J+1} \|\mathbf{e}_j(\mathcal{G}_j^-)\|_{\ell_2}^2. \tag{4.9}$$

Proof: Recall that $\partial\mathcal{T}^-$ denotes the set of *outer leaves* of the tree $\mathcal{T}^\circ$, i.e. $\lambda^\circ \in \partial\mathcal{T}^-$ means that λ° does not belong to $\mathcal{T}^\circ$, yet its parent does. As the support cubes associated with $\partial\mathcal{T}^-$ form a partition of Ω, by (2.1) we have with $\mathbf{c}^{\mathrm{R}}_{j_0} = \mathbf{d}^{\mathrm{R}}_{j_0-1}$

$$\|\mathbf{c}_{j_0} - \mathbf{c}^{\mathrm{R}}_{j_0}\|^2_{\ell_2} + \|\mathbf{d}(\mathcal{T}) - \mathbf{d}^{\mathrm{R}}(\mathcal{T})\|^2_{\ell_2} \le c_\Theta^{-1}\|E_\mathcal{T}\|^2_{L_2(\Omega)} = c_\Theta^{-1}\sum_{\lambda^\circ\in\partial\mathcal{T}^-}\|E_\mathcal{T}\|^2_{L_2(\square_{\lambda^\circ})}. \tag{4.10}$$

The well-gradedness of $\mathcal{T}$ implies that for every $\lambda^\circ \in \partial\mathcal{T}^-_j$ one has that $(\tilde\Phi_l^T\mathbf{R}_l)|_{\square_{\lambda^\circ}} \not\equiv 0$ holds only for a uniformly bounded number of levels $|l-j| \le K$. Since the two scale matrices define uniformly bounded mappings on ℓ_2, the assertion follows. ∎

Note that the main difference from earlier results in [17] lies in the fact that the estimate involves no assumptions e.g. concerning the equilibration of local polynomial approximation errors of g, or in the case $g = G\circ v$, of v.

In fact, suppose the $q_{j,k}$ have the form $q_{j,k} = \langle Q_{j,k}(g), \phi_{j,k}\rangle$ where $Q_{j,k}(g)$ is a polynomial approximation of g on the support $S_{j,k}$ of $\phi_{j,k}$ of some order m, say. Suppose that e.g. a discrete least squares fit to g on a fixed sufficiently dense sample set in $S_{j,k}$ even assures (for the give g) that $\|g - Q_{j,k}(g)\|_{L_2(S_{j,k})} \le C\inf_{P\in\mathbb{P}_m}\|g-P\|_{L_2(S_{j,k})} =: \eta_{j,k}(g)$. Moreover suppose that the tree $\mathcal{T}$ is *essentially balanced* with respect to the $\eta_{j,k}$. This means that for some $\delta > 0$ and all $(j,k) \in \partial\mathcal{T}^-$ we have $\eta_{i,k}(g) \le \delta$ while for a subset Λ of $\partial\mathcal{T}$ with $\#\Lambda \ge c\#\partial\mathcal{T}$ one has $\eta_{j,k}(g) \ge \delta$, $(j,k)\in\Lambda$. Then, whenever $g \in B^s_q(L_p)$ with $1/p < s/d + 1/2$ for some $s \le m$ one has (see [17])

$$\left(\sum_{(j,k)\in\partial\mathcal{T}^-}\eta^2_{j,k}\right)^{1/2} \le C(\#\mathcal{T})^{-s/d}|g|_{B^s_q(L_p)}, \tag{4.11}$$

see e.g [21] for the definition of the Besov spaces $B^s_q(L_p)$ and their semi-norms. One immediately derives from (4.9) and (4.11)

$$\left(\|\mathbf{c}_{j_0} - \mathbf{c}^{\mathrm{R}}_{j_0}\|^2_{\ell_2} + \|\mathbf{d}(\mathcal{T}) - \mathbf{d}^{\mathrm{R}}(\mathcal{T})\|^2_{\ell_2}\right)^{1/2} \le C(\#\mathcal{T})^{-s/d}|g|_{B^s_q(L_p)}. \tag{4.12}$$

The interest in estimates of this type lies in the following points. The right hand side of (4.12) involves measuring smoothness of g in L_p which for the above admissible range of p imposes much weaker conditions than measuring smoothness in L_2. Thus a proper choice of a tree can make up for the lack of smoothness so as to retain an optimal order relating accuracy to the number of degrees of freedom even in the presence of singularities. Moreover, under the above assumptions the truncation error can be shown to be also bounded by the right hand side of (4.12) (although it could actually happen to be smaller), so that at least for the *class* of functions $g \in B^s_q(L_p)$ one obtains optimal error estimates. However, the line of argument of [17] works only under certain assumptions on the distribution of local errors and does not quite meet the demands arising in the context of adaptive solution schemes, see Section 1.2 and the notion of s^*-sparsity in [8]. Recall from **Task R2** that ideally one would like to relate the errors created by RECOVER directly to the truncation error $\|g - P_\mathcal{T} g\|_{L_2(\Omega)}$ or equivalently to $\|\mathbf{d} - \mathbf{d}(\mathcal{T})\|_{\ell_2}$ (or bounds for those terms). Note that one has

$$\|\mathbf{d} - \mathbf{d}(\mathcal{T})\|^2_{\ell_2} = \sum_{\lambda^\circ\in\partial\mathcal{T}}\sum_{\mu:\mu^\circ\succ\lambda^\circ}|d_\mu|^2. \tag{4.13}$$

Recall that $\mu^\circ \succ \lambda^\circ$ means that μ° is a descendant of (not equal to) λ°. Clearly, the "local remainders"

$$r_{\lambda^\circ} = r_{\lambda^\circ}(\mathbf{d},\mathcal{T}) := (\sum_{\mu:\mu^\circ\succ\lambda^\circ}|d_\mu|^2)^{1/2}, \quad \lambda^\circ \in \partial\mathcal{T}, \tag{4.14}$$

are just local L_2-errors of the approximation $P_{\mathcal{T}}g$ to g. Thus it would be ideal if the quadrature errors e_{λ° (see (4.2)) could be estimated in terms of r_{λ°, or at most by a finite number of nearby local remainders, which would entail $\|\mathbf{d}-\mathbf{d}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2} \lesssim \|\mathbf{d}-\mathbf{d}(\mathcal{T})\|_{\ell_2}$.

4.3 Dual Norms

As indicated before, when dealing with adaptive schemes for operator equations such as PDE's the relevant function space is most often not L_2 but some other Hilbert space $\mathcal{H}'$, typically a Sobolev space of positive order, see Section 1.2 and [8]. What will matter here is that the primal norms can be localized, i.e. restrictions of $f\in\mathcal{H}$ to sub-domains $\Omega'\subset\Omega$ (of suitable regularity) belong to a localized version $\mathcal{H}(\Omega')$ and that one has for any partition $\mathcal{P}$ of Ω into cells Δ

$$\sum_{\Delta\in\mathcal{P}}\|f\|^2_{\mathcal{H}(\Delta)} \lesssim \|f\|^2_{\mathcal{H}},\quad f\in\mathcal{H}. \tag{4.15}$$

This is, for instance, the case for $\mathcal{H}=H^t$ with $t\geq 0$. Under the assumption (2.10) that a Riesz basis Ψ is obtained from an L_2-Riesz basis Θ through scaling we can still apply the above scheme RECOVER to obtain

$$\mathbf{g}^{\mathrm{R}}(\mathcal{T}) := \mathbf{D}_{\mathcal{T}}\mathbf{d}^{\mathrm{R}}(\mathcal{T}) \tag{4.16}$$

as an approximation to the desired array $\mathbf{g}(\mathcal{T})=(\langle\psi_\lambda,g\rangle)_{\lambda\in\mathcal{T}}$, cf. 3.1. Of course, this scaling is incorporated in the loop of RECOVER, i.e. (3.13) is replaced by $\mathbf{d}^{\mathrm{R}}_j := \mathbf{D}_{\mathcal{T}_j}\mathbf{G}_{j,\tilde{\Theta}}\bar{\mathbf{c}}_{j+1}$. Again our objective is to estimate $\|\mathbf{g}(\mathcal{T})-\mathbf{g}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2}$ and ultimately $\|\mathbf{g}-\mathbf{g}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2}$.

4.3.1 Main obstruction and inherent problems

Estimating $\|\mathbf{g}(\mathcal{T})-\mathbf{g}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2}$ poses a certain principal difficulty which is perhaps worth pointing out first. The above estimate in Theorem 4.1 provides a bound in terms of local L_2-errors resulting from approximating the function g in L_2. The relevant function space here, however, is $\mathcal{H}'$. So in order to estimate the accuracy of the scaled array in (4.16) one would have to relate it to the error in $\mathcal{H}'$. This is not straightforward as Sobolev norms of negative order do not localize in a simple way.

