Adaptive Optimization of Convex Functionals in Banach Spaces

Claudio Canuto and Karsten Urban

Preprint Series: 2004-01

Fakultät für Mathematik und Wirtschaftswissenschaften
UNIVERSITÄT ULM

ADAPTIVE OPTIMIZATION OF CONVEX FUNCTIONALS IN BANACH SPACES*

CLAUDIO CANUTO† AND KARSTEN URBAN‡

Abstract. This paper is concerned with optimization or minimization problems that are governed by operator equations, such as partial differential or integral equations, and thus are naturally formulated in an infinite dimensional function space V. We first construct a prototype algorithm of steepest descent type in V and prove its convergence. By using a Riesz basis in V we can transform the minimization problem into an equivalent one posed in a sequence space of type ℓ_p. We convert the prototype algorithm into an adaptive method in ℓ_p. This algorithm is shown to be convergent under mild conditions on the parameters that appear in the algorithm. Under more restrictive assumptions we are also able to establish the rate of convergence of our algorithm and prove that the work/accuracy-balance is asymptotically optimal. At last, we give two particular examples.

Keywords: optimization, convex analysis, steepest descent method, adaptive methods, nonlinear approximation, wavelet bases.
AMS MSC (2000): 65K10, 65T60

1. Introduction. Optimization or minimization problems arise in many areas of modern science and technology. As examples let us mention control theory, image processing and segmentation, drag reduction, shape optimization, and so on. Constrained and non-smooth optimization, such as in modelling American Options and elastoplastic hardening and softening pose additional challenges and require special care. Here we are particularly interested in problems that are governed by operator equations, such as partial differential or integral equations, and thus are naturally formulated in infinite dimensional function spaces (see, e.g., [2, 8, 25]).

The numerical solution of minimization problems in function spaces is traditionally based upon the choice of a suitable discretization of the spaces, leading to similar problems in finite dimension; these are then solved by some of the available optimization algorithms in Euclidean spaces. Such approach may incorporate an adaptive strategy, which is highly appropriate for those problems whose minimizers contain well localized structures. With the aid of a-posteriori error analysis, a refinement or derefinement (coarsening) of the current discretization can be constructed yielding to an adaptively generated sequence of finite dimensional discretizations. Even though this kind of adaptive methods have been successfully used in many different applications, the literature on the numerical analysis of adaptive methods for minimization problems seems fairly limited. Not much is known on the convergence of the adaptive iterations and even less on the convergence speed.

Starting from recent investigations concerning adaptive wavelet methods [11, 12, 13], it is by now clear that using as much information from the original infinite-dimensional variational problem as possible gives in particular a strong mathematical tool to prove convergence and convergence rate results. The new philosophy consists of combining an infinite-dimensional iteration with an approximate finite application of the underlying exact operators. By using an analogous approach, similar results for adaptive finite element methods have been obtained, [4, 24, 28]. An important ingredient in the design and analysis of such algorithms has been provided by recent results

*This work was partly supported by the *European Commission* within the Research Training Network *Breaking Complexity*, No. 2-2001-00574
†Politecnico di Torino, Dipartimento di Matematica, Corso Duca degli Abruzzi 24, I-10129 Torino, Italy, ccanuto@polito.it
‡University of Ulm, Abteilung Numerik, D-89069 Ulm, Germany, kurban@mathematik.uni-ulm.de

in Nonlinear Approximation Theory, [23]. They give guidelines for the definition of refinement and coarsening strategies based upon a rigorous control of the resulting errors as well as the number of degrees of freedom. They set the appropriate notion of optimality by indicating the best possible relation between accuracy and number of degrees of freedom in approximating the solution to the particular problem at hand. It turns out that the correct functional setting for this notion is provided by certain approximation spaces. In many cases, these approximation spaces are close to Besov spaces in certain scales, in which the summability index decreases as the regularity index increases (as opposed to traditional scales of Sobolev spaces for non-adaptive methods).

The purpose of the present paper is to take a first step towards the adaptive numerical treatment of infinite dimensional minimization problems following the new philosophy described above. To this end, we start with a fairly classical situation in which we want to minimize a strictly convex, Fréchet differentiable functional J defined on a reflexive Banach space V. The assumption of convexity guarantees the existence of a global minimizer and global convergence towards it, allowing us to concentrate on the central issues related to the infinite-dimensional setting. Since the ultimate goal of our investigations will be to handle a broad class of problems including constrained or non-smooth optimization (see, e.g., [26]), the minimization strategy will be based upon a general method of steepest descent type, coupled with a line search. The advantage is that we do not require strong regularity assumptions such as existence or boundedness of the Hessian of the functional.

After making suitable assumptions on the well-posedness of the minimization problem, we construct a steepest descent algorithm in the Banach space V which is proven to be convergent. This algorithm will serve as a prototype for the adaptive algorithms constructed next. By introducing a Riesz basis in V and by considering the sequence of the expansion coefficients of a function with respect to the chosen basis, we transform the minimization problem into an equivalent one posed in a sequence space of type ℓ_p. This transformation is a crucial step towards the concrete realization of the algorithm and for the introduction of adaptive concepts.

Next, we convert the previous algorithm into a steepest descent method defined in the sequence space. While the exact minimizer is in general represented by a sequence with infinitely many non-vanishing entries, the new algorithm only acts on finite vectors (i.e., sequences having only a finite number of non-vanishing entries); hence, it is numerically feasible. Since the number of active coefficients may grow in the descent stages, we incorporate a coarsening procedure in order to remove unneccessary details from time to time. This adaptive algorithm is shown to be convergent under mild conditions on the parameters that appear in its definition.

Under more restrictive assumptions we are also able to investigate the rate of convergence of our algorithm. Precisely, we prove that the error between the exact and the approximate minimizer decays at least in a geometric manner as the number of iterations increases. The number of approximate evaluations of the gradient of the functional applied to a finite vector (which is usually the most expansive part of the algorithm) grows in an asymptotically optimal way, i.e., at most logarithmically in the accuracy. Finally, we can prove that the output of the algorithm is optimal in the sense of Nonlinear Approximation Theory (asymptotically optimal work/accuracy balance) under the condition that the gradient can be efficiently approximated in a sparse way and an optimal thresholding procedure is available to realize the coarsening.

The paper is organized as follows. We start from a fairly general framework of convex optimization in Banach spaces. Under these weak assumptions, we formulate the abstract algorithm in Section 2 and prove its convergence. Section 3 is devoted to the transformation of the algorithm into an equivalent one set in an ℓ_p-space. This abstract setting yields a method in infinite dimension which is in general not computable. Hence, in Section 4 we investigate the ingredients that are needed to define a computable adaptive version of our general algorithm and then we prove the convergence of the resulting method.

In the second part of the paper, starting with Section 5, we specialize our setting to the ℓ_2-case and we assume Lipschitz continuity of the gradient. In this framework we indicate a precise choice of the parameters in our abstract algorithm and construct more efficient adaptive algorithmic ingredients in order to obtain also a rate of convergence. In Section 6, we investigate the optimality of the algorithm. At last, two examples are provided in Section 7.

We will frequently use the notation $A \lesssim B$ which means that there exists a constant $c >$ such that $A \leq c\, B$, uniformly in all parameters on which A and B may depend. The notation $A \sim B$ means $A \lesssim B$ and $B \lesssim A$.

2. Convex Optimization in Banach Spaces. Our setting is as follows. Let V be a reflexive Banach space normed by $\|\cdot\|_V$ and let us denote by $\langle\cdot,\cdot\rangle$ the duality pairing between V and V'. Let $J : V \to \mathbb{R}$ be a Fréchet differentiable functional. We consider the following minimization problem: Find $u \in V$ such that

$$J(u) = \min_{v\in V} J(v). \tag{2.1}$$

We start by collecting the general assumptions on the functional J, that will hold throughout the paper. Then we will recall the concept of admissible descent directions and stepsizes adapted to the Banach space setting. Finally, we formulate our abstract algorithm and prove its convergence.

2.1. General Assumptions.

Assumption 2.1. *The functional J satisfies the following conditions:*

(i) The Fréchet-derivative $J' : V \to V'$ is uniformly continuous on each bounded subset of V;

(ii) J is V-elliptic in the sense that there exist $c_J > 0$ and $p > 1$ such that

$$\langle J'(w) - J'(v), w - v\rangle \geq c_J \|w - v\|_V^p$$

for all $w, v \in V$.

Let us collect some consequences of the latter assumption.

Lemma 2.2. *Let Assumption 2.1 be satisfied. Then, the following statements hold.*

a.) For all $w, v \in V$ we have

$$J(v) - J(w) \geq \langle J'(w), v - w\rangle + \frac{c_J}{p}\|v - w\|_V^p. \tag{2.2}$$

b.) J is strictly convex and bounded from below.

c.) Let $u^{(0)} \in V$ be arbitrary, then the set $\mathcal{R}(u^{(0)}) := \{v \in V : J(v) \leq J(u^{(0)})\}$ is bounded.

The straightforward proof can be found in Appendix A (see also [9]). In particular, we see from b.) that our assumptions are slightly stronger than convexity.

Under Assumption 2.1, there exists a unique solution to the minimization problem (2.1) (see, e.g., [8] for a proof).

2.2. Descent Directions and Stepsizes. As already said, we aim at formulating an algorithm of steepest descent type that leads to the solution u of the optimization problem (2.1). Since we are working in abstract Banach spaces, we cannot expect to have orthogonal descent directions available, no matter what inner product is used. Hence, we have to specify what directions are admissible in order to yield a possible descent. We require that the direction is *not* orthogonal to the current gradient.

DEFINITION 2.3. *Given $v \in V$, we call $s \in V$ an* admissible descent direction *for v if $\|s\| = 1$ and $\langle J'(v), s\rangle < 0$.*

Once an admissible descent direction is determined, one has to find the minimum of the functional J along the search direction. Again, we cannot hope to be able to determine this minimum exactly, even though this is a 1D minimization problem. In order to ensure that the line search in fact yields a smaller value of the functional then the starting one, we identify a possible range of stepsizes that guarantees this descent. We adapt the classical concept of admissible stepsizes as follows (see, e.g., [22, §6.3]).

DEFINITION 2.4. (Wolfes Condition) *Let α, β be fixed constants satisfying $0 < \alpha < \beta < 1$. For any $v \in V$ and any admissible descent direction $s \in V$, define $\mathcal{A}(J; v, s)$ as the set of all those $\mu \in \mathbb{R}_+$ satisfying the following conditions*

$$J(v + \mu s) \leq J(v) + \alpha\, \mu\, \langle J'(v), s\rangle, \tag{2.3}$$

$$\langle J'(v + \mu s), s\rangle \geq \beta\, \langle J'(v), s\rangle. \tag{2.4}$$

We call $\mathcal{A}(J; v, s)$ the set of admissible stepsizes*.*

In order to understand the meaning of the previous conditions, it is convenient to introduce the auxiliary univariate function

$$\varphi(\mu) := J(v + \mu s), \qquad \mu \in \mathbb{R}, \tag{2.5}$$

which is a strictly convex function satisfying $\varphi(0) = J(v)$, $\varphi'(0) = \langle J'(v), s\rangle < 0$, and $\varphi(\mu) \to +\infty$ for $\mu \to +\infty$. Then, condition (2.3) reads

$$\varphi(\mu) \leq \varphi(0) + \alpha\, \varphi'(0)\, \mu \tag{2.6}$$

and, due to the convexity of φ, it identifies an interval of the form $(0, \mu_{\max}]$. Conversely, condition (2.4) reads

$$\varphi'(\mu) \geq \beta \varphi'(0) \tag{2.7}$$

and identifies an interval of the form $[\mu_{\min}, +\infty)$. Then, $\mathcal{A}(J; v, s)$ is the intersection of these two intervals. It is not empty since $\alpha < \beta$, see also Figure 2.1.

2.3. A Convergent Steepest Descent Algorithm. We are now ready to introduce the steepest descent algorithm for solving the optimization problem (2.1). In this first version, we assume that all evaluations of both the functional J and its gradient J' are done exactly. Hence, we term this version the *Exact Algorithm*.

We need the following notation.

DEFINITION 2.5. *Let $R : V' \to V$ denote the* Riesz operator*, i.e., $R(f)$ is defined for $f \in V'$ as the unique element in V such that $\|R(f)\|_V = 1$ and $\|f\|_{V'} = \langle f, R(f)\rangle$.*

Obviously, $R(f)$ can also be characterized by

$$\langle f, R(f)\rangle = \sup_{v \in V} \frac{\langle f, v\rangle}{\|v\|}.$$

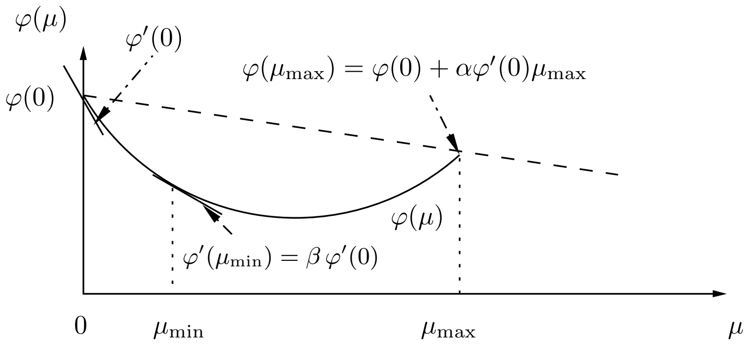

FIG. 2.1. *Conditions on the stepsize imposed by Definition 2.4.*

ALGORITHM 2.6. *(Exact Algorithm) Let $u^{(0)} \in V$ be given. Then, for $k = 0, 1, 2, \ldots$, while $J'(u^{(k)}) \neq 0$, do*

1. *choose the search direction as $s^{(k)} := -R(J'(u^{(k)}))$.*
2. *determine an admissible stepsize $\mu^{(k)} \in \mathcal{A}(J; u^{(k)}, s^{(k)})$.*
3. *update: $u^{(k+1)} := u^{(k)} + \mu^{(k)} s^{(k)}$.* $\square$

PROPOSITION 2.7. *Under Assumption 2.1, the Exact Algorithm converges to u.*

Proof. Without loss of generality we assume that the algorithm produces an infinite sequence of vectors. We do the proof in several steps by adapting the convergence proof of the classical steepest descent method for a convex functional in *finite* dimension, see, e.g., [9].

(i) Using Lemma 2.2 a.), we get

$$\frac{c_J}{p}\|u^{(k+1)} - u^{(k)}\|_V^p \leq [J(u^{(k+1)}) - J(u^{(k)})] - \langle J'(u^{(k)}), u^{(k+1)} - u^{(k)}\rangle$$

Next, condition (2.3) can be equivalently written as

$$-\langle J'(u^{(k)}), u^{(k+1)} - u^{(k)}\rangle \leq \frac{1}{\alpha}[J(u^{(k)}) - J(u^{(k+1)})],$$

thus

$$\frac{c_J}{p}\|u^{(k+1)} - u^{(k)}\|_V^p \leq (\frac{1}{\alpha} - 1)[J(u^{(k)}) - J(u^{(k+1)})]. \tag{2.8}$$

(ii) By construction, we have that $\{J(u^{(k)})\}_{k\in\mathbb{N}_0}$ is monotonically decreasing and bounded from below by $J(u)$, hence

$$\lim_{k\to\infty}[J(u^{(k)}) - J(u^{(k+1)})] = 0.$$

Using (2.8), we obtain $\lim_{k\to\infty}\|u^{(k+1)} - u^{(k)}\|_V = 0$.

(iii) Now we have by condition (2.4)

$$\begin{aligned}
\|J'(u^{(k)})\|_{V'} = \sup_{\|\varphi\|_V=1}\langle J'(u^{(k)}), \varphi\rangle \;&=\; -\langle J'(u^{(k)}), s^{(k)}\rangle \\
&= \langle J'(u^{(k+1)}) - J'(u^{(k)}), s^{(k)}\rangle - \langle J'(u^{(k+1)}), s^{(k)}\rangle \\
&\leq \|J'(u^{(k+1)}) - J'(u^{(k)})\|_{V'}\,\|s^{(k)}\|_V - \beta\langle J'(u^{(k)}), s^{(k)}\rangle \\
&= \|J'(u^{(k+1)}) - J'(u^{(k)})\|_{V'} + \beta\,\|J'(u^{(k)})\|_{V'},
\end{aligned}$$

and hence $\|J'(u^{(k)})\|_{V'} \leq \frac{1}{1-\beta}\|J'(u^{(k+1)}) - J'(u^{(k)})\|_{V'}$.

