

# Adaptive Wavelet Frame Domain Decomposition Methods for Nonlinear Elliptic Equations

Dominik Lellek

September 15, 2011

## Abstract

In this paper we are concerned with the numerical treatment of nonlinear elliptic boundary value problems. Our method of choice is a domain decomposition strategy. Partially following the lines from [CDD3, Kap, Lui1, SW2], we develop an adaptive additive Schwarz method using wavelet frames. We show that the method converges with an asymptotically optimal rate and support our theoretical results with numerical tests in one and two space dimensions.

**Subject Classification:** 35A35, 35J60, 65N12, 65N55, 65T60

**Keywords:** Nonlinear elliptic boundary value problems, wavelet frames, nonlinear approximation, adaptivity, overlapping domain decomposition, additive Schwarz method

## 1 Introduction

The theory of linear elliptic partial differential equations (PDEs) is well-explored. However, when dealing with real-life-problems in science and technology, realistic models often lead to nonlinear PDEs. Their analytical treatment is much more difficult than in the linear case, see e.g. [AA, BS, Tay] for an introduction. The computational effort for the numerical treatment of these problems is usually high, especially when working on large and irregular domains  $\Omega \subset \mathbb{R}^d$ . One natural way to overcome these difficulties and to reduce its complexity would be to decompose the problem into several smaller ones by a *domain decomposition approach*. This means to split the problem into a series of subproblems on easier subdomains  $\Omega_i$  with  $\Omega = \cup_{i=0}^{m-1} \Omega_i$ , see [QV, TW] for an overview of these methods. The application of domain decomposition methods to nonlinear problems has first been investigated in [Lio], interpreting the equation as a minimization problem. In an abstract setting, domain decomposition methods for linear and nonlinear problems were analyzed in [DH, TE, Xu]. Further investigations were made applying Schwarz methods to the nonlinear problems, see [Bad, HE, Lui1, Lui2, Mar] and the references therein. In these papers, the equation is mostly discretized using finite elements. Furthermore, exact solution of the subproblems is usually assumed.

As an alternative to finite elements, quite recently adaptive methods using wavelet bases for linear elliptic problems have been developed. We refer e.g. to [CDD1, DDHS] for details. These methods can be proved to be convergent and *asymptotically optimal*. By optimality we mean that the method converges with the same rate as a best  $N$ -term-approximation with respect to the given wavelet basis. These optimality results show the principle advantages of using adaptive methods, since non-adaptive

methods can in many cases only achieve the same rate as classical linear approximation schemes, which can be significantly lower especially if the exact solution has singularities. To prove optimality results of that kind, extensive use of the advantages of wavelet bases is made: They can be designed to be smooth and compactly supported, have vanishing moments, characterize function spaces and allow for a sparse representation of a broad range of operators. However, wavelet bases may be hard to construct and ill-conditioned for realistic domains, especially if non-convexities occur. These difficulties can be ameliorated by using wavelet frames instead of bases. A frame is a stable yet redundant generating system. In the wavelet case, such a frame can be constructed by decomposing the domain into easier overlapping subdomains and collecting bases on the subdomains. These bases can be easier to construct and better conditioned if the subdomains are properly chosen. In [DFR, Ste], adaptive wavelet frame methods for linear elliptic problems were developed. With additional measures to control redundancies caused by the frame approach, this Richardson-based method was shown to converge with optimal rate. Later on to further increase efficiency, the wavelet frame idea was combined with Schwarz domain decomposition methods, see [SW2]. This approach lead to significant improvements regarding computational time and sparsity of the solution. Hence, it seems promising to follow this direction.

Adaptive wavelet methods can also be applied to nonlinear problems. Asymptotically optimal methods using wavelet bases were designed in [CDD3, XZ]. In [Kap], the approach from [CDD3] was generalized to frames, and a Richardson-based algorithm was developed and shown to be convergent and asymptotically optimal. Having in mind the significant improvements achieved by Schwarz methods for linear problems, it seems natural to apply this strategy to nonlinear problems as well. However, to fully benefit from the structure of the frame, we need to linearize the subproblems. In [Lui1], it was shown that a simple linearization strategy with exact solvers for the subproblems does indeed lead to a convergent algorithm provided that the nonlinearity is sufficiently small. In this paper, we extend this approach and develop an adaptive implementable domain decomposition method for elliptic PDEs with nonlinearities. We show its convergence for an arbitrary number of subdomains and prove that the rate of convergence is asymptotically optimal. We then confirm the theoretical results with numerical examples in one and two space dimensions.

We organize this paper as follows: In Section 2, we first describe our basic setting and the class of problems we are going to work with. We then introduce the concepts of frames and Riesz bases. Afterwards, we shortly summarize how to construct wavelet frames on a domain from wavelet bases on the interval. Using the wavelet frame, we then discretize the nonlinear elliptic PDE. In Section 3, we collect the main building blocks for our adaptive algorithm. First, we describe an additional tree structure on the set of wavelet indices. With the help of this additional structure, a method for the evaluation of the nonlinear part as in [CDD3, Kap] can then be applied. We also describe the tools we need for evaluating the linear part of the equation and the right-hand side, as well as a coarsening method that is essential to guarantee an optimal balance between accuracy and the degrees of freedom consumed. In Section 4, we develop our algorithm. At first, we show convergence for a non-adaptive method based on the ideas from [Lui1]. Starting from this, we develop an adaptive method and show its convergence. We then prove our main result which says that the convergence rate is asymptotically optimal regarding the degrees of freedom consumed and we analyze the computational effort. In the last section, we test our algorithm in one and two spaces dimensions. We see that the theoretically predicted results can indeed be observed in practice. In particular, for classical test cases such as the semilinear Poisson equation on the L-shaped domain, the expected convergence rates are achieved by our numerical method.

In the remainder of this paper, we often write  $a \lesssim b$  for inequalities  $a \leq C \cdot b$  with a global constant  $C > 0$ . We write  $a \approx b$  if both  $a \lesssim b$  and  $b \lesssim a$  hold.

## 2 Basic setting

### 2.1 Weak formulation

Let  $\Omega$  be a Lipschitz domain in  $\mathbb{R}^d$ . In this paper, we consider equations of the form

$$-\Delta u + G(u) = f \text{ in } \Omega, \quad u = 0 \text{ on } \partial\Omega,$$

where  $G : \mathbb{R} \rightarrow \mathbb{R}$  is a nonlinearity specified later. In order to ensure existence and uniqueness of the solution and to use concepts from functional analysis, we will work with a weak formulation. Hence, we will assume that  $G(u) \in H^{-1}(\Omega)$  for all  $u \in H_0^1(\Omega)$ . For example, if the nonlinearity is of the form  $G(u) = u^s$  with  $s > 1$ , we have  $G(u) \in H^{-1}(\Omega)$  for all  $u \in H_0^1(\Omega)$  if  $d \leq 2 \frac{s+1}{s-1}$ . This is because by Sobolev's embedding theorem, it is  $u^s \in L^{\frac{2d}{s(d-2)}}(\Omega)$ . Using Hölder's inequality, we can ensure that  $u^s v \in L_1(\Omega)$  for all  $v \in H_0^1(\Omega)$  and  $\|u^s v\|_{L_1(\Omega)} \lesssim \|u\|_{H^1(\Omega)} \|v\|_{H^1(\Omega)}$  if  $\frac{s(d-2)}{2d} + \frac{d-2}{2d} \leq 1$ , which yields the above bound for the dimension  $d$ . The following result gives a more general criterion:

**Proposition 2.1** ([CDD2, Proposition 4.1]) *Let  $p \geq 0$  and  $n^* \in \mathbb{N}$  such that*

$$|G^{(n)}(x)| \lesssim (1 + |x|)^{\max\{0, p-n\}}, \quad x \in \mathbb{R}, \quad n = 0, \dots, n^*.$$

*If  $d \leq 2$  or  $0 \leq p \leq \frac{d+2}{d-2}$ , then  $G(u) \in H^{-1}(\Omega)$  for all  $u \in H_0^1(\Omega)$ . If in addition  $n^* > 1$ , we have*

$$\|G(u) - G(v)\|_{H^{-1}(\Omega)} \leq C \|u - v\|_{H^1(\Omega)},$$

*where  $C = C(\max\{\|u\|_{H^1(\Omega)}, \|v\|_{H^1(\Omega)}\})$  is non-decreasing in the argument.*

For a right-hand side  $f \in H^{-1}(\Omega)$  and  $a(u, v) = \int_{\Omega} \nabla u \nabla v$ , the weak formulation is then given by:

$$\text{Find } u \in H_0^1(\Omega) \text{ such that } a(u, v) + G(u)(v) = f(v) \text{ for all } v \in H_0^1(\Omega). \quad (1)$$

In principle, any other symmetric elliptic operator could be used instead of the Laplace operator, since they lead to equivalent norms. However, for simplicity of notation we stick to this model case. Existence and uniqueness of the solution to (1) is non-trivial and depends on the form of the nonlinearity. Using Banach's fix-point Theorem, a short calculation from [Lui1] shows that (1) has exactly one solution if there exists a constant  $c < 1$  such that

$$\|G(u) - G(v)\| \leq c \|u - v\|, \quad (2)$$

where by  $\|\cdot\|$  we denote the *energy norm*  $\|u\| := a(u, u)^{1/2}$ , which is equivalent to  $\|\cdot\|_{H^1(\Omega)}$  on  $H_0^1(\Omega)$ , and the operator norm on  $H^{-1}(\Omega)$  induced by the energy norm. Contraction properties of that kind will later be used to prove convergence of the domain decomposition algorithms.

### 2.2 Frames and Riesz bases

In order to discretize equation (1), we are going to use a *frame* for  $H_0^1(\Omega)$ , which is a redundant generating system that fulfills a stability condition. In general, allowing redundancies to enter the construction leads

to more flexibility so that additional desirable properties can be incorporated into the design. The basic definitions can be formulated for abstract Hilbert spaces, see [Chr] for an overview. Here, we roughly follow the notation and presentation from [Ste]. From now on, let  $\mathcal{H}$  be a separable Hilbert space and  $\mathcal{N}$  a countable index set. We denote the dual space of  $\mathcal{H}$  by  $\mathcal{H}'$  and often write the dual pair as  $\langle \cdot, \cdot \rangle_{\mathcal{H}' \times \mathcal{H}}$  or  $\langle \cdot, \cdot \rangle_{\mathcal{H} \times \mathcal{H}'}$ .

**Definition 2.2** A set  $\mathcal{G} = \{g_n\}_{n \in \mathcal{N}}$  is called a Riesz basis for  $\mathcal{H}$  if  $\overline{\text{span } \mathcal{G}} = \mathcal{H}$  and there exist constants  $A_{\mathcal{G}}, B_{\mathcal{G}} > 0$  such that for all  $\mathbf{c} \in \ell_2(\mathcal{N})$  the inequalities

$$A_{\mathcal{G}} \|\mathbf{c}\|_{\ell_2(\mathcal{N})} \leq \|\mathbf{c}^\top \mathcal{G}\|_{\mathcal{H}} \leq B_{\mathcal{G}} \|\mathbf{c}\|_{\ell_2(\mathcal{N})}$$

hold, where  $\mathbf{c}^\top \mathcal{G}$  is shorthand for  $\sum_{n \in \mathcal{N}} c_n g_n$ . The numbers  $A_{\mathcal{G}}, B_{\mathcal{G}}$  are called Riesz constants.