At the first glance an easy remedy is offered by the norm equivalences (2.1). In fact, we recall that one can deduce from (2.1), (2.10) and (2.11) that for some $\hat{c},\hat{C}$ one has

$$\hat{c}\|\sum_{\lambda\in\mathcal{J}}\langle\psi_\lambda,w\rangle\tilde{\psi}_\lambda\|_{\mathcal{H}'}\leq\|\|\sum_{\lambda\in\mathcal{J}}\omega_\lambda\langle\theta_\lambda,w\rangle\tilde{\theta}_\lambda\|_{L_2}\leq\hat{C}\|\sum_{\lambda\in\mathcal{J}}\langle\psi_\lambda,w\rangle\tilde{\psi}_\lambda\|_{\mathcal{H}'}. \tag{4.17}$$

This tells us that approximation of some expansion $w=\sum_{\lambda\in\mathcal{J}}\langle\psi_\lambda,w\rangle\tilde{\psi}_\lambda$ in $\mathcal{H}'$ is equivalent to approximating the scaled expansion $\Sigma(w):=\sum_{\lambda\in\mathcal{J}}\omega_\lambda\langle\theta_\lambda,w\rangle\tilde{\theta}_\lambda$ in L_2, which by (2.1) yields an estimate for the approximations to $\langle\psi_\lambda,w\rangle$ in terms of an L_2 error for the approximant. For simplicity we shall assume that the scaling weights ω_λ depend only on the scale $|\lambda|$, $\omega_\lambda=\omega_{|\lambda|}$. Accordingly, it is suggested in [22] to approximate

$$\Sigma(g)_{\mathcal{T}} := \sum_{\lambda\in\mathcal{T}}\omega_\lambda\langle\theta_\lambda,g\rangle\tilde{\theta}_\lambda = \omega_{j_0}P_{j_0}g+\sum_{j=j_0}^{J}\omega_j(P_{j+1}-P_j)g$$

in L_2 and then use the above error estimation. However, that would require approximating the inner products $\langle\Sigma(g),\phi_{j,k}\rangle$. This, in turn, would require knowing the coefficients $\langle\Sigma(g),\phi_{j',k'}\rangle$ for all (j',k') such that the support of $\tilde{\phi}_{j',k'}$ intersects that of $\phi_{j,k}$, as the level dependent scaling

factor ω_j prevents cancellation inside the tree. This difficulty could be seen as a 'lacking locality' which, due to the negative norm, is inherent to the problem. Thus the approximate scaling function coefficients $q_{j,k}$ cannot be computed by just sampling g locally on the support of $\phi_{j,\mathbf{k}}$ and a related approach (see [22]) therefore does *not* seem to be computationally feasible.

In order to get around this difficulty and use only computationally feasible approximations to $\langle g, \phi_{j,k}\rangle$ for indices in $\mathcal{G}^-$, one could attempt to employ the same decomposition strategy as before trying to "hollow" the tree. To this end, note first that one has

$$\begin{aligned} g_{\mathcal{T}} &= \omega_{j_0} P_{j_0} g + \sum_{j=j_0}^{J} \omega_j (P_{j+1} - P_j) g_{\mathcal{T}} \\ &= \sum_{j=j_0+1}^{J+1} \tilde{\Phi}_j^T \Big([\omega_{j-1} - \omega_j] \mathbf{c}_j(\mathcal{T}_j^\circ) + \omega_{j-1} \Big(\mathbf{c}_j - \mathbf{M}_{j-1,\tilde{\Phi}} \mathbf{c}_{j-1}(\mathcal{G}_{j-1}^-) \Big)(\mathcal{G}_j^-) \Big), \end{aligned} \tag{4.18}$$

where we have used (3.6), see Section 3.1. In order to turn this into an approximation one needs approximations to the arrays $\mathbf{c}_j(\mathcal{T}_j^\circ)$ and $\mathbf{c}_j(\mathcal{G}_j^-)$. These approximations are already produced by the scheme RECOVER. Thus, we are led to define

$$\begin{aligned} P_{\mathcal{T}} g &:= \sum_{j=j_0+1}^{J+1} \tilde{\Phi}_j^T \Big([\omega_{j-1} - \omega_j] \check{\mathbf{c}}_j(\mathcal{T}_j^\circ) + \omega_{j-1} \Big(\mathbf{q}_j - \mathbf{M}_{j-1,\tilde{\Phi}} \mathbf{q}_{j-1}(\mathcal{G}_{j-1}^-) \Big)(\mathcal{G}_j^-) \Big) \\ &= \sum_{j=j_0+1}^{J+1} \tilde{\Phi}_j^T \Big(\omega_{j-1} \bar{\mathbf{c}}_j(\mathcal{G}_j) - \omega_j \check{\mathbf{c}}_j(\mathcal{T}_j^\circ) \Big). \end{aligned} \tag{4.19}$$

The main difference from the previous situation, reflected by (4.3) and (4.4), lies now in the additional term $[\omega_{j-1} - \omega_j]\check{\mathbf{c}}_j(\mathcal{T}_j^\circ)$ due to the new scaling in front of the telescoping summands. This means that the error terms will no longer live only on $\mathcal{G}^-$ but that multiscale decompositions of higher level error components are transported into the interior of $\mathcal{T}$. It does not seem to be clear how this will effect the overall error. These observations seem to indicate that error estimation for L_2 does not simply carry over to more general norms. Thus, in summary, the computational complexity of the approach proposed in [22] is therefore not clear to us.

4.3.2 A general estimate

On account of the remarks brought up in the previous section, we prefer to stay here with the original order of operation, namely to apply RECOVER to g and then scale the resulting coefficients, see (4.16), as opposed to scaling g first into $\Sigma(g)$ and then applying RECOVER. It is now less obvious though how to estimate the resulting array in a proper way and, based on such estimates, how to realize **Task R2**. One probably cannot expect a complete analysis that works in the generality considered so far. Therefore the objective of this section is to further explore some basic aspects of **Task R2** and to derive from these considerations suitable algorithmic ingredients. This is to identify more specific requirements on the quadrature that may lead to the desired asymptotically optimal computational performance. In particular, it will be seen that, in a strict sense, this requires generally more than just accuracy bounds for the individual coefficients $q_{j,\mathbf{k}}$.

For technical simplicity we shall continue assuming in what follows that the scaling weights in (2.10) depend only on the scales, i.e. $\omega_\lambda = \omega_{|\lambda|}$. The following reasoning, however, can be extended to more general situations. One easily concludes from (2.11) that then for (well behaved sub-domains Ω' of Ω)

$$\inf_{v_j \in S_j} \|v - v_j\|_{L_2(\Omega')} \lesssim \omega_j \|v\|_{\mathcal{H}(\Omega')}, \quad v \in \mathcal{H}. \tag{4.20}$$

Our goal is again to estimate the scaled arrays $\|\mathbf{g}(\mathcal{T})-\mathbf{g}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2}$ in terms of the quadrature errors e_{λ°, $\lambda^\circ\in\mathcal{G}^-$. One might think that the ideal estimate would be

$$\|\mathbf{g}(\mathcal{T})-\mathbf{g}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2}^2\le C\sum_{j=j_0+1}^{J+1}\omega_j^2\|\mathbf{e}_j(\mathcal{G}_j^-)\|_{\ell_2}^2, \tag{4.21}$$

i.e., the error contributions scale like wavelet coefficients. This would indeed be the case if the error components were scaling function coefficients of some underlying error function, i.e., $e_{\lambda^\circ}=\langle\phi_{\lambda^\circ},E_{\mathcal{T}}\rangle$ which would entail the relations

$$\mathbf{G}_{j,\tilde\Phi}\mathbf{e}_{j+1}=\mathbf{e}_j. \tag{4.22}$$

This, in turn, is equivalent to $\mathbf{G}_{j,\tilde\Phi}\mathbf{q}_{j+1}=\mathbf{q}_j$, which, of course, will generally not be true. So it seems that all we can hope for is an estimate that involves the deviation from (4.22) which then hopefully is relatively small. In order to make this more precise, we introduce the following notation. Let

$$\Omega_j:=\bigcup_{(j,\mathbf{k})\in\mathcal{G}_j^-}\operatorname{supp}\tilde\phi_{j,\mathbf{k}},\quad P_j(w|\mathcal{G}_j^-):=\sum_{(j,\mathbf{k})\in\mathcal{G}_j^-}\langle\phi_{j,\mathbf{k}},w\rangle\tilde\phi_{j,\mathbf{k}}. \tag{4.23}$$

We can use (2.11) to conclude that

$$\|\mathbf{g}(\mathcal{T})-\mathbf{g}^{\mathrm{R}}(\mathcal{T})\|_{\ell_2}\le C_\Psi\|E_{\mathcal{T}}\|_{\mathcal{H}'}, \tag{4.24}$$

where $E_{\mathcal{T}}$ is given by (4.4). Furthermore let $P_j^*(\cdot|\mathcal{G}_j^-)$ denote the adjoint of $P_j(\cdot|\mathcal{G}_j^-)$ and

$$\|v\|_{(\mathcal{H}(\Omega'))'}:=\sup_{w\in\tilde{\mathcal{H}}(\Omega')}\frac{\langle w,v\rangle_{\Omega'}}{\|w\|_{\mathcal{H}(\Omega')}}$$

where $\tilde{\mathcal{H}}(\Omega')$ consists for $\Omega'\subset\Omega$ of those $w\in\mathcal{H}(\Omega')$ whose extension by zero to all of Ω is still in $\mathcal{H}$. In order to develop suitable quadrature techniques, it will be convenient to employ the mappings

$$L_jg:=\sum_{(j,\mathbf{k})\in\mathcal{G}_j^-}q_{j,\mathbf{k}}(g)\tilde\phi_{j,\mathbf{k}}, \tag{4.25}$$

which always involve only indices from $\mathcal{G}_j^-$. As before, the coefficients $q_{j,\mathbf{k}}(g)$ forming the arrays $\mathbf{q}_j$ on $\mathcal{G}_j^-$ are to approximate the exact scaling function coefficients $c_{j,\mathbf{k}}=\langle\phi_{j,\mathbf{k}},g\rangle$ for $(j,\mathbf{k})\in\mathcal{G}_j^-$. We shall now present some general estimates for the error in the dual norm in order to obtain upper bounds for (4.24).