(iv) The sequence $\{u^{(k)}\}_k$ lies in $\mathcal{R}(u^{(0)})$ since $J(u^{(k)}) \le J(u^{(k-1)}) \le \cdots \le J(u^{(0)})$. Hence by Assumption 2.1 and Lemma 2.2, c.), we obtain

$$\|J'(u^{(k+1)}) - J'(u^{(k)})\|_{V'} \overset{k\to\infty}{\longrightarrow} 0,$$

which by (iii) implies

$$\lim_{k\to\infty} J'(u^{(k)}) = 0. \tag{2.9}$$

(v) Now using Assumption 2.1 and the fact that $J'(u) = 0$, we get

$$\begin{aligned} c_J \|u^{(k)} - u\|_V^p &\le \langle J'(u^{(k)}) - J'(u), u^{(k)} - u\rangle \; = \; \langle J'(u^{(k)}), u^{(k)} - u\rangle \\ &\le \|J'(u^{(k)})\|_{V'} \, \|u^{(k)} - u\|_V \end{aligned}$$

and finally by (2.9)

$$\|u^{(k)} - u\|_V \le \left(\frac{1}{c_J}\|J'(u^{(k)})\|_{V'}\right)^{1/(p-1)} \overset{k\to\infty}{\longrightarrow} 0.$$

This proves the assertion. □

3. Optimization in Sequence Spaces. So far, we considered a minimization problem in an *arbitrary* reflexive Banach space V. In many cases of interest, V is equipped with a Riesz basis Ψ, i.e., the norm in V of an expansion in Ψ is equivalent to a discrete norm of the expansion coefficients (see also (3.1) below). One may think of V being a function space and the basis being the Fourier basis. However, our considerations have been motivated by recent results in wavelet theory, [10, 17]. In particular, we were driven by the results in [11, 12], where an adaptive wavelet method for solving certain operator equations was proven to be asymptotically optimally convergent. One main ingredient for defining the adaptive approximation and for analyzing it, is the transformation of the operator equation into an equivalent discrete problem still on an infinite-dimensional space. We will mimic this approach for our minimization problem.

To this end, let us now consider a basis $\Psi := \{\psi_\lambda : \lambda \in \mathcal{J}\}$ in V and, for any $v = \sum_{\lambda\in\mathcal{J}} v_\lambda \psi_\lambda \in V$, let us denote by $\boldsymbol{v} = (v_\lambda)_{\lambda\in\mathcal{J}}$ the (possibly infinite) sequence of its coefficients; we will use the notation $v = \boldsymbol{v}^T\Psi$. We assume that Ψ is a Riesz basis in V, in the sense that there exists $1 < p < \infty$ and constants $0 < c_\Psi \le C_\Psi < \infty$ such that

$$c_\Psi \, \|\boldsymbol{v}\|_{\ell_p} \le \|v\|_V \le C_\Psi \, \|\boldsymbol{v}\|_{\ell_p}, \quad \text{for all } v = \boldsymbol{v}^T\Psi = \sum_{\lambda\in\mathcal{J}} v_\lambda \, \psi_\lambda \in V. \tag{3.1}$$

Let us also denote by $\langle\cdot,\cdot\rangle$ the duality pairing between $\ell_p = \ell_p(\mathcal{J})$ and $\ell_{p'} = \ell_{p'}(\mathcal{J})$, where $\frac{1}{p} + \frac{1}{p'} = 1$. The isomorphism $v \mapsto \boldsymbol{v}$ induces an isomorphism $F \mapsto \boldsymbol{F}$ between V' and $\ell_{p'}$, by setting $\langle \boldsymbol{F}, \boldsymbol{v}\rangle = \langle F, v\rangle$ for all $\boldsymbol{v} \in \ell_{p'}$. The previous norm equivalences (3.1) yield

$$C_\Psi^{-1} \, \|\boldsymbol{F}\|_{\ell_{p'}} \le \|F\|_{V'} \le c_\Psi^{-1} \, \|\boldsymbol{F}\|_{\ell_{p'}}, \quad \text{for all } F \in V'. \tag{3.2}$$

Thus, we can transfer the minimization in V to a minimization in the sequence space ℓ_p. To this end, let us introduce the functional $\mathbb{J} : \ell_p \to \mathbb{R}$ defined as

$$\mathbb{J}(\boldsymbol{v}) := J(\boldsymbol{v}^T\Psi). \tag{3.3}$$

We use $\mathbb{J}$ (and not $\boldsymbol{J}$) here to indicate that $\mathbb{J}(\boldsymbol{v}) \in \mathbb{R}$ is a number whereas boldface characters always stand for sequences. Then, the minimization problem (2.1) can equivalently be formulated as follows: find $\boldsymbol{u} \in \ell_p$ such that

$$\mathbb{J}(\boldsymbol{u}) = \min_{\boldsymbol{v} \in \ell_p} \mathbb{J}(\boldsymbol{v}). \tag{3.4}$$

Note that the solutions $u \in V$ of (2.1) and $\boldsymbol{u} \in \ell_p$ of (3.4) are related by $u = \boldsymbol{u}^T \Psi$.

The operator $\mathbb{J}$ inherits all the properties of the operator J. In particular, it satisfies the conditions in Assumption 2.1. Its derivative is given by

$$\boldsymbol{J}'(\boldsymbol{v}) = \langle J'(\boldsymbol{v}^T \Psi), \Psi \rangle,$$

where, for any $f \in V'$, we use the notation $\langle f, \Psi \rangle := (\langle f, \psi_\lambda \rangle)_{\lambda \in \mathcal{J}}$. In fact,

$$\begin{aligned} \langle \boldsymbol{J}'(\boldsymbol{v}), \boldsymbol{w} \rangle = \lim_{t \to 0+} \frac{\mathbb{J}(\boldsymbol{v} + t\boldsymbol{w}) - \mathbb{J}(\boldsymbol{v})}{t} &= \lim_{t \to 0+} \frac{J(\boldsymbol{v}^T \Psi + t\boldsymbol{w}^T \Psi) - J(\boldsymbol{v}^T \Psi)}{t} \\ = \langle J'(\boldsymbol{v}^T \Psi), \boldsymbol{w}^T \Psi \rangle \ = \ \boldsymbol{w}^T \langle J'(\boldsymbol{v}^T \Psi), \Psi \rangle \ &= \ \langle \langle J'(\boldsymbol{v}^T \Psi), \Psi \rangle, \boldsymbol{w} \rangle. \end{aligned}$$

Let us now define the operator $\boldsymbol{R} : \ell_{p'} \setminus \{0\} \to \ell_p$, $\frac{1}{p} + \frac{1}{p'} = 1$, by

$$(\boldsymbol{R}(\boldsymbol{f}))_\lambda := \frac{f_\lambda \, |f_\lambda|^{p'-2}}{\|\boldsymbol{f}\|_{\ell_{p'}}^{p'-1}}.$$

We have

$$\langle \boldsymbol{f}, \boldsymbol{R}(\boldsymbol{f}) \rangle = \|\boldsymbol{f}\|_{\ell_{p'}}^{1-p'} \sum_\lambda f_\lambda^2 \, |f_\lambda|^{p'-2} = \|\boldsymbol{f}\|_{\ell_{p'}};$$

since $p = \frac{p'}{p'-1}$, we also have

$$\|\boldsymbol{R}(\boldsymbol{f})\|_{\ell_p} = \|\boldsymbol{f}\|_{\ell_{p'}}^{1-p'} \Big(\sum_\lambda |f_\lambda|^{p(p'-1)} \Big)^{1/p} = \|\boldsymbol{f}\|_{\ell_{p'}}^{1-p'} \Big(\sum_\lambda |f_\lambda|^{p'} \Big)^{(p'-1)/p'} = 1.$$

Note that $\boldsymbol{R}$ coincides with the normalized Riesz operator in the Hilbert case $p = p' = 2$, otherwise the operator is nonlinear.

Now, we can formulate the discrete counterpart of Algorithm 2.6 which reads as follows

ALGORITHM 3.1. *(Fully infinite-dimensional steepest descent method)*
Let $\boldsymbol{u}^{(0)} \in V$ be given.
Then, for $k = 0, 1, 2, \ldots$, while $\boldsymbol{J}'(\boldsymbol{u}^{(k)}) \neq \boldsymbol{0}$, do

1. *determine the search direction $\boldsymbol{s}^{(k)}$ by $\boldsymbol{s}^{(k)} := -\boldsymbol{R}(\boldsymbol{J}'(\boldsymbol{u}^{(k)}))$;*
2. *determine an admissible stepsize $\mu^{(k)} \in \mathcal{A}(\mathbb{J}; \boldsymbol{u}^{(k)}, \boldsymbol{s}^{(k)})$;*
3. *update: $\boldsymbol{u}^{(k+1)} := \boldsymbol{u}^{(k)} + \mu^{(k)} \boldsymbol{s}^{(k)}$.*

The convergence of the latter algorithm to $\boldsymbol{u}$ follows by Proposition 2.7.

4. A General Adaptive Algorithm. The above fully infinite-dimensional algorithm is posed in general terms and is in general not computable. The next step is to replace all non-computable operations by finite approximations. We start by identifying certain routines that are needed within the algorithm. To this end, for any sequence $\boldsymbol{v} \in \ell(\mathcal{J})$, we define supp $\boldsymbol{v}$ as the set of indices corresponding to the non-zero entries, i.e.,

$$\text{supp } \boldsymbol{v} := \{\lambda \in \mathcal{J} : v_\lambda \neq 0\}, \tag{4.1}$$

and we call a vector *compactly (or finitely) supported*, if $\#\,\text{supp } \boldsymbol{v} < \infty$. We will often add the index Λ and use the notation $\boldsymbol{v}_\Lambda$ in order to indicate a compactly supported vector. Note that Λ sometimes (but not necessarily) coincides with supp $\boldsymbol{v}_\Lambda$. When this is the case, we will clearly state it.

There are several issues that we need to treat in order to define a computable version of the descent algorithm. First of all, even if $\boldsymbol{v}_\Lambda$ is a compactly supported vector, in general the gradient $\boldsymbol{J}'(\boldsymbol{v}_\Lambda)$ has infinitely many components, so it cannot be computed exactly. Next, once an approximate descent direction has been found, an admissible stepsize has to be computed. This may require several evaluations of the functional $\mathbb{J}$. Finally, in order to achieve an algorithm of optimal complexity, the negligible components of the current approximation of the minimizer should be removed (possibly not at each iteration), by using a coarsening procedure.

We start by assuming the availability of three basic procedures for approximating the functional and its gradient as well as for thresholding a given vector.

ASSUMPTION 4.1. *A procedure* **EVAL-GRAD:** $[\boldsymbol{v}_\Lambda, \varepsilon] \mapsto \boldsymbol{w}_\Lambda$ *is available: Given a compactly supported vector* $\boldsymbol{v}_\Lambda$ *and a tolerance* $\varepsilon > 0$*, a compactly supported vector* $\boldsymbol{w}_\Lambda$ *is computed, such that* $\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{w}_\Lambda\|_{\ell_{p'}} \leq \varepsilon$.

ASSUMPTION 4.2. *A procedure* **EVAL-J:** $[\boldsymbol{v}_\Lambda, \varepsilon] \mapsto g$ *is available: Given a compactly supported vector* $\boldsymbol{v}_\Lambda$ *and a tolerance* $\varepsilon > 0$*, a real number* g *is computed, such that* $|\mathbb{J}(\boldsymbol{v}_\Lambda) - g| \leq \varepsilon$.

ASSUMPTION 4.3. *A procedure* **THRESH:** $[\boldsymbol{v}_\Lambda, \varepsilon] \mapsto \boldsymbol{z}_\Lambda$ *is available: Given a compactly supported vector* $\boldsymbol{v}_\Lambda$ *and a tolerance* $\varepsilon > 0$*, a compactly supported vector* $\boldsymbol{z}_\Lambda$ *is computed, such that* $\|\boldsymbol{v}_\Lambda - \boldsymbol{z}_\Lambda\|_{\ell_p} \leq \varepsilon$ *and* supp $\boldsymbol{z}_\Lambda \subseteq$ supp $\boldsymbol{v}_\Lambda$ *has minimal cardinality, possibly subject to certain constraints on the distribution of its entries.*

Now we indicate how to use these routines in order to realize the main ingredients of the adaptive algorithm. We use **EVAL-GRAD** to create a procedure, called **APPROX-GRAD**, which yields an admissible descent direction. Next, we combine **EVAL-GRAD** and **EVAL-J** to construct a procedure, called **LINE-SEARCH**, which defines an admissible stepsize. In turns, the routines **APPROX-GRAD** and **LINE-SEARCH** are combined to form **DESCENT** which realizes one descent step of the algorithm. On the other hand, by **THRESH** and **APPROX-GRAD** we construct a procedure **COARSE** in order to coarsen the current iterate. Finally, **DESCENT** and **COARSE** are used to define our general adaptive algorithm **MINIMIZE**. For the sake of clarity, we show the hierarchy of these routines in Figure 4.1.

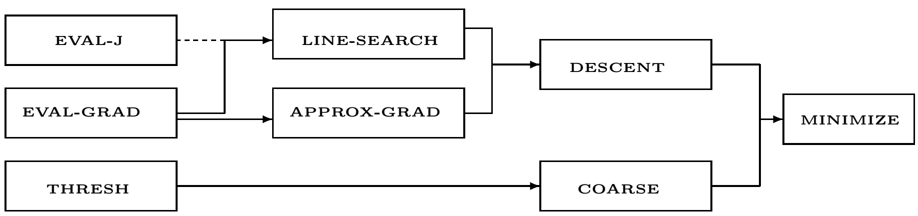

FIG. 4.1. *Hierarchy of procedures.*

4.1. Approximation of the Gradient. We first note that **EVAL-GRAD** may not yield an admissible descent direction. In fact, it may very well happen that the approximate evaluation of the gradient $\boldsymbol{J}'(\boldsymbol{v}_\Lambda)$ gives $\mathbf{0}$ even though $\boldsymbol{v}_\Lambda$ is *not* the minimum of $\mathbb{J}$. Indeed, the approximation may discard many very small entries of $\boldsymbol{J}'(\boldsymbol{v}_\Lambda)$ so that $\boldsymbol{v}_\Lambda$ may even be far away from the minimum. Hence, we have to ensure theoretically that $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ vanishes if and only if the exact gradient vanishes. This is accomplished by the following procedure **APPROX-GRAD**. Besides the input vector $\boldsymbol{v}_\Lambda$ and the accuracy $\varepsilon > 0$, **APPROX-GRAD** takes an extra input parameter, namely $0 < \gamma < 1$ whose meaning will be discussed later on. Finally, $\nu \in (0,1)$ denotes any arbitrarily fixed constant.

APPROX-GRAD: $[\boldsymbol{v}_\Lambda, \varepsilon, \gamma] \mapsto [\boldsymbol{G}(\boldsymbol{v}_\Lambda), \eta]$
1. set $\eta^{(1)} := \varepsilon$;
2. for $n = 1, 2, \ldots$ do
   (a) $\boldsymbol{w}_\Lambda^{(n)} :=$ **EVAL-GRAD**$[\boldsymbol{v}_\Lambda, \eta^{(n)}]$;
   (b) if $\|\boldsymbol{w}_\Lambda^{(n)}\|_{\ell_2} \geq \frac{1+\gamma}{1-\gamma}\eta^{(n)}$, set $\boldsymbol{G}(\boldsymbol{v}_\Lambda) := \boldsymbol{w}^{(n)}$, $\eta := \eta^{(n)}$, RETURN;
   else set $\eta^{(n+1)} := \nu\, \eta^{(n)}$;
3. $\boldsymbol{G}(\boldsymbol{v}_\Lambda) := \mathbf{0}$, $\eta := 0$.

Note that statement 3. is reached only after an infinite loop in 2. Of course, in the computable version of the algorithm, we will insert a stopping criterion in 2., see Section 5.6 below.

The following statement is an immediate consequence of the definition.

Proposition 4.4. *The procedure* **APPROX-GRAD:** $[\boldsymbol{v}_\Lambda, \varepsilon, \gamma] \mapsto [\boldsymbol{G}(\boldsymbol{v}_\Lambda), \eta]$ *has the following property: Given* $0 < \gamma < 1$, $\varepsilon > 0$ *and a finitely supported vector* $\boldsymbol{v}_\Lambda$*, then a vector* $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ *with finite support and a number* $\eta = \eta(\boldsymbol{v}_\Lambda) \in [0, \varepsilon]$ *are computed, such that*

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \eta, \tag{4.2}$$

$$\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \geq \frac{1+\gamma}{1-\gamma}\,\eta. \qquad \square \tag{4.3}$$

Remark 4.5. *Even though* ε *does not appear in* (4.2) *and* (4.3)*, it serves as upper bound for the tolerance* η *determined inside the routine* **APPROX-GRAD**.

The inequalities (4.2) and (4.3) show that the routine **APPROX-GRAD** defines an admissible descent direction whenever $\boldsymbol{J}'(\boldsymbol{v}_\Lambda) \neq \mathbf{0}$. Precisely, the following results, which will play a crucial role in the subsequent analysis, hold.

Proposition 4.6. $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ *satisfies the following properties:*

a.) the inequality

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} - \eta \leq \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} + \eta. \tag{4.4}$$

holds.

b.) $\boldsymbol{J}'(\boldsymbol{v}_\Lambda) = \mathbf{0}$ *if and only if* $\boldsymbol{G}(\boldsymbol{v}_\Lambda) = \mathbf{0}$.

c.) If $\boldsymbol{J}'(\boldsymbol{v}_\Lambda) \neq \mathbf{0}$*, then for* $\boldsymbol{s}_\Lambda := -\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))$*, one has*

$$\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda \rangle \leq -\gamma\, \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}, \tag{4.5}$$

i.e., $\boldsymbol{s}_\Lambda$ *is an admissible descent direction.*

d.) Finally, we have

$$\gamma \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq |\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle| \leq \frac{2}{1+\gamma}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}. \tag{4.6}$$

Proof. The first assertion is an immediate consequence of the triangle inequality. As for b.) if $\boldsymbol{J}'(\boldsymbol{v}_\Lambda) = \mathbf{0}$, then

$$\frac{1+\gamma}{1-\gamma}\,\eta \leq \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \eta$$

which is only possible if $\eta = 0$ and consequently $\boldsymbol{G}(\boldsymbol{v}_\Lambda) = \mathbf{0}$. The converse is trivial.