An equivalent formulation is to say that the mapping  $\mathbf{c} \mapsto \mathbf{c}^\top \mathcal{G}$  is well-defined and an isomorphism from  $\ell_2(\mathcal{N})$  to  $\mathcal{H}$ . One can show (see e.g. [Chr]) that for a Riesz basis  $\mathcal{G} = \{g_n\}_{n \in \mathcal{N}}$  for  $\mathcal{H}$ , there exists a dual Riesz basis  $\tilde{\mathcal{G}} = \{\tilde{g}_n\}_{n \in \mathcal{N}}$  for  $\mathcal{H}'$  that fulfills  $\langle \tilde{g}_n, g_j \rangle_{\mathcal{H}' \times \mathcal{H}} = \delta_{n,j}$ . Hence, for  $g = \sum_{n \in \mathcal{N}} c_n g_n$  we have  $c_j = \langle \sum_{n \in \mathcal{N}} c_n g_n, \tilde{g}_j \rangle_{\mathcal{H} \times \mathcal{H}'} = \langle g, \tilde{g}_j \rangle_{\mathcal{H} \times \mathcal{H}'}$ , i.e. the expansion coefficients can be computed via the dual Riesz basis. In practice, Riesz bases may be hard to construct. An important weaker concept which we are going to use instead is now defined:

**Definition 2.3** A set  $\mathcal{F} = \{f_n\}_{n \in \mathcal{N}}$  is called a frame for  $\mathcal{H}$  if there exist  $A_{\mathcal{F}}, B_{\mathcal{F}} > 0$  such that for all  $f \in \mathcal{H}'$  the inequalities

$$A_{\mathcal{F}} \|f\|_{\mathcal{H}'}^2 \leq \sum_{n \in \mathcal{N}} |\langle f, f_n \rangle_{\mathcal{H}' \times \mathcal{H}}|^2 \leq B_{\mathcal{F}} \|f\|_{\mathcal{H}'}^2$$

hold. The numbers  $A_{\mathcal{F}}, B_{\mathcal{F}}$  are called frame bounds.

Note that we define a frame using the dual Hilbert space for testing as in [Ste], which is more natural in our case. It is also common to take  $f \in \mathcal{H}$  and to replace the dual pair with the inner product  $\langle f, f_n \rangle$  on  $\mathcal{H}$ . However, by identifying  $\mathcal{H}$  with  $\mathcal{H}'$  via the Riesz isomorphism, this is essentially the same definition and the results from [Chr] carry over. A similar characterization as for Riesz bases can also be shown for frames: The set  $\mathcal{F} = \{f_n\}_{n \in \mathcal{N}} \subset \mathcal{H}$  is a frame for  $\mathcal{H}$  if and only if the synthesis operator

$$F^* : \ell_2(\mathcal{N}) \rightarrow \mathcal{H}, \quad \mathbf{c} \mapsto \mathbf{c}^\top \mathcal{F}$$

is well-defined, surjective and bounded. From the equivalent characterizations, it is obvious that every Riesz basis for  $\mathcal{H}$  is a frame for the same space. One can also show that for a frame the analysis operator

$$F : \mathcal{H}' \rightarrow \ell_2(\mathcal{N}), \quad f \mapsto \{f(f_n)\}_{n \in \mathcal{N}}$$

is bounded and injective, see [Chr, Ste]. Therefore, the frame operator  $S := F^* \circ F$  is boundedly invertible and, as  $F^*$  is the dual of  $F$ , self-adjoint. The collection  $S^{-1} \mathcal{F}$  is a frame for  $\mathcal{H}'$ , the canonical dual frame. For  $f \in \mathcal{H}$ , we have

$$f = S S^{-1} f = \sum_{n \in \mathcal{N}} \langle S^{-1} f, f_n \rangle_{\mathcal{H}' \times \mathcal{H}} f_n = \sum_{n \in \mathcal{N}} \langle f, S^{-1} f_n \rangle_{\mathcal{H} \times \mathcal{H}'} f_n. \quad (3)$$

By this formula, we can compute expansion coefficients of an element  $f \in \mathcal{H}$  in the frame. On the other hand, we can reconstruct an element  $g \in \mathcal{H}'$  using the canonical dual frame via

$$g = S^{-1} S g = \sum_{n \in \mathcal{N}} \langle g, f_n \rangle_{\mathcal{H}' \times \mathcal{H}} S^{-1} f_n.$$

The coefficients  $c_n$  in an expansion  $f = \sum_{n \in \mathcal{N}} c_n f_n$  are in general non-unique. There may be other *dual frames*, which are frames  $\tilde{\mathcal{F}} = \{\tilde{f}_n\}_{n \in \mathcal{N}}$  for  $\mathcal{H}'$  that fulfill an analogous formula

$$f = \sum_{n \in \mathcal{N}} \langle \tilde{f}_n, f \rangle_{\mathcal{H}' \times \mathcal{H}} f_n, \quad f \in \mathcal{H}$$

as in (3). In practice, the canonical dual frame is often hardly accessible, therefore one resorts to other dual frames, often called *non-canonical duals*.

## 2.3 Wavelet bases

In this section, we shortly discuss properties and construction principles of wavelet bases for  $L_2([0, 1])$ . For a more extensive introduction to the construction of wavelets by *Multiresolution Analysis (MRA)*, see e.g. [Dau] or [Mey].

In [DKU] and [Pri], spline wavelet Riesz bases  $\Psi = \{\psi_{j,k}, k \in \nabla_j, j \geq j_0 - 1\}$  and  $\tilde{\Psi} = \{\tilde{\psi}_{j,k}, k \in \nabla_j, j \geq j_0 - 1\}$  for  $L_2([0, 1])$  are constructed. The parameter  $j$  usually is a dilation index and called the *level* of the wavelet. The wavelets on level  $j_0 - 1$  are called *generators*, the corresponding indices are called *generator indices*. Given a suitable pair of integers  $l, \tilde{l}$ , the bases are constructed fulfilling the following properties:

- $\Psi$  and  $\tilde{\Psi}$  are biorthogonal, i.e.  $\langle \psi_{j,k}, \tilde{\psi}_{j',k'} \rangle_{L_2([0,1])} = \delta_{j,j'} \delta_{k,k'}$ ,
- the wavelets  $\psi_{j,k}$  are globally  $(l - 2)$ -times continuously differentiable,
- except for the generators, the wavelets  $\psi_{j,k}, j \geq j_0$  have  $\tilde{l}$  vanishing moments,
- their support is compact and decaying with the level by  $\text{diam supp } \psi_{j,k} \lesssim 2^{-j}$ ,
- they characterize Sobolev spaces, i.e. a scaled version  $\Psi_{\mathbf{D}} := \mathbf{D}^{-1} \Psi := \{2^{-sj} \psi_{j,k}\}$  is a Riesz bases for  $H^s([0, 1])$  for parameters  $s > 0$  in a certain range.

Dirichlet boundary conditions can be incorporated into the construction such that bases for  $H_0^s([0, 1])$  are obtained.

With a tensor product approach as in [DS], from these bases one can construct a wavelet basis  $\Psi_{\Lambda} := \{\psi_{\lambda}\}_{\lambda \in \Lambda \square}$  for  $L_2([0, 1]^d)$ . The corresponding indices  $\lambda \in \Lambda$  are of the form  $\lambda = (j, \mathbf{e}, \mathbf{k})$ , where by  $|\lambda| := j$  we denote the level of the wavelet,  $\mathbf{e} \in \{0, 1\}^d$  is referred to as the *type* and the parameter  $\mathbf{k} \in \mathbb{Z}^d$  encodes a multidimensional translation parameter. Again, a scaled version  $\{2^{-|\lambda|s} \psi_{\lambda}\}_{\lambda \in \Lambda \square}$  is a basis for  $H^s([0, 1]^d)$ . Zero boundary conditions carry over to the tensorized wavelets, hence one can also construct wavelet bases for  $H_0^s([0, 1]^d)$ . In our numerical experiments in Section 5, we will resort to the construction from [Pri], which leads to well-conditioned wavelet bases.

## 2.4 Construction principles of wavelet frames

We now give a short overview of the construction of wavelet frames from wavelet bases. First, we decompose the domain  $\Omega \subset \mathbb{R}^d$  into  $m$  overlapping subdomains  $\Omega = \cup_{i=0}^{m-1} \Omega_i$ , where the  $\Omega_i$  usually are isomorphic to the open unit cube  $(0, 1)^d$ . We require the subdomains to be *overlapping* in the sense that  $H_0^1(\Omega)$  can be split in the following way:

**Definition 2.4** A set  $\{\sigma_i\}_{i=0}^{m-1}$  of functions  $\sigma_i : \Omega \rightarrow \mathbb{R}$  is called a partition of unity with respect to  $\Omega = \cup_{i=0}^{m-1} \Omega_i$  if

- (i)  $\text{supp } \sigma_i \subset \overline{\Omega_i}$ ,
- (ii)  $\sum_{i=0}^{m-1} \sigma_i \equiv 1$  in  $\Omega$ ,
- (iii)  $\sigma_i v \in H_0^1(\Omega_i)$  for all  $v \in H_0^1(\Omega)$ ,
- (iv)  $\|\sigma_i v\|_{H^1(\Omega_i)} \lesssim \|v\|_{H^1(\Omega)}$ ,  $v \in H_0^1(\Omega)$ .

Now, if we have frames or Riesz bases for  $H_0^1(\Omega_i)$ , it is sufficient to collect them in order to obtain a frame for  $H_0^1(\Omega)$ :

**Lemma 2.5** ([Wer, Proposition 2.8]) Assume that there exists a partition of unity with respect to  $\Omega = \cup_{i=0}^{m-1} \Omega_i$  and let  $\Psi_i$  be frames or Riesz bases for  $H_0^1(\Omega_i)$ ,  $i = 0, \dots, m-1$ . Set  $\Psi := \cup_{i=0}^{m-1} E_i \Psi_i$ , where  $E_i$  is the zero extension from  $H_0^1(\Omega_i)$  to  $H_0^1(\Omega)$ . Then  $\Psi$  is a frame for  $H_0^1(\Omega)$ .

The index set belonging to the frame is of the form

$$\Lambda = \cup_{i=0}^{m-1} \{i\} \times \Lambda_i = \{(i, j, \mathbf{e}, \mathbf{k}) : (j, \mathbf{e}, \mathbf{k}) \in \Lambda_i\},$$

where  $\Lambda_i$ ,  $i = 0, \dots, m-1$ , are the indices from the Riesz bases on  $\Omega_i$ . Usually, these are transformed versions of the unit cube bases as in Section 2.3, hence  $\Lambda_i = \Lambda^\square$ .

## 2.5 Discretization with wavelet frames

Having at hand a suitable wavelet frame  $\Psi = \{\psi_\lambda\}_{\lambda \in \Lambda}$  for  $H_0^1(\Omega)$  constructed from wavelet Riesz bases  $\Psi_i = \{\psi_\lambda\}_{\lambda \in \Lambda_i}$  for  $H_0^1(\Omega_i)$ ,  $i = 0, \dots, m-1$ , we can now discretize equation (1), see [CDD1], [Ste] and [CDD3], [Kap] for a more detailed discussion in the linear and nonlinear case, respectively. Note that we simplify the notation by taking  $\Psi$  as a frame for  $H_0^1(\Omega)$  instead of  $L_2(\Omega)$ , thus avoiding to carry along the scaling coefficients  $2^{-|\lambda|}$  that switch from  $L_2(\Omega)$  to  $H_0^1(\Omega)$ .

We first define the matrix  $\mathbf{A} := \{a(\psi_\lambda, \psi_\mu)\}_{\lambda, \mu \in \Lambda}$ . By boundedness of  $a(\cdot, \cdot)$ , the infinite-dimensional matrix-vector multiplication  $\mathbf{v} \mapsto \mathbf{A}\mathbf{v}$  is a bounded operator from  $\ell_2(\Lambda)$  to  $\ell_2(\Lambda)$  representing the continuous Laplace operator. For  $v = \mathbf{v}^\top \Psi$ ,  $\mathbf{v} \in \ell_2(\Lambda)$ , we then set  $\mathbf{G}(\mathbf{v}) := \{(G(v))(\psi_\lambda)\}_{\lambda \in \Lambda} \in \ell_2(\Lambda)$ , which is the discrete version of the nonlinearity. The right-hand side will be discretized by  $\mathbf{f} := \{f(\psi_\lambda)\}_{\lambda \in \Lambda} \in \ell_2(\Lambda)$ . Since  $\overline{\text{span } \Psi} = H_0^1(\Omega)$ ,  $\mathbf{u} \in \ell_2(\Lambda)$  is a solution of

$$\mathbf{A}\mathbf{u} + \mathbf{G}(\mathbf{u}) = \mathbf{f} \tag{4}$$

if and only if  $u = \mathbf{u}^\top \Psi$  is a solution of (1). However, if (1) has a unique solution, the solution to (4) is only unique up to elements in  $\ker F^*$ . These redundancies have to be controlled in an algorithm in order to achieve an optimal work/precision-ratio.