Proposition 4.2 *Under the above assumptions we have the following estimates of the quadrature error $E_{\mathcal{T}}$ defined in (4.4).*

a) One has

$$\|E_{\mathcal{T}}\|_{\mathcal{H}'}\ \lesssim\ (\sum_{j=j_0}^{J+1}\|\tilde\Phi_j^T\mathbf{e}_j\|_{\mathcal{H}(\Omega_j)'}^2)^{1/2}. \tag{4.26}$$

b) Recalling the definition of P_j (3.4) we have for any L_j given by (4.25)

$$\|E_{\mathcal{T}}\|_{\mathcal{H}'}\ \lesssim\ (\sum_{j=j_0}^{J+1}\|L_jg-P_jg\|_{\mathcal{H}(\Omega_j)'}^2)^{1/2}. \tag{4.27}$$

Proof: As for a), we recall the definition (4.4) and obtain, upon using (2.2) and biorthogonality of $\Phi_j, \tilde{\Phi}_j$ several times,

$$\begin{aligned}
\|E_{\mathcal{T}}\|_{\mathcal{H}'} &= \sup_w \frac{\langle E_{\mathcal{T}}, w\rangle}{\|w\|_{\mathcal{H}}} = \sup_w \frac{\sum_{j=j_0+1}^{J+1}\langle \tilde{\Phi}_j^T\Big(\mathbf{e}_j - \mathbf{M}_{j-1,\tilde{\Phi}}\mathbf{e}_{j-1}(\mathcal{G}_{j-1}^-)\Big)(\mathcal{G}_j^-), w\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}}} \\
&= \sup_w \frac{\sum_{j=j_0+1}^{J+1}\langle \tilde{\Phi}_j^T\Big(\mathbf{e}_j - \mathbf{M}_{j-1,\tilde{\Phi}}\mathbf{e}_{j-1}(\mathcal{G}_{j-1}^-)\Big)(\mathcal{G}_j^-), P_j^*(w|\mathcal{G}_j^-)\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}}} \\
&= \sup_w \frac{\sum_{j=j_0+1}^{J+1}\langle \tilde{\Phi}_j^T\Big(\mathbf{e}_j - \mathbf{M}_{j-1,\tilde{\Phi}}\mathbf{e}_{j-1}(\mathcal{G}_{j-1}^-)\Big), P_j^*(w|\mathcal{G}_j^-)\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}}} \\
&= \sup_w \frac{\sum_{j=j_0+1}^{J+1}\langle \tilde{\Phi}_j^T\mathbf{e}_j - \tilde{\Phi}_{j-1}^T\mathbf{e}_{j-1}, P_j^*(w|\mathcal{G}_j^-)\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}}} \\
&\leq \sup_w \frac{\sum_{j=j_0+1}^{J+1}\|\tilde{\Phi}_j^T\mathbf{e}_j - \tilde{\Phi}_{j-1}\mathbf{e}_{j-1}\|_{(\mathcal{H}(\Omega_j))'}\|P_j^*(w|\mathcal{G}_j^-)\|_{\mathcal{H}(\Omega_j)}}{\|w\|_{\mathcal{H}}} \\
&\lesssim \sup_w \frac{\sum_{j=j_0+1}^{J+1}\|\tilde{\Phi}_j^T\mathbf{e}_j - \tilde{\Phi}_{j-1}\mathbf{e}_{j-1}\|_{(\mathcal{H}(\Omega_j))'}\|w\|_{\mathcal{H}(\Omega_j)}}{\|w\|_{\mathcal{H}}},
\end{aligned} \tag{4.28}$$

where we have also used that the $P_j^*(\cdot|\mathcal{G}_j^-)$ are bounded in $\mathcal{H}$. Since $\mathbf{e}_{j-1}$ is supported in $\mathcal{G}_{j-1}^-$, we have that $\|\tilde{\Phi}_{j-1}\mathbf{e}_{j-1}\|_{(\mathcal{H}(\Omega_j))'} = \|\tilde{\Phi}_{j-1}\mathbf{e}_{j-1}\|_{(\mathcal{H}(\Omega_j\cap\Omega_{j-1}))'} \leq \|\tilde{\Phi}_{j-1}\mathbf{e}_{j-1}\|_{(\mathcal{H}(\Omega_{j-1}))'}$, since the dual norm is monotone in the domain. Using this, applying Cauchy-Schwarz, bearing in mind that, by well-gradedness, only a uniformly bounded finite number of Ω_j overlap at any given point and employing (4.15), yields the assertion a).
Concerning b), we have the representation, cf. (4.2)

$$e_{j,\mathbf{k}} = q_{j,\mathbf{k}} - c_{j,\mathbf{k}} = \langle \phi_{j,\mathbf{k}}, L_j g - P_j g\rangle, \quad \text{i.e} \quad \mathbf{e}_j = \langle \Phi_j, L_j g - P_j(g|\mathcal{G}_j^-)\rangle. \tag{4.29}$$

Clearly, (4.29) says that $\tilde{\Phi}_j^T\mathbf{e}_j = P_j(L_j g|\mathcal{G}_j^-) - P_j(g|\mathcal{G}_j^-)) = L_j g - P_j(g|\mathcal{G}_j^-)$. Hence, using the biorthogonality of $\Phi_j, \tilde{\Phi}_j$, we can rewrite the terms in (4.26) as follows.

$$\begin{aligned}
\|\tilde{\Phi}_j^T\mathbf{e}_j\|_{\mathcal{H}(\Omega_j)'} &= \sup_{w\in\mathcal{H}(\Omega_j)} \frac{\langle \tilde{\Phi}_j^T\mathbf{e}_j, P_j^*(w|\mathcal{G}_j^*)\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}(\Omega_j)}} = \sup_{w\in\mathcal{H}(\Omega_j)} \frac{\langle L_j(g|\mathcal{G}_j^-) - P_j(g|\mathcal{G}_j^-), P_j^*(w|\mathcal{G}_j^*)\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}(\Omega_j)}} \\
&= \sup_{w\in\mathcal{H}(\Omega_j)} \frac{\langle L_j g - P_j g, P_j^*(w|\mathcal{G}_j^*)\rangle_{\Omega_j}}{\|w\|_{\mathcal{H}(\Omega_j)}},
\end{aligned} \tag{4.30}$$

which confirms the claim b) by the same arguments as used in a). ∎

5 Quadrature

So far we have not specified how to compute in RECOVER the approximate scaling function coefficients $q_{j,k}$. So the objective of this section is to find relevant requirements on quadrature. We will start by identifying desirable properties of L_j from (4.25).

5.1 A wishlist for the mappings L_j

In what follows we shall make frequent use of the following fact.

Remark 5.1 *For any sub-domain Ω' of Ω one has $\|g\|^2_{(\mathcal{H}(\Omega'))'} \lesssim \sum_{\lambda:S_\lambda\cap\Omega'\neq\emptyset}|\langle\psi_\lambda,g\rangle|^2$, where $S_\lambda = \operatorname{supp}\tilde\psi_\lambda$.*

Proof: For $w\in\tilde{\mathcal{H}}(\Omega')$ one has

$$\begin{aligned}|\langle w,g\rangle| &= |\langle w,\sum_{\lambda\in\mathcal{J}}\langle\psi_\lambda,g\rangle\tilde\psi_\lambda\rangle| = |\sum_{\lambda:S_\lambda\cap\Omega'\neq\emptyset}\langle w,\tilde\psi_\lambda\rangle\langle\psi_\lambda,g\rangle|\\ &\lesssim \|w\|_{\mathcal{H}(\Omega')}\Big(\sum_{\lambda:S_\lambda\cap\Omega'\neq\emptyset}|\langle\psi_\lambda,g\rangle|^2\Big)^{1/2},\end{aligned}$$

whence the assertion follows. ∎

Recalling that $P_jg=\sum_{(j,\mathbf{k})\in\mathcal{I}_j}\langle\phi_{j,\mathbf{k}},g\rangle\tilde\phi_{j,\mathbf{k}}=\sum_{(j,\mathbf{k})\in\mathcal{I}_j}c_{j,\mathbf{k}}\tilde\phi_{j,\mathbf{k}}$, the subsequent discussion will be guided by the following simple observations based on Proposition 4.2 b).