If $\boldsymbol{J}'(\boldsymbol{v}_\Lambda) \neq \mathbf{0}$, then

$$\begin{aligned}
\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle &= \langle \boldsymbol{G}(\boldsymbol{v}_\Lambda) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))\rangle - \langle \boldsymbol{G}(\boldsymbol{v}_\Lambda), \boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))\rangle \\
&\leq \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} - \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \\
&\leq \eta - (1+\gamma)\eta - \gamma\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \\
&= -\gamma\,(\eta + \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}) \;\leq\; -\gamma\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}
\end{aligned}$$

by (4.4) and Proposition 4.4, which proves c.) Note that with this choice the angle between the gradient and the descent direction is bounded away from 90°.

The first inequality in d.) follows directly from the proof of c.) taking into account that $\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle$ is negative. In fact, as above, we have

$$|\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle| \geq \gamma\,(\eta + \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}) \geq \gamma\,\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}.$$

As for the second inequality, we have

$$\begin{aligned}
|\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle| &\leq |\langle \boldsymbol{G}(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle| + |\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle| \\
&= \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} + \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}.
\end{aligned}$$

By (4.2) and (4.3), we get

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \eta \leq \frac{1-\gamma}{1+\gamma}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}},$$

from which d.) follows. □

Note that **APPROX-GRAD** gives the desired result after a finite number of steps if and only if $\boldsymbol{J}'(\boldsymbol{v}_\Lambda) \neq \mathbf{0}$.

REMARK 4.7. *Proposition 4.4 shows that* **APPROX-GRAD** *gives an approximation of the gradient up to a* relative *accuracy:*

$$\frac{\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}}{\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}} \leq \frac{1-\gamma}{2\gamma}.$$

In fact, using (4.4) *as well as* (4.3) *gives*

$$\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} + \eta \leq \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} + \frac{1-\gamma}{1+\gamma}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}},$$

which implies

$$\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \frac{1+\gamma}{2\gamma}\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}. \tag{4.7}$$

Then, we conclude by (4.2)

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \eta \leq \frac{1-\gamma}{1+\gamma}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}} \leq \frac{1-\gamma}{2\gamma}\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}.$$

4.2. Approximate Descent Step. Let $\boldsymbol{v}_\Lambda$ be a given compactly supported vector, and let $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ be an approximation of the gradient $\boldsymbol{J}'(\boldsymbol{v}_\Lambda)$ produced by **APPROX-GRAD**; let us set $\boldsymbol{s}_\Lambda = -\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))$. Hereafter, we describe how to select an admissible step size. The proposed algorithm is based on a bisection procedure, which yields the admissible step $\mu \in \mathcal{A}(\mathbb{J}; \boldsymbol{v}_\Lambda, \boldsymbol{s}_\Lambda)$ after a finite number of bisections, and only requires the approximate evaluation of the functional $\mathbb{J}$.

Setting as in (2.5) $\varphi(\mu) := \mathbb{J}(\boldsymbol{v}_\Lambda + \mu \boldsymbol{s}_\Lambda)$, we look for μ satisfying (2.6) and the similar condition

$$\varphi(\mu) \geq \varphi(0) + \beta\, \varphi'(0)\, \mu;$$

indeed, the latter condition implies (2.7), thanks to Lagrange's theorem and the monotonicity of φ'. Thus, setting

$$Q(\mu) := \frac{\varphi(\mu) - \varphi(0)}{\mu\, \varphi'(0)} = \frac{\varphi(0) - \varphi(\mu)}{\mu\, |\varphi'(0)|},$$

we look for μ satisfying

$$\alpha \leq Q(\mu) \leq \beta. \tag{4.8}$$

This is possible, since $\lim_{\mu \to 0+} Q(\mu) = 1$ and Q is monotonically decreasing to $-\infty$ for $\mu \to +\infty$. On the other hand, only a computable approximation of $Q(\mu)$ can be used in the search. So, we approximate $Q(\mu)$ by

$$\tilde{Q}(\mu) := \frac{\tilde{\varphi}(0) - \tilde{\varphi}(\mu)}{\mu\, \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}},$$

where $\tilde{\varphi}(0) = \textbf{EVAL-J}[\boldsymbol{v}_\Lambda, \varepsilon]$, $\tilde{\varphi}(\mu) = \textbf{EVAL-J}[\boldsymbol{v}_\Lambda + \mu \boldsymbol{s}_\Lambda, \varepsilon]$ and $\varepsilon := \varrho\, \mu\, \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_{p'}}$, $\varrho > 0$ being a constant to be determined later on. It is easily seen that

$$\frac{2}{1+\gamma} \tilde{Q}(\mu) - \frac{4\varrho}{1+\gamma} \leq Q(\mu) \leq \frac{1}{\gamma} \tilde{Q}(\mu) + \frac{2\varrho}{\gamma}.$$

Thus, (4.8) holds if $\tilde{Q}(\mu)$ satisfies

$$\tilde{\alpha} \leq \tilde{Q}(\mu) \leq \tilde{\beta} \tag{4.9}$$

with

$$\tilde{\alpha} := \frac{1+\gamma}{2} \alpha + 2\varrho \qquad \text{and} \qquad \tilde{\beta} := \gamma\beta - 2\varrho.$$

Note that these bounds are meaningful provided that α, β and γ are chosen such that

$$\beta > \frac{1+\gamma}{2\gamma} \alpha \tag{4.10}$$

holds. In fact, then we have $\gamma\beta - \frac{1+\gamma}{2}\alpha > 0$ and we can choose $\varrho > 0$ such that

$$\varrho < \frac{1}{4}(\gamma\beta - \frac{1+\gamma}{2}\alpha) \tag{4.11}$$

and thus $\tilde{\alpha} = \frac{1+\gamma}{2}\alpha + 2\varrho < \gamma\beta - 2\varrho = \tilde{\beta}$.

Starting from any tentative stepsize and possibly halving or doubling the current stepsize a finite number of times, either we satisfy (4.9), or we find two values $\mu_0^- < \mu_0^+$ such that $\tilde{Q}(\mu_0^-) > \tilde{\beta}$ and $\tilde{Q}(\mu_0^+) < \tilde{\alpha}$. In this case, we can start the classical bisection procedure applied to $\tilde{Q}$. Let us prove that it leads to satisfy (4.9) in a finite number of steps. We argue by contradiction, assuming that the procedure generates an infinite sequence of values $\mu_n^- < \mu_n^+$ such that $\tilde{Q}(\mu_n^+) < \tilde{\alpha}$, $\tilde{Q}(\mu_n^-) > \tilde{\beta}$, $\lim_{n\to\infty}(\mu_n^+ - \mu_n^-) = 0$. Then, there exists $\bar{\mu}$ such that $\lim_{n\to\infty} \mu_n^+ = \lim_{n\to\infty} \mu_n^- = \bar{\mu}$. By continuity, we have $\lim_{n\to\infty}(Q(\mu_n^-) - Q(\mu_n^+)) = 0$. On the other hand,

$$Q(\mu_n^-) \geq \frac{2}{1+\gamma}\,\tilde{Q}(\mu_n^-) - \frac{4\varrho}{1+\gamma} > \frac{2\tilde{\beta} - 4\varrho}{1+\gamma}$$

and

$$Q(\mu_n^+) \leq \frac{1}{\gamma}\,\tilde{Q}(\mu_n^+) + \frac{2\varrho}{\gamma} < \frac{2\varrho + \tilde{\alpha}}{\gamma}$$

so that

$$Q(\mu_n^-) - Q(\mu_n^+) > \omega - \nu\,\varrho$$

with

$$\omega = \omega(\alpha,\beta,\gamma) := \frac{2\gamma}{1+\gamma}\,\beta - \frac{1+\gamma}{2\gamma}\,\alpha, \qquad \text{and} \qquad \nu = \nu(\gamma) := \frac{\gamma}{1+\gamma} + \frac{4}{\gamma}.$$

Choosing α, β and γ such that in addition to (4.10) we also have

$$\beta > \frac{(1+\gamma)^2\alpha + 4\varrho(1+3\gamma)}{4\gamma^2} \tag{4.12}$$

we obtain $\omega - \varrho\nu > 0$, i.e., a contradiction. In fact, a straightforward calculation shows that (4.12) yields

$$\varrho < \frac{4\gamma^2\beta - (1+\gamma)^2\alpha}{4(1+3\gamma)} = \frac{\omega}{\nu}.$$

We summarize the result as follows.

Proposition 4.8. *Given any compactly supported vector $\boldsymbol{v}_\Lambda$, let us set $\boldsymbol{s}_\Lambda = -\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))$), where $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ is an approximation of $\boldsymbol{J}'(\boldsymbol{v}_\Lambda)$ satisfying conditions* (4.2) *and* (4.3). *In addition, let Assumption 4.2 be satisfied. Then, choosing α, β, γ according to* (4.10) *and* (4.12), *we can compute an admissible stepsize $\mu \in \mathcal{A}(\mathbb{J}; \boldsymbol{v}_\Lambda, \boldsymbol{s}_\Lambda)$.* $\square$

A different algorithm for choosing the stepsize will be described under more restrictive assumptions in Section 5.1 below. Both algorithms are particular realizations of a general procedure **LINE-SEARCH** which we define as follows.

Definition 4.9. *We call* **LINE-SEARCH:** $[\boldsymbol{v}_\Lambda, \boldsymbol{G}(\boldsymbol{v}_\Lambda), \alpha, \beta] \mapsto \eta$ *any procedure with the following property: Given α, β satisfying $0 < \alpha < \beta < 1$, a finitely supported vector $\boldsymbol{v}_\Lambda$ and an approximation $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ of its gradient produced by* **APPROX-GRAD**, *and setting $\boldsymbol{s}_\Lambda = -\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))$, then an admissible stepsize $\eta \in \mathcal{A}(\mathbb{J}; \boldsymbol{v}_\Lambda, \boldsymbol{s}_\Lambda)$ is computed.*

By the routines **APPROX-GRAD** and **LINE-SEARCH**, we perform one step of descent, which is detailed in the following routine.

DESCENT: $[\boldsymbol{v}_\Lambda, \varepsilon] \mapsto \boldsymbol{w}_\Lambda$
1. $[\boldsymbol{G}(\boldsymbol{v}_\Lambda), \eta] :=$ **APPROX-GRAD**$[\boldsymbol{v}_\Lambda, \varepsilon, \gamma]$
2. if $\eta = 0$, STOP ($\boldsymbol{v}_\Lambda = \boldsymbol{u}$); else
3. $\boldsymbol{s}_\Lambda := -\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))$;
4. $\mu :=$ **LINE-SEARCH**$[\boldsymbol{v}_\Lambda, \boldsymbol{G}(\boldsymbol{v}_\Lambda), \alpha, \beta]$;
5. $\boldsymbol{w}_\Lambda := \boldsymbol{v}_\Lambda + \mu \boldsymbol{s}_\Lambda$.

4.3. Coarsening. The problem of coarsening a vector is one of the central issues of nonlinear approximation theory (see [10, 23]). Several routines of this type are available in the literature, see, e.g., [11, 12, 13]. However, we are interested in a procedure which guarantees the possibility of significantly reducing the support of $\boldsymbol{v}_\Lambda$ while preserving the value of the functional $\mathbb{J}$ up to a small pertubation. The particular realization of the coarsening that we propose is based on the following result.

LEMMA 4.10. *Let $\boldsymbol{v}_\Lambda$, $\boldsymbol{w}_\Lambda$ be finitely supported vectors such that* supp $\boldsymbol{w}_\Lambda \subseteq$ supp $\boldsymbol{v}_\Lambda =: \Lambda$. *Furthermore, let* $[\boldsymbol{G}(\boldsymbol{w}_\Lambda, \eta] =$ **APPROX-GRAD**$[\boldsymbol{w}_\Lambda, \varepsilon, \gamma]$ *for some* $\varepsilon > 0$ *and* $0 < \gamma < 1$. *Then,*

$$\mathbb{J}(\boldsymbol{w}_\Lambda) - \mathbb{J}(\boldsymbol{v}_\Lambda) \leq \Big(\|\boldsymbol{G}(\boldsymbol{w}_\Lambda)|_\Lambda\|_{\ell_{p'}} + \eta \Big) \|\boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\|_{\ell_p}.$$

Proof. By the convexity of $\mathbb{J}$, we obtain

$$\mathbb{J}(\boldsymbol{w}_\Lambda) - \mathbb{J}(\boldsymbol{v}_\Lambda) \leq \langle \boldsymbol{J}'(\boldsymbol{w}_\Lambda), \boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda \rangle \leq \|\boldsymbol{J}'(\boldsymbol{w})|_\Lambda\|_{\ell_{p'}} \, \|\boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\|_{\ell_p}. \tag{4.13}$$

Next, by (4.2), we have

$$\|\boldsymbol{J}'(\boldsymbol{w})|_\Lambda - \boldsymbol{G}(\boldsymbol{w})|_\Lambda\|_{\ell_{p'}} \leq \|\boldsymbol{J}'(\boldsymbol{w}) - \boldsymbol{G}(\boldsymbol{w})\|_{\ell_{p'}} \leq \eta,$$

whence the result immediately follows. ☐

We are ready to introduce the routine **COARSE**. It depends on a parameter $\vartheta > 0$, which bounds from above the error in the functional, as clarified in the next Proposition 4.11 below.

COARSE: $[\boldsymbol{v}_\Lambda, \vartheta] \to \boldsymbol{w}_\Lambda$
1. $[\boldsymbol{G}(\boldsymbol{v}_\Lambda), \eta^{(0)}] =$ **APPROX-GRAD**$[\boldsymbol{v}_\Lambda, \vartheta, \gamma]$;
2. $A^{(0)} = \|\boldsymbol{G}(\boldsymbol{v})|_\Lambda\|_{\ell_{p'}} + \eta^{(0)}$.
3. For $k = 0, \dots$ do
   (a) $\boldsymbol{w}_\Lambda^{(k)} =$ **THRESH**$[\boldsymbol{v}_\Lambda, \frac{\vartheta}{A^{(k)}}]$;
   (b) $[\boldsymbol{G}(\boldsymbol{w}_\Lambda^{(k)}), \eta^{(k)}] =$ **APPROX-GRAD**$[\boldsymbol{w}_\Lambda^{(k)}, \vartheta, \gamma]$;
   (c) $B^{(k)} = \|\boldsymbol{G}(\boldsymbol{w}_\Lambda^{(k)})|_\Lambda\|_{\ell_{p'}} + \eta^{(k)}$;
   (d) if $B^{(k)} \leq A^{(k)}$, set $\boldsymbol{w}_\Lambda := \boldsymbol{w}_\Lambda^{(k)}$, RETURN;
   (e) else $A^{(k+1)} = \max(B^{(k)}, 2A^{(k)})$.

PROPOSITION 4.11. *Given a finitely supported vector $\boldsymbol{v}_\Lambda$ and a number $\vartheta > 0$, the procedure* **COARSE:** $[\boldsymbol{v}_\Lambda, \vartheta] \mapsto \boldsymbol{w}_\Lambda$ *produces a finitely supported vector $\boldsymbol{w}_\Lambda$ obtained by thresholding $\boldsymbol{v}_\Lambda$ such that $\mathbb{J}(\boldsymbol{w}_\Lambda) < \mathbb{J}(\boldsymbol{v}_\Lambda) + \vartheta$.*

Proof. Thanks to Lemma 4.10 and the fact that the sequence $A^{(k)}$ is geometrically increasing, it is immediate to see that this procedure terminates in a finite number of iterations leading to the claimed inequality. $\square$

An improved version of **COARSE** will be given in Section 5.4 below under more restrictive assumptions.

4.4. The General Convergent Adaptive Algorithm. We are now ready to define a general adaptive algorithm. It depends on the choice of various parameters. A strategy for their selection will be detailed in the next section.

ALGORITHM 4.12. **MINIMIZE**
Let $\boldsymbol{u}^{(0)} = \boldsymbol{u}_\Lambda^{(0)} \in \ell_p$ *be given. Fix constants* α, β, γ *satisfying* $0 < \alpha < \beta < 1$ *and* $0 < \gamma < 1$, *which enter into the definition of* **DESCENT**.