By  $\mathbf{A}^{(i,i)} := \{a(\psi_\lambda, \psi_\mu)\}_{\lambda, \mu \in \Lambda_i}$ , we denote the  $i$ -th diagonal block of  $\mathbf{A}$ . Since the  $\Psi_i$  are Riesz-bases and  $a(\cdot, \cdot)$  is  $H_0^1(\Omega)$ -elliptic, the matrix-vector-multiplication with  $\mathbf{A}^{(i,i)}$  is a boundedly invertible operator on  $\ell_2(\Lambda_i)$ . Moreover, the matrices  $\mathbf{A}^{(i,i)}$  are symmetric and positive definite. Hence,  $\|\mathbf{v}\|_{\mathbf{A}^{(i,i)}} := \langle \mathbf{A}^{(i,i)} \mathbf{v}, \mathbf{v} \rangle_{\ell_2(\Lambda_i)}^{1/2}$  is equivalent to  $\|\mathbf{v}\|_{\ell_2(\Lambda_i)}$  on  $\ell_2(\Lambda_i)$ . Since the  $\Psi_i$  are Riesz bases,  $\|\mathbf{v}\|_{\mathbf{A}^{(i,i)}}$  is also equivalent to  $\|\mathbf{v}^\top \Psi_i\|_{H^1(\Omega_i)}$  and  $\|\mathbf{v}^\top \Psi_i\|$ .

In order to be able to develop numerical methods for the discretized equation, we need methods that approximately evaluate the infinite-dimensional matrix-vector-product  $\mathbf{A}\mathbf{v}$ , the discrete nonlinearity  $\mathbf{G}(\mathbf{v})$  and the discrete right-hand side  $\mathbf{f}$ . Furthermore, we require the output to be sparse in a certain sense. For measuring sparsity in the context of wavelet methods for nonlinear equations, a structural tool presented in the next section is needed. The above-mentioned methods will be introduced afterwards.

### 3 Building blocks for the adaptive algorithm

We have so far formulated the discretized version of the elliptic problem (1). In this section we are going to collect the tools we need to deal with the discretized equation. These building blocks appear in all kinds of adaptive wavelet algorithms as in [CDD1, CDD3, DFR, Kap, Ste, SW2]. Roughly speaking, it is in these methods where the adaptivity of the algorithm is established.

#### 3.1 Tree approximation

The most difficult part is to efficiently evaluate the nonlinearity  $\mathbf{G}(\mathbf{v})$  in wavelet coefficients. That is because for an exact calculation of  $(\mathbf{G}(\mathbf{v}))_\lambda = \langle G(\mathbf{v}^\top \Psi), \psi_\lambda \rangle_{H^{-1}(\Omega) \times H_0^1(\Omega)}$ , one would have to compute  $v = \mathbf{v}^\top \Psi$ , apply  $G$  to  $v$  and then evaluate the dual pair. This is costly, and all the steps have to be repeated if the argument  $\mathbf{v}$  is changed. In order to efficiently evaluate the nonlinearity, we need an additional structure on the set of wavelet indices. Following [CDD3] and [Kap], we will assume that the underlying wavelet indices possess a certain tree structure. Hence, we shortly describe the properties of such trees and aggregated trees. The concept follows the same lines as the construction of wavelet frames: First, trees for wavelet bases are defined which are subsequently collected in order to obtain aggregated trees for wavelet frames.

For a wavelet bases  $\{\psi_{j,k}\}$  on  $L_2(\mathbb{R})$ , we can simply define the indices  $(j+1, k)$  and  $(j+1, k+1)$  as the successors of the index  $(j, k)$ . For more general construction of a wavelet basis, a partial order  $\succ$  can be defined on the set  $\Lambda^0 = \{(j, \mathbf{k}), \mathbf{k} \in \nabla_j\}$  using *reference cubes*, see [Kap] for details. Hence, we call an index  $\lambda^0$  *successor* of  $\mu^0$  if  $\lambda^0 \succ \mu^0$ . Alternatively, we say that  $\mu^0$  is a *predecessor* of  $\lambda^0$ . An index set  $T^0 \subset \Lambda^0$  is then called a *(local) tree*, if  $\lambda^0 \in T^0$  implies that all predecessors of  $\lambda^0$  also belong to  $T^0$ . This definition can be extended to a set  $T \subset \Lambda$  of wavelet indices by saying that  $T$  is a *(local) tree*, if the corresponding set  $T^0 = \{(j, \mathbf{k}), (j, \mathbf{e}, \mathbf{k}) \in \Lambda\}$  forms a tree in the above sense.

Once such a tree structure is obtained, we can measure approximation properties with regard to this structure. Hence, we first define  $\Sigma_{N, \mathcal{T}}$  as the set of all  $\mathbf{v} \in \ell_2(\Lambda)$  whose support has tree structure and consists of at most  $N$  entries. By setting

$$\sigma_{N, \mathcal{T}}(\mathbf{v}) := \inf_{\mathbf{w} \in \Sigma_{N, \mathcal{T}}} \|\mathbf{v} - \mathbf{w}\|_{\ell_2(\Lambda)},$$

we obtain the approximation space

$$\mathcal{A}_{\mathcal{T}}^s := \{\mathbf{v} \in \ell_2(\Lambda) : \sigma_{N, \mathcal{T}}(\mathbf{v}) \lesssim N^{-s}\}$$

of vectors that can be approximated with rate  $s$  in tree structure equipped with the quasi-norm

$$\|\mathbf{v}\|_{\mathcal{A}_{\mathcal{T}}^s} := \sup_{N \in \mathbb{N}} N^s \sigma_{N, \mathcal{T}}(\mathbf{v}).$$

In a similar fashion, we can define tree approximation spaces for wavelet frames: A set  $T = \cup_{i=0}^{m-1} \{i\} \times T_i$  with  $T_i \subset \Lambda_i$  is called *aggregated tree*, if all the  $T_i$  have (local) tree structure. As above, we define  $\Sigma_{N, \mathcal{AT}}$  as the set of all  $\mathbf{v} \in \ell_2(\Lambda)$  with  $\#\text{supp } \mathbf{v} \leq N$  such that  $\text{supp } \mathbf{v}$  has aggregated tree structure. Defining  $\sigma_{N, \mathcal{AT}}(\mathbf{v}) := \inf_{\mathbf{w} \in \Sigma_{N, \mathcal{AT}}} \|\mathbf{v} - \mathbf{w}\|_{\ell_2(\Lambda)}$ , we now have the approximation space

$$\mathcal{A}_{\mathcal{AT}}^s := \{\mathbf{v} \in \ell_2(\Lambda) : \sigma_{N, \mathcal{AT}}(\mathbf{v}) \lesssim N^{-s}\}$$

with the analogous quasi-norm  $\|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s} := \sup_{N \in \mathbb{N}} N^s \sigma_{N, \mathcal{AT}}(\mathbf{v})$ .

### 3.2 Adaptive evaluation and coarsening

We are now going to collect the basic building blocks needed to construct an adaptive wavelet algorithm. These tools were mostly developed in [CDD1], then adapted to frames e.g. in [Ste] and extended to the concept of trees and aggregated trees in [CDD3] and [Kap], respectively.

In order to achieve an optimal work/accuracy-ratio, it is necessary to remove very small entries from the iterates. Hence, we would like to have a method that for a given vector  $\mathbf{v} \in \ell_2(\Lambda)$  and  $\varepsilon > 0$  determines  $\mathbf{w}_\varepsilon \in \ell_2(\Lambda)$  such that  $\|\mathbf{v} - \mathbf{w}_\varepsilon\|_{\ell_2(\Lambda)} \leq \varepsilon$ ,  $\text{supp } \mathbf{w}_\varepsilon$  has tree structure and is minimal, i.e.  $\#\text{supp } \mathbf{w}_\varepsilon = \#T(\varepsilon, \mathbf{v})$  with

$$\#T(\varepsilon, \mathbf{v}) := \{\min \#\text{supp } \mathbf{w} : \|\mathbf{v} - \mathbf{w}\|_{\ell_2(\Lambda)} \leq \varepsilon \text{ and } \text{supp } \mathbf{w} \text{ has aggregated tree structure}\}.$$

An algorithm of that kind, however, would require too much computational effort because one would have to sort all the entries of  $\mathbf{v}$  by their size. Therefore, we only require that a method

$$\mathbf{ATCOARSE}[\mathbf{v}, \varepsilon] \mapsto \mathbf{w}_\varepsilon$$

exists with  $\|\mathbf{v} - \mathbf{w}_\varepsilon\|_{\ell_2(\Lambda)} \leq \varepsilon$  such that  $\text{supp } \mathbf{w}_\varepsilon$  is an aggregated tree and there exists a global constant  $C^* > 1$  such that  $\#\text{supp } \mathbf{w}_\varepsilon \leq C^* \#T(\frac{\varepsilon}{C^*})$ . Based on the ideas from [CDD3], such a method was developed in [Kap] and it was shown that the following properties hold, which will be essential to the proof of optimality in Section 4.2:

**Lemma 3.1** ([Kap, Lemma 4.1]) *Let  $\mathbf{v} \in \mathcal{A}_{\mathcal{AT}}^s$ ,  $\mathbf{w} \in \ell_2(\Lambda)$  such that  $\#\text{supp } \mathbf{w} < \infty$  and  $\|\mathbf{v} - \mathbf{w}\|_{\ell_2(\Lambda)} \leq \frac{\varepsilon}{2C^*+1}$ , then for  $\mathbf{w}_\varepsilon := \mathbf{ATCOARSE}[\mathbf{v}, \frac{2C^*\varepsilon}{2C^*+1}]$  it holds that*

$$\begin{aligned} \|\mathbf{v} - \mathbf{w}_\varepsilon\|_{\ell_2(\Lambda)} &\leq \varepsilon, \\ \#\text{supp } \mathbf{w}_\varepsilon &\lesssim \varepsilon^{-1/s} \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s}, \\ \|\mathbf{w}_\varepsilon\|_{\mathcal{A}_{\mathcal{AT}}^s} &\lesssim \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}. \end{aligned}$$

*The number of operations required to compute  $\mathbf{w}_\varepsilon$  is bounded by a constant multiple of  $\#\text{supp } \mathbf{w}$ . The constants only depend on  $s$  when  $s \rightarrow 0$ , and on  $C^*$ .*

The next building block we need is the efficient approximate evaluation of the right-hand side  $\mathbf{f}$ , i.e. we assume that for  $\mathbf{f} \in \mathcal{A}_{\mathcal{AT}}^s$ , we have a method

$$\mathbf{ATRHS}[\mathbf{f}, \varepsilon] \mapsto \mathbf{f}_\varepsilon$$

with  $\|\mathbf{f} - \mathbf{f}_\varepsilon\|_{\ell_2(\Lambda)} \leq \varepsilon$ ,  $\#\text{supp } \mathbf{f}_\varepsilon \lesssim \varepsilon^{-1/s} \|\mathbf{f}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s}$ ,  $\|\mathbf{f}_\varepsilon\|_{\mathcal{A}_{\mathcal{AT}}^s} \lesssim \|\mathbf{f}\|_{\mathcal{A}_{\mathcal{AT}}^s}$  that requires at most a constant multiple of  $\varepsilon^{-1/s} \|\mathbf{f}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s}$  operations. A method like this is usually constructed by evaluating the right-hand side as it is done in adaptive methods for linear problems and then applying **ATCOARSE** to the



result to obtain a vector in aggregated tree structure.