Proposition 5.2 *a) If*

$$\|L_jg-P_jg\|_{(\mathcal{H}(\Omega_j))'}\le C\|g-P_jg\|_{(\mathcal{H}(\Omega_j))'} \tag{5.1}$$

uniformly in j, then the approximate array $\mathbf{g}^R(\mathcal{T})$ on the well-graded tree $\mathcal{T}$ produced by RECOVER *satisfies*

$$\|\mathbf{g}(\mathcal{T})-\mathbf{g}^R(\mathcal{T})\|_{\ell_2}\le C^*\|\mathbf{g}-\mathbf{g}(\mathcal{T})\|_{\ell_2}, \tag{5.2}$$

where C^* is independent of $\mathcal{T}$ but depends on C from (5.1) and the constant in (4.27).
b) Suppose that $\mathcal{T}$ satisfies

$$\|\mathbf{g}-\mathbf{g}(\mathcal{T})\|_{\ell_2}\le\varepsilon \tag{5.3}$$

and that J is the highest level appearing in $\mathcal{T}$. If for $j_0\le j\le J+1$

$$\|L_jg-P_jg\|_{(\mathcal{H}(\Omega_j))'}\le C\varepsilon\sqrt{\#\mathcal{G}_j^-/\#\mathcal{G}^-} \tag{5.4}$$

uniformly in $j\le J$, one has $\|\mathbf{g}-\mathbf{g}^R(\mathcal{T})\|_{\ell_2}\le(1+C^*)\varepsilon$, where C^* depends on the constants in (3.15) and (4.27), but is independent of $\mathcal{T}$.

Proof: Applying Remark 5.1 to $g-P_jg$, (5.1) says that for $g_\lambda=\langle\psi_\lambda,g\rangle$

$$\|L_jg-P_jg\|^2_{(\mathcal{H}(\Omega_j))'}\lesssim\sum_{\substack{|\lambda|>j\\ \operatorname{supp}\tilde\psi_\lambda\cap\Omega_j\neq\emptyset}}|g_\lambda|^2. \tag{5.5}$$

Since by well-gradedness of $\mathcal{T}$ only finitely many of the Ω_j overlap at a given point and since by (3.15) $\#\mathcal{G}^-\lesssim\#\mathcal{T}\lesssim\#\partial\mathcal{T}^-$, we conclude that the right hand side of (4.27) can be bounded in the following way

$$\|E_{\mathcal{T}}\|_{\mathcal{H}'}\lesssim\Big(\sum_{j=j_0}^{J+1}\|L_jg-P_jg\|^2_{\mathcal{H}(\Omega_j)'}\Big)^{1/2}\lesssim\Big(\sum_{j=j_0}^{J+1}\sum_{\substack{|\lambda|>j\\ \operatorname{supp}\tilde\psi_\lambda\cap\Omega_j\neq\emptyset}}|g_\lambda|^2\Big)^{1/2}\lesssim\Big(\sum_{\lambda\notin\mathcal{T}}|g_\lambda|^2\Big)^{1/2}=\|\mathbf{g}-\mathbf{g}(\mathcal{T})\|_{\ell_2},$$

which confirms a). The proof of the second claim is analogous. ∎

Obviously, (5.1) is the stronger assumption which would yield the best possible result, namely that the quadrature error is dominated by the prediction error. Recall, however, that the scenario given by part b) in Proposition 5.2 is sufficient for **Task R2** which warrants optimal complexity estimates in the adaptive context described in Section 1.2. This leads us to the following definition.

Definition 5.3 *We shall call a quadrature* reliable for, *respectively* ε-reliable for *g when the corresponding mappings L_j satisfy (5.1), respectively, when (5.4) holds whenever (5.3) is provided.*

5.2 Reliable Quadrature

We still have to detail on how to set up a reliable quadrature routine, cf. Definition 5.3. Before going into this, we comment briefly on the fact that we are actually dealing with approximating g in spaces on which point evaluations are generally not continuous. One should keep in mind that the main context where RECOVER applies has been outlined in Section 1.2. In this case we have $g = F(v)$ where v is a *finite* wavelet expansion and where F is a local operator. Moreover, v itself has usually the form $v = \sum_{\lambda\in\mathcal{T}_v} v_\lambda\psi_\lambda$ where $\mathcal{T}_v$ is a tree that is contained in the prediction tree $\mathcal{T}$. Since quadrature affects only terms $\langle\phi_{j,\mathbf{k}}, F(v)\rangle$ where $(j,\mathbf{k}) \in \mathcal{G}^-$ and since $\mathcal{T}$ is a well graded tree $\phi_{j,\mathbf{k}}$ "sees" only the image of lower scale wavelets in the expansion of v under the mapping F. Thus, on the support of $\phi_{j,\mathbf{k}}$, $F(v)$ is roughly speaking "finite dimensional" and can only oscillate with a frequency comparable at most with the diameter of $\phi_{j,\mathbf{k}}$. Moreover, the composition actually has some pointwise smoothness even of some positive Hölder degree. (Of course, when $\varepsilon \to 0$ these norms may eventually grow unboundedly because $F(v)$ is only stable in $\mathcal{H}'$). Thus it is justified to assume always that point evaluations are well defined and that, for any fixed $\mathcal{T} = \mathcal{T}_\varepsilon$, refining quadrature locally, does provide increasing accuracy of approximations to quantities like $\langle\phi_{j,\mathbf{k}}, g\rangle$.

The approximations $q_{j,\mathbf{k}}(g)$ approximating the exact scaling function coefficients $c_{j,\mathbf{k}} = \langle\phi_{j,\mathbf{k}}, g\rangle$ on $\mathcal{G}_j^-$ can be obtained, for instance, by making use of the fact that $\int \phi_{j,k}(x)f(x)dx$ can often be computed *exactly* for certain functions f such as arbitrary other *refinable* functions or polynomials of any degree, see [14]. A simple way then is to determine some local approximation $Q_{j,\mathbf{k}}(g)$ on the support of $\phi_{j,\mathbf{k}}$, either by interpolation or by a least squares fit using proper oversampling. Then $\langle Q_{j,\mathbf{k}}(g), \phi_{j,\mathbf{k}}\rangle$ can be computed exactly providing

$$q_{j,\mathbf{k}} := \int_\Omega Q_{j,\mathbf{k}}(g)(x)\phi_{j,\mathbf{k}}(x)dx, \qquad (j,\mathbf{k}) \in \mathcal{G}_j^-. \tag{5.6}$$

Exactness in $\tilde{S}_j$: By the above remarks, $Q_{j,\mathbf{k}}(g)$ might be a polynomial of possibly high fixed order, or a linear combination of high order B-splines. However, recall from (5.1) that the main issue is to make $L_j g$ close to $P_j g$ in $(\mathcal{H}(\Omega_j))'$. This suggests to make L_j map into $\tilde{S}_j = \operatorname{span}\tilde{\Phi}_j$, cf. Section 2.1. Especially, for fixed but sufficiently dense sampling sets $Y_{j,\mathbf{k}}$ one could define $Q_{j,\mathbf{k}}(g)$ by the *least squares* fit

$$\begin{aligned} Q_{j,\mathbf{k}}(g) &:= \sum_{(j,\mathbf{k}'):\operatorname{supp}\tilde{\phi}_{j,\mathbf{k}'}\cap\operatorname{supp}\phi_{j,\mathbf{k}}\neq\emptyset} q^*_{j,\mathbf{k}'}\tilde{\phi}_{j,\mathbf{k}'}, \quad \text{where} \\ \mathbf{q}^* &= \operatorname{argmin}_{\tilde{\mathbf{q}}} \sum_{y\in Y_{j,\mathbf{k}}} (g(y) - \sum_{(j,\mathbf{k}')} \tilde{q}_{j,\mathbf{k}'}\tilde{\phi}_{j,\mathbf{k}'}(y))^2, \end{aligned} \tag{5.7}$$

so that, by biorthogonality, $q_{j,\mathbf{k}} = q^*_{j,\mathbf{k}}$. In this case the mapping L_j would be indeed exact on $\tilde{S}_j$.

Boundedness in $(\mathcal{H}(\Omega_j))'$: Moreover, "ideally" the mapping L_j would be a linear projector onto $\tilde{S}_j(\mathcal{G}_j^-) := \operatorname{span}\{\tilde{\phi}_{j,\mathbf{k}} : (j,\mathbf{k}) \in \mathcal{G}_j^-\}$ which is bounded in $(\mathcal{H}(\Omega_j))'$. In fact, one could then write $L_j g - P_j(g|\mathcal{G}_j^-) = L_j(g - P_j g)$ so that the boundedness of L_j in $(\mathcal{H}(\Omega_j))'$ would immediately give (5.1) hence Proposition (5.2) a) would apply and provide the desired estimate (5.2). Such a requirement will generally not be feasible. Without any further assumptions on g this cannot be guaranteed by any fixed rule for forming the $q_{j,\mathbf{k}}$. In fact, L_j can ultimately be based only on discrete information on some finest resolution level. On the other hand, in the above mentioned "locally finite dimensional case" one can expect to contrive a mechanism for improving the quality

of the $q_{j,\mathbf{k}}$ if necessary. This means, one can realize the necessary closeness of $L_j g$ to the particular projector $P_j g$ in $(\mathcal{H}(\Omega_j))'$ required by Proposition 5.2. We shall describe next some ingredients of such a mechanism.