For $m = 0, 1, \ldots,$ *do*
$\quad \boldsymbol{v}_\Lambda^{(m,0)} := \boldsymbol{u}_\Lambda^{(m)}$
$\quad$ *choose an integer* $K^{(m)} \geq 1$
$\quad$ *For* $k = 0, 1, \ldots, K^{(m)} - 1$, *do*
$\quad\quad$ *choose* $\varepsilon^{(m,k)} > 0$
$\quad\quad \boldsymbol{v}_\Lambda^{(m,k+1)} := \textbf{DESCENT}[\boldsymbol{v}_\Lambda^{(m,k)}, \varepsilon^{(m,k)}]$
$\quad$ *End*
$\quad$ *choose* $\vartheta^{(m)} > 0$
$\quad \boldsymbol{u}_\Lambda^{(m+1)} := \textbf{COARSE}[\boldsymbol{v}_\Lambda^{(m,K^{(m)})}, \vartheta^{(m)}]$
End

THEOREM 4.13. *Let the sequence* $(K^{(m)})_m$ *be arbitrary. Assume that the sequence* $(\varepsilon^{(m,k)})_{m,k}$ *satisfies*

$$\lim_{m\to\infty} \sup_k \varepsilon^{(m,k)} = 0, \tag{4.14}$$

whereas the sequence $(\vartheta^{(m)})_m$ *satisfies*

$$\vartheta^{(m)} \leq \frac{1}{2}[\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})})]. \tag{4.15}$$

Then, Algorithm 4.12 **MINIMIZE** *either yields* $\boldsymbol{u}$ *after a finite number of steps, or produces an infinite sequence* $(\boldsymbol{u}_\Lambda^{(m)})_m$ *which converges to* $\boldsymbol{u}$.

Proof. If, for some m and k, the procedure **DESCENT** stops, then $\boldsymbol{v}_\Lambda^{(m,k)} = \boldsymbol{u}$. Otherwise, the algorithm produces an infinite sequence of vectors. Assuming to be in the latter case, we first show that the sequence $(\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}))_m$ is strictly decreasing. Let m be fixed. By definition of **DESCENT**, we have

$$\mathbb{J}(\boldsymbol{v}_\Lambda^{(m,k+1)}) < \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,k)}), \qquad k = 0, 1, \ldots, K^{(m)} - 1, \tag{4.16}$$

whence

$$\mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) < \mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}). \tag{4.17}$$

Furthermore, by definition of **COARSE** and assumption (4.15), we get

$$\mathbb{J}(\boldsymbol{u}_\Lambda^{(m+1)}) \leq \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) + \vartheta^{(m)} \leq \frac{1}{2}[\mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) + \mathbb{J}(\boldsymbol{u}_\Lambda^{(m)})], \tag{4.18}$$

from which we obtain the desired result

$$\mathbb{J}(\boldsymbol{u}_\Lambda^{(m+1)}) - \mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) \leq \frac{1}{2}[\mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) - \mathbb{J}(\boldsymbol{u}_\Lambda^{(m)})] < 0. \tag{4.19}$$

Next, since the functional $\mathbb{J}$ is bounded from below, we deduce that

$$\lim_{m\to\infty}[\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) - \mathbb{J}(\boldsymbol{u}_\Lambda^{(m+1)})] = 0. \tag{4.20}$$

Using (4.18) again, we easily get $\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) < 2[\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) - \mathbb{J}(\boldsymbol{u}_\Lambda^{(m+1)})]$. Hence, we also have that

$$\lim_{m\to\infty}\vartheta^{(m)} = \lim_{m\to\infty}\left[\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})})\right] = 0. \tag{4.21}$$

Thanks to (4.16), we obtain

$$\lim_{m\to\infty}\sup_k[\mathbb{J}(\boldsymbol{v}_\Lambda^{(m,k)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,k+1)})] = 0.$$

Exactly as in the proof of Proposition 2.7, item (i), the latter result implies

$$\lim_{m\to\infty}\sup_k\|\boldsymbol{v}_\Lambda^{(m,k+1)} - \boldsymbol{v}_\Lambda^{(m,k)}\|_{\ell_p} = 0. \tag{4.22}$$

We now observe that, by (4.16) and (4.19), the sequence $(\boldsymbol{v}_\Lambda^{(m,k)})_{m,k}$ is contained in the bounded set $\mathcal{R}(u^{(0)})$. Using the uniform continuity of $\boldsymbol{J}'$ on bounded sets, we obtain from (4.22)

$$\lim_{m\to\infty}\sup_k\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k+1)}) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} = 0. \tag{4.23}$$

Next, setting $[\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)}), \eta^{(m,k)}] := \textbf{APPROX-GRAD}[\boldsymbol{v}_\Lambda^{(m,k)}, \varepsilon^{(m,k)}, \gamma]$ and using (4.2) as well as step 3 of **DESCENT**, we have, for $0 \le k < K^{(m)}$,

$$\begin{aligned}
\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} - \eta^{(m,k)} \le \|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} &= -\langle \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)}), \boldsymbol{s}_\Lambda^{(m,k)}\rangle \\
&= \langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)}) - \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)}), \boldsymbol{s}_\Lambda^{(m,k)}\rangle \\
&\quad + \langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k+1)}) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)}), \boldsymbol{s}_\Lambda^{(m,k)}\rangle \\
&\quad - \langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k+1)}), \boldsymbol{s}_\Lambda^{(m,k)}\rangle.
\end{aligned}$$

Now, (2.4) and (4.2) yield

$$\begin{aligned}
&\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} - \eta^{(m,k)} \le \\
&\quad \le \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)}) - \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(k)})\|_{\ell_{p'}} + \|\boldsymbol{J}'(\boldsymbol{v}^{(m,k+1)}) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} \\
&\qquad - \beta\ \langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)}), \boldsymbol{s}_\Lambda^{(m,k)}\rangle \\
&\quad \le \eta^{(m,k)} + \|\boldsymbol{J}'(\boldsymbol{v}^{(m,k+1)}) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} + \beta\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}},
\end{aligned}$$

thus

$$(1-\beta)\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} \le \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)}) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k+1)})\|_{\ell_{p'}} + 2\eta^{(m,k)}.$$

By definition of **APPROX-GRAD** and assumption (4.14), we get $\eta^{(m,k)} \le \varepsilon^{(m,k)} \to 0$ as $m \to \infty$, uniformly in k. This and (4.23) imply

$$\lim_{m\to\infty}\sup_k\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}} = 0.$$

Since $\|\boldsymbol{v}_\Lambda^{(m,k)} - \boldsymbol{u}\|_{\ell_p} \le \left(\frac{1}{c_{J,\Psi}}\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}}\right)^{1/(p-1)}$, we conclude that

$$\lim_{m\to\infty} \sup_k \|\boldsymbol{v}_\Lambda^{(m,k)} - \boldsymbol{u}\|_{\ell_p} = 0.$$

In particular, since $\boldsymbol{v}_\Lambda^{(m,0)} = \boldsymbol{u}_\Lambda^{(m)}$, this implies the claimed result. □

A few remarks are in order.

REMARK 4.14. (i) Fulfilling assumption (4.15) may be accomplished as follows. By Wolfes condition, the value of the functional is decreased when going from $\boldsymbol{v}_\Lambda^{(m,k)}$ to $\boldsymbol{v}_\Lambda^{(m,k+1)}$, provided the gradient is not yet zero causing the algorithm to stop. Precisely, setting $\boldsymbol{v}_\Lambda^{(m,k+1)} = \boldsymbol{v}_\Lambda^{(m,k)} + \mu^{(k)}\boldsymbol{s}^{(k)}$, we have by (2.3), (4.5) and (4.7)

$$\begin{aligned}\boldsymbol{J}(\boldsymbol{v}_\Lambda^{(m,k)}) - \boldsymbol{J}(\boldsymbol{v}_\Lambda^{(m,k+1)}) &\ge -\alpha\mu^{(k)}\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)}), \boldsymbol{s}_\Lambda^{(m,k)}\rangle \;\ge\; \alpha\gamma\,\mu^{(k)}\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}}\\ &\ge \alpha\frac{2\gamma^2}{1+\gamma}\mu^{(k)}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}}.\end{aligned}$$

Then, we obtain by using a telescopic sum

$$\begin{aligned}\mathbb{J}(\boldsymbol{u}_\Lambda^{(m)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) = \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,0)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) &= \sum_{k=0}^{K^{(m)}-1} \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,k)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,k+1)})\\ &\ge \frac{2\alpha\gamma^2}{1+\gamma}\sum_{k=0}^{K^{(m)}-1} \mu^{(k)}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}},\end{aligned}$$

i.e., we can choose

$$\vartheta^{(m)} := \frac{\alpha\gamma^2}{1+\gamma}\sum_{k=0}^{K^{(m)}-1} \mu^{(k)}\|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_{p'}},$$

which is in fact computable.

(ii) In view of (4.21), a natural way to satisfy assumption (4.14) is to set

$$\varepsilon^{(m,k)} := \begin{cases} 1 & \text{if } m = 0,\\ \vartheta^{(m-1)} & \text{if } m \ge 1,\end{cases} \qquad k = 0, 1, \ldots, K^{(m)} - 1.$$

5. An Adaptive Algorithm with Convergence Rate. From now on, we specialize our analysis to the case $p = 2$. Furthermore, we assume that the Fréchet derivative J' is Lipschitz continuous on each bounded subset of V. Precisely, recalling that the set $\mathcal{R}(u^{(0)})$ (see Lemma 2.2) is bounded, we assume the existence of a constant $L_J > 0$, possibly depending on $u^{(0)}$, such that

$$\|J'(v) - J'(w)\|_{V'} \le L_J\,\|v - w\|_V, \quad v, w \in \mathcal{R}(u^{(0)}). \tag{5.1}$$

Recalling the norm equivalences (3.1) and (3.2) and setting $L_{J,\Psi} = L_J C_\Psi^2$, we obtain the discrete form of the previous inequality, namely

$$\|\boldsymbol{J}'(\boldsymbol{v}) - \boldsymbol{J}'(\boldsymbol{w})\|_{\ell_2} \le L_{J,\Psi}\,\|\boldsymbol{v} - \boldsymbol{w}\|_{\ell_2}, \quad \boldsymbol{v}, \boldsymbol{w} \in \mathcal{R}(u^{(0)}). \tag{5.2}$$

Recalling Assumption 2.1 (ii) and setting $c_{J,\Psi} = c_J c_\Psi^2$, we also get

$$(5.3)\; c_{J,\Psi}\|\boldsymbol{v}-\boldsymbol{w}\|_{\ell_2}^2 \le \langle \boldsymbol{J}'(\boldsymbol{v}) - \boldsymbol{J}'(\boldsymbol{w}), \boldsymbol{v}-\boldsymbol{w}\rangle \le L_{J,\Psi}\|\boldsymbol{v}-\boldsymbol{w}\|_{\ell_2}^2, \quad \boldsymbol{v},\boldsymbol{w} \in \mathcal{R}(u^{(0)}).$$

In this section, we present a particular realization of the abstract adaptive algorithm **MINIMIZE**, which exploits the extra properties of the functional $\mathbb{J}$ stated above. The convergence result of the algorithm will be supplemented by a precise estimate of the rate of decay of the error. This will allow us to determine the number of iterations needed to reach a target tolerance, as well as to relate the error of the algorithm to the best approximation error of the solution.

We start by describing a more efficient line-search algorithm than the one prescribed in Section 4.2; it neither requires the evaluation of the functional, nor additional approximate evaluations of the gradient other then the already computed descent direction.

5.1. Line Search. Given a compactly supported vector $\boldsymbol{v}_\Lambda \in \mathcal{R}(u^{(0)})$ and an approximation $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ of $\boldsymbol{J}'(\boldsymbol{v}_\Lambda)$ which satisfies conditions (4.2) and (4.3), we set $\boldsymbol{s}_\Lambda = -\boldsymbol{R}(\boldsymbol{G}(\boldsymbol{v}_\Lambda))$ and we define a closed interval with computable endpoints, contained in the interval $\mathcal{A}(\mathbb{J};\boldsymbol{v}_\Lambda,\boldsymbol{s}_\Lambda)$. To this end, let us set $\varphi(\mu) = \mathbb{J}(\boldsymbol{v}_\Lambda + \mu\boldsymbol{s}_\Lambda)$, so that $\varphi'(\mu) = \langle \mathbb{J}(\boldsymbol{v}_\Lambda + \mu\boldsymbol{s}_\Lambda), \boldsymbol{s}_\Lambda\rangle$ and let us recall that $\mu \in \mathcal{A}(\mathbb{J};\boldsymbol{v}_\Lambda,\boldsymbol{s}_\Lambda)$ if and only if conditions (2.6) and (2.7) are satisfied. In this subsection, we actually assume that (5.2), and consequently (5.3), holds indeed in a bounded set larger than $\mathcal{R}(u^{(0)})$, namely in the neighborhood of $\mathcal{R}(u^{(0)})$ of radius $\bar{\mu} := \mathrm{diam}(\mathcal{R}(u^{(0)}))$ (in the ℓ_2-distance); obviously, this is not restrictive at all, since it amounts to properly (re-)defining the constant L_J. Thus, if we choose $\boldsymbol{v} = \boldsymbol{v}_\Lambda$ and $\boldsymbol{w} = \boldsymbol{v}_\Lambda + \mu\boldsymbol{s}_\Lambda$ in (5.3), we easily obtain

$$c_{J,\Psi}\,\mu \le \varphi'(\mu) - \varphi'(0) \le L_{J,\Psi}\,\mu, \qquad \mu \in [0,\bar{\mu}]. \tag{5.4}$$

Using the right-hand side of (5.4), we get for some $\theta \in (0,1)$

$$\varphi(\mu) = \varphi(0) + \varphi'(\theta\,\mu)\mu \le \varphi(0) + \varphi'(0)\,\mu + \theta\,L_{J,\Psi}\,\mu^2.$$

Hence, (2.6) is fulfilled if μ satisfies

$$\varphi'(0)\,\mu + L_{J,\Psi}\,\mu^2 \le \alpha\,\varphi'(0)\,\mu, \qquad \text{i.e.,} \qquad L_{J,\Psi}\,\mu \le (1-\alpha)\,|\varphi'(0)|.$$

Taking into account the left-hand side of (4.6), we get the sufficient condition for the validity of (2.6)

$$\mu \le \mu^* := \gamma\,\frac{1-\alpha}{L_{J,\Psi}}\,\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_2},$$

provided $\mu^* \le \bar{\mu}$. This is indeed the case, since by (4.7) and (5.2) we have

$$\mu^* \le \frac{1-\alpha}{L_{J,\Psi}}\,\frac{1+\gamma}{2}\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \le (1-\alpha)\,\frac{1+\gamma}{2}\,\|\boldsymbol{v}_\Lambda - \boldsymbol{u})\|_{\ell_2} < \bar{\mu}.$$

On the other hand, using the inequality on the left-hand side, condition (2.7) is fulfilled if μ satisfies

$$\varphi'(0) + c_{J,\Psi}\,\mu \ge \beta\,\varphi'(0), \qquad \text{i.e.,} \qquad c_{J,\Psi}\,\mu \ge (1-\beta)\,|\varphi'(0)|.$$

By the right-hand side of (4.6), we get the sufficient condition for the validity of (2.7)

$$\mu \geq \mu_* := \frac{2}{1+\gamma}\frac{1-\beta}{c_{J,\Psi}}\,\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_2}.$$

Obviously, we have to require that $\mu_* \leq \mu^*$, i.e,

$$(1-\beta) \leq \frac{\gamma(\gamma+1)}{2}\frac{c_{J,\Psi}}{L_{J,\Psi}}(1-\alpha), \tag{5.5}$$

which can always be satisfied, e.g., by fixing α and choosing β as close to 1 as needed.

Thus, we have obtained the following result.

PROPOSITION 5.1. *For any $\alpha \in (0,1)$ and any $\gamma \in (0,1)$, there exists β^* satisfying $\alpha \leq \beta^* < 1$ such that for all $\beta \in [\beta^*, 1)$, the interval*

$$\left[\frac{2}{1+\gamma}\frac{1-\beta}{c_{J,\Psi}}\,\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_2}, \gamma\,\frac{1-\alpha}{L_{J,\Psi}}\,\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_2}\right] \tag{5.6}$$

is non-empty and contained in $\mathcal{A}(\mathbb{J}; \boldsymbol{v}_\Lambda, \boldsymbol{s}_\Lambda)$. $\square$

Consequently, the output of the procedure **LINE-SEARCH**$[\boldsymbol{v}_\Lambda, \alpha, \beta]$ can be defined by picking any value in this interval.

5.2. Error Reduction in One Descent Step. Let $\boldsymbol{v}_\Lambda \in \mathcal{R}(u^{(0)})$ be any compactly supported approximation of the exact minimizer $\boldsymbol{u}$. We apply the routine **DESCENT** to it and, assuming $\boldsymbol{v}_\Lambda \neq \boldsymbol{u}$, we get a new approximation $\boldsymbol{w}_\Lambda$. Hereafter, we are interested in studying the behaviour of the approximation error in going from $\boldsymbol{v}_\Lambda$ to $\boldsymbol{w}_\Lambda$. In the analysis, the following definition will be useful.

DEFINITION 5.2. *For any compactly supported vector $\boldsymbol{v}_\Lambda$, we set*

$$\mathbb{E}(\boldsymbol{v}_\Lambda) := \mathbb{J}(\boldsymbol{v}_\Lambda) - \mathbb{J}(\boldsymbol{u}).$$

The quantity $\mathbb{E}(\boldsymbol{v}_\Lambda)$ is an a-priori error bound; indeed, by (2.2) and (3.1), we get

$$\|\boldsymbol{v}_\Lambda - \boldsymbol{u}\|_{\ell_2}^2 \leq \frac{2}{c_{J,\Psi}}\mathbb{E}(\boldsymbol{v}_\Lambda). \tag{5.7}$$

In order to compare $\mathbb{E}(\boldsymbol{w}_\Lambda)$ to $\mathbb{E}(\boldsymbol{v}_\Lambda)$, we recall the definition of $\boldsymbol{w}_\Lambda = \boldsymbol{v}_\Lambda + \mu\,\boldsymbol{s}_\Lambda$ given at point 5. of **DESCENT**; from condition (2.3) and inequality (4.5), we immediately get

$$\mathbb{E}(\boldsymbol{w}_\Lambda) \leq \mathbb{E}(\boldsymbol{v}_\Lambda) - \alpha\gamma\mu\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2}. \tag{5.8}$$

We now establish two technical results which will be used in the sequel.