In order to evaluate the linear part of the equation and to implement a further projection step into the algorithm, we require an approximate infinite-dimensional matrix-vector multiplication

$$\mathbf{ATAPPLY}[\mathbf{A}, \mathbf{v}, \varepsilon] \mapsto \mathbf{w}_\varepsilon$$

that has the following properties:

$$\|\mathbf{A}\mathbf{v} - \mathbf{w}_\varepsilon\|_{\ell_2(\Lambda)} \leq \varepsilon, \quad (5)$$

$$\#\text{supp } \mathbf{w}_\varepsilon \lesssim \varepsilon^{-1/s} \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s}, \quad (6)$$

$$\|\mathbf{w}_\varepsilon\|_{\mathcal{A}_{\mathcal{AT}}^s} \lesssim \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}, \quad (7)$$

and needs not more than a constant multiple of  $\varepsilon^{-1/s} \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + \#\text{supp } \mathbf{v}$  operations. A sufficient criterion for such a method to exist is fulfilled if the matrix can be well approximated by sparse matrices. Here, the concept of *compressibility* comes into play:

**Definition 3.2** Let  $s^* > 0$ . A matrix  $\mathbf{A} : \ell_2(\Lambda) \rightarrow \ell_2(\Lambda)$  is called  $s^*$ -compressible, if there exist numbers  $\alpha_j, C_j$  and matrices  $\mathbf{A}_j, j \in \mathbb{N}$ , such that  $\mathbf{A}_j$  has at most  $a_j 2^j$  non-zero entries in each column,  $\|\mathbf{A} - \mathbf{A}_j\| \leq C_j, \sum_j \alpha_j < \infty$  and  $\sum_j C_j 2^{sj} < \infty$  for all  $s \in (0, s^*)$ .

Compressibility properties in wavelet frame coordinates can be shown for a broad range of elliptic operators, including the Laplace operator, see [CDD1, SW] for details. The proofs make extensive use of the nice properties of wavelet frames such as their local support, smoothness and vanishing moments. The results from [Ste] and [Kap, Remark 4.2] show that a method **ATAPPLY** with the above properties exists for many matrices and can be practically implemented:

**Lemma 3.3** Let  $\mathbf{A}$  be  $s^*$ -compressible,  $0 < s < s^*$ , and bounded on  $\mathcal{A}_{\mathcal{AT}}^s$ . Then, a method **ATAPPLY** fulfilling properties (5), (6), (7) exists and the evaluation of  $\mathbf{w}_\varepsilon = \mathbf{ATAPPLY}[\mathbf{A}, \mathbf{v}, \varepsilon]$  requires at most  $\mathcal{O}(\varepsilon^{-1/s} \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + \#\text{supp } \mathbf{v})$  operations.

To apply Lemma 3.3, boundedness of  $\mathbf{A}$  on the approximation space  $\mathcal{A}_{\mathcal{AT}}^s$  still has to be shown. However, this can be guaranteed if the matrices decay fast enough from the main diagonal, see Lemma 4.2 in [Kap].

The last building block is the evaluation of the discretized nonlinearity. For this purpose, we assume that there exists a method

$$\mathbf{EVAL}[\mathbf{G}, \mathbf{v}, \varepsilon] \mapsto \mathbf{w}_\varepsilon$$

such that  $\|\mathbf{G}(\mathbf{v}) - \mathbf{w}_\varepsilon\|_{\ell_2(\Lambda)} \leq \varepsilon, \#\text{supp } \mathbf{w}_\varepsilon \lesssim \varepsilon^{-1/s} (\|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1)$  and  $\|\mathbf{w}_\varepsilon\|_{\mathcal{A}_{\mathcal{AT}}^s} \lesssim \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s} + 1$ . For input arguments in aggregated tree structure, we assume that **EVAL** requires at most a constant multiple of  $\varepsilon^{-1/s} (\|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1) + \#\text{supp } \mathbf{v}$  operations. The constant terms stem from the indices on the lowest level, see [CDD3] and [Kap], that are required for building up the tree structure in which the nonlinearity is evaluated. We will not go into the details of the construction of this method here. In [Kap], for a range of nonlinearities a routine **EVAL** of that kind has been constructed for wavelet frames based on the construction for wavelet bases in [CDD3]. This method will be applied in our numerical tests. It is based on a two-step approach: First, using wavelet properties such as vanishing moments, the support of the result is predicted. On the predicted support, the wavelet coefficients are then approximately evaluated.

We have now collected the building blocks for the adaptive algorithm which will be constructed in the next section.

## 4 Additive Schwarz algorithm

We are now ready to formulate an additive Schwarz algorithm for our problem (1). In order to construct an adaptive and implementable algorithm, we first define a non-adaptive version and show its convergence. In a second step, we introduce coarsening and projection steps and propose an adaptive solver for the subproblems. We then show that the algorithm is asymptotically optimal.

### 4.1 Basic algorithm

We assume again that (2) holds and that we find a *relaxation parameter*  $\omega > 0$  with

$$\rho := \|I - \omega(P_0 + \dots + P_{m-1})\| + m\omega c < 1. \quad (8)$$

Here, we denote by  $P_k$ ,  $k = 0, \dots, m-1$ , the  $a(\cdot, \cdot)$ -orthogonal projector on  $H_0^1(\Omega_k)$ . It is known that  $\|I - \omega(P_0 + \dots + P_{m-1})\| < 1$  for  $\omega \in (0, 2/\lambda_{\max}(\sum_{k=0}^{m-1} P_k))$ , where  $\lambda_{\max}(\sum_{k=0}^{m-1} P_k)$  is the largest eigenvalue of  $\sum_{k=0}^{m-1} P_k$ , see [Wer]. Hence, inequality (8) basically says that we demand the nonlinearity to be sufficiently small. By  $u \in H_0^1(\Omega)$ , we denote the exact solution to (1) and by  $A_k$  we denote the weak (negative) Laplace operator

$$\langle A_k(u), v \rangle_{H^{-1}(\Omega_i) \times H_0^1(\Omega_i)} = a(u, v) = \int_{\Omega_i} \nabla u \nabla v$$

considered as a mapping from  $H_0^1(\Omega_i)$  to  $H^{-1}(\Omega_i)$ , which is again boundedly invertible. As the index set  $\Lambda$  is of the form  $\Lambda = \cup_{i=0}^{m-1} \{i\} \times \Lambda_i$ , we will refer to the restriction of  $\mathbf{w} \in \ell_2(\Lambda)$  to the indices belonging to  $\Lambda_i$  as  $\mathbf{w}_{|\Lambda_i}$ . By partially following the lines from [DW] and [Lui1], we now define the following algorithm:

---

#### Algorithm 1 AddSchw1[ $\varepsilon$ ]

---

Let  $\tilde{\rho} := \frac{1}{2}(1 + \rho)$ ,  $u^{(0)} := 0$ ,  $M \geq \|u\|$ ,  $\varepsilon_n := \tilde{\rho}^n M$ .  
 $n := 0$   
**while**  $\varepsilon_n > \varepsilon$  **do**  
  **for**  $k = 0, \dots, m-1$  **do**  
    Determine  $\tilde{e}_k^{(n)}$  as an approximation to  $e_k^{(n)}$  from  
     $a(e_k^{(n)}, v) = -a(u^{(n)}, v) + f(v) - G(u^{(n)})(v)$ ,  $v \in H_0^1(\Omega_k)$   
    with tolerance  $\|e_k^{(n)} - \tilde{e}_k^{(n)}\| \leq \frac{1-\rho}{2m\omega} \tilde{\rho}^n M$ .  
  **end for**  
 $u^{(n+1)} := u^{(n)} + \omega \sum_{k=0}^{m-1} \tilde{e}_k^{(n)}$   
 $n := n + 1$   
**end while**  
 $u_\varepsilon := u^{(n)}$

---

Note that the  $m$  subproblems in each iteration are independent so they can be solved in parallel. Having chosen geometrically decreasing tolerances, we can ensure that the convergence result from [Lui1] can be extended to this algorithm:

**Proposition 4.1** *Let  $\varepsilon > 0$  and assume that (8) is valid. Then, for the iterates  $u^{(n)}$  from Algorithm 1 it holds that*

$$\|u^{(n)} - u\| \leq \tilde{\rho}^n M = \varepsilon_n,$$

hence  $\|u_\varepsilon - u\| \leq \varepsilon$ .

*Proof.* We show the statement by induction over  $n$ . The case  $n = 0$  is clear. Let us assume the statement is valid for some  $n \in \mathbb{N}$ . For all  $v \in H_0^1(\Omega_k)$  it holds that

$$\begin{aligned} a(e_k^{(n)}, v) &= -a(u^{(n)}, v) + f(v) - G(u^{(n)})(v) \\ &= -a(u^{(n)}, v) + a(u, v) + G(u)(v) - G(u^{(n)})(v) \\ &= a(u - u^{(n)}, v) + G(u)(v) - G(u^{(n)})(v). \end{aligned}$$

Hence  $e_k^{(n)} = P_k(u - u^{(n)}) + A_k^{-1}(G(u) - G(u^{(n)}))$ . We set  $\bar{u}^{(n+1)} := u^{(n)} + \omega \sum_{k=0}^{m-1} e_k^{(n)}$  and obtain

$$\begin{aligned} \|u - \bar{u}^{(n+1)}\| &= \|u - u^{(n)} - \omega \sum_{k=0}^{m-1} e_k^{(n)}\| \\ &= \|u - u^{(n)} - \omega \sum_{k=0}^{m-1} (P_k(u - u^{(n)}) + A_k^{-1}(G(u) - G(u^{(n)})))\| \\ &\leq \|(I - \omega \sum_{k=0}^{m-1} P_k)\| \|u - u^{(n)}\| + m\omega c \|u - u^{(n)}\| \\ &= \rho \|u - u^{(n)}\|. \end{aligned}$$

By induction it now follows that

$$\begin{aligned} \|u^{(n+1)} - u\| &\leq \|u - \bar{u}^{(n+1)}\| + \|\bar{u}^{(n+1)} - u^{(n+1)}\| \\ &\leq \rho \|u - u^{(n)}\| + \omega \left\| \sum_{k=0}^{m-1} (\tilde{e}_k^{(n)} - e_k^{(n)}) \right\| \\ &\leq \rho \tilde{\rho}^n M + \omega m \frac{1 - \rho}{2m\omega} \tilde{\rho}^n M \\ &= \tilde{\rho}^{n+1} M. \end{aligned}$$

□

Since  $\tilde{\rho} < 1$ , the iterates converge towards the solution  $u$ . In fact, a slightly more moderate assumption on the nonlinearity is sufficient:

**Remark 4.2** *For the proof of convergence, the contraction property (2) is only used for the difference  $u^{(n)} - u$ . Since  $u^{(0)} = 0$  and the series  $u^{(n)}$  converges geometrically towards  $u$ , we have  $\|u^{(n)}\| \leq 2\|u\|$ . Hence, if we know that the solution  $u$  is unique, we only need the contraction property in a  $2\|u\|$ -ball around 0 measured in the energy norm.*

## 4.2 Adaptive algorithm

In order to design an asymptotically optimal algorithm, it is necessary to keep possible redundancies under control. These redundancies can be caused by the frame approach because the representation of an element  $v \in H_0^1(\Omega)$  by a vector  $\mathbf{v} \in \ell_2(\Lambda)$  with  $v = \mathbf{v}^\top \Psi$  is only unique up to elements in the kernel of the synthesis operator  $F^*$  belonging to the frame  $\Psi$ . When approximately solving the subproblems, these elements in  $\ker F^*$  can occur and cause unnecessary work in the following iterations. Hence, as in

