SIAM J. NUMER. ANAL.
Vol. 61, No. 2, pp. 431–456

© 2023 Society for Industrial and Applied Mathematics

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

ADAPTIVE FINITE ELEMENT APPROXIMATIONS FOR ELLIPTIC PROBLEMS USING REGULARIZED FORCING DATA[*]

LUCA HELTAI[†] AND WENYU LEI[‡]

Abstract. We propose an adaptive finite element algorithm to approximate solutions of elliptic problems whose forcing data is locally defined and is approximated by regularization (or mollification). We show that the energy error decay is quasi-optimal in two-dimensional space and suboptimal in three-dimensional space. Numerical simulations are provided to confirm our findings.

Key words. finite elements, interface problems, immersed boundary method, Dirac delta approximations, a posteriori error estimates, adaptivity

MSC codes. 65N15, 65N30, 65N50

DOI. 10.1137/21M1455991

1. Introduction.

Let us consider the numerical approximation of the following elliptic problem with rough data: given a bounded domain $\Omega \subset \mathbb{R}^d$ with $d = 2$ or 3, we seek a distribution u satisfying

$$\begin{aligned} -\nabla\cdot(A(x)\nabla u) + c(x)u &= F \quad \text{in } \Omega, \\ u &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{1.1}$$

Here $A(x)$ is a $d \times d$ symmetric positive definite matrix with all entries in $C^1(\overline{\Omega})$. We further assume that there exist positive constants a_0 and a_1 satisfying

$$a_0|\nu|^2 \le \nu^\intercal A(x)\nu \le a_1|\nu|^2 \quad \text{for all } \nu \in \mathbb{R}^d \text{ and } x \in \overline{\Omega}. \tag{1.2}$$

The lower order coefficient $c(x)$ is set to be nonnegative and Lipschitz in $\overline{\Omega}$. We consider rough forcing data F that can be written as

$$F(x) := \int_B \delta(x-y)f(y)\,\mathrm{d}y \quad \text{with } B \subset \Omega,$$

where δ denotes the d-dimensional Dirac distribution and $B \subset \mathbb{R}^d$ is an immersed domain. If the co-dimension of B is zero, $F(x) = \chi_B(x)f(x)$ with χ_B denoting the indicator function of B. If the co-dimension of B is one, F can be written as a distribution. That is,

$$\langle F, \phi\rangle := \int_B f(y)\phi(y)\,\mathrm{d}y \quad \text{for all } \phi \in C_c^\infty(\overline{\Omega}). \tag{1.3}$$

[*] Received by the editors October 28, 2021; accepted for publication (in revised form) October 25, 2022; published electronically March 10, 2023.
https://doi.org/10.1137/21M1455991

Funding: The work of the authors was partially supported by the National Research Projects (PRIN 2017) “Numerical Analysis for Full and Reduced Order Methods for the Efficient and Accurate Solution of Complex Systems Governed by Partial Differential Equations,” funded by the Italian Ministry of Education, University, and Research.

[†] Mathematics Area, SISSA – International School for Advanced Studies, via Bonomea 265, 34136, Trieste, Italy (luca.heltai@sissa.it).

[‡] School of Mathematical Sciences, University of Electronic Science and Technology of China, No2006, Xiyuan Ave, West Hi-Tech Zone, 611731, Chengdu, China (wenyulei5515@gmail.com).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

In the rest of the paper, our discussion on the numerical approximation of (1.1) will be restricted to the co-dimension one case.

The above elliptic problem is a prototype of governing differential equations for interface problems, phase transitions, and fluid-structure interactions problems using the immersed boundary method [39, 11, 40, 44]. Many works exist that concentrate on the study of (adaptive) finite element methods with point Dirac sources [7, 29, 1]. The relevant literature for more complex distributions of singularities is more limited [33, 32]. The motivation for such methods lies in the possibly complex geometry of the immersed domain, such as thin vascular structures in tissues [22, 23, 15] or fibers in isotropic materials [2], for which it is difficult to obtain a bulk mesh of Ω matching the embedded domain.