LEMMA 5.3. *Let $\mu =$ **LINE-SEARCH**$[\boldsymbol{v}_\Lambda, \alpha, \beta]$. Then the following inequality holds*

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \leq \frac{L_{J,\Psi}}{\gamma(1-\beta)}\,\mu.$$

Proof. We start by the trivial identity

$$-\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle = \langle \boldsymbol{J}'(\boldsymbol{w}_\Lambda) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle - \langle \boldsymbol{J}'(\boldsymbol{w}_\Lambda), \boldsymbol{s}_\Lambda\rangle.$$

Using (5.3) and condition (2.4), we obtain

$$-\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle \leq L_{J,\Psi}\|\boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\|_{\ell_2} - \beta\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{s}_\Lambda\rangle,$$

i.e.,

$$-\langle \boldsymbol{J}'(\boldsymbol{w}_\Lambda), \boldsymbol{s}_\Lambda\rangle \le \frac{L_{J,\Psi}}{1-\beta}\,\|\boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\|_{\ell_2}.$$

We conclude by (4.5). □

Lemma 5.4. *The following inequality holds*

$$\mathbb{E}(\boldsymbol{v}_\Lambda) \le \frac{1}{2c_{J,\Psi}}\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|^2_{\ell_2}.$$

Proof. We apply again (2.2) and (3.1) to get

$$\begin{aligned}
\mathbb{J}(\boldsymbol{u}) - \mathbb{J}(\boldsymbol{v}_\Lambda) &\ge \langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{u}-\boldsymbol{v}_\Lambda\rangle + \frac{c_{J,\Psi}}{2}\|\boldsymbol{u}-\boldsymbol{v}_\Lambda\|^2_{\ell_2}\\
&\ge \min_{\boldsymbol{z}\in\ell_2}\left\{\langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{z}\rangle + \frac{c_{J,\Psi}}{2}\|\boldsymbol{z}\|^2_{\ell_2}\right\}\\
&= \min_{\boldsymbol{z}\in\ell_2}\sum_{\lambda\in\mathcal{J}}\left\{\boldsymbol{J}'(\boldsymbol{v}_\Lambda)_\lambda\, z_\lambda + \frac{c_{J,\Psi}}{2} z_\lambda^2\right\}.
\end{aligned}$$

We note that we can minimize along each component λ independently for each $\lambda \in \mathcal{J}$; the minimum is attained at $z_\lambda = -\frac{1}{c_{J,\Psi}}\,\boldsymbol{J}'(\boldsymbol{v}_\Lambda)_\lambda$. We conclude that

$$\mathbb{J}(\boldsymbol{u}) - \mathbb{J}(\boldsymbol{v}_\Lambda) \ge -\frac{1}{2c_{J,\Psi}}\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|^2_{\ell_2},$$

which is precisely the thesis. □

We are now ready to estimate the error reduction guaranteed by one application of **DESCENT**.

Proposition 5.5. *Given a compactly supported vector $\boldsymbol{v}_\Lambda \in \mathcal{R}(u^{(0)})$ and any tolerance ε, define $\boldsymbol{w}_\Lambda :=$ **DESCENT**$(\boldsymbol{v}_\Lambda, \varepsilon)$. Let α, β be any fixed constants satisfying Wolfes condition in Definition 2.4. Then, setting*

$$\sigma := 1 - 2\frac{c_{J,\Psi}}{L_{J,\Psi}}\,\gamma^2\,(1-\beta)\,\alpha < 1, \tag{5.9}$$

we have

$$\mathbb{E}(\boldsymbol{w}_\Lambda) \le \sigma\,\mathbb{E}(\boldsymbol{v}_\Lambda).$$

Proof. By the two previous lemmas, we obtain

$$\mu\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \ge \frac{\gamma\,(1-\beta)}{L_{J,\Psi}}\,\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|^2_{\ell_2} \ge 2\frac{c_{J,\Psi}}{L_{J,\Psi}}\,\gamma\,(1-\beta)\,\mathbb{E}(\boldsymbol{v}_\Lambda).$$

The result follows from (5.8). □

Remark 5.6. The expression (5.9) for the error reduction factor σ suggests a simple strategy for selecting the two parameters α and β which appear in (2.3) and (2.4). Indeed, we observe that σ is monotonically decreasing as $1-\beta$ increases. Recalling condition (5.5), we are lead to choose

$$1-\beta = \frac{\gamma(\gamma+1)}{2}\,\frac{c_{J,\Psi}}{L_{J,\Psi}}\,(1-\alpha). \tag{5.10}$$

In this case, the interval (5.6) reduces to a point, which is the chosen stepsize, i.e., we set

$$\mu := \gamma \, \frac{1-\alpha}{L_{J,\Psi}} \, \|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\ell_2}.$$

Substituting (5.10) into (5.9), we see that σ is minimized with respect to α by the choice $\alpha = \frac{1}{2}$. With this value for α, we finally obtain the error reduction rate

$$\sigma_{\mathrm{opt}} := 1 - \left(\frac{c_{J,\Psi}}{L_{J,\Psi}}\right)^2 \frac{\gamma^3\,(1+\gamma)}{4}. \tag{5.11}$$

As a by-product of the previous analysis, we obtain next result, which describes the interplay between the available error control quantities.

Proposition 5.7. *The following chain of inequalities holds*

$$\|\boldsymbol{v}_\Lambda - \boldsymbol{u}\|^2_{\ell_2} \le \frac{2}{c_{J,\Psi}}\, \mathbb{E}(\boldsymbol{v}_\Lambda) \le \frac{2}{c^2_{J,\Psi}}\, \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|^2_{\ell_2} \le 2\left(\frac{L_{J,\Psi}}{c_{J,\Psi}}\right)^2 \|\boldsymbol{v}_\Lambda - \boldsymbol{u}\|^2_{\ell_2},$$

for all $\boldsymbol{v}_\Lambda \in \mathcal{R}(u^{(0)})$.

Proof. It is enough to combine (5.7), Lemma 5.4, and (5.2) with $\boldsymbol{v} = \boldsymbol{v}_\Lambda$ and $\boldsymbol{w} = \boldsymbol{u}$. □

The result means that $(\mathbb{E}(\boldsymbol{v}_\Lambda))^{1/2}$ is an a-priori error bound both from above and from below; similarly $\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2}$ is an upper and lower a-posteriori error estimator. Of course, in general, none of them is computable in practice. However, by using inequalities (4.4), we can easily obtain from the latter one a computable a-posteriori error estimator.

5.3. Improving the Error Estimate. In this section, we show that the application of **APPROX-GRAD** within the procedure **DESCENT** may lead to the improvement of the currently available estimate of the quantity $\mathbb{E}$. This feature will be crucial in the design of the adaptive algorithm considered in the next section.

Let $\boldsymbol{v}_\Lambda$ be any compactly supported vector and let $\mathbb{E}_\Lambda$ be any upper estimate of $\mathbb{E}(\boldsymbol{v}_\Lambda)$, i.e., $\mathbb{E}(\boldsymbol{v}_\Lambda) \le \mathbb{E}_\Lambda$. By Proposition 5.7, there exists a constant $\mathcal{C}_1$ (whose expression is easily computable) such that

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \le \mathcal{C}_1 \mathbb{E}(\boldsymbol{v}_\Lambda)^{1/2} \le \mathcal{C}_1 \mathbb{E}_\Lambda^{1/2}. \tag{5.12}$$

Let us enter **APPROX-GRAD**$[\boldsymbol{v}_\Lambda, \varepsilon, \gamma]$ with the choice $\varepsilon = \mathcal{C}_1 \mathbb{E}_\Lambda^{1/2}$. Observe that whenever the loop in **APPROX-GRAD** does not stop, i.e., whenever for some n we have $\|\boldsymbol{w}_\Lambda^{(n)}\|_{\ell_2} < \frac{1+\gamma}{1-\gamma}\eta^{(n)}$, then we get a new a-priori information on the norm of the gradient, namely

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \le \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda) - \boldsymbol{w}_\Lambda^{(n)}\|_{\ell_2} + \|\boldsymbol{w}_\Lambda^{(n)}\|_{\ell_2} < \eta^{(n)} + \frac{1+\gamma}{1-\gamma}\eta^{(n)} = \frac{2}{1-\gamma}\eta^{(n)}.$$

Thus, if $\bar{n}$ denotes the number of applications of **EVAL-GRAD** within **APPROX-GRAD**, then we know that

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \le \frac{2}{1-\gamma}\nu^{\bar{n}-1}\varepsilon = \mathcal{C}_1 \frac{2}{1-\gamma}\nu^{\bar{n}-1}\mathbb{E}_\Lambda^{1/2}\,. \tag{5.13}$$

Again by Proposition 5.7 we have, for a suitable constant $\mathcal{C}_2$,

$$\mathbb{E}(\boldsymbol{v}_\Lambda) \le \mathcal{C}_2 \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2}^2 \le \mathcal{C}_2 \mathcal{C}_1^2 \frac{4}{(1-\gamma)^2} \nu^{2\bar{n}-2}\, \mathbb{E}_\Lambda. \tag{5.14}$$

If we define n_0 as the smallest integer ≥ 1 such that

$$\mathcal{C}_2 \mathcal{C}_1^2 \frac{4}{(1-\gamma)^2} \le \nu^{-2(n_0-1)}, \tag{5.15}$$

then we obtain

$$\mathbb{E}(\boldsymbol{v}_\Lambda) \le \nu^{2(\bar{n}-n_0)}\, \mathbb{E}_\Lambda.$$

Obviously, we have to take the most accurate estimate between this one and the initial one $\mathbb{E}(\boldsymbol{v}_\Lambda) \le \mathbb{E}_\Lambda$. Denoting by $(z)_+$ the positive part of a number z, we conclude that at the output of an application of **APPROX-GRAD**, in which $\bar{n}$ applications of **EVAL-GRAD** have been made, we know that

$$\mathbb{E}(\boldsymbol{v}_\Lambda) \le \nu^{2(\bar{n}-n_0)_+}\, \mathbb{E}_\Lambda. \tag{5.16}$$

5.4. Coarsening. The coarsening procedure **COARSE:** $[\boldsymbol{v}_\Lambda, \vartheta] \mapsto \boldsymbol{w}_\Lambda$ as described in Section 4.3 can be simplified and optimized by exploiting the assumption (5.1). Under the sole condition that an estimate of $\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2}$ is known, we can get $\boldsymbol{w}_\Lambda$ without any application of **APPROX-GRAD**, by calling once the procedure **THRESH** introduced therein. Indeed, by the convexity of $\mathbb{J}$ and by (5.1), we have for any $\boldsymbol{w}_\Lambda \in \mathcal{R}(u^{(0)})$

$$\begin{aligned}\mathbb{J}(\boldsymbol{w}_\Lambda) - \mathbb{J}(\boldsymbol{v}_\Lambda) &\le \langle \boldsymbol{J}'(\boldsymbol{w}_\Lambda), \boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\rangle \\ &= \langle \boldsymbol{J}'(\boldsymbol{w}_\Lambda) - \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\rangle + \langle \boldsymbol{J}'(\boldsymbol{v}_\Lambda), \boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\rangle \\ &\le L_{J,\Psi} \|\boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\|_{\ell_2}^2 + \|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \|\boldsymbol{w}_\Lambda - \boldsymbol{v}_\Lambda\|_{\ell_2}.\end{aligned}$$

Assume that $\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda)\|_{\ell_2} \le \xi$ for some $\xi > 0$. Then, setting

$$\varepsilon_\vartheta := \min\left(\left(\frac{\vartheta}{2L_{J,\Psi}} \right)^{1/2}, \frac{\vartheta}{2\xi} \right) \qquad \text{and} \qquad \boldsymbol{w}_\Lambda := \textbf{THRESH}[\boldsymbol{v}_\Lambda, \varepsilon_\vartheta], \tag{5.17}$$

we obtain the desired coarsened vector $\boldsymbol{w}_\Lambda$.

5.5. The Adaptive Algorithm. Denote by $\mathbb{E}_0$ any computable estimate of the initial error $\mathbb{E}(\boldsymbol{u}_\Lambda^{(0)})$, i.e., choose $\mathbb{E}_0$ so that

$$\mathbb{E}(\boldsymbol{u}_\Lambda^{(0)}) := \mathbb{J}(\boldsymbol{u}_\Lambda^{(0)}) - \mathbb{J}(\boldsymbol{u}) \le \mathbb{E}_0. \tag{5.18}$$

Let $\nu \in (0,1)$ be a fixed constant. We now prove by recursion that the parameters appearing in Algorithm 4.12 **MINIMIZE** can be chosen in such a way that either the exact solution $\boldsymbol{u}$ is obtained after a finite number of steps, or there exists a strictly increasing sequence of integers e_m with $e_0 = 0$ such that

$$\mathbb{E}(\boldsymbol{u}_\Lambda^{(m)}) \le \nu^{2e_m}\, \mathbb{E}_0; \tag{5.19}$$

furthermore, denoting by N_m the number of applications of **EVAL-GRAD** to get $\boldsymbol{u}_\Lambda^{(m)}$, we will relate N_m to e_m.

For $m = 0$, inequality (5.19) is precisely (5.18). By induction, we assume that (5.19) holds up to some $m \geq 0$. We set $e_{m,0} = 0$ and we prove by recursion that there exists a non-decreasing sequence of integers $e_{m,k}$ such that

$$\mathbb{E}(\boldsymbol{v}_\Lambda^{(m,k)}) \leq \mathbb{E}_{m,k}, \quad \text{with } \mathbb{E}_{m,k} := \sigma^k \nu^{2(e_m + e_{m,k})}\, \mathbb{E}_0, \qquad k = 0, 1, \ldots \tag{5.20}$$

By (5.19), this inequality holds for $k = 0$. By induction, assume that it holds up to some $k \geq 0$. Set

$$\varepsilon^{(m,k)} := \mathcal{C}_1 \mathbb{E}_{m,k}^{1/2} \tag{5.21}$$

and apply **DESCENT**$[\boldsymbol{v}_\Lambda^{(m,k)}, \varepsilon^{(m,k)}]$. If the output η of **APPROX-GRAD** within **DESCENT** is zero, we have found the exact solution and the algorithm stops. Otherwise, let $\bar{n}_{m,k}$ denote the number of applications of **EVAL-GRAD** inside the routine **DESCENT**$[\boldsymbol{v}_\Lambda^{(m,k)}, \varepsilon^{(m,k)}]$. Recalling (5.16) and Proposition 5.5, we obtain

$$\mathbb{E}(\boldsymbol{v}_\Lambda^{(m,k+1)}) \leq \sigma \mathbb{E}(\boldsymbol{v}_\Lambda^{(m,k)}) \leq \sigma \nu^{2(\bar{n}_{m,k} - n_0)_+}\, \mathbb{E}_{m,k}. \tag{5.22}$$

Setting $e_{m,k+1} := e_{m,k} + (\bar{n}_{m,k} - n_0)_+$, we obtain (5.20) with k replaced by $k+1$. This completes the inner recursion argument.

Let $K^{(m)}$ be an integer to be determined. The coarsening step **COARSE** yields

$$\mathbb{E}(\boldsymbol{u}_\Lambda^{(m+1)}) - \mathbb{E}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) = \mathbb{J}(\boldsymbol{u}_\Lambda^{(m+1)}) - \mathbb{J}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})}) \leq \vartheta^{(m)}.$$

Choose

$$\vartheta^{(m)} := \bar{\mathcal{C}}\, \mathbb{E}_{m,K^{(m)}}, \tag{5.23}$$

where $\bar{\mathcal{C}} > 0$ is by now a constant which can be freely chosen, but in the next section will be chosen in order to guarantee the optimality of the thresholding procedure. Then, recalling the definition in (5.20), we obtain

$$\mathbb{E}(\boldsymbol{u}_\Lambda^{(m+1)}) \leq (1 + \bar{\mathcal{C}})\, \sigma^{K^{(m)}} \nu^{2(e_m + e_{m,K^{(m)}})}\, \mathbb{E}_0.$$

This suggests to choose $K^{(m)} \geq 1$ as the smallest integer for which we have

$$g_{m,K^{(m)}} \geq 1, \text{ where } g_{m,K^{(m)}} \text{ satisfies } \nu^{2g_{m,K^{(m)}}} := (1 + \bar{\mathcal{C}})\, \sigma^{K^{(m)}} \nu^{2e_{m,K^{(m)}}}. \tag{5.24}$$

Note that $K^{(m)}$ is bounded by $\bar{K}$, where $\bar{K} \geq 1$ is the smallest integer satisfying $(1 + \bar{\mathcal{C}})\, \sigma^{\bar{K}} \leq \nu^2$. Setting

$$\bar{e}_m := [g_{m,K^{(m)}}] \geq 1 \qquad \text{and} \qquad e_{m+1} := e_m + \bar{e}_m, \tag{5.25}$$

we obtain (5.19) with m replaced by $m+1$. This completes the outer recursion argument.