[Ste] and [Wer], we are going to introduce a projector  $\mathbf{P} : \ell_2(\Lambda) \rightarrow \ell_2(\Lambda)$  with  $\ker \mathbf{P} = \ker F^*$  such that  $\|\mathbf{P}\mathbf{v}\|_{\ell_2(\Lambda)} \approx \|\mathbf{v}^\top \Psi\|_{H^1(\Omega)}$ . Applying this projector to the current discrete iterate removes the undesired redundancies. The projector can be constructed with the help of a partition of unity  $\{\sigma_i\}_{i=0}^{m-1}$  as in Definition 2.4. Let  $v = \mathbf{v}^\top \Psi$ . Then we take  $v_i := \sigma_i v \in H_0^1(\Omega_i)$  and denote by  $\mathbf{w}_i$  the unique expansion coefficients of  $v_i$  in the Riesz basis  $\Psi_i$  for  $H_0^1(\Omega_i)$ . We then set

$$\mathbf{P}\mathbf{v} := (\mathbf{w}_0, \dots, \mathbf{w}_{m-1}).$$

With the help of the dual Riesz bases on the patches, we can write down the coefficients in the matrix representation of  $\mathbf{P}$  explicitly: Let  $\lambda \in \Lambda$  belong to patch  $i$ ,  $\lambda \in \{i\} \times \Lambda_i$ , and  $\tilde{\psi}_\lambda$  be taken from the dual Riesz basis on  $\Lambda_i$ . Then we have

$$(\mathbf{P})_{\lambda,\mu} = (\langle \tilde{\psi}_\lambda, \sigma_i \psi_\mu \rangle_{H^{-1}(\Omega_i) \times H_0^1(\Omega_i)})$$

because of

$$(\mathbf{P}\mathbf{v})_\lambda = \sum_{\mu \in \Lambda} \langle \tilde{\psi}_\lambda, \sigma_i \psi_\mu \rangle_{H^{-1}(\Omega_i) \times H_0^1(\Omega_i)} \mathbf{v}_\mu = \langle \tilde{\psi}_\lambda, \sigma_i v \rangle_{H^{-1}(\Omega_i) \times H_0^1(\Omega_i)}$$

and by definition of the dual Riesz basis. It is obvious that  $\mathbf{P}^2 = \mathbf{P}$ , and it holds that  $\ker \mathbf{P} = \ker F^*$ . This is because by the Riesz basis property of the  $\Lambda_i$ , we have  $\mathbf{w}_i = 0$  if and only if  $\sigma_i v = 0$ , and all the  $\sigma_i v$  being zero is equivalent to  $v = 0$ . From  $\ker \mathbf{P} = \ker F^*$  it follows that  $F^* \mathbf{P}\mathbf{u} = F^* \mathbf{u}$  for  $\mathbf{u} \in \ell_2(\Lambda)$ . Hence, the exact application of  $\mathbf{P}$  only changes the coefficients in the representation of  $u = \mathbf{u}^\top \Psi$ , but not  $u$  itself. This projector has the following useful properties:

**Lemma 4.3** *For  $v \in H_0^1(\Omega)$ ,  $v = \mathbf{v}^\top \Psi$ ,  $\mathbf{v} \in \ell_2(\Lambda)$  it holds that*

$$\begin{aligned} \|\mathbf{P}\mathbf{v}\|_{\ell_2(\Lambda)} &\approx \|v\|, \\ \|\mathbf{P}\mathbf{v}\|_{\ell_2(\Lambda)} &\lesssim \|\mathbf{v}\|_{\ell_2(\Lambda)}. \end{aligned}$$

*Proof.* By the Riesz basis property and Definition 2.4, we have

$$\|\mathbf{P}\mathbf{v}\|_{\ell_2(\Lambda)} \approx \sum_{i=0}^{m-1} \|(\mathbf{P}\mathbf{v})|_{\Lambda_i}\|_{\ell_2(\Lambda_i)} \approx \sum_{i=0}^{m-1} \|\sigma_i v\|_{H^1(\Omega_i)} \approx \|v\|_{H^1(\Omega)}$$

and  $\|v\|_{H^1(\Omega)} \lesssim \|\mathbf{v}\|_{\ell_2(\Lambda)}$  and  $\|v\|_{H^1(\Omega)} \approx \|v\|$ . □

In order to define an adaptive version of Algorithm 1, we need to fix some constants. According to Lemma 4.3, there exists a constant  $L > 0$  such that  $\|\mathbf{P}\mathbf{v}\|_{\ell_2(\Lambda)} \leq L\|v\|$  for all  $\mathbf{v} \in \ell_2(\Lambda)$ . Since the synthesis operator  $F^*$  from Section 2.2 is bounded and  $\|v\|_{H^1(\Omega)} \approx \|v\|$  for  $v \in H_0^1(\Omega)$ , there is another constant  $K > 0$  such that  $\|\mathbf{v}^\top \Psi\| \leq K\|\mathbf{v}\|_{\ell_2(\Lambda)}$  for all  $\mathbf{v} \in \ell_2(\Lambda)$ . We are now ready to formulate our algorithm. In this version, we introduce coarsening and the application of the projector  $\mathbf{P}$ . We will later specify how the local subproblems are to be solved.

---

**Algorithm 2** AddSchw2 [ $\varepsilon$ ]

---

Let  $\tilde{\rho} := \frac{1}{2}(1 + \rho)$ ,  $u^{(0)} := 0$ ,  $M \geq \|u\|$ .

Let  $C^*$  be the constant in Lemma 3.1.

Let  $l^* \in \mathbb{N}$  be minimal with  $\tilde{\rho}^{l^*} \leq \frac{1}{2KL} \frac{1}{2C^*+1} \tilde{\rho}$ .

Let  $\varepsilon_n := \tilde{\rho}^n M$ ,  $n \in \mathbb{N}$ .

**while**  $\varepsilon_n > \varepsilon$  **do**

$v^{(n,0)} := u^{(n)}$

**for**  $l = 0, \dots, l^* - 1$  **do**

**for**  $k = 0, \dots, m - 1$  **do**

            Determine  $\tilde{e}_k^{(n,l)}$  as an approximation to  $e_k^{(n,l)}$  from

$a(e_k^{(n,l)}, v) = -a(v^{(n,l)}, v) + f(v) - G(v^{(n,l)})(v)$ ,  $v \in H_0^1(\Omega_k)$

            with tolerance  $\|e_k^{(n,l)} - \tilde{e}_k^{(n,l)}\| \leq \frac{1-\rho}{2m\omega} \varepsilon_n \tilde{\rho}^l$ .

**end for**

$v^{(n,l+1)} := v^{(n,l)} + \omega \sum_{k=0}^{m-1} \tilde{e}_k^{(n,l)}$ ,  $\mathbf{v}^{(n,l+1)} = (\mathbf{v}^{(n,l+1)})^\top \Psi$

**end for**

$\tilde{\mathbf{u}}^{(n+1)} := \mathbf{ATAPPLY}[\mathbf{P}, \mathbf{v}^{(n,l^*)}, \frac{1}{2K} \frac{1}{2C^*+1} \varepsilon_{n+1}]$

$\mathbf{u}^{(n+1)} := \mathbf{ATCOARSE}[\tilde{\mathbf{u}}^{(n+1)}, \frac{1}{K} \frac{2C^*}{2C^*+1} \varepsilon_{n+1}]$ ,  $u^{(n+1)} = (\mathbf{u}^{(n+1)})^\top \Psi$

$n := n + 1$

**end while**

$\mathbf{u}_\varepsilon := \mathbf{u}^{(n)}$ ,  $u_\varepsilon := \mathbf{u}_\varepsilon^\top \Psi$

---

Note that although we write the local subproblems in the continuous formulation, they are of course solved on the discrete level so that the expansion coefficients of the iterates are available at no extra cost. We first prove that the algorithm is convergent:

**Proposition 4.4** *For the iterates  $u^{(n)}$  from Algorithm 2 it holds that*

$$\|u^{(n)} - u\| \leq \varepsilon_n$$

and thus  $\|u_\varepsilon - u\| \leq \varepsilon$ .

*Proof.* We show the statement by induction over  $n \geq 0$ . For  $n = 0$ , we have  $\varepsilon_0 = M$ , hence the statement is true by choice of  $M$ . Assume the statement holds for some  $n \in \mathbb{N}$ . As in the proof of Theorem 4.1, by induction we have

$$\|v^{(n,l)} - u\| \leq \tilde{\rho}^l \varepsilon_n$$

for all  $l = 0, \dots, l^*$ . Thus

$$\|v^{(n,l^*)} - u\| \leq \tilde{\rho}^{l^*} \varepsilon_n \leq \frac{1}{2KL} \frac{1}{2C^*+1} \tilde{\rho} \varepsilon_n = \frac{1}{2KL} \frac{1}{2C^*+1} \varepsilon_{n+1}.$$

Denoting by  $\mathbf{u} \in \ell_2(\Lambda)$  any expansion coefficients of the exact solution  $u$ ,  $u = \mathbf{u}^\top \Psi$ , from this it follows that

$$\begin{aligned}
\|u - u^{(n+1)}\| &\leq K\|\mathbf{P}\mathbf{u} - \mathbf{u}^{(n+1)}\|_{\ell_2(\Lambda)} \\
&\leq K(\|\mathbf{P}\mathbf{u} - \mathbf{P}\mathbf{v}^{(n,l^*)}\|_{\ell_2(\Lambda)} + \|\mathbf{P}\mathbf{v}^{(n,l^*)} - \tilde{\mathbf{u}}^{(n+1)}\|_{\ell_2(\Lambda)} + \|\tilde{\mathbf{u}}^{(n+1)} - \mathbf{u}^{(n+1)}\|_{\ell_2(\Lambda)}) \\
&\leq K(L\|u - v^{(n,l^*)}\| + \frac{1}{2K} \frac{1}{2C^* + 1} \varepsilon_{n+1} + \frac{1}{K} \frac{2C^*}{2C^* + 1} \varepsilon_{n+1}) \\
&\leq K(L \frac{1}{2KL} \frac{1}{2C^* + 1} \varepsilon_{n+1} + \frac{1}{2K} \frac{1}{2C^* + 1} \varepsilon_{n+1} + \frac{1}{K} \frac{2C^*}{2C^* + 1} \varepsilon_{n+1}) \\
&= \varepsilon_{n+1}.
\end{aligned}$$

□

We can now show that this algorithm is asymptotically optimal with respect to the support size of the iterates  $\mathbf{u}^{(n)}$ . In order to do so, we need to assume that the projector  $\mathbf{P}$  is bounded on  $\mathcal{A}_{\mathcal{A}\mathcal{T}}^s$ . To ensure this, fast decay of the entries in the matrix representation is required, see [Kap] for a more detailed discussion.

**Proposition 4.5** *Assume that  $\mathbf{P}$  is bounded on  $\mathcal{A}_{\mathcal{A}\mathcal{T}}^s$  and let  $\mathbf{u} \in \mathcal{A}_{\mathcal{A}\mathcal{T}}^s$ . Then, for the iterates in Algorithm 2 it holds that*

$$\begin{aligned}
\#\text{supp } \mathbf{u}^{(n)} &\lesssim \varepsilon_n^{-1/s} \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s}, \\
\|\mathbf{u}^{(n)}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} &\lesssim \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}.
\end{aligned}$$

*Proof.* By definition of Algorithm 2 and as in the proof of Proposition 4.1, we have

$$\begin{aligned}
\|\mathbf{P}\mathbf{u} - \tilde{\mathbf{u}}^{(n+1)}\|_{\ell_2(\Lambda)} &\leq \|\mathbf{P}\mathbf{u} - \mathbf{P}\mathbf{v}^{(n,l^*)}\|_{\ell_2(\Lambda)} + \|\mathbf{P}\mathbf{v}^{(n,l^*)} - \tilde{\mathbf{u}}^{(n+1)}\|_{\ell_2(\Lambda)} \\
&\leq L\|u - v^{(n,l^*)}\| + \frac{1}{2K} \frac{1}{2C^* + 1} \varepsilon_{n+1} \\
&\leq L \frac{1}{2KL} \frac{1}{2C^* + 1} \varepsilon_{n+1} + \frac{1}{2K} \frac{1}{2C^* + 1} \varepsilon_{n+1} \\
&= \frac{1}{K} \frac{1}{2C^* + 1} \varepsilon_{n+1}.
\end{aligned}$$

With Lemma 3.1 it now follows that

$$\#\text{supp } \mathbf{u}^{(n+1)} \lesssim \varepsilon_{n+1}^{-1/s} \|\mathbf{P}\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} \lesssim \varepsilon_{n+1}^{-1/s} \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s}.$$

The second estimate follows from the same lemma. □

We are now going to discuss how exactly the local problems can be solved such that we obtain a similar result as in Proposition 4.5 for the computational effort of our algorithm. Since the subproblems are linear and the matrices  $\mathbf{A}^{(k,k)}$  are positive definite, we use a Richardson iteration, following the lines from [Raa, Ste, Wer]. Remind that the convergence rate of the exact Richardson iteration for the matrices  $\mathbf{A}^{(k,k)}$  with relaxation parameter  $\alpha$  measured in the norm  $\|\cdot\|_{\mathbf{A}^{(k,k)}} = \langle \mathbf{A}^{(k,k)}, \cdot \rangle_{\ell_2(\Lambda)}^{1/2}$  is given by  $\mathbf{I} - \alpha \mathbf{A}^{(k,k)}$  measured in the operator norm induced by  $\|\cdot\|_{\mathbf{A}^{(k,k)}}$ . For  $\alpha > 0$  sufficiently small, this rate is known to be smaller than 1.