On the the other hand, when considering a nonmatching bulk mesh to approximate problem (1.1), it is necessary to evaluate F on the quadrature points of Ω or to compute (1.3) when ϕ is a test function in a finite dimensional space. The implementation of the former strategy was introduced by Peskin in the early seventies (see [39] for a review) in the context of finite differences and later adopted to finite volume and finite element approaches [35]. The latter approximation strategy, usually referred to as the "variational formulation," was introduced in [10] and later works, for example, [24].

When computing $\int_B f\phi$ in the variational formulation, one has a choice to make: (i) either evaluate f and ϕ on the quadrature points derived from a fixed subdivision of B which is independent on the subdivision of Ω (using a single quadrature scheme on B), or (ii) evaluate f and ϕ on the nonzero intersections of cells $K \subset B$ and $T \subset \Omega$ (using a custom quadrature formula for the generally polygonal intersection).

The first approach is cheaper to compute, but it introduces some errors due to integration of nonsmooth functions using quadrature rules. It is a two-step process that requires first the exact identification of the cells that contain quadrature points of B, and then the computation of the inverse of the mapping from the reference cell $\widehat{T}$ to the cell T in the subdivision of Ω that contains the quadrature points. Such inverse mapping is nonlinear in unstructured quad- or hex-meshes, or when using higher order mappings.

The second approach requires a much more expensive computation, and its efficient implementation is the subject of active research (see, e.g., [30, 9]). If one wants to perform such integration exactly, it would require first the computation of the intersection between cells $K \subset B$ and $T \subset \Omega$, then the definition of a quadrature scheme on the (possibly polygonal or curved) intersection, in addition to the computation of the inverse of the mapping from the reference cells to the intersection part.

To avoid the complexity related to the evaluation of inverse mappings and possibly the computation of nonmatching grid intersections, here we consider an alternative approach by approximating F with its regularization (or mollification) [28, 45]. That is, we replace δ with a family of Dirac delta approximations δ^r, where r denotes the regularization parameter so that the regularized data, denoted by F^r, satisfies a certain smoothness property.

In the proposed finite element algorithm, we compute a regularized right-hand-side $\int_B f\phi^r \,\mathrm{d}x$ for a fixed parameter r. This computation requires the evaluation of the double integral

$$\int_\Omega \int_B f(x)\delta^r(x-y)\phi(y)\,\mathrm{d}x\,\mathrm{d}y.$$

When applying quadrature schemes both on the support of ϕ and on B, we only evaluate f and ϕ on (independent sets of) quadrature points of B and of Ω, respectively, weighted by the regularized Dirac distribution. This computation need be performed

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

only when the integration cells are at a distance smaller than r, and it does not require any special implementation.

The error between the exact solution u and its regularized counterpart $\mathtt{u}^r$ is analyzed in [25] in both the H^1 and the L^2 sense. The finite element approximation of (1.1) using quasi-uniform subdivisions is also discussed in [25], where we also show (see [25, Figure 7]) that the computational cost and the accuracy of the regularization approach are comparable to the corresponding nonregularized approach, at least in the first case described above. The regularization in this case has the advantage of being trivial to implement, a fact that contributed significantly to the success of the immersed boundary method in the literature, which remains one of the most used methods in the finite difference and finite volume community for the computation of nonmatching couplings.

In this paper, we consider the finite element approximation of (1.1) with the regularized data F^r under adaptive subdivisions. We show that the regularization approach not only is trivial to implement but also lends itself quite well to adaptive finite element methods (AFEMs) and to a posteriori errror analysis. AFEMs have been widely used for decades; see [38] for a survey of AFEMs for elliptic problems. In terms of the singular data $F \in H^{-1}(\Omega)$, we refer to [42, 41] for piecewise constant approximation of F and [17] using surrogate data indicators. We also refer to [37, 31] on AFEM for more complex singularities.

The approximation error based on regularized data consists of two parts: the regularization error for u and the finite element approximation error for $\mathtt{u}^r$. The analysis of adaptive algorithms applied to the regularized problem is complicated by the facts that optimal choices of the regularization parameter r depend on the local mesh size h (see [25]) and that the error estimates depend both on the local mesh size and on the regularization parameter r.

We present our algorithm in section 3. We control each error in a separate routine: the routine `INTERFACE` controls the first error using the perturbation theory built in [25] (see also Proposition 2.6) and returns the optimal regularization parameter r to use in the routine `SOLVE`, which controls the error of the regularized problem using classic AFEM results based on [17].