We note that each coarsening step $\boldsymbol{u}_\Lambda^{(m+1)} :=$ **COARSE**$[\boldsymbol{v}_\Lambda^{(m,K^{(m)})}, \vartheta^{(m)}]$ can be accomplished as described in Section 5.4. Recalling (5.12), we have

$$\|\boldsymbol{J}'(\boldsymbol{v}_\Lambda^{(m,K^{(m)})})\|_{\ell_2} \leq \mathcal{C}_1 \mathbb{E}(\boldsymbol{v}_\Lambda^{(m,K^{(m)})})^{1/2} \leq \mathcal{C}_1 \mathbb{E}_{m,K^{(m)}}^{1/2} =: \xi^{(m)}.$$

Since we have chosen $\vartheta^{(m)} = \bar{\mathcal{C}}\,\mathbb{E}_{m,K^{(m)}}$, by (5.17) we obtain $\boldsymbol{u}_\Lambda^{(m+1)}$ as the output of $\mathbf{THRESH}[\boldsymbol{v}_\Lambda^{(m,K^{(m)})}, \varepsilon_\vartheta^{(m)}]$ with

$$\varepsilon_\vartheta^{(m)} := \min\left(\left(\frac{\bar{\mathcal{C}}}{2L_{J,\Psi}}\right)^{1/2}, \frac{\bar{\mathcal{C}}}{2\mathcal{C}_1}\right)\mathbb{E}_{m,K^{(m)}}^{1/2}. \tag{5.26}$$

Finally, let us count the number of applications of **EVAL-GRAD** in our algorithm. By the previous considerations, they only occur within **DESCENT**. Denote by $N_{m+1,m} = \bar{n}_{m,0} + \cdots + \bar{n}_{m,K^{(m)}-1}$ the number of applications of **EVAL-GRAD** needed to compute $\boldsymbol{u}_\Lambda^{(m+1)}$ from $\boldsymbol{u}_\Lambda^{(m)}$. We observe that surely $e_{m,K^{(m)}} \leq \bar{e}_m + \delta\bar{K}$ for some $\delta \geq 1$. Thus,

$$\begin{aligned} N_{m+1,m} &= \sum_{k=0}^{K^{(m)}-1} (\bar{n}_{m,k} - n_0) + n_0 K^{(m)} \leq \sum_{k=0}^{K^{(m)}-1} (\bar{n}_{m,k} - n_0)_+ + n_0\bar{K} \\ &= e_{m,K^{(m)}} + n_0\bar{K} \leq \bar{e}_m + (n_0+\delta)\bar{K} \\ &\leq (n_0+\delta+1)\bar{K}\bar{e}_m = q\bar{K}(e_{m+1} - e_m) \end{aligned}$$

with $q := n_0 + \delta + 1$ and n_0 defined in (5.15). We conclude that the number N_{m+1} of applications of **EVAL-GRAD** needed to compute $\boldsymbol{u}_\Lambda^{(m+1)}$ from $\boldsymbol{u}_\Lambda^{(0)}$ satisfies

$$N_{m+1} = \sum_{\mu=0}^{m} N_{\mu+1,\mu} \leq q\bar{K}\sum_{\mu=0}^{m}(e_{\mu+1} - e_\mu) = q\bar{K}e_{m+1}.$$

REMARK 5.8. In view of the subsequent optimality analysis, we observe that a version of the algorithm can be given in which each $\bar{n}_{m,k}$ (and hence $N_{m+1,m}$) is guaranteed to be uniformly bounded. In fact, (5.14) shows that the output $\boldsymbol{w}_\Lambda^{(n)}$ of the routine $\mathbf{EVAL-GRAD}[\boldsymbol{v}_\Lambda, \nu^{n-1}\varepsilon]$ satisfies

$$\mathbb{E}(\boldsymbol{w}_\Lambda^{(n)}) \leq \mathcal{C}_2\mathcal{C}_1^2\frac{4}{(1-\gamma)^2}\nu^{2n-2}\,\mathbb{E}_\Lambda,$$

where again $\mathbb{E}_\Lambda$ is an estimate for $\mathbb{E}(\boldsymbol{v}_\Lambda)$. Choosing $\bar{n}$ as the smallest integer satisfying

$$\mathcal{C}_2\mathcal{C}_1^2\frac{4}{(1-\gamma)^2}\nu^{2n-2} \leq \sigma,$$

we obtain $\mathbb{E}(\boldsymbol{w}_\Lambda^{(n)}) \leq \sigma\,\mathbb{E}_\Lambda$. Hence, we can restrict the loop in **APPROX-GRAD** to $n = 1, 2, \ldots, \bar{n}$, with an absolute constant $\bar{n}$. If (4.3) is satisfied before the end of the loop, we leave **APPROX-GRAD** and proceed by **LINE-SEARCH** (resulting in some $\boldsymbol{v}_\Lambda$) and obtain the error reduction $\mathbb{E}(\boldsymbol{v}_\Lambda) \leq \sigma\,\mathbb{E}_\Lambda$ by the above arguments. Otherwise, after at most $\bar{n}$ steps, we leave **APPROX-GRAD**, set $\boldsymbol{v}_\Lambda := \boldsymbol{w}_\Lambda^{(\bar{n})}$ avoiding the necessity of the line search in this case. This again results in the same error reduction.

Let us summarize our result in the following theorem.

THEOREM 5.9. *Let $p = 2$ and let* (5.1) *hold. We consider Algorithm 4.12* **MINIMIZE**. *We fix a constant $\nu \in (0,1)$ and we choose a constant $\mathbb{E}_0$ satisfying* (5.18). *Then, for the choice* (5.24) *of the parameters $K^{(m)}$ (with $K^{(m)} \lesssim 1$), $\vartheta^{(m)}$ as in* (5.23) *and $\varepsilon^{(m,k)}$ as in* (5.21), *either the Algorithm yields the exact solution $\boldsymbol{u}$ after a* *finite number of steps, or there exists a strictly increasing sequence* $\{e_m\}$ *of integers* (5.25), *with* $e_0 = 0$, *such that the sequence of approximations* $\boldsymbol{u}_\Lambda^{(m)}$ *produced by the Algorithm satisfies*

$$\mathbb{E}(\boldsymbol{u}_\Lambda^{(m)}) \le \nu^{2e_m}\,\mathbb{E}_0, \qquad m = 0, 1, \ldots$$

Furthermore, the number N_m *of applications of* **EVAL-GRAD** *needed to compute* $\boldsymbol{u}_\Lambda^{(m)}$ *from* $\boldsymbol{u}_\Lambda^{(0)}$ *satisfies* $N_m \lesssim e_m$*, i.e., it grows at most as the logarithm of the obtained accuracy.* $\square$

Recalling (3.1) and Proposition 5.7, we immediately obtain an error estimate in the V-norm.

COROLLARY 5.10. *Under the same conditions of the previous theorem, we have*

$$\|u_\Lambda^{(m)} - u\|_V \le \nu^{e_m}\, C_\Psi \left(\frac{2\mathbb{E}_0}{c_{J,\Psi}}\right)^{1/2}, \qquad m = 0, 1, \ldots \qquad \square$$

5.6. Stopping Criteria. The previous algorithm produces arbitrarily close approximations to the exact solution u. In practice, one fixes a tolerance TOL and wishes to stop the algorithm as soon as the inequality $\|u_\Lambda^{(m)} - u\|_V < TOL$ is guaranteed to hold. According to the previous estimate, one can take as m the smallest integer for which

$$\nu^{e_m} < \frac{1}{C_\Psi}\left(\frac{c_{J,\Psi}}{2\mathbb{E}_0}\right)^{1/2} TOL.$$

On the other hand, the stopping test needs to be included also within **APPROX-GRAD**, in order to prevent an infinite loop, or simply to avoid unnecessary applications of **EVAL-GRAD** therein. Precisely, recalling (5.13), the loop in **APPROX-GRAD** is stopped if, for some n,

$$\frac{2}{1-\gamma}\nu^{n-1}\varepsilon < \omega\, TOL,$$

where $\omega \in (0,1)$ can be easily determined from Proposition 5.7 so that $\|\boldsymbol{v}_\Lambda - \boldsymbol{u}\|_{\ell_2} < TOL/(2C_\Psi)$. Setting $\boldsymbol{w}_\Lambda := \mathbf{THRESH}[\boldsymbol{v}_\Lambda, TOL/(2C_\Psi)]$, the triangle inequality yields $\|\boldsymbol{w}_\Lambda - \boldsymbol{u}\|_{\ell_2} < TOL/C_\Psi$, whence $\|w_\Lambda - u\|_V < TOL$ and the algorithm is stopped.

6. Optimality Properties of the Algorithm. We continue the discussion of the algorithm described in Section 5. Here, we investigate how the number of active basis functions in $u_\Lambda^{(m)}$ (i.e., the cardinality of the support of $\boldsymbol{u}_\Lambda^{(m)}$) and the overall computational complexity needed to get $u_\Lambda^{(m)}$ (number of arithmetic operations and sortings) are related to the accuracy of the approximation.

The relationship between cardinality of the active degrees of freedom (i.e., the number of active basis functions) and accuracy is a central topic in Nonlinear Approximation Theory, [23]. This involves the best N-term or the best η-accurate approximation to a given $\boldsymbol{v} \in \ell_2$ defined as follows. Let

$$\Sigma_N := \{\boldsymbol{w} = \sum_{\lambda\in\Lambda} w_\lambda\, \psi_\lambda : |\Lambda| \le N\}$$

(where $|\Lambda|$ denotes the cardinality of $\Lambda \subset \mathcal{J}$) be the nonlinear manifold of all linear combinations of basis functions containing at most N terms. Then a best N-term approximation $\boldsymbol{v}_N$ to $\boldsymbol{v}$ is defined by

$$\|\boldsymbol{v} - \boldsymbol{v}_N\|_{\ell_2} = \inf_{\boldsymbol{w}_N \in \Sigma_N^{\mathrm{con}}} \|\boldsymbol{v} - \boldsymbol{w}_N\|_{\ell_2} =: \varrho_N^{\mathrm{con}}(\boldsymbol{v}),$$

where $\Sigma_N^{\mathrm{con}} \subseteq \Sigma_N$ takes possible constraints in the choice of the active degrees of freedom into account (see below for an example).

Based on this, approximation spaces $\mathcal{A}^s$ are defined as the quasi-normed sequence space consisting of all those elements whose error $\varrho_N^{\mathrm{con}}(\boldsymbol{v})$ decays at least as N^{-s}. The quasi-norm is defined by

$$\|\boldsymbol{v}\|_{\mathcal{A}^s} := \sup_{N>0} N^s \, \varrho_N^{\mathrm{con}}(\boldsymbol{v}).$$

On the other hand, a best η-accurate approximation $\boldsymbol{v}_\eta$ to $\boldsymbol{v}$ is a vector of smallest support compatible with the constraints, such that

$$\|\boldsymbol{v} - \boldsymbol{v}_\eta\|_{\ell_2} \le \eta.$$

A typical result is as follows: any best η-accurate approximation $\boldsymbol{v}_\eta$ satisfies the inequality $|\operatorname{supp} \boldsymbol{v}_\eta| \lesssim \eta^{-1/s}\|\boldsymbol{v}\|_{\mathcal{A}^s}^{1/s}$ as $\eta \to 0$.

In this framework, the behaviour of best N-term or best η-accurate approximations to $\boldsymbol{v}$ provides a benchmark to evaluate the quality of any compactly supported approximation of $\boldsymbol{v}$. Precisely, a family $\{\boldsymbol{v}_\Lambda\}$ of compactly supported vectors converging to $\boldsymbol{v}$ in the ℓ_2-norm is said to be *asymptotically optimal* if it satisfies $\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s} \lesssim \|\boldsymbol{v}\|_{\mathcal{A}^s}$ and

$$\|\boldsymbol{v} - \boldsymbol{v}_\Lambda\|_{\ell_2} \lesssim |\Lambda|^{-s}\|\boldsymbol{v}\|_{\mathcal{A}^s},$$

or, equivalently

$$|\Lambda| \lesssim \|\boldsymbol{v} - \boldsymbol{v}_\Lambda\|_{\ell_2}^{-1/s}\|\boldsymbol{v}\|_{\mathcal{A}^s}^{1/s}.$$

The latter condition is surely satisfied if for any $\boldsymbol{v}_\Lambda$ there exists $\eta = \eta_\Lambda > 0$ such that

$$\|\boldsymbol{v} - \boldsymbol{v}_\Lambda\|_{\ell_2} \le \eta \qquad \text{and} \qquad |\Lambda| \lesssim \eta^{-1/s}\|\boldsymbol{v}\|_{\mathcal{A}^s}^{1/s}.$$

Note that in this context the subscript Λ refers to the actual support of the vector. We remark that these results are known for L_p-spaces as well, see e.g. [15, 23].

Before we proceed, let us give two concrete examples for the space $\mathcal{A}^s$. If no constraint is imposed on the choice of the active degrees of freedom, then $\mathcal{A}^s = \ell_\tau^{\mathrm{w}}$, where ℓ_τ^{w} (with $\frac{1}{\tau} = s + \frac{1}{2}$) is the Lorentz space of sequences $\boldsymbol{v} = \{v_\lambda\} \in \ell_2$ whose non-increasing rearrangement $|v_{\lambda_1}| \ge |v_{\lambda_2}| \ldots \ge |v_{\lambda_n}| \ge \ldots$ satisfies

$$|\boldsymbol{v}|_{\ell_\tau^{\mathrm{w}}} := \sup_n n^{1/\tau}|v_{\lambda_n}| < \infty.$$

This expression can be viewed as a quantitative measurement of the sparseness of the sequence $\boldsymbol{v}$. The space ℓ_τ^{w} is equipped with the (quasi-)norm $\|\boldsymbol{v}\|_{\ell_\tau^{\mathrm{w}}} := \|\boldsymbol{v}\|_{\ell_2} + |\boldsymbol{v}|_{\ell_\tau^{\mathrm{w}}}$. If V is the Besov space $B_{p,p}^r(\Omega)$ of functions defined in a domain $\Omega \subset \mathbb{R}^d$, then $\boldsymbol{v} \in \mathcal{A}^s$ is implied by the regularity condition $v = \boldsymbol{v}^T\Psi \in B_{\tau,\tau}^{r+ds}(\Omega)$, provided the basis Ψ characterizes this space (see [6, 7, 16, 18, 19, 21] for more details, where the construction is detailed in $L_2(\Omega)$ but it can easily be extended to the L_p-case).

As a second example, we consider a basis Ψ of compactly supported wavelets in Ω. Suppose that the set of active degrees of freedom is constrained to have a tree-structure (i.e., $\lambda \in \operatorname{supp} \boldsymbol{v}_N$ implies $\mu \in \operatorname{supp} \boldsymbol{v}_N$ for all wavelets ψ_μ whose support contains the support of ψ_λ). Then, $\mathcal{A}^s = \mathcal{A}^{s,\text{tree}}$ is the space of sequences whose best tree N-term approximation converge at a rate N^{-s}, i.e. the space of all ℓ_2-sequences $\boldsymbol{v}$ such that

$$\varrho_N^{\text{tree}}(\boldsymbol{v}) \lesssim N^{-s},$$

which is a quasi-normed space under the quasi-norm

$$\|\boldsymbol{v}\|_{\mathcal{A}^{s,\text{tree}}} := \sup_{n\in\mathbb{N}} N^s \, \varrho_N^{\text{tree}}(\boldsymbol{v}).$$

Here $\boldsymbol{v} \in \mathcal{A}^s$ holds provided $v = \boldsymbol{v}^T\Psi \in B^{r+ds}_{\tau^*,\tau^*}(\Omega)$ holds for some $\tau^* > \tau$ with τ as above. We now suppose that the procedure **THRESH** defined in Section 4.3 satisfies the following condition.

ASSUMPTION 6.1. *There exists a constant $C^* \geq 1$ such that if $\boldsymbol{v} \in \mathcal{A}^s$ and if $\boldsymbol{v}_\Lambda$ satisfies $\|\boldsymbol{v} - \boldsymbol{v}_\Lambda\|_{\ell_2} \leq \varepsilon$, then $\boldsymbol{z}_\Lambda :=$ **THRESH**$[\boldsymbol{v}_\Lambda, C^*\varepsilon]$ satisfies*

$$\|\boldsymbol{v} - \boldsymbol{z}_\Lambda\|_{\ell_2} \leq (1 + C^*)\,\varepsilon, \quad |\operatorname{supp} \boldsymbol{z}_\Lambda| \lesssim \varepsilon^{-1/s}\|\boldsymbol{v}\|_{\mathcal{A}^s}, \quad \|\boldsymbol{z}_\Lambda\|_{\mathcal{A}^s} \lesssim \|\boldsymbol{v}\|_{\mathcal{A}^s}. \tag{6.1}$$

This condition holds both for the unconstrained thresholding (e.g. with $C^* = 4$, see [11, 12]) and for the tree-thresholding (with a possibly larger C^*, see [5, 13]). Obviously, with a larger constant C^*, the results are still valid.