To this end, denote by $\mathcal{G}_{j,r}^-$ the set of those indices in $\mathcal{I}_{j+r}$ which arise from an r-fold subdivision of the $\tilde{\phi}_{j,\mathbf{k}}, (j,\mathbf{k}) \in \mathcal{G}_j^-$. These indices enter local refinements of $\tilde{S}_j$ over Ω_j. Let for $r \geq 0$

$$L_j^r g := \sum_{(j+r,\mathbf{k}) \in \mathcal{G}_{j,r}^-} q_{j+r,\mathbf{k}}^r(g) \tilde{\phi}_{j+r,\mathbf{k}}$$

be a mapping into $\tilde{S}_{j+r}(\mathcal{G}_{j,r}^-)$. The L_j^r could be defined by any of the above ways but with respect to higher resolution level $j+r$. If $L_j^r g$ provides better approximations to g with increasing r, then its projection back to $\tilde{S}_j(\mathcal{G}_j^-)$ can be expected to yield better approximations to $P_j g$, provided these approximations are stable in $(\mathcal{H}(\Omega_j))'$. Thus, given $L_j^r g$ we then set

$$L_j g := P_j(L_j^r g). \tag{5.8}$$

Writing $L_j g = (\mathbf{q}^r)_j^T \tilde{\Phi}_j$ in the form (4.25), a repeated application of (2.13) yields that the array $\mathbf{q}_j$ is given by

$$\mathbf{q}_j = \mathbf{q}_j^r := \mathbf{G}_{j,\tilde{\Phi}} \cdots \mathbf{G}_{j+r-1,\tilde{\Phi}} \mathbf{q}_{j+r}^r. \tag{5.9}$$

As indicated above, the rationale is that the closer $L_j^r g$ gets to g, due to the better resolution offered by projecting into $\tilde{S}_{j+r}$, the closer should its projection by P_j be to $P_j g$.

A possible obstruction that has to be taken into account is that high frequency components of g might be picked up through the quadrature so as to render L_j^r unstable in $(\mathcal{H}(\Omega_j))'$. We shall discuss possible remedies under the general assumption that quadrature is feasible. More precisely, we shall always make the following

Working Assumption (WA): Let $\mathcal{T} = \mathcal{T}_\varepsilon$ be the prediction tree for which $\|\mathbf{g} - \mathbf{g}(\mathcal{T})\|_{\ell_2} \leq \varepsilon$ and let $J = J(\mathcal{T})$ be the highest scale occurring in $\mathcal{T}$. For any $\rho > 0$ there exists an $R \in \mathbb{N}$ such that one has for all $j \leq J(\mathcal{T}_\varepsilon)$ uniformly in ε

$$\inf_{v \in \tilde{S}_{j+R}} \|g - v\|_{(\mathcal{H}(\Omega_j))'} \leq \rho \|g - P_j g\|_{(\mathcal{H}(\Omega_j))'}, \tag{5.10}$$

i.e. the wavelet coefficients of g with respect to $\tilde{\Psi}$ have some fixed scale-wise decay and best tree approximation is a near best N-term approximation.

This assumption can be verified for many operators F in the second scenario in Section 1.2 when $g = F(v)$ and v is a finite tree expansion where the tree for v is contained in the prediction tree $\mathcal{T}$ for $F(v)$. We shall refer to this situation, in a slight abuse of terminology, as the "locally finite dimensional case" on every support of $\phi_{j,\mathbf{k}}$ with $(j,\mathbf{k}) \in \mathcal{G}^-$. This follows from the construction of the prediction trees in [9] and the corresponding error analysis which shows how expanding the tree decreases the error. Since

$$\|L_j g - P_j g\|_{(\mathcal{H}(\Omega_j))'} = \|P_j(L_j^r g - g)\|_{(\mathcal{H}(\Omega_j))'} \lesssim \|L_j^r g - g\|_{(\mathcal{H}(\Omega_j))'}$$

the condition (5.1) would be satisfied for $r \leq R$ provided that the mapping L_j^r is *exact* on $\tilde{S}_{j+r}$ and *stable* in $(\mathcal{H}(\Omega_j))'$. Here are some ingredients towards this aim. Any linear combination

$v_{j+r} := \sum_{(j+r,\mathbf{k}')\in\mathcal{G}_{j,r}^-} \mathbf{q}_{j+r,\mathbf{k}'}\tilde{\phi}_{j+r,\mathbf{k}'}$ can be written as

$$v_{j+r} = P_j v_{j+r} + (I-P_j)v_{j+r} = \mathbf{p}_j^T\tilde{\Phi}_j + \sum_{l=j}^{j+r-1} \bar{\mathbf{d}}_l^T\Theta_l,$$

where, by (5.9)

$$\mathbf{p}_j = \mathbf{G}_{j,\tilde{\Phi}}\cdots\mathbf{G}_{j+r-1,\tilde{\Phi}}\mathbf{q}_{j+r}, \quad \bar{\mathbf{d}}_l = \mathbf{G}_{l,\tilde{\Theta}}\mathbf{G}_{l+1,\tilde{\Phi}}\cdots\mathbf{G}_{j+r-1,\tilde{\Phi}}\mathbf{q}_{j+r}. \tag{5.11}$$

Now let $Y_{j+r}(\mathcal{G}_{j,r}^-)$ be a set of sampling points in Ω_j whose cardinality is larger than $\#\mathcal{G}_{j,r}^-$ but of the same order. Fix a positive weight α and consider the least squares problem

$$\begin{aligned}
\mathbf{q}_{j+r}^* &:= \operatorname{argmin}_{\mathbf{q}_{j+r}}\{\alpha \sum_{y\in Y_{j+r}(\mathcal{G}_{j,r}^-)} h^d(g(y) - \sum_{(j+r,\mathbf{k}')\in\mathcal{G}_{j,r}^-} q_{j+r,\mathbf{k}'}\tilde{\phi}_{j+r,\mathbf{k}'}(y))^2 \\
&\quad + \sum_{l=j}^{j+r-1} \omega_j^2 \|\mathbf{G}_{l,\tilde{\Theta}}\mathbf{G}_{l+1,\tilde{\Phi}}\cdots\mathbf{G}_{j+r-1,\tilde{\Phi}}\mathbf{q}_{j+r}\|_{\ell_2}^2\} \\
&=: \operatorname{argmin}_{\mathbf{q}_{j+r}}\{K_0(\mathbf{q}_{j+r}) + K_1(\mathbf{q}_{j+r})\},
\end{aligned} \tag{5.12}$$

where h measures the spacing between the sampling points. First of all, for suitable $Y_{j+r}(\mathcal{G}_{j,r}^-)$ (5.12) has a unique solution. Moreover, if $g\in\tilde{S}_j(\mathcal{G}_j^-)$ the minimum of the quadratic functional is zero and $P_j(L_j^r g) = g$ on Ω_j. In fact the first part $K_0(\mathbf{q}_{j+r})$ of the functional can be made zero through the proper fit and the second part vanishes since the wavelet coefficients of g vanish. In general, we have $K_1(\mathbf{q}_{j+r}) \sim \|(I-P_j)v_{j+r}\|^2_{(\mathcal{H}(\Omega_j))'}$ controls the dual norm of the fit whose quality is ensured by $K_0(\mathbf{q}_{j+r})$. In fact, $K_0(\mathbf{q}_{j+r})$ can be viewed as a weighted L_2-approximation, especially in the above mentioned locally finite dimensional case. For instance, with $\alpha\sim\omega_j^2$, $K_0(\mathbf{q}_{j+r})^{1/2}$ becomes then a good upper bound for $\|g-v_{j+r}\|_{(\mathcal{H}(\Omega_j))'}$.

In practical terms it is not essential to solve (5.12) exactly. What matters is that $K_1(\mathbf{q}_{j+r})$ is controlled thereby stabilizing the $(\mathcal{H}(\Omega_j))'$ approximation. Therefore one can proceed as follows.

(i) For $(j,\mathbf{k})\in\mathcal{G}_j^-$ determine $q_{j,\mathbf{k}}$ according to (5.7).

(ii) Set $\mathbf{q}_{j+r} := \mathbf{M}_{j+r-1,\tilde{\Phi}}\cdots\mathbf{M}_{j,\tilde{\Phi}}\mathbf{q}_j$ so that the corresponding $\bar{\mathbf{d}}_l$, $l=j,\dots,j+r-1$, vanish, see (5.11).

(iii) With this $\mathbf{q}_{j+r}$ as an initial guess one carries out a few gradient descent steps to drive $K_0(\mathbf{q}_{j+r}) + K_1(\mathbf{q}_{j+r})$ towards the minimum.

Again, when $g\in\tilde{S}_j(\mathcal{G}_j^-)$ one already has (up to round off) $K_0(\mathbf{q}_{j+r})+K_1(\mathbf{q}_{j+r}) = 0$, i.e. exactness is ensured. Step (iii) provides the desired $(\mathcal{H}(\Omega_j))'$-stability where one could vary the weight α in the course of the iteration. Clearly, for fixed r the overall computational work remains proportional to $\#\mathcal{G}_j^-$.