Given a target tolerance, the `INTERFACE` routine refines a priori the cells around the immersed domain so that the regularization error can be properly controlled. This procedure ensures that the regularization parameter r is suitable for the local mesh size around the immersed domain. Given the regularization parameter r, the `SOLVE` routine will then approximate the regularized problem using AFEM based on [17] so that the finite element error can also be reduced below the desired tolerance. Our complete algorithm is based on the iteration of the two routines above with a decaying target tolerance.

The performance of our adaptive algorithm is studied adapting the theories from [17, 12] to our regularized problem. The major point to take into account is that all the estimates one obtains are generally dependent on the regularization parameter r, which in turn is generally chosen according to the local mesh size h. More precisely speaking, the following two issues must be analyzed carefully:

- For any $r > 0$, the regularized solutions $\mathtt{u}^r$ are in some approximation class $\mathcal{A}^s$ for some $s \in (0, \frac{1}{d}]$ (see section 4.2 for the definition) and the corresponding quasi-seminorms are uniformly bounded.
- Since regularized data F^r is in $L^2(\Omega)$, we can guarantee that there exists an adaptive method to approximate F^r with a quasi-optimal rate (cf. [17,

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Assumption $\widetilde{A}(s)$]). That is, starting from a subdivision $\mathcal{T}$ and applying the bulk chasing strategy to obtain a refinement $\mathcal{T}^*$ of $\mathcal{T}$, the data indicator (defined in section 3.2) is less than the tolerance τ and

$$\#(\mathcal{T}^*) - \#(\mathcal{T}) \leq C\tau^{-d};$$

see [17, Theorem 7.3]. However, the constant C above depends on the regularization parameter r, i.e., on the local mesh size h, and may lead to a deterioration of the convergence rates.

To resolve the first issue, we follow the arguments from [12]. Thanks to the a priori refinements from the `INTERFACE` part of the algorithm, Lemma 3.2 of [12] allows us to measure the complexity of the `SOLVE` stage independently of r. To remedy the second issue, in Lemma 4.7, we revisit [17, Theorem 7.3] and provide a finer estimate for the constant C above which can be shown to be $C \sim r^{1-d/2}$ by exploiting the fact that F^r is supported in the neighborhood of the immersed domain. It turns out that we can still obtain optimal convergence rates in the two-dimensional case, while we get suboptimal rates in the three-dimensional case. We show this in Theorem 4.18 and Remark 4.19.

The rest of this article is organized as follows. In section 2 we provide some essential notation to define our model problem in the variational sense, and we introduce the data regularization (or data mollification) as well as a regularized version of the model problem. In section 3 we review the AFEM for elliptic problems with $L^2(\Omega)$ forcing data. Following this approach, we then propose our adaptive algorithm for the model problem. The analysis of the adaptive algorithm is presented in section 4. In section 5 we provide some numerical experiments to illustrate the performance of our proposed algorithm. We conclude with some remarks in section 6.

Notation and Sobolev spaces. Let $\Omega \subset \mathbb{R}^d$ be a bounded Lipschitz domain. We write $A \lesssim B$ if $A \leq cB$ for some constant c independent of A, B as well as other discretization parameters. We say $A \sim B$ if $A \lesssim B$ and $B \lesssim A$.

Given a Hilbert space X, we denote with $(\cdot,\cdot)_X$ its inner product and with X' its dual space with the induced norm

$$\|F\|_{X'} = \sup_{\|v\|_X = 1} \langle F, v\rangle_{X',X},$$

where $\langle\cdot,\cdot\rangle_{X',X}$ denotes the duality pairing.