We are now going to see why we only need a fixed number of iterations in each step of the local solver. As in the proof of Proposition 4.1, we have

$$\begin{aligned}
\|e_k^{(n,l)}\| &= \|P_k(u - v^{(n,l)}) + A_k^{-1}(G(u) - G(v^{(n,l)}))\| \\
&\leq \|P_k(u - v^{(n,l)})\| + c\|u - v^{(n,l)}\| \\
&\leq (1 + c)\varepsilon_n \hat{\rho}^l.
\end{aligned}$$

Comparing this to the tolerance  $\frac{1-\rho}{2m\omega}\varepsilon_n\tilde{\rho}^l$  for the local solvers, we see that the error only has to be reduced by the *constant* factor

$$R := (1+c)\frac{2m\omega}{1-\rho}$$

in the energy norm if we start the local solvers with approximation 0 to  $e_k^{(n,l)}$ . Since the  $P_k$  are orthogonal projectors with respect to the energy norm, we have

$$\rho = \|I - \omega(P_0 + \dots + P_{m-1})\| + m\omega c \geq 1 - \omega\|P_0 + \dots + P_{m-1}\| + m\omega c \geq 1 - m\omega + m\omega c = 1 - (1-c)m\omega.$$

Hence it is indeed  $R \geq 2\frac{1+c}{1-c} > 1$ . Because for  $\mathbf{v} \in \ell_2(\Lambda_k)$ , we have  $\|\mathbf{v}\|_{\mathbf{A}^{(k,k)}} \approx \|\mathbf{v}^\top \Psi_k\|$ , the error measured in the discrete norm  $\|\cdot\|_{\mathbf{A}^{(k,k)}}$  only has to be reduced by a given constant factor  $\tilde{R}$ . This fact allows us to use a *constant* number of iterations in each local solver because in each Richardson step, the error is reduced by a constant factor in the norm  $\|\cdot\|_{\mathbf{A}^{(k,k)}}$ . Hence we define the local solvers as follows, similarly to [Wer]:

---

**Algorithm 3** LocSolve $[\delta]$

---

Let  $\xi := \|\mathbf{I} - \alpha\mathbf{A}^{(k,k)}\|_{\mathbf{A}^{(k,k)}} < 1$ , choose  $p \in \mathbb{N}$  minimal with  $2\xi^p(\tilde{R} + \frac{1}{2}) \leq \frac{1}{2}$ .

$$tol := \frac{1}{6}\|(\mathbf{A}^{(k,k)})^{-1}\|_{\ell_2(\Lambda_k) \rightarrow \ell_2(\Lambda_k)}^{-1/2} \delta$$

$$tol_2 := \frac{\xi^p(\tilde{R} + \frac{1}{2})\delta}{\alpha p \|\mathbf{A}^{(k,k)}\|_{\ell_2(\Lambda_k) \rightarrow \ell_2(\Lambda_k)}^{1/2}}$$

$$\mathbf{r} := -\mathbf{ATAPPLY}[\mathbf{A}, \mathbf{v}^{(n,l)}, tol]_{|\Lambda_k} + \mathbf{ATRHS}[\mathbf{f}, tol]_{|\Lambda_k} - \mathbf{EVAL}[\mathbf{G}, \mathbf{v}^{(n,l)}, tol]_{|\Lambda_k}$$

$$\mathbf{w}^{(0)} := \mathbf{0}$$

**for**  $r = 0, \dots, p-1$  **do**

$$\mathbf{w}^{(r+1)} := \mathbf{w}^{(r)} + \alpha(\mathbf{r} - \mathbf{ATAPPLY}[\mathbf{A}^{(k,k)}, \mathbf{w}^{(r)}, tol_2])$$

**end for**

$$\tilde{\mathbf{e}}_k^{(n,l)} := \mathbf{w}^{(p)}, \tilde{e}_k^{(n,l)} := (\mathbf{w}^{(p)})^\top \Psi_k$$


---

By the above remarks, it suffices to show the following lemma in order to ensure convergence of the entire Algorithm 2, where the subproblems are solved by LocSolve $[\delta]$ , with  $\delta$  being the tolerance for the local solver:

**Lemma 4.6** *Assume that for the exact solution  $e_k^{(n,l)} = (\mathbf{e}_k^{(n,l)})^\top \Psi_k$  of the local problem it holds that  $\|\mathbf{e}_k^{(n,l)}\|_{\mathbf{A}^{(k,k)}} \leq \delta\tilde{R}$ . Then, the discrete approximation  $\tilde{\mathbf{e}}_k^{(n,l)}$  from Algorithm 3 fulfills*

$$\|\mathbf{e}_k^{(n,l)} - \tilde{\mathbf{e}}_k^{(n,l)}\|_{\mathbf{A}^{(k,k)}} \leq \delta.$$

*Proof.* By definition of  $\mathbf{e}_k^{(n,l)}$  and  $\mathbf{r}$  it holds that

$$\|\mathbf{A}^{(k,k)}\mathbf{e}_k^{(n,l)} - \mathbf{r}\|_{\ell_2(\Lambda_k)} \leq 3 \cdot tol.$$

From this and the Cauchy-Schwarz inequality it follows that

$$\begin{aligned} \|\mathbf{e}_k^{(n,l)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} &= \left( \langle \mathbf{A}^{(k,k)}\mathbf{e}_k^{(n,l)} - \mathbf{r}, \mathbf{e}_k^{(n,l)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r} \rangle_{\ell_2(\Lambda_k)} \right)^{1/2} \\ &= \left( \langle \mathbf{A}^{(k,k)}\mathbf{e}_k^{(n,l)} - \mathbf{r}, (\mathbf{A}^{(k,k)})^{-1}(\mathbf{A}^{(k,k)}\mathbf{e}_k^{(n,l)} - \mathbf{r}) \rangle_{\ell_2(\Lambda_k)} \right)^{1/2} \\ &\leq \|(\mathbf{A}^{(k,k)})^{-1}\|_{\ell_2(\Lambda_k) \rightarrow \ell_2(\Lambda_k)}^{1/2} \|\mathbf{A}^{(k,k)}\mathbf{e}_k^{(n,l)} - \mathbf{r}\|_{\ell_2(\Lambda_k)} \\ &\leq 3\|(\mathbf{A}^{(k,k)})^{-1}\|_{\ell_2(\Lambda_k) \rightarrow \ell_2(\Lambda_k)}^{1/2} \cdot tol \\ &= \frac{1}{2}\delta. \end{aligned}$$

Hence it is

$$\begin{aligned}
\|\mathbf{w}^{(0)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} &= \|(\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} \\
&\leq \|\mathbf{e}_k^{(n,l)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} + \|\mathbf{e}_k^{(n,l)}\|_{\mathbf{A}^{(k,k)}} \\
&\leq \left(\frac{1}{2} + \tilde{R}\right)\delta.
\end{aligned}$$

Since the error of the exact Richardson iteration measured in the  $\|\cdot\|_{\mathbf{A}^{(k,k)}}$ -norm is reduced by the factor  $\xi$  in each step, we obtain

$$\|\mathbf{w}^{(1)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} \leq \xi\|\mathbf{w}^{(0)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} + \alpha\|\mathbf{A}^{(k,k)}\|_{\ell_2(\Lambda_k) \rightarrow \ell_2(\Lambda_k)}^{1/2} \text{tol}_2.$$

From this, as in [Wer], it follows by induction that

$$\begin{aligned}
\delta\|\mathbf{w}^{(p)} - (\mathbf{A}^{(k,k)})^{-1}\mathbf{r}\|_{\mathbf{A}^{(k,k)}} &\leq \alpha p\|\mathbf{A}^{(k,k)}\|_{\ell_2(\Lambda_k) \rightarrow \ell_2(\Lambda_k)}^{1/2} \text{tol}_2 + \xi^p\left(\tilde{R} + \frac{1}{2}\right)\delta \\
&= 2\xi^p\left(\tilde{R} + \frac{1}{2}\right)\delta \\
&\leq \frac{1}{2}\delta.
\end{aligned}$$

Altogether we have  $\|\mathbf{e}_k^{(n,l)} - \tilde{\mathbf{e}}_k^{(n,l)}\|_{\mathbf{A}^{(k,k)}} \leq \delta$ . □

We can now show that the number of operations needed and the support size of the inner iterates is optimal up to some additional constants.

**Lemma 4.7** *Assume that  $\mathbf{A}$  is bounded on  $\mathcal{A}_{\mathcal{AT}}^s$ . For the inner iterates  $\mathbf{v}^{(n,l)} = (\mathbf{v}^{(n,l)})^\top \Psi_k$  from Algorithm 3 it holds that*

$$\#\text{supp } \mathbf{v}^{(n,l)} \lesssim \varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1), \quad (9)$$

$$\|\mathbf{v}^{(n,l)}\|_{\mathcal{A}_{\mathcal{AT}}^s} \lesssim \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s} + 1. \quad (10)$$

*The computation of  $\mathbf{v}^{(n,l)}$ ,  $l \geq 1$  from  $\mathbf{v}^{(n,l-1)}$  requires at most  $\mathcal{O}(\varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1))$  operations.*

*Proof.* We use induction over  $l$ . For  $l = 0$  the statement is valid by Proposition 4.5. Assume that the statement is true some  $l$  with  $0 \leq l \leq l^* + 1$ . By definition of an aggregated tree it holds that for  $\mathbf{v} \in \mathcal{A}_{\mathcal{AT}}^s$  the restriction  $\mathbf{v}|_{\Lambda_k}$  also belongs to  $\mathcal{A}_{\mathcal{AT}}^s$  and  $\|\mathbf{v}|_{\Lambda_k}\|_{\mathcal{A}_{\mathcal{AT}}^s} \leq \|\mathbf{v}\|_{\mathcal{A}_{\mathcal{AT}}^s}$ . In Algorithm 2, the local solvers are called with tolerance  $\delta = \frac{1-\rho}{2m\omega} \varepsilon_n \tilde{\rho}^l$ . Because  $l$  is bounded between 0 and  $l^* - 1$ , it is  $\delta \approx \varepsilon_n$ . Hence, we also have  $\text{tol} \approx \varepsilon_n$ . From the properties of **ATRHS**, **ATAPPLY** and **EVAL**, it then follows that the vector  $\mathbf{r}$  from Algorithm 3 fulfills

$$\#\text{supp } \mathbf{r} \lesssim \varepsilon_n^{-1/s} (\|\mathbf{f}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + \|\mathbf{v}^{(n,l)}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1) \lesssim \varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1)$$

and

$$\|\mathbf{r}\|_{\mathcal{A}_{\mathcal{AT}}^s} \lesssim \|\mathbf{f}\|_{\mathcal{A}_{\mathcal{AT}}^s} + \|\mathbf{v}^{(n,l)}\|_{\mathcal{A}_{\mathcal{AT}}^s} + 1 \lesssim \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s} + 1.$$

Here, we used the boundedness of  $\mathbf{A}$  on  $\mathcal{A}_{\mathcal{AT}}^s$  to see that  $\|\mathbf{f}\|_{\mathcal{A}_{\mathcal{AT}}^s} = \|\mathbf{A}\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s} \lesssim \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s}$ . Since the number  $p$  of inner loops is constant, by the properties of **ATAPPLY** and  $\text{tol}_2 \approx \varepsilon_n$  for the inner iterates  $\mathbf{w}^{(r)}$ ,  $r = 0, \dots, p-1$ , it holds that

$$\#\text{supp } \mathbf{w}^{(r+1)} \lesssim \#\text{supp } \mathbf{w}^{(r)} + \#\text{supp } \mathbf{r} + \varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{AT}}^s}^{1/s} + 1)$$



and

$$\|\mathbf{w}^{(r+1)}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} \lesssim \|\mathbf{w}^{(r)}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} + \|\mathbf{r}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}.$$

From this, by an inner induction over  $r = 0, \dots, p$  we obtain that

$$\#\text{supp } \mathbf{w}^{(r)} \lesssim \varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1)$$

and

$$\|\mathbf{w}^{(r)}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} \lesssim \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} + 1$$

for all  $0 \leq r \leq p$ . Note that, because  $p$  is constant, the constants involved do not mount up unboundedly. Hence, we have  $\#\text{supp } \tilde{\mathbf{e}}_k^{(n,l)} \lesssim \varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1)$  and  $\|\tilde{\mathbf{e}}_k^{(n,l)}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} \lesssim \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s} + 1$ . By induction and the definition of  $\mathbf{v}^{(n,l+1)}$ , we now obtain (9) and (10). Since all the inner iterates have aggregated tree structure, from the properties of **ATRHS**, **ATAPPLY** and **EVAL** and the above estimates it immediately follows that the number of operations for calculating  $\mathbf{v}^{(n,l+1)}$  from  $\mathbf{v}^{(n,l)}$  is bounded by a constant multiple of  $\varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1)$ .  $\square$

Summing up what we have shown so far, we can now conclude our main result:

**Theorem 4.8** *Let  $s^* > 0$ ,  $\mathbf{A}$  and  $\mathbf{P}$  be  $s^*$ -compressible and bounded on  $\mathcal{A}_{\mathcal{A}\mathcal{T}}^s$ . Assume that (8) is valid. Then, for the output from Algorithm 2 with LocSolve as a local solver and for all  $\varepsilon > 0$  it holds that*

$$\|u_\varepsilon - u\| \leq \varepsilon. \quad (11)$$

If  $u = \mathbf{u}^\top \Psi$  with  $\mathbf{u} \in \mathcal{A}_{\mathcal{A}\mathcal{T}}^s$ ,  $s \in (0, s^*)$  we have

$$\#\text{supp } \mathbf{u}_\varepsilon \lesssim \varepsilon^{-1/s} \|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} \quad (12)$$

and the calculation of  $\mathbf{u}_\varepsilon$  requires at most a constant multiple of  $\mathcal{O}(\varepsilon^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1))$  operations.