In order to guarantee the desired accuracy of the approximation one could solve (5.12) for several increasing values of r (always using the result of the previous step as initial guess). This procedure is stopped when either some fixed upper bound for r is exceeded or when the following criteria are fulfilled for some $r'>r$ and some fixed constant $c<1$

$$\begin{aligned}
\|\mathbf{q}_j^{r'} - \mathbf{q}_j^r\|_{\ell_2} &\le c\varepsilon(\#\mathcal{G}_j^-/\#\mathcal{G}^-), \\
(\sum_{l=j+r}^{j+r'-1} \omega_j^2\|\mathbf{G}_{l,\tilde{\Theta}}\mathbf{G}_{l+1,\tilde{\Phi}}\cdots\mathbf{G}_{j+r-1,\tilde{\Phi}}\mathbf{q}_{j+r'}\|_{\ell_2}^2)^{1/2} &\le c\varepsilon(\#\mathcal{G}_j^-/\#\mathcal{G}^-).
\end{aligned} \tag{5.13}$$

An alternative criterion is to increase r until

$$(K_0(\mathbf{q}^*_{j+r}) + K_1(\mathbf{q}^*_{j+r}))^{1/2} \leq C\varepsilon(\#\mathcal{G}_j^- / \#\mathcal{G}^-), \tag{5.14}$$

where now C could be a fixed (larger) constant. To explain these criteria note that, under Assumption **WA**, $\|g - L_j^r g\|_{(\mathcal{H}(\Omega_j))'}$ gets small compared with $\|g - P_j g\|_{(\mathcal{H}(\Omega_j))'}$ for increasing r so that the $\mathbf{q}_j^r$ become stationary for increasing r while the higher order wavelet coefficients become negligible. Thus the stopping criteria (5.13) are met after a finite number of steps depending only on the choice of the constants c in (5.13), (5.14). As for (5.14), a prediction error $\|\mathbf{g} - \mathbf{g}(\mathcal{T})\|_{\ell_2} \leq \varepsilon$ suggests to expect a portion like $\varepsilon(\#\mathcal{G}_j^- / \mathcal{G}^-)$ to be attributed to $\mathcal{G}_j^-$. On account of the norm equivalence (1.2) one expects that $\|g - P_j(g|\mathcal{G}_j^-)\|_{(\mathcal{H}(\Omega_j))'} \lesssim \varepsilon(\#\mathcal{G}_j^- / \mathcal{G}^-)$. Thus the value of $(K_0(\mathbf{q}^*_{j+r}) + K_1(\mathbf{q}^*_{j+r}))^{1/2}$ would be of the order $\varepsilon(\#\mathcal{G}_j^- / \mathcal{G}^-)$ for $\mathbf{q}^*_{j+r} := \mathbf{M}_{j+r-1,\tilde{\Phi}} \cdots \mathbf{M}_{j,\tilde{\Phi}} \mathbf{c}_j$. Hence, when (5.14) is met one has an $\mathcal{H}'$-stable sufficiently good approximation to $P_j(\cdot|\mathcal{G}_j^-)$.

Remark 5.4 *Recall that* RECOVER *works from top to bottom, i.e., $\mathcal{G}^-$ is not known beforehand and is in fact never assembled. However, on account of (3.15) we can replace $\#\mathcal{G}^-$ in (5.13), (5.14) by the known quantity $\#\mathcal{T}^\circ$.*

Thus, in principle, for a wide range of cases an asymptotically optimal work count, as required by Remark 3.2, can be achieved. Nevertheless, the quantitative performance of a strategy outlined above may still be unacceptable. It is therefore important to see whether the full scope of such a strategy is generally necessary. The numerical experiments in the next section are to shed some light on this issue.

6 Numerical Experiments

Referring primarily to the second scenario in Section 1.2, we shall apply now the recovery scheme RECOVER to compositions $g = (F \circ v)$ as a simple representative of a nonlinear operator. Here F is just a smooth nonlinear scalar valued function and v is some finite wavelet expansion that approximates some function u referred to as the *original function*. Thus g is always pointwise defined. We are not interested in any specific application but in a quantitative validation of the above theoretical investigations which guides the selection of u, F. In order to be able to access true errors the role of the original function u will actually be played by some fixed highly accurate approximation u_J of u, namely its (approximate) projection to $S(\Phi_J)$. This is done by first computing the scaling function coefficients by (highly accurate) quadrature on a uniform grid. The standard wavelet transform yields then the finite array of wavelet coefficients $\mathbf{u}_J$ of u_J with maximal level J. Likewise we compute a highly accurate approximation $g_{J'}$ to $F \circ u_J$ for some $J' \geq J$. u_J and $g_{J'}$ will serve as fully known reference quantities. The test routine reads as follows:

- For a sequence of decreasing parameters $\varepsilon \in \{\varepsilon_i, i = 1, 2, \ldots\}$ determine approximations $v = u^\varepsilon$ to u_J by thresholding the wavelet coefficients in $\mathbf{u}_J$ yielding $\mathbf{u}^\varepsilon = \mathbf{u}(\Lambda^\varepsilon)$ to $\mathbf{u}_J$.
- Construct the prediction set $\mathcal{T}^\varepsilon = \mathcal{T}(\Lambda^\varepsilon)$ for g based on Λ^ε according to the methods described in [17, 9].
- Execute RECOVER and validate the results by comparing the reference $\mathbf{g}(\mathcal{T}^\varepsilon)$ with $\mathbf{g}^R = \mathbf{g}^R(\mathcal{T}^\varepsilon)$ and with $\mathbf{g}_t^R$ which is obtained by the top-to-bottom thresholding on the fly as described at the end of Section 3.2.

All computation were carried out on an *AMD Athlon(Tm)XP 2000+* processor with 1.26GHz and 512MB of memory, and we always used Cardinal B-spline wavelet bases. In all cases, CPU times scaled linearly with the cardinality, see [2] for more details.

6.1 The L_2 case

We shall treat first approximation in L_2 and consider the example

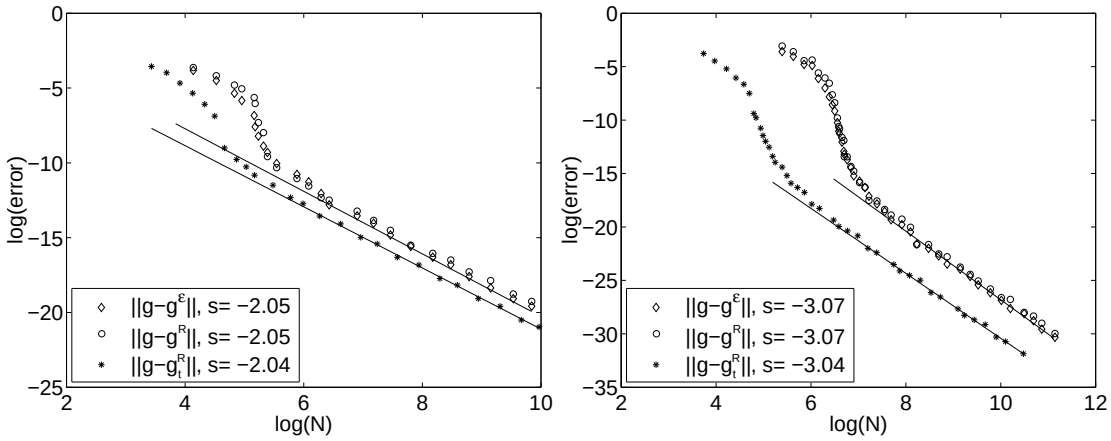
$$u_1 = \begin{cases} exp(x) & x \in (-1,1) \\ 0 & x \in (-2,2) \setminus (-1,1) \end{cases}, \qquad F(\cdot) = 4sin(2\cdot). \tag{6.1}$$

We employ simple Gauss-type formulas according to [1] for the computation of a uniform scaling function approximation to u_1 on level $J = 13$ (32769 coefficients). The composition is approximated on $(-2,2)$ in this case on level $J' = 15$ (131073 coefficients). Due to the jumps of u_1 at $-1, 1$ we expect significant wavelet coefficients with spatial indices near these points in $\mathbf{u}_J$, and therefore also in Λ^ε and $\mathcal{T}^\varepsilon$.

To unify the representation we shall denote the arrays of wavelet coefficients always by $\mathbf{g}, \mathbf{g}(\mathcal{T})$, etc. where it is understood that in the L_2-case the diagonal matrix in (4.16) is just the identity. The task of the recovery scheme is to compute approximations $\mathbf{g}^R$ to $\mathbf{g}$ that are - up to a constant C - as accurate as $\mathbf{g}(\mathcal{T}^\varepsilon)$. In the next table we exemplarily display cardinalities and errors, where we used the biorthogonal cardinal B-spline systems $N(m, m^*)$ according to the spline of order $m = 2$ and its dual $m^* = 2$. Concerning the cardinalities of the involved sets, note that as expected, the prediction set $\#\mathcal{T}^\varepsilon$ is significantly larger then the set of actually relevant coefficients and hence the set resulting from RECOVER (including top-to-bottom thresholding as described in Section 3.2) $\#\Lambda^R$, yet the difference decreases with growing size of $\#\mathcal{T}$. Recall that, according to **Task R**, the reference accuracy is given by $\|g - g^\varepsilon\|$, with $\mathbf{g}^\varepsilon = \mathbf{g}(\mathcal{T}^\varepsilon)$, where here and below we write $\|\cdot\| := \|\cdot\|_{\ell_2}$.