From now on, let us suppose that $\boldsymbol{u} \in \mathcal{A}^s$ for some $s > 0$. We recall that, for any $m \geq 0$, the vector $\boldsymbol{v}_\Lambda^{(m,K^{(m)})}$ satisfies by Proposition 5.7 and by (5.20)

$$\|\boldsymbol{v}_\Lambda^{(m,K^{(m)})} - \boldsymbol{u}\|_{\ell_2} \leq \left(\frac{2}{c_{J,\Psi}}\right)^{1/2} \mathbb{E}^{1/2}_{m,K^{(m)}} =: \overline{\varepsilon}^{(m)}. \tag{6.2}$$

Furthermore, we recall that $\boldsymbol{u}_\Lambda^{(m+1)} =$ **THRESH**$[\boldsymbol{v}_\Lambda^{(m,K^{(m)})}, \varepsilon_\vartheta^{(m)}]$ where $\varepsilon_\vartheta^{(m)}$ is defined in (5.26). Now we choose the constant $\bar{\mathcal{C}}$ introduced in (5.23) in such a way that $\varepsilon_\vartheta^{(m)} = C^*\overline{\varepsilon}^{(m)}$, i.e.,

$$\min\left(\left(\frac{\bar{\mathcal{C}}}{2L_{J,\Psi}}\right)^{1/2}, \frac{\bar{\mathcal{C}}}{2\mathcal{C}_1}\right) = C^* \left(\frac{2}{c_{J,\Psi}}\right)^{1/2}. \tag{6.3}$$

In this way (6.1) applies to $\boldsymbol{v} = \boldsymbol{u}$, $\boldsymbol{v}_\Lambda = \boldsymbol{v}_\Lambda^{(m,K^{(m)})}$ and $\boldsymbol{z}_\Lambda = \boldsymbol{u}_\Lambda^{(m+1)}$. Finally, observing that $\mathbb{E}^{1/2}_{m,K^{(m)}} \sim \nu^{e_{m+1}}\mathbb{E}_0^{1/2}$ and shifting m into $m-1$, we get the following result.

PROPOSITION 6.2. *Let the the assumptions of Theorem 5.9 and Assumption 6.1 hold. Then, the iterates $\boldsymbol{u}_\Lambda^{(m)}$ generated by Algorithm 4.12* **MINIMIZE** *with the choice* (6.3) *satisfy*

$$|\operatorname{supp} \boldsymbol{u}_\Lambda^{(m)}| \lesssim (\nu^{e_m}\mathbb{E}_0^{1/2})^{-1/s}\|\boldsymbol{u}\|^{1/s}_{\mathcal{A}^s} \qquad \text{and} \qquad \|\boldsymbol{u}_\Lambda^{(m)}\|_{\mathcal{A}^s} \lesssim \|\boldsymbol{u}\|_{\mathcal{A}^s},$$

for $m = 0, 1, \ldots$ $\square$

Since we have $\|\boldsymbol{u} - \boldsymbol{u}_\Lambda^{(m)}\|_{\ell_2} \lesssim \nu^{e_m}\mathbb{E}_0^{1/2}$ by Theorem 5.9, this proves the optimality of the approximation as far as the number of active degrees of freedom is concerned.

In order to control the possible growth of the supports of the vectors in the intermediate stages of the algorithm, as well as the computational complexity, we suppose that the procedure **EVAL-GRAD** defined in Section 4.1 satisfies the following condition.

ASSUMPTION 6.3. *Given any tolerance $\varepsilon > 0$ and any compactly supported vector $\boldsymbol{v}_\Lambda$, then the output $\boldsymbol{w}_\Lambda :=$ **EVAL-GRAD**$[\boldsymbol{v}_\Lambda, \varepsilon]$ satisfies*

$$
\begin{aligned}
|\operatorname{supp} \boldsymbol{w}_\Lambda| &\lesssim \varepsilon^{-1/s}(\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1),\\
\|\boldsymbol{w}_\Lambda\|_{\mathcal{A}^s} &\lesssim \|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s} + \|\boldsymbol{u}\|_{\mathcal{A}^s} + 1.
\end{aligned}
$$

The number $\operatorname{ops} \boldsymbol{w}_\Lambda$ *of operations needed to compute* $\boldsymbol{w}_\Lambda$ *satisfies*

$$
\operatorname{ops} \boldsymbol{w}_\Lambda \lesssim \varepsilon^{-1/s}(\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1) + |\operatorname{supp} \boldsymbol{v}_\Lambda|. \tag{6.4}
$$

This condition is fulfilled in a number of relevant cases, as described in [12, 13]. The second term on the right-hand side of (6.4) can be neglected in certain situations (e.g., in the linear and certain nonlinear cases). When it is needed, it accounts for computing a chain of near-best trees for the given input index set. It was shown in [14, Theorem 3.4] that the number $\log_2(\|\boldsymbol{v}_\Lambda\|/\varepsilon)$ of these trees depends only on $\|\boldsymbol{v}_\Lambda\|$ and the target accuracy. Moreover, it was shown there, that the overall cost to compute these trees remains proportional to $\varepsilon^{-1/s}(\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s}^{1/s} + 1) + |\mathcal{T}(\boldsymbol{v}_\Lambda)|$, where $\mathcal{T}(\boldsymbol{v})$ denotes the smallest tree containing $\operatorname{supp} \boldsymbol{v}$. It is not restrictive to assume in the nonlinear case, that the input as well as all intermediate vectors do already have tree structures. This shows that applying **EVAL-GRAD** to a sequence of decreasing tolerances for the *same* input vector simply may require to compute additional near-best trees. This in turns means that $|\mathcal{T}(\boldsymbol{v}_\Lambda)| = |\operatorname{supp} \boldsymbol{v}_\Lambda|$ has to be counted only once in the operation count.

From the previous property, we deduce similar bounds for the output of the routine **APPROX-GRAD**. Precisely, recalling that **EVAL-GRAD** is recursively applied in **APPROX-GRAD** with the *same* input vector $\boldsymbol{v}_\Lambda$, we immediately deduce that

$$
|\operatorname{supp} \boldsymbol{G}(\boldsymbol{v}_\Lambda)| \lesssim \eta^{-1/s}(\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1), \tag{6.5}
$$

$$
\|\boldsymbol{G}(\boldsymbol{v}_\Lambda)\|_{\mathcal{A}^s} \lesssim \|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s} + \|\boldsymbol{u}\|_{\mathcal{A}^s} + 1, \tag{6.6}
$$

where the constants on the right-hand side do not depend on the number of applications of **EVAL-GRAD**.

Concerning the operation count, we have to take into account that the accuracy in the calls to **EVAL-GRAD** is reduced at a geometric rate, i.e., we have $\eta^{(k)} = \varepsilon\,\nu^{k-1}$ after k calls of **EVAL-GRAD**. Furthermore, we recall that the second term on the right-hand side of (6.4) can be counted only once. Hence, denoting again by $\bar{n}$ the number of calls to **EVAL-GRAD**, and observing that $\eta = \eta^{(\bar{n})}$, we have

$$
\begin{aligned}
\operatorname{ops} \boldsymbol{G}(\boldsymbol{v}_\Lambda) &\lesssim \sum_{k=1}^{\bar{n}} \operatorname{ops}\,(\textbf{EVAL-GRAD}[\boldsymbol{v}_\Lambda, \eta^{(k)}])\\
&\lesssim \sum_{k=1}^{\bar{n}} (\eta^{(k)})^{-1/s}(\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1) + |\operatorname{supp} \boldsymbol{v}_\Lambda|\\
&\lesssim \eta^{-1/s}(\|\boldsymbol{v}_\Lambda\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1) + |\operatorname{supp} \boldsymbol{v}_\Lambda|, \qquad (6.7)
\end{aligned}
$$

where again the constants on the right-hand side do not depend on $\bar{n}$. Recalling Remark 5.8, we point out that the number of applications of **EVAL-GRAD** in each iteration can be uniformly bounded. We stress the fact that the crucial point is –as already pointed out earlier– that **EVAL-GRAD** is called within **APPROX-GRAD** with the *same* input vector $\boldsymbol{v}_\Lambda$ with a series of tolerences that are geometrically decreasing. This implies as shown in (6.7) a uniform bound on the number of operations in terms of the error which is actually reachable for the input $\boldsymbol{v}_\Lambda$.

Consequently, we obtain similar estimates if, on the left-hand sides, we replace $\boldsymbol{G}(\boldsymbol{v}_\Lambda)$ by the output $\boldsymbol{w}_\Lambda := \mathbf{DESCENT}[\boldsymbol{v}_\Lambda, \varepsilon]$; in this case, η is the quantity determined within **APPROX-GRAD**.

We now apply these results within the Algorithm **MINIMIZE**, with the choice of parameters described in Sections 5 and 6. For any $m \geq 0$ and $k = 0, \ldots, K^{(m)} - 1$, set $[\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)}), \eta^{(m,k)}] := \mathbf{APPROX\text{-}GRAD}[\boldsymbol{v}_\Lambda^{(m,k)}, \varepsilon^{(m,k)}]$. Recalling (5.20) and (5.21) as well as the definitions of $\bar{n}_{m,k}$ and $e_{m,k+1}$, it is straightforward to check that

$$\eta^{(m,k)} \geq \mathcal{C}_1 \sigma^{-1/2} \mathbb{E}_{m,k+1}^{1/2} \begin{cases} \nu^{n_0}, & \text{if } \bar{n}_{m,k} \geq n_0, \\ 1, & \text{if } \bar{n}_{m,k} < n_0. \end{cases}$$

In both cases, $\eta^{(m,k)} \gtrsim \mathbb{E}_{m,k+1}^{1/2}$ uniformly in m and k. Thus, by (6.5) to (6.7) we obtain

$$|\operatorname{supp} \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})| \lesssim (\mathbb{E}_{m,k+1}^{1/2})^{-1/s} (\|\boldsymbol{v}_\Lambda^{(m,k)}\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1) \tag{6.8}$$

$$\|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\mathcal{A}^s} \lesssim \|\boldsymbol{v}_\Lambda^{(m,k)}\|_{\mathcal{A}^s} + \|\boldsymbol{u}\|_{\mathcal{A}^s} + 1, \tag{6.9}$$

$$\operatorname{ops} \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)}) \lesssim (\mathbb{E}_{m,k+1}^{1/2})^{-1/s} (\|\boldsymbol{v}_\Lambda^{(m,k)}\|_{\mathcal{A}^s}^{1/s} + \|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1) + |\operatorname{supp} \boldsymbol{v}_\Lambda^{(m,k)}|. \tag{6.10}$$

Next, from the definition of $\boldsymbol{v}_\Lambda^{(m,k+1)} = \boldsymbol{v}_\Lambda^{(m,k)} + \mu^{(m,k)} \boldsymbol{s}_\Lambda^{(m,k)}$, we deduce that

$$|\operatorname{supp} \boldsymbol{v}_\Lambda^{(m,k+1)}| \leq |\operatorname{supp} \boldsymbol{v}_\Lambda^{(m,k)}| + |\operatorname{supp} \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})|, \tag{6.11}$$

$$\operatorname{ops}(\boldsymbol{v}_\Lambda^{(m,k+1)}, \boldsymbol{v}_\Lambda^{(m,k)}) \lesssim \operatorname{ops} \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)}) + |\operatorname{supp} \boldsymbol{v}_\Lambda^{(m,k+1)}|, \tag{6.12}$$

where $\operatorname{ops}(\boldsymbol{w}_\Lambda, \boldsymbol{v}_\Lambda)$ means the number of operations needed to compute $\boldsymbol{w}_\Lambda$, given $\boldsymbol{v}_\Lambda$. We observe that, by the particular realization of line search we have chosen, we have

$$\mu^{(m,k)} \boldsymbol{s}_\Lambda^{(m,k)} = \zeta^{(m,k)} \|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_2} \frac{\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})}{\|\boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})\|_{\ell_2}} = \zeta^{(m,k)} \boldsymbol{G}(\boldsymbol{v}_\Lambda^{(m,k)})$$

for some $\zeta^{(m,k)}$ uniformly bounded from above and below (see (5.6)). Thus, we have

$$\|\boldsymbol{v}_\Lambda^{(m,k+1)}\|_{\mathcal{A}^s} \lesssim \|\boldsymbol{v}_\Lambda^{(m,k)}\|_{\mathcal{A}^s} + \|\boldsymbol{u}\|_{\mathcal{A}^s} + 1. \tag{6.13}$$

Now, we apply a recursion argument; taking into account Proposition 6.2, using again the property of geometric series and the uniform boundedness of $K^{(m)}$, from (6.8) to (6.13), we deduce the estimates

$$\begin{aligned} |\operatorname{supp} \boldsymbol{v}_\Lambda^{(m,k+1)}| + \operatorname{ops}(\boldsymbol{v}_\Lambda^{(m,k+1)}, \boldsymbol{u}^{(m)}) &\lesssim (\mathbb{E}_{m,k+1}^{1/2})^{-1/s} (\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1) \\ &\lesssim (\nu^{e_{m+1}} \mathbb{E}_0^{1/2})^{-1/s} (\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s} + 1), \\ \|\boldsymbol{v}_\Lambda^{(m,k+1)}\|_{\mathcal{A}^s} &\lesssim \|\boldsymbol{u}\|_{\mathcal{A}^s} + 1. \end{aligned} \tag{6.14}$$

Finally, recall that also the number of sortings needed in $\mathbf{THRESH}[\boldsymbol{v}_\Lambda^{(m,K^{(m)})}, \varepsilon_\vartheta^{(m)}]$ can be made proportional to $|\operatorname{supp} \boldsymbol{v}_\Lambda^{(m,K^{(m)})}|$ (see [3]). Shifting m into $m-1$, we get the following result.

PROPOSITION 6.4. *Under the assumptions of Proposition 6.2 and Assumption 6.3, the cardinality of the supports of all vectors involved in the computation of* $\boldsymbol{u}_\Lambda^{(m)}$ *starting from* $\boldsymbol{u}_\Lambda^{(0)}$*, as well as the complexity to compute* $\boldsymbol{u}_\Lambda^{(m)}$*, are bounded by* $C(\nu^{e_m}\mathbb{E}_0^{1/2})^{-1/s}(\|\boldsymbol{u}\|_{\mathcal{A}^s}^{1/s}+1)$. $\square$

This completes the assessment of the optimality of the algorithm.

7. Examples. We conclude the paper with two concrete examples.

7.1. The Non-linear Laplacian. Let $\Omega \subset \mathbb{R}^n$ be a bounded domain with Lipschitz boundary $\partial\Omega$. Given some $p > 2$, let $\mathcal{V} = W_0^{1,p}(\Omega)$ be the closed subspace of the Sobolev space $W^{1,p}(\Omega)$ of the functions vanishing on $\partial\Omega$, equipped with the norm $\|v\|_{W_0^{1,p}(\Omega)} = \left(\sum_{i=1}^n \|\frac{\partial v}{\partial x_i}\|_{L^p(\Omega)}^p\right)^{1/p}$. Let f be an element in $\mathcal{V}' = W^{-1,p'}(\Omega)$, and let $\langle f, v\rangle_{\mathcal{V}'\times\mathcal{V}}$ denote the duality pairing between $\mathcal{V}'$ and $\mathcal{V}$.

We consider the functional $J : \mathcal{V} \to \mathbb{R}$ defined as

$$J(v) = \frac{1}{p}\|v\|_{\mathcal{V}}^p - \langle f, v\rangle_{\mathcal{V}'\times\mathcal{V}}.$$

Its Fréchet derivative is given by $J'(w) = A(w) - f$, where

$$A(w) = -\sum_{i=1}^n \frac{\partial}{\partial x_i}\left(\left|\frac{\partial w}{\partial x_i}\right|^{p-2}\frac{\partial w}{\partial x_i}\right)$$

is known as the p-Laplacian. The functional J satisfies Assumption 2.1. Indeed, condition (i) can be proven by a repeated application of Hölder's inequality in L^p-spaces, whereas condition (ii) easily follows from the existence of a constant $c_p > 0$ such that

$$(|s|^{p-2}s - |t|^{p-2}t)(s-t) \geq c_p|s-t|^p, \qquad \forall s,t \in \mathbb{R}. \tag{7.1}$$

Thus, there exists a unique minimizer of the functional J on $\mathcal{V}$, i.e., a unique solution of the Dirichlet problem $A(u) = f$ in Ω, $u = 0$ on $\partial\Omega$ (see, e.g., [27] for more details).

In view of the numerical discretization of such problem, we introduce a wavelet basis $\Psi^* := \{\psi_\lambda^* : \lambda \in \mathcal{J}\}$ in $L^p(\Omega)$, such that $\|\psi_\lambda^*\|_{L^p(\Omega)} \sim 1$ for all λ. Here λ is a multi-index, containing all the relevant parameters of the wavelet, including the level index $j =: |\lambda|$. Furthermore, we denote by $B_{p,p,0}^r(\Omega)$ the closure of $\mathcal{C}_0^\infty(\Omega)$ in the Besov space $B_{p,p}^r(\Omega)$, and we assume that there exists $r^* > 1$ such that, for all r satisfying $0 < r < r^*$, Ψ^* is also a basis in $B_{p,p,0}^r(\Omega)$ and the norm equivalence

$$\|v\|_{B_{p,p,0}^r(\Omega)} \sim \left(\sum_{\lambda\in\mathcal{J}} 2^{pr|\lambda|}\,|v_\lambda^*|^p\right)^{1/p} \tag{7.2}$$

holds for all $v = \sum_{\lambda\in\mathcal{J}} v_\lambda^*\psi_\lambda^* \in B_{p,p,0}^r(\Omega)$. Examples of such bases can be found, e.g., in [6, 7, 16, 18, 19, 21].