*Proof.* Estimate (11) follows from Proposition 4.4 and Lemma 4.6. Let  $N$  be minimal such that  $\varepsilon_N \leq \varepsilon$ . Then, we have  $\varepsilon_N \approx \varepsilon$ , hence (12) is a direct consequence of Proposition 4.5. By Lemma 4.7 and because  $l^*$  is constant, the calculation of  $\mathbf{v}^{(n,l^*)}$  from  $\mathbf{u}^{(n)}$  via LocSolve requires  $\mathcal{O}(\varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1))$  operations. Due to (10) and  $\varepsilon_{n+1} \approx \varepsilon_n$ , the number of operations needed for **ATAPPLY** and **ATCOARSE** in Algorithm 2 can be bounded by the same expression. Hence, the calculation of  $\mathbf{u}^{(n+1)}$  from  $\mathbf{u}^{(n)}$  requires  $\mathcal{O}(\varepsilon_n^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1))$  operations. Since  $\sum_{n=0}^N \varepsilon_n^{-1/s} \approx \varepsilon_N^{-1/s} \approx \varepsilon^{-1/s}$ , the total number of operations needed can be bounded by  $\mathcal{O}(\varepsilon^{-1/s} (\|\mathbf{u}\|_{\mathcal{A}_{\mathcal{A}\mathcal{T}}^s}^{1/s} + 1))$ .  $\square$

## 5 Numerical tests

In this section we are going to test Algorithm 2. Our test case will be the weak formulation of the typical model problem

$$\begin{aligned} -\Delta u + u^3 &= f & \text{in } \Omega, \\ u &= 0 & \text{on } \partial\Omega. \end{aligned} \quad (13)$$

Although the nonlinearity does not fulfill property (2) globally, by Remark 4.2 our approach is applicable to this problem. The wavelets we are going to use stem from the construction in [Pri]. In the practical

implementation, we slightly modify Algorithm 2 for instance by leaving out the application of projector  $\mathbf{P}$ . Even though the projection step is required for the strict proof of optimality, in practice it does not improve the overall performance. The highest convergence rate we can expect from nonlinear approximation with wavelets is governed by the *Besov* regularity of the function to approximate, see [DeV, Ste]. However, the range in which this holds depends on the order of the underlying wavelet bases. If the Besov regularity is sufficiently high, the maximal rate that can be obtained in general is

$$s = \frac{l-1}{d}, \quad (14)$$

where  $l$  denotes the order of the spline wavelets and  $d$  is the space dimension. That is why we cannot expect a discrete solution  $\mathbf{u}$  with  $u = \mathbf{u}^\top \Psi$  to be in  $\mathcal{A}_{\mathcal{AT}}^s$  for  $s > \frac{l-1}{d}$ . Remind that the rate produced by uniform methods is typically governed by the Sobolev regularity of the solution.

## 5.1 Tests in one space dimension

At first, we investigate the easiest case of the interval  $\Omega = (0, 1)$  with overlapping domain decomposition  $\Omega_0 = (0, 0.7)$ ,  $\Omega_1 = (0.3, 1)$ . The wavelet frame is composed as described in Section 2 using the wavelet basis from [Pri]. According to Lemma 2.5, we have to ensure that a partition of unity exists. A common tool used in [Wer] is the function

$$\xi_{a,b} := \frac{\omega(b-x)}{\omega(x-a) + \omega(b-x)} \quad (15)$$

with

$$\omega(x) := \begin{cases} 0, & x \leq 0 \\ \exp(-x^{-2}), & x > 0 \end{cases}.$$

It holds that  $\xi_{a,b}(x) = 1$  for  $x \leq a$  and  $\xi_{a,b}(x) = 0$  for  $x \geq b$ , and  $\xi_{a,b} \in C^\infty(\mathbb{R})$ . By setting  $\sigma_0 := \xi_{0.3,0.7}$  and  $\sigma_1 := 1 - \sigma_0$ , we have constructed the desired partition of unity. In practical computations, one limits the maximal level of wavelets considered. Here, we choose a maximal level  $j_{max} := 15$ , where the minimal level is  $j_0 := 3$  for the bases of order  $(l, \tilde{l}) = (2, 2)$ ,  $(3, 3)$  and  $j_0 := 5$  for the basis of order  $(l, \tilde{l}) = (4, 6)$ . In general, choosing a lower maximal level significantly speeds up the algorithm at the price of a lower maximal accuracy. We fix the relaxation parameters as  $\omega = \alpha = 0.25$  and set  $\rho = 0.8$ . It is important to note that one does not have to know  $\rho$  exactly. Choosing a lower value for  $\rho$  generally leads to a lower number of iterations at the price of a faster grow of the support sizes of the iterates.

For our test, we fix the exact solution

$$u(x) := -\sin(3\pi x) + \begin{cases} 2x^2, & x \in (0, \frac{1}{2}) \\ 2(1-x)^2, & x \in [\frac{1}{2}, 1) \end{cases}.$$

This is an appropriate test case for our adaptive algorithm because this function belongs to all Besov spaces  $B_{\tau,\tau}^\alpha(\Omega)$ ,  $\alpha > 0$ ,  $\tau^{-1} = \alpha - \frac{1}{2}$ , but is only contained in  $H^\alpha(\Omega)$  for  $\alpha < \frac{3}{2}$ , see [SW2]. Hence, we can expect a benefit from using an adaptive algorithm. We set the tolerance for the discrete residual to  $\varepsilon := 10^{-3}$  and limit the number of iterations to a maximum of 100. The graphs in Figure 1 show that even without a projection step, the algorithm indeed converges with the optimal rate from (14). It also becomes obvious that the best results can be achieved choosing  $(l, \tilde{l}) = (3, 3)$  or  $(l, \tilde{l}) = (4, 6)$ . For these bases, the algorithm terminated after 94 iterations. With the parameters  $(2, 2)$ , the residual was around  $2.3 \cdot 10^{-3}$  after 100 iterations. Hence, although the choice  $(l, \tilde{l}) = (4, 6)$  leads to the highest convergence rate, the constants involved are better when choosing  $(l, \tilde{l}) = (3, 3)$ , which from now we are going to fix.

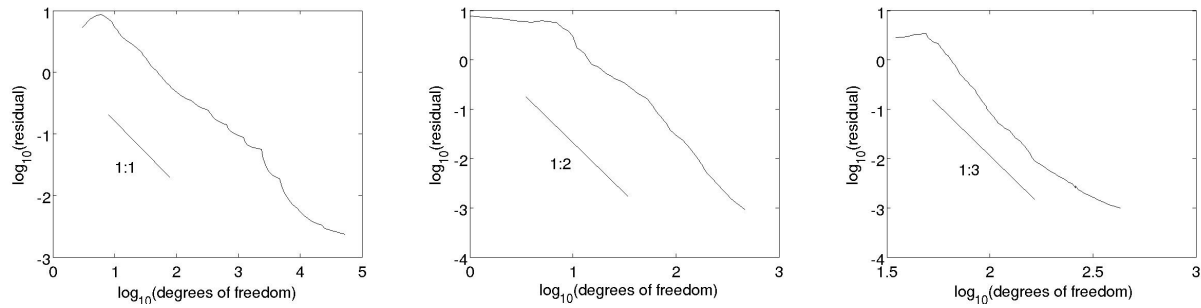


Figure 1:  $\ell_2$ -Norm of the residual vs. degrees of freedom for bases  $(l, \tilde{l}) = (2, 2), (3, 3), (4, 6)$

In Figure 2, we see that the maximal pointwise error corresponding to a residual of  $10^{-3}$  in this case is around  $10^{-4}$ . Apparently, the error is largest around the singularity of the solution  $u(x)$  at the point  $x = 0.5$ .

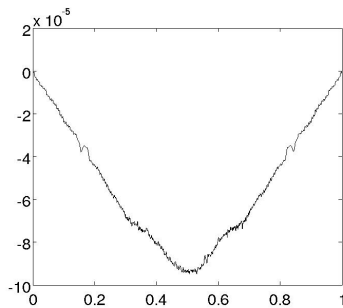


Figure 2: Pointwise error

In Figure 3, we compare the exact solution to the local parts  $(\mathbf{u}_{\varepsilon|\Lambda_0})^\top \Psi_0$  and  $(\mathbf{u}_{\varepsilon|\Lambda_1})^\top \Psi_1$  produced by our algorithm. The figure shows that the redundancies in the overlapping region  $\Omega_0 \cap \Omega_1$  remain moderate. Hence, it seems reasonable to leave out the projection step via the application of  $\mathbf{P}$  that explicitly removes the redundancies at the price of more computational effort.

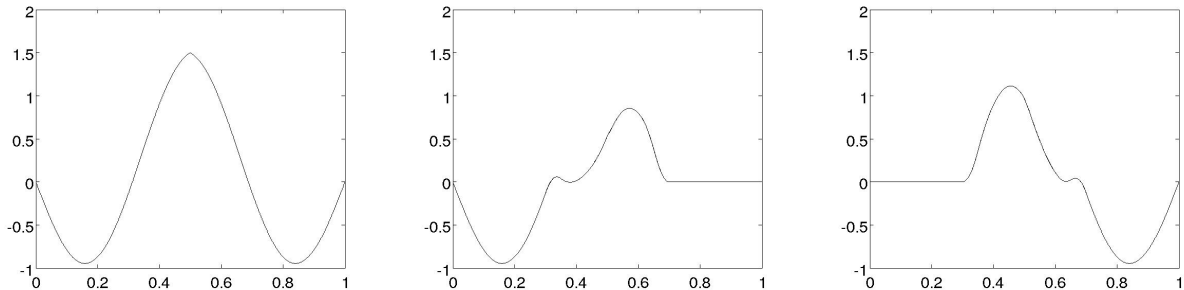


Figure 3: Exact solution and the local parts produced by Algorithm 2

## 5.2 Tests in two space dimensions

After the encouraging results from the tests in one space dimension, we now test our algorithm with problem (13) on the two-dimensional L-domain  $\Omega = (-1, 1)^2 \setminus [0, 1]^2$  with domain decomposition  $\Omega_0 = (-1, 0) \times (-1, 1)$  and  $\Omega_1 = (-1, 1) \times (-1, 0)$ . This is a typical example of a polygonal domain with a reentrant corner. Solutions to elliptic equations on domains of this kind typically have a singularity, see [Gri]. That is why one would like to use an adaptive algorithm. Due to the reentrant corner it is difficult to construct well-conditioned wavelet bases for the L-domain, which on the contrary is possible for the rectangular subdomains  $\Omega_0$  and  $\Omega_1$ . Therefore, it is reasonable to use a wavelet frame. In order to obtain a wavelet frame from unifying bases on the subdomains, according to Lemma 2.5, we need to make sure that a partition of unity  $\{\sigma_0, \sigma_1\}$  with respect to this domain decomposition exists. Because  $\sigma_0$  and  $\sigma_1$  are bound to be discontinuous at the origin, the construction is more difficult than in the one-dimensional test case. However, in [Wer] it is shown that by setting  $\sigma_0 := \xi_{\frac{\pi}{2}, \pi} \circ \theta$ , where  $\xi_{a,b}$  is the function from (15) and  $\theta$  denotes the angle in polar coordinates with respect to the origin, and again  $\sigma_1 := 1 - \sigma_0$ , we indeed obtain a partition of unity according to Definition 2.4 for the L-domain. We fix the exact solution

$$S(r, \theta) := \xi_{0,1}(r)r^{2/3} \sin\left(\frac{2}{3}\theta\right) \quad (16)$$

in polar coordinates. The solution  $S$  has a singularity at the origin due to which it belongs to  $H^\alpha(\Omega)$  only for  $\alpha < \frac{5}{3}$ , see [Gri]. The Besov regularity of the solution is much higher, namely  $S \in B_{\tau,\tau}^\alpha$  for all  $\alpha > 0$ ,  $\tau^{-1} = \frac{\alpha-1}{2} + \frac{1}{2}$ , see [Dah]. Hence, it is preferable to use an adaptive method.