$\#\mathcal{T}^\varepsilon$	104	332	3664	13908	23256	27040
$\|\mathbf{g} - \mathbf{g}^\varepsilon\|$	2.90e-3	7.41e-5	2.93e-7	4.40e-9	1.57e-11	2.60e-13
$\|\mathbf{g} - \mathbf{g}^R\|$	2.90e-3	7.92e-5	3.21e-7	4.51e-9	1.60e-11	2.61e-13
$\|\mathbf{g} - \mathbf{g}_t^R\|$	2.92e-3	7.92e-5	3.3e-7	4.56e-9	1.71e-11	2.61e-13
$\#\Lambda^R$	25	112	1585	8771	15711	18527
$\|\mathbf{g} - \mathbf{g}^R\|/\|\mathbf{g} - \mathbf{g}^\varepsilon\|$	1.0	1.06	1.1	1.02	1.1	1.0
$\#\mathcal{T}^\varepsilon/\#\Lambda^R$	4.16	2.96	2.31	1.58	1.48	1.45

The following graphs show the log-log plots of the approximation $\mathbf{g}^\varepsilon$ to $\mathbf{g}$ in comparison with the result $\mathbf{g}^R$ of the recovery of $\mathbf{g}^\varepsilon$ for the bases according to $m = 2$, $m^* = 2$ and $m = 3$, $m^* = 3$.



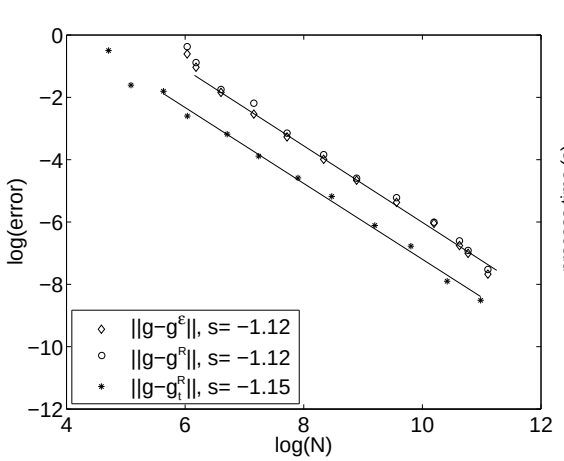
(a) Error rates using $N(2,2)$ (b) Error rates using $N(3,3)$

Figure 6.1: 1D-Tests of RECOVER: $g = F \circ u = 4sin(2u_1)$

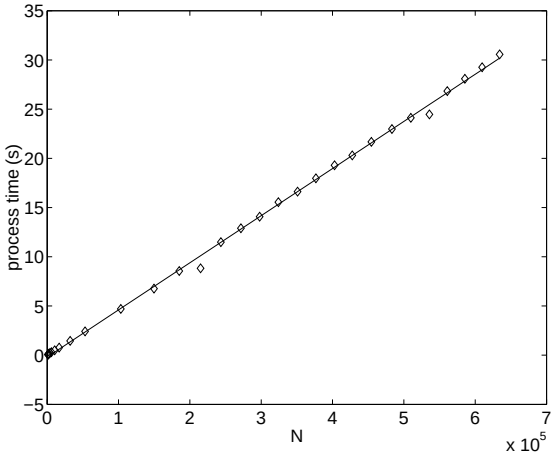
The expected error rate for these tests, i.e. the theoretical slope in the log-log graph is m, which is well reflected by our experiments as soon as the problem reaches a certain size. For smaller problems, we observe a *tune-in behavior*. However, the asymptotic (optimal order) regime is entered at a relatively early stage. CPU times scaled linearly. As a two dimensional test example, we choose u_2 defined by

$$u_2(x_1, x_2) := exp(-100 * (x_1^2 + x_2^2)), \tag{6.2}$$

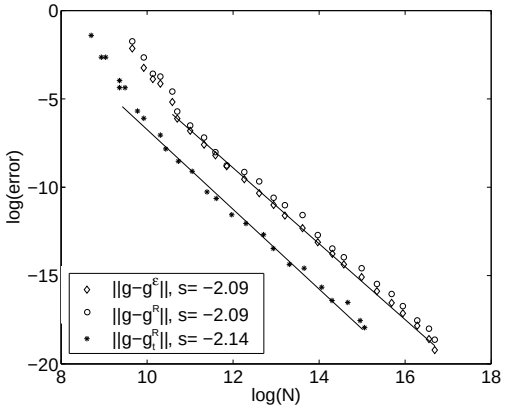
and retain the nonlinearity $F(\cdot) = 4sin(2\cdot)$. The results concerning RECOVER are similar to the one dimensional case including the constant C. The expected asymptotical error rate is $s = m^{-d/2}$, which is matched fairly well. The plots (b), (d) in Figure 6.2 confirm that the CPU-time indeed scales linearly with the size of the problem.



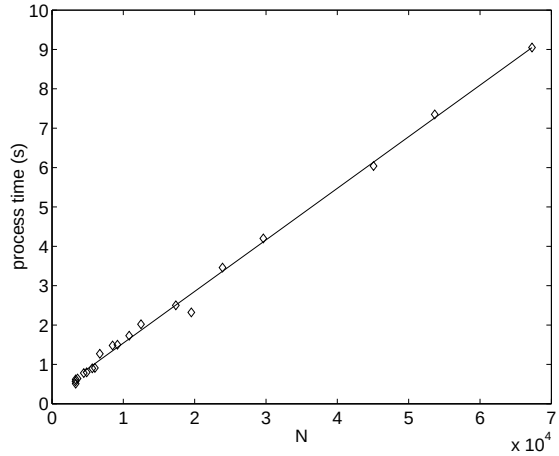
(a) Error rates using $N(2,2)$



(b) CPU time for recovery with $N(2,2)$



(c) Error rates using $N(4,4)$



(d) CPU time for recovery with $N(4,4)$

Figure 6.2: 2D-Tests of RECOVER: $g = F \circ u = 4sin(2u_2)$

6.2 Dual norms

In the following tests, the original function u is subject to a nonlinear mapping (of polynomial type) $F : H^t \to H^{-t}$. As before the examples are designed in such a way that $F \circ u$ is in L_2 and pointwise smooth, yet the H^t- and L_2-norms are very large compared to the (dual) H^{-t}-norm where the recovery is supposed to take place.

6.2.1 Gauss quadrature

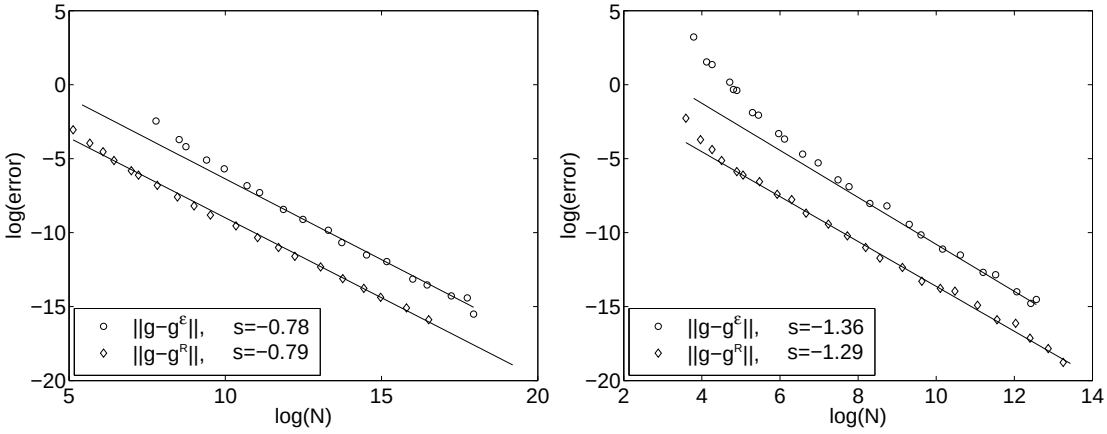
We use again first the Gauss-type formulas from [1] for quadrature. One of our goals here is to illustrate, that this suffices in practical applications involving pointwise smooth or piecewise smooth objects. The test (2D) will be concerned with the nonlinearity $F(\cdot) = \cdot^5$ and $u_4(x_1, x_2) := u_3(x_1)u_3(x_2)$.

The next figures show the result of our tests executed in $(\mathcal{H}, \tilde{\mathcal{H}}) = (H^{0.667}, H^{-0.667})$ in a similar fashion as before and, though dealing with larger constants, also show the expected behavior. Note

that in $\mathbb{R}^d$ according to [9], $F : u \to u^p$ maps H^t into H^{-t} for $t < d/2$ if

$$p < p^* = \frac{d+2t}{d-2t}. \tag{6.3}$$

For $t = 0.667$ and $d = 2$ respectively, $F(\cdot) = \cdot^5$ is therefore close to the limit case. These choices have no practical meaning but are merely to test the dual norm case in a regime where the nonlinear mapping affects the norms in a visible way. The size of the various norms are record as follows $\|u_4\|_{L_2} = 0.501$, $\|u_4\|_{H^{0.667}} = 10.548$ and $\|F(u_4)\|_{L_2} = 57.395$, $\|F(u_4)\|_{H^{-0.667}} = 0.399$.



(a) Error rates using $N(2,2)$

(b) Error rates using $N(3,3)$

Figure 6.3: 2D-Tests of RECOVER: $g = F \circ u = (u_4)^5, (\mathcal{H}, \tilde{\mathcal{H}}) = (H^{0.667}, H^{-0.667})$.

6.2.2 Least squares quadrature

Next we will be concerned with the pointwise smooth but oscillatory function

$$u_5(x) = (exp(-100 * (x - 0.5)) * sin(500x))^3,$$

which is to be recovered for test reasons in H^{-1}. Concerning the H^{-1} norm we have $H^{-1} \approx 0.0877$, again determined by the weighted sum of a (finite) reference wavelet approximation obtained by means of a scaling function representation on a uniform grid (level 15). In order to validate the results, the inner products for this reference are computed *exactly* by means of automatic, symbolic integration using MAPLE 9.0.