We indicate with $L^2(\Omega)$, $H^1(\Omega)$, and $H^2(\Omega)$ the usual Sobolev spaces and use $(\cdot,\cdot)_\Omega$ to indicate the $L^2(\Omega)$-inner product. For $s \in (0,1)$, we denote the fractional Sobolev spaces $H^s(\Omega)$ using the Sobolev–Slobodeckij norm

$$\|v\|_{H^s(\Omega)} := \left(\|v\|^2_{L^2(\Omega)} + \int_\Omega \int_\Omega \frac{(v(x) - v(y))^2}{|x-y|^{d+2s}} \,\mathrm{d}x\,\mathrm{d}y \right)^{1/2}.$$

For $s \in (1,2)$,

$$\|v\|_{H^s(\Omega)} = \left(\|v\|^2_{L^2(\Omega)} + \|\nabla v\|^2_{H^{s-1}(\Omega)} \right)^{1/2}.$$

For $s \in (\frac{1}{2}, 1]$, we set $H^s_0(\Omega)$ to be the collection of functions in $H^1(\Omega)$ vanishing on $\partial\Omega$. It is well known that $H^s_0(\Omega)$ is the closure of $C^\infty_c(\overline{\Omega})$ (the space of infinitely differentiable functions with compact support in $\overline{\Omega}$) with respect to the norm of $H^s(\Omega)$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

(cf. [21]). Also, $H_0^s(\Omega)$ is an interpolation space between $L^2(\Omega)$ and $H_0^1(\Omega)$ using the real method. Finally for $s \in (\frac{1}{2}, 1]$, we set $H^{-s}(\Omega) = H_0^s(\Omega)'$.

2. Model problem and its regularization.

In this section, we will introduce the variational formulation of our model problem as well as a formulation when the forcing data F is approximated by regularization.

2.1. The forcing data.

Let $\omega \subset \Omega$ be a bounded domain and let $\gamma := \partial\omega$ be its boundary, which we take to be Lipschitz. In what follows, we only consider the case when γ is away from $\partial\Omega$, i.e., there exists a positive constant c_γ such that

$$\operatorname{dist}(\gamma, \partial\Omega) > c_\gamma. \tag{2.1}$$

We assume that the data function $f \in L^\infty(\gamma)$. For a technicality (cf. Lemma 4.12), we further assume that there exists a finite collection of nonoverlapping nonempty open sets $\{\gamma_j \subset \gamma\}_{j=1}^{M_\gamma}$ such that $\sum_{j=1}^{M_\gamma} |\gamma_j| = |\gamma|$ and f does not change its sign on each γ_j. We define I to be the set where f changes sign, i.e.,

$$I := \bigcup_{i=1}^{M_\gamma} \partial\gamma_i. \tag{2.2}$$

The above limitation on the sign change is used only in Lemma 4.12 and allows us to simplify its proof, without sacrificing too much on the generality of the admissible data. In particular, a sufficient condition for the above statement to be true is that the co-dimension two measure of I is bounded, i.e., I consists of a finite number of points for one-dimensional curves embedded in two dimensions, or a collection of curves with finite length for two-dimensional surfaces embedded in three dimensions.

We then consider a forcing data that can be formally written as

$$F = \mathcal{M}f := \int_\gamma \delta(x-y) f(y)\, \mathrm{d}\sigma_y. \tag{2.3}$$

The variational definition of F (see, i.e., [25]) implies that $F \in H^{-s}(\Omega) \subset H^{-1}(\Omega)$ with any fixed $s \in (\frac{1}{2}, 1]$. In fact, for any $v \in H_0^1(\Omega)$, there holds

$$\begin{aligned} \langle F, v\rangle_{H^{-1}(\Omega), H_0^1(\Omega)} &= \int_\gamma f v \,\mathrm{d}\sigma \\ &\le \|f\|_{L^2(\gamma)} \|v\|_{L^2(\gamma)} \\ &\lesssim \|f\|_{L^2(\gamma)} \|v\|_{H^s(\omega)} \lesssim \|f\|_{L^2(\gamma)} \|v\|_{H^s(\Omega)}, \end{aligned} \tag{2.4}$$

where for the first inequality above we applied the Schwarz inequality and for the second inequality we used the trace inequality.

2.2. Weak formulation.

The variational formulation of (1.1) reads, Given a function $f \in L^2(\gamma)$, we seek $u \in H_0^1(\Omega)$ such that

$$A(u, v) = \langle F, v\rangle_{H^{-1}(\Omega), H_0^1(\Omega)} \qquad \text{for all } v \in H_0^1(\Omega), \tag{2.5}$$

where

$$A(v, w) = \int_\Omega \nabla v^{\intercal} A(x) \nabla w + c(x) v w \,\mathrm{d}x \qquad \text{for all } v, w \in H_0^1(\Omega).$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Assumption (1.2) and the nonnegativity of $