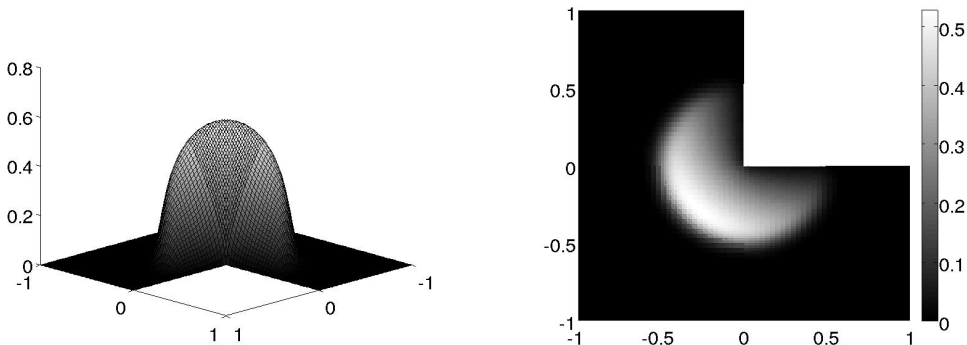


Figure 4: Exact solution  $S(r, \theta)$  from (16).

Since the systems are larger in the two-dimensional case, we lower the relaxation parameter  $\alpha$  for the local solvers to 0.1 and we fix the minimum and maximum level of the wavelets as  $j_0 := 3$  and  $j_{max} := 6$ . Again, we set the discrete tolerance to  $\varepsilon := 10^{-3}$ , which is achieved after 113 iterations. In Figure 5, we see the local parts  $(\mathbf{u}_\varepsilon|_{\Lambda_i})^\top \Psi_i$ ,  $i = 0, 1$  produced by Algorithm 2. The redundancies in the overlapping region are very moderate. Therefore, from a numerical point of view, the application of the projector  $\mathbf{P}$  is not necessary here. As expected, the pointwise error is again largest around the singularity at the origin. Comparing the degrees of freedom to the residual, we see that in this test case the rate is slightly better than predicted by formula (14), see Figure 6.

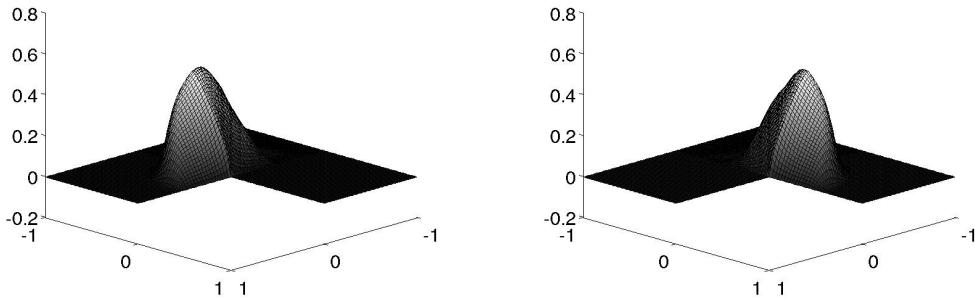


Figure 5: Local parts produced by Algorithm 2

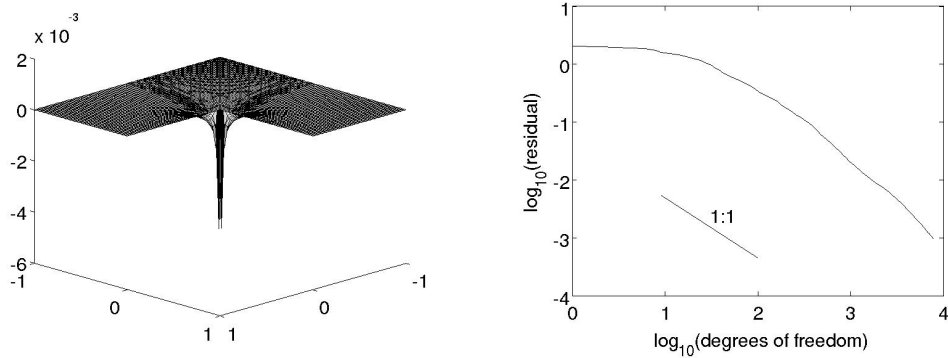


Figure 6: Pointwise error and convergence rate for Algorithm 2 in the two-dimensional case

All in all, we conclude that the algorithm indeed shows the theoretically predicted behavior. The constants involved might still be improved by using more sophisticated sparsening strategies. Further research will go into this direction. It also seems promising to study the multiplicative Schwarz method and more advanced linearization techniques.

## References

- [AA] A. Ambrosetti and D. Arcoya. *An Introduction to Nonlinear Functional Analysis and Elliptic Problems*. Birkhäuser, Boston/Basel/Berlin, 2011
- [Bad] L. Badea. On the Schwarz alternating method with more than two subdomains for non-linear monotone problems. *SIAM J. Numer. Anal.* **28** (1991), no. 1, 179–204
- [BS] M. Badiale and B. Serra. *Semilinear Elliptic Equations for Beginners*. Springer, London, 2011
- [CDD1] A. Cohen, W. Dahmen and R. DeVore. Adaptive wavelet methods for elliptic operator equations: convergence rates. *Math. Comp.* **70** (2001), no. 233, 27–75
- [CDD2] A. Cohen, W. Dahmen and R. DeVore. Sparse evaluation of compositions of functions using multiscale expansions. *SIAM J. Math. Anal.* **35** (2003), no. 2, 279–303

- [CDD3] A. Cohen, W. Dahmen and R. DeVore. Adaptive wavelet schemes for nonlinear variational problems. *SIAM J. Numer. Anal.* **41** (2003), no. 5, 1785–1823
- [Chr] O. Christensen. *An Introduction to Frames and Riesz Bases*. Birkhäuser, Boston, 2003
- [Dah] S. Dahlke. Besov regularity for elliptic boundary value problems in polygonal domains. *Appl. Math. Lett.* **12** (1999), no. 6, 31–36
- [Dau] I. Daubechies. *Ten Lectures on Wavelets*. CBMS-NSF Regional Conference Series in Applied Mathematics, 61, SIAM, Philadelphia, 1992
- [DDHS] S. Dahlke, W. Dahmen, R. Hochmuth and R. Schneider. Stable multiscale bases and local error estimation for elliptic problems. *Appl. Numer. Math.* **23** (1997), no. 1, 21–47.
- [DeV] R. DeVore. Nonlinear approximation. *Acta Numer.* **7** (1998), 51–150
- [DFR] S. Dahlke, M. Fornasier and T. Raasch. Adaptive frame methods for elliptic operator equations. *Adv. Comput. Math.* **27** (2007), no. 1, 27–63
- [DH] M. Dryja and W. Hackbusch. On the nonlinear domain decomposition method. *BIT* **37** (1997), no. 2, 296–311
- [DKU] W. Dahmen, A. Kunoth and K. Urban. Biorthogonal spline wavelets on the interval – stability and moment conditions. *Appl. Comput. Harmon. Anal.* **6** (1999), no. 2, 132–196
- [DS] W. Dahmen and R. Schneider. Wavelets with complementary boundary conditions – function spaces on the cube. *Results Math.* **34** (1998), no. 3–4, 255–293
- [DW] M. Dryja and O. Widlund. An additive variant of the Schwarz alternating method for the case of many subregions. New York University, Courant Institute of Mathematical Sciences, Computer Science Technical Report no. 339 (1987)
- [Gri] P. Grisvard. *Elliptic Problems in Nonsmooth Domains*. Pitman, Boston, 1985
- [HE] Q. He and D. J. Evans. Monotone-Schwarz parallel algorithm for nonlinear elliptic equations. *Int. J. Parallel Emergent Distrib. Syst.* **9** (1996), 173–184
- [Kap] J. Kappei. Adaptive frame methods for nonlinear elliptic problems. *Appl. Anal.* **90** (2011), no. 8, 1323–1353.
- [Lio] P. L. Lions. On the Schwarz alternating method I. *First International Symposium on Domain Decomposition Methods for Partial Differential Equations (Paris, 1987)*, 1–42, SIAM, Philadelphia, 1988.
- [Lui1] S. H. Lui. On Schwarz alternating methods for nonlinear elliptic PDEs. *SIAM J. Sci. Comput.* **21** (2000), no. 4, 1506–1523
- [Lui2] S. H. Lui. On linear monotone iteration and Schwarz methods for nonlinear elliptic PDEs. *Numer. Math.* **93** (2002), no. 1, 109–129
- [Mar] L. Marcinkowski. Additive Schwarz method for quasilinear partial differential equations. Technical Report, Institute of Appl. Math. and Mechanics, Dept. of Math., Informatics and Mechanics, Warsaw University, no. 13 (1996)

- [Mey] Y. Meyer. *Ondelettes et Opérateurs. I*. Hermann, Paris, 1990
- [Pri] M. Primbs. *Stabile biorthogonale Spline-Waveletbasen auf dem Intervall*. Dissertation, Universität Duisburg-Essen, 2006
- [QV] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford University Press, New York, 1999
- [Raa] T. Raasch. *Adaptive Wavelet and Frame Schemes for Elliptic and Parabolic Equations*. Logos Verlag Berlin, 2007
- [Ste] R. Stevenson. Adaptive solution of operator equations using wavelet frames. *SIAM J. Numer. Anal.* **41** (2003), no. 3, 1074–1100
- [SW] R. Stevenson and M. Werner. Computation of differential operators in aggregated wavelet frame coordinates. *IMA J. Numer. Anal.* **28** (2008), no. 2, 354–381
- [SW2] R. Stevenson and M. Werner. A multiplicative Schwarz adaptive wavelet method for elliptic boundary value problems. *Math. Comp.* **78** (2009), no. 266, 619–644
- [Tay] M. E. Taylor. *Partial Differential Equations III: Nonlinear Equations*. Second edition. Applied Mathematical Sciences, 117, Springer, New York, 2011.
- [TE] X.-C. Tai and M. Espedal. Applications of a space decomposition method to linear and nonlinear elliptic problems. *Numer. Methods Partial Differential Equations* **14** (1998), no. 6, 717–737
- [TW] A. Toselli and O. Widlund. *Domain Decomposition Methods – algorithms and theory*. Springer Series in Computational Mathematics, 34, Springer, Berlin, 2005.
- [Wer] M. Werner. *Adaptive Wavelet Frame Domain Decomposition Methods for Elliptic Operator Equations*. Logos Verlag Berlin, 2009
- [Xu] J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.* **34** (1992), no. 2, 581–613
- [XZ] Y. Xu and Q. Zou. Adaptive wavelet methods for elliptic operator equations with nonlinear terms. *Adv. Comput. Math.* **19** (2003), no. 1-3, 99–146