For the above setting, we will compare the recovery scheme using two different quadratures, namely the Gauss-quadrature by [1] and the following quadrature method which is a simplification of (5.12). Given an overall target accuracy $\varepsilon > 0$, for each level j do:

(i) Let r=1.

(ii) Determine $\mathbf{q}_j^r$ as described in (5.9) and compute $\mathbf{d}_j^r(g) := \mathbf{G}_{j,\tilde{\Theta}} \cdots \mathbf{G}_{j+r-1,\tilde{\Psi}} \mathbf{q}_{j+r}^r$.

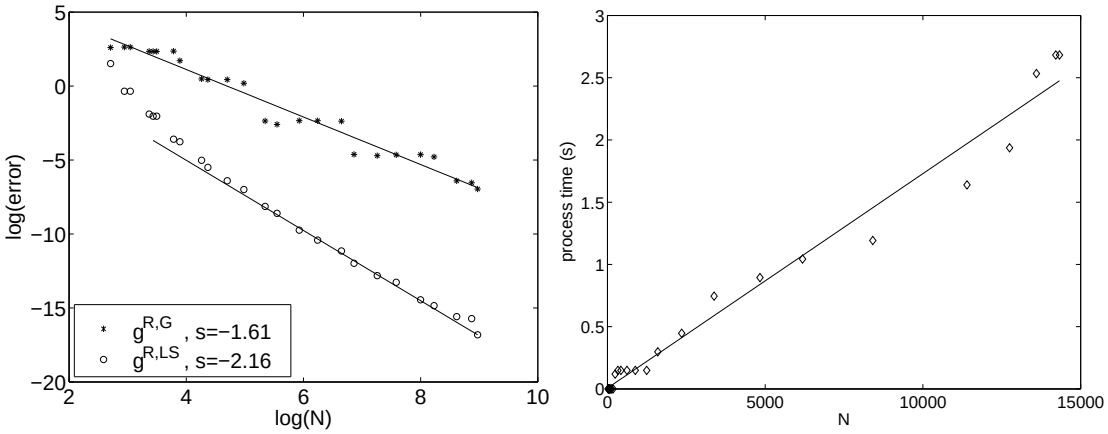
(iii) If

$$(\sum_{l=j+r}^{j+r'-1} \omega_j^2 \|\mathbf{G}_{l,\tilde{\Theta}} \mathbf{G}_{l+1,\tilde{\Phi}} \cdots \mathbf{G}_{j+r-1,\tilde{\Phi}} \mathbf{q}_{j+r'}\|_{\ell_2}^2)^{1/2} > \varepsilon(\#\Lambda_j/\#\Lambda). \tag{6.4}$$

set $r \to r+1$ and go to i) else accept $\mathbf{q}_j^r$, cf. (5.13). Here we take $r' \in \{r+1, r+2\}$.

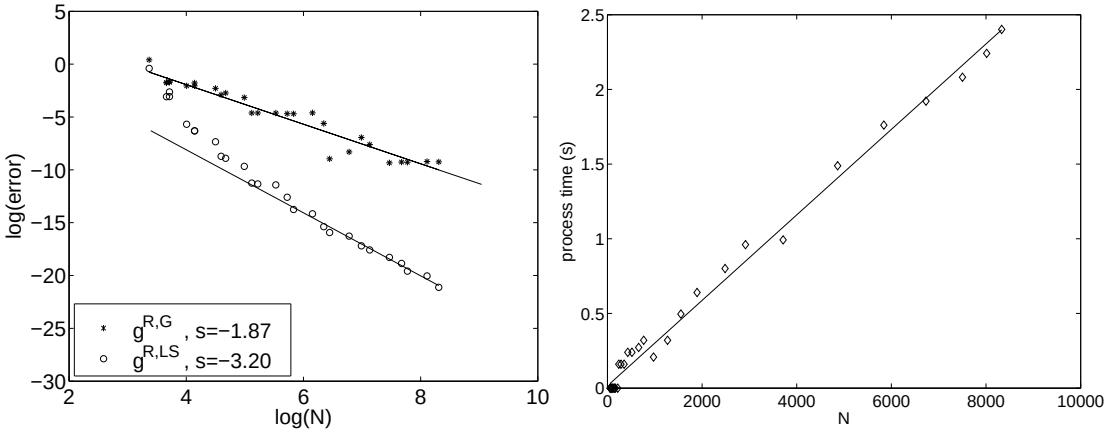
The following graphs show the results of our computations with $N(2,2)$ and $N(3,3)$. We used Gauss quadratures with two points, the number of sampling points for the least-squares formula is $|Y(N(2,2))| = |Y(N(3,3))| = 4$. The observed average refinement parameter is $r = 2$.

Note, that the error of the recovery scheme using the Gauss quadrature, labeled $(F \circ u)^{R,G}$ in the plot, does not show the desired behavior. It stagnates from time to time and even though we can observe an overall error decay, the rate is certainly insufficient. The reason is, that the error stemming from the Gaussian quadrature dominates the error of the recovery scheme. On the other hand, the performance of the recovery scheme using the least-squares method described above, labeled $(F \circ u)^{R,LS}$ is satisfactory as (6.4) ensures, that the quadrature error does not dominate. On the right hand side, we displayed the CPU-times referring to to *least-squares* computation. More than 70% of the CPU time is consumed by checking the error criterion (6.4), namely to apply $\mathbf{G}_{j,\tilde{\Theta}}$ and to compute the norm in (6.4), yet CPU time again scales linearly with N.



(a) Error rates using $N(2,2)$

(b) CPU time for recovery with $N(2,2)$



(c) Error rates using $N(3,3)$

(d) CPU time for recovery with $N(3,3)$

Figure 6.4: 1D-Tests of the recovery using least-squares quadrature, $u = u_5$, $F \circ u = (u_5)^3$

6.3 Conclusions

We have described and analyzed a new scheme for the fast computation of sparse arrays of wavelet coefficients. The design of the algorithm is tailored to the needs of recent adaptive wavelet concepts addressing the main obstructions encountered in this context. First numerical tests, dealing with compositions as simple representers of nonlinear operators, are presented to illustrate the quantitative behavior of the scheme, in particular with regard to requirements on the quadrature routines.

References

[1] A. Barinka, T. Barsch, S. Dahlke, M. Konik, Some Remarks on Quadrature Formulas for Refinable Functions and Wavelets, ZAMM 81 (2001) 12, 839–855.

[2] A. Barinka, W. Dahmen, R. Schneider, Fast computation of adaptive wavelet expansions (extended version), IGPM Report #244, RWTH Aachen, July 2004.

[3] G. Beylkin, J.M. Keiser, An adaptive pseudo-wavelet approach for solving nonlinear partial differential equations *Multiscale Wavelet Methods for PDE's* W. Dahmen, A.J. Lurdila, P. Oswald (eds.), Academic Press,1997, 137-197.

[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.

[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, IGPM Report # 221, RWTH Aachen, July 2002. SIAM J. Numer. Anal., 2003.

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

[10] A. Cohen, W. Dahmen, R. DeVore, Adaptive wavelet techniques in Numerical Simulation, to appear in *Encyclopedia of Computational Mechanics*, (R. De Borste, T. Hughes, E. Stein, eds.), 2004.

[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, K. Urban, Adaptive wavelet methods for saddle point problems – Convergence rates, SIAM J. Numer. Anal., 40 (no. 4) (2002), 1230–1262.

[13] W. Dahmen, Multiscale and Wavelet Methods for Operator Equations, C.I.M.E. Lecture Notes, Springer–Verlag, Heidelberg.

[14] W. Dahmen, C.A. Micchelli, Using the refinement equation for evaluating integrals of wavelets, SIAM J. Numer. Anal., 30(1993), 507-537.

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

[16] W. Dahmen and R. Schneider, Wavelets on Manifolds I: Construction and Domain Decomposition, SIAM J. Math. Anal., 31 (1999), 184–230.

[17] W. Dahmen, R. Schneider, Y. Xu, Nonlinear functions of wavelet expansions – Adaptive reconstruction and fast evaluation, Numer. Math., 86 (2000), 49–101.

[18] J. Fröhlich and K. Schneider, A Fast Algorithm for Lacunary Wavelet Bases related to the Solution of PDEs, C.R. Math. Rep. Acad. Sci. Canada XVII (6), pp. 283-286, 1995.

[19] J. Liandrat, Ph. Tchamitchian, On the fast approximation of some nonlinear operators in non regular wavelet space *Advances in Computational Mathematics*, 8 (1998), 179–192. Numer. Math., 86 (2000), 49–101.

[20] Y. Maday, V. Perrier, J.C. Ravel, Adaptivité dynamique sur base d'ondelettes pour l'approximation d'équation aux dérivée partielles *C.R. Acad. Sci. Paris, Sér. I Math.*, 312 (1991), 405–410. Numer. Math., 86 (2000), 49–101.

[21] R. DeVore, Nonlinear Approximation, *Acta Numerica*, 7 (1998), Cambridge University Press, 51–150.

[22] Yuesheng Xu, Qingsong Zou, Adaptive Wavelet Methods for Elliptic Operator Equations with Nonlinear Terms *Advances in Computational Mathematics*, 19(1) (2003), 99–146.