It is well known (see, e.g., [30]) that $B_{p,p}^r(\Omega) = W^{r,p}(\Omega)$ algebrically and topologically for all $r \notin \mathbb{N}$, whereas unfortunately $B_{p,p}^k(\Omega) \neq W^{k,p}(\Omega)$ for $k \in \mathbb{N}$; one only has

$B^{k+\varepsilon}_{p,p}(\Omega) \subset W^{k,p}(\Omega) \subset B^k_{p,p}(\Omega)$ for all $\varepsilon > 0$ with strict topological inclusions. Thus, we cannot apply the results of Sections 3 and 4 taking $W^{1,p}_0(\Omega)$ as V therein.

We circumvent such a drawback by resorting to a perturbation argument. To this end, we assume that $u \in B^{1+\varepsilon_0}_{p,p,0}(\Omega)$ for some $\varepsilon_0 > 0$. Then, for any fixed $\varepsilon \in (0, \varepsilon_0]$, we set $V = V_\varepsilon = B^{1+\varepsilon}_{p,p,0}(\Omega)$ and we normalize the wavelets in the V_ε-norm, i.e., we set $\psi_\lambda = 2^{-(1+\varepsilon)|\lambda|}\,\psi^*_\lambda$ for all $\lambda \in \mathcal{J}$; then, (7.2) yields precisely (3.1). Next, we introduce the perturbed functional $J_\varepsilon : V_\varepsilon \to \mathbb{R}$, defined as

$$J_\varepsilon(v) = \frac{\varepsilon}{p}\|\boldsymbol{v}\|^p_{\ell_p} + J(v), \qquad \forall v = \boldsymbol{v}^T\Psi \in V_\varepsilon.$$

Its Fréchet derivative $J'_\varepsilon : V_\varepsilon \to V'_\varepsilon$ satisfies $\langle J'_\varepsilon(v), w\rangle_{V'_\varepsilon \times V_\varepsilon} = \varepsilon \sum_{\lambda\in\mathcal{J}} |v_\lambda|^{p-2} v_\lambda w_\lambda + \langle J'(v), w\rangle_{\mathcal{V}'\times\mathcal{V}}$ for all $v, w \in V_\varepsilon$. This functional, too, satifies Assumption 2.1. Actually, condition (i) follows from the continuous inclusion $V_\varepsilon \subset W^{1,p}_0(\Omega)$, whereas condition (ii) is satisfied since (7.1) and the similar condition for J imply for all $w, v \in V_\varepsilon$

$$\langle J'_\varepsilon(w) - J'_\varepsilon(v), w - v\rangle_{V'_\varepsilon \times V_\varepsilon} \geq \varepsilon c_p \|\boldsymbol{w} - \boldsymbol{v}\|^p_{\ell_p} + c_J\|w - v\|^p_{W^{1,p}_0(\Omega)}. \tag{7.3}$$

Using (7.2), we obtain the existence of a constant c'_J (independent of ε) such that

$$\langle J'_\varepsilon(w) - J'_\varepsilon(v), w - v\rangle_{V'_\varepsilon \times V_\varepsilon} \geq c'_J(\varepsilon\|w - v\|^p_{B^{1+\varepsilon}_{p,p,0}(\Omega)} + \|w - v\|^p_{W^{1,p}_0(\Omega)}).$$

We can also express the ellipticity bound of the functional in terms of wavelet coefficients. Indeed, since $W^{1,p}_0(\Omega) \subset B^1_{p,p,0}(\Omega)$ with continuos inclusion, by (7.2) with $r = 1$ there exists a constant c''_J (again independent of ε) such that

$$\langle J'_\varepsilon(w) - J'_\varepsilon(v), w - v\rangle_{V'_\varepsilon \times V_\varepsilon} \geq c''_J \sum_{\lambda\in\mathcal{J}} (\varepsilon + 2^{-p\varepsilon|\lambda|})|w_\lambda - v_\lambda|^p.$$

Let $u_\varepsilon \in V_\varepsilon$ be the unique minimizer of J_ε in V_ε. Since all assumptions are fulfilled, we can apply Algorithm 4.12 **MINIMIZE** described in Sections 3 and 4 to the functional J_ε.

Next result proves that u_ε can be made arbitrarily close to u by choosing ε small enough.

PROPOSITION 7.1. *The following estimate holds*

$$\|u - u_\varepsilon\|_{W^{1,p}_0(\Omega)} \lesssim \varepsilon^{1/p}, \qquad 0 < \varepsilon \leq \varepsilon_0.$$

Proof. Set $N(v) := \frac{1}{p}\|\boldsymbol{v}\|^p_{\ell_p}$ and note that $J'_\varepsilon(u_\varepsilon) = 0$ and $J'_\varepsilon(u) = J'(u) + \varepsilon N'(u) = \varepsilon N'(u)$. Applying (7.3) with $w = u$ and $v = u_\varepsilon$, we obtain

$$\begin{aligned} c_J\|u - u_\varepsilon\|^p_{W^{1,p}_0(\Omega)} + \varepsilon c_p\|\boldsymbol{u} - \boldsymbol{u}_\varepsilon\|^p_{\ell_p} &\lesssim \langle J'_\varepsilon(u) - J'_\varepsilon(u_\varepsilon), u - u_\varepsilon\rangle_{V'_\varepsilon \times V_\varepsilon} \\ &\lesssim \varepsilon\,\langle N'(u), u - u_\varepsilon\rangle_{V'_\varepsilon\times V_\varepsilon} = \varepsilon\,\langle \boldsymbol{N}'(\boldsymbol{u}), \boldsymbol{u} - \boldsymbol{u}_\varepsilon\rangle_{\ell_{p'}\times\ell_p}. \end{aligned}$$

By Hölder's inequality,

$$\langle \boldsymbol{N}'(\boldsymbol{u}), \boldsymbol{u} - \boldsymbol{u}_\varepsilon\rangle_{\ell_{p'}\times\ell_p} \leq \|\boldsymbol{N}'(\boldsymbol{u})\|_{\ell_{p'}}\|\boldsymbol{u} - \boldsymbol{u}_\varepsilon\|_{\ell_p} \leq \frac{1}{p' c_p^{p'/p}}\|\boldsymbol{N}'(\boldsymbol{u})\|^{p'}_{\ell_{p'}} + \frac{c_p}{p}\|\boldsymbol{u} - \boldsymbol{u}_\varepsilon\|^p_{\ell_p},$$

whence the result easily follows from the assumption $u \in B^{1+\varepsilon_0}_{p,p,0}(\Omega)$. $\square$

The proposition provides a guideline for selecting the perturbation parameter ε. Suppose that we wish to approximate u, in the $W^{1,p}_0(\Omega)$-norm, within an accuracy of order TOL. Then, it is enough to choose $\varepsilon = (TOL)^p$ and stop Algorithm **MINIMIZE** as soon as u_ε is itself approximated with an accuracy of order TOL.

7.2. The Non-linear Reaction-Diffusion Problem. Let $G : \mathbb{R} \to \mathbb{R}^+$ be a smooth strictly convex function, such that $G(s) \lesssim |s|^p$ for all $s \in \mathbb{R}$, where $p \geq 2$ is chosen so that $H_0^1(\Omega) \hookrightarrow L^p(\Omega)$; $V = H_0^1(\Omega)$. We consider the functional

$$J(v) := \frac{1}{2}\int_\Omega |\nabla v|^2 + \int_\Omega G(v) - \langle f, v\rangle_{V' \times V},$$

which is strictly convex and unbounded. The Fréchet derivative is given by

$$J'(v) = -\Delta + g(v)I - f =: A(v) - f,$$

where $g(s) := G'(s)$ is strictly monotone in $\mathbb{R}$. It is readily seen that A is continuous and strictly monotone. Moreover, A is Lipschitz continuous and one can show the following bounds hold

$$\|v\|_V \lesssim \|A(v)\|_{V'} \lesssim \|v\|_V^{p-1}, \quad v \in V.$$

Thus, Assumption 2.1 (i) is satisfied. Moreover, due to the monotonicity of g, for all $v, w \in V$, we have $(g(v) - g(w), v - w)_{0,\Omega} \geq 0$ and thus

$$\langle J'(v) - J'(w), v - w\rangle = \|\nabla(v - w)\|_{0,\Omega}^2 + (g(v) - g(w), v - w)_{0,\Omega} \gtrsim \|v - w\|_V^2,$$

so that (ii) holds for $p = 2$.

Let us take any wavelet basis Ψ of $H_0^1(\Omega)$, [6, 7, 16, 18, 19, 21], so that the following norm equivalence holds

$$\|v\|_V \sim \left(\sum_{\lambda\in\mathcal{J}} 2^{2|\lambda|}|v_\lambda|^2\right)^{1/2}, \qquad v = \sum_{\lambda\in\mathcal{J}} v_\lambda\, \psi_\lambda.$$

If the nonlinear function G is a global polynomial, then all required ingredients can be taken from the literature. Indeed, **EVAL-GRAD** and **EVAL-J** consist of a linear part which is described in [11, 12] and a nonlinearity of polynomial type. A corresponding evaluation scheme can be found in [14, 20]. Also a tree-coarsening routine is available in [13]. Finally, the gradient consists of the Laplacian and again a nonlinearity of polynomial type which can be realized using the methods describe in [11, 12, 14, 20]. Moreover, note that if in addition the Fréchet derivative J'' of J' is well-posed in the sense that $\|J''(v)w\|_{V'} \sim \|w\|_V$ for all v in a neighborhood of the solution u of $J'(u) = 0$, then one can also use the adaptive wavelet method presented in [13] for the Euler-Lagrange equation in order to determine the minimizer u of J.

However, the function G may have a much more complicated structure, e.g. it may be defined piecewise. In this case, our general approach can also be used, provided corresponding routines for **EVAL-J**, **EVAL-GRAD** and **THRESH** are available. It is a subject of future research to construct and analyze such schemes.

Appendix A. Proof of Lemma 2.2. As for a.), the proof is straightforward setting $w := u + t(v - u)$, i.e., $v - u = \frac{1}{t}(w - u)$ and using (ii) to obtain

$$\begin{aligned} J(v) - J(u) &= \int_0^1 \frac{d}{dt} J(u + t(v - u))\, dt \\ &= \int_0^1 \langle J'(u + t(v - u)), v - u\rangle\, dt \end{aligned}$$

$$
\begin{aligned}
&= \langle J'(u), v-u\rangle + \int_0^1 \langle J'(u+t(v-u)) - J'(u), v-u\rangle \, dt \\
&\geq \langle J'(u), v-u\rangle + c_J \int_0^1 \frac{1}{t}\|w-u\|_V^p \, dt \\
&= \langle J'(u), v-u\rangle + c_J \int_0^1 \frac{1}{t}\|t(u-v)\|_V^p \, dt \\
&= \langle J'(u), v-u\rangle + c_J \int_0^1 t^{p-1}\|u-v\|_V^p \, dt \\
&= \langle J'(u), v-u\rangle + \frac{c_J}{p}\|u-v\|_V^p .
\end{aligned}
$$

Now we prove b.). Indeed,

$$
J(v) \geq J(u) + \langle J'(u), v-u\rangle + \frac{c_J}{p}\|v-u\|^p,
$$

hence, $J(v) > J(u) + \langle J'(u), v-u\rangle$ for all $u \neq v \in V$. For the boundedness, take again $u = 0$ in a.), then

$$
J(v) \geq J(0) + \langle J'(0), v\rangle + \frac{c_J}{p}\|v\|_V^p \geq J(0) - \|J'(0)\|_{V'}\,\|v\|_V + \frac{c_J}{p}\|v\|_V^p,
$$

which is bounded from below by the constant $J(0) - \|J'(0)\|_{V'}$.

As for c.), take $u = 0$ in a.). Then, for $v \in \mathcal{R}(u^{(0)})$, we have

$$
\frac{c_J}{p}\|v\|_V^p \leq J(v) - J(0) - \langle J'(0), v\rangle \leq |J(u^{(0)})| + |J(0)| + \|J'(0)\|_{V'}\,\|v\|_V .
$$

Since $p > 1$, we have $\|v\|_V \leq C(p, u^{(0)})$. $\square$

Acknowledgement. The authors wish to thank one of the referees for suggesting significant improvements over the original version of the paper.

REFERENCES

[1] R.A. Adams, Sobolev Spaces, Academic Press, 1978.
[2] J.-P. Aubin, Optima and equilibria : an introduction to nonlinear analysis , Springer, Berlin 1993.
[3] A. Barinka, Fast Computational Tools for Adaptive Wavelet Schemes, PhD Thesis, RWTH Aachen, in preparation.
[4] P. Binev, W. Dahmen and R.A. DeVore, *Adaptive Finite Element Methods with Convergence Rates*, Numer. Math., online first, Jan. 2004.
[5] P. Binev and R.A. DeVore, *Fast Computation in Adaptive Tree Approximation*, Numer. Math., online first, Feb. 2004.
[6] C. Canuto, A. Tabacco, and K. Urban, *The Wavelet Element Method, Part I: Construction and Analysis*, Appl. Comp. Harm. Anal. **6** (1999), 1–52.
[7] C. Canuto, A. Tabacco, and K. Urban, *The Wavelet Element Method, Part II: Realization and additional features in 2D and 3D*, Appl. Comp. Harm. Anal. **8** (2000), 123–165.
[8] J. Céa, Optimisation: théorie et algorithmes, Dunod, Paris 1971.
[9] P.G. Ciarlet, Introduction to numerical linear algebra and optimization, Cambridge University Press, 1988.
[10] A. Cohen, Wavelet methods in Numerical Analysis, in: Ciarlet, P.G., Lions, J.L. (eds), Handbook of Numerical Analysis. Elsevier North-Holland, 417–711, 2000.
[11] A. Cohen, W. Dahmen, and R.A. DeVore, *Adaptive Wavelet Schemes for Elliptic Operator Equations — Convergence Rates*, Math. Comput., **70**, No. 233 (2001), 27–75.

[12] A. Cohen, W. Dahmen, and R.A. DeVore, *Adaptive Wavelet Methods II — Beyond the Elliptic Case*, Found. Comput. Math. **2** (2002), 203–245.

[13] A. Cohen, W. Dahmen, and R.A. DeVore, *Adaptive Wavelet Schemes for Nonlinear Variational Problems*, SIAM J. Numer. Anal. **41**, No. 5 (2003), 1785–1823.

[14] A. Cohen, W. Dahmen, and R.A. DeVore, *Sparse Evaluation of Compositions of Functions Using Multiscale Expansions*, SIAM J. Math. Anal. **35**, No. 2 (2003), 279–303.

[15] A. Cohen, W. Dahmen, I. Daubechies, and R.A. DeVore, *Tree approximation and optimal encoding*, Appl. Comput. Harmon. Anal. **11**, No. 2 (2001), 192–226.

[16] A. Cohen and R. Masson *Wavelet Adaptive Methods for Second Order Elliptic Problems — Boundary Conditions and Domain Decomposition*, Numer. Math., **86**, No. 2 (2000) 193–238.

[17] W. Dahmen, *Wavelet and Multiscale Methods for Operator Equations*, Acta Numerica **6** (1997), 55–228.

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

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

[20] W. Dahmen, R. Schneider, and Y. Xu, *Nonlinear functionals of wavelet expansions — adaptive reconstruction and fast evaluation*, Numer. Math. **86** (2000), 49–101.

[21] W. Dahmen and R. Stevenson, *Element-by-element construction of wavelets satisfying stability and moment conditions*, SIAM J. Numer. Anal., **37** (1999), 319-325.

[22] J.E. Dennis and R.B. Schnabel, Numerical methods for unconstrained optimization and nonlinear equations, Prentice-Hall Series in Computational Mathematics, 1983.

[23] R.A. DeVore, *Nonlinear Approximation*, Acta Numerica, **7** (1998), 51–150.

[24] W. Dörfler, *A convergent adaptive algorithm for Poisson's equation*, SIAM J. Numer. Anal. **33** (1996), no. 3, 1106–1124.

[25] I. Ekeland and R. Temam, Convex analysis and variational problems, North-Holland, Amsterdam 1976.

[26] J.-B. Hiriart-Urruty and C. Lemaréchal, Convex analysis and minimization algorithms, Springer, Berlin 1993.

[27] J.-L. Lions, Quelques méthodes de résolution des problèmes aux limites non linéaires, Dunod, Paris 1969.

[28] P. Morin, R.H. Nochetto, and K.G. Siebert, *Data oscillation and convergence of adaptive FEM*, SIAM J. Numer. Anal. **38** (2000), no. 2, 466–488.

[29] E. Polak, Optimization: Algorithms and Consistent Approximations, Springer 1997.

[30] H. Triebel, Interpolation Theory, Function Spaces, Differential Operators, North-Holland, Amsterdam 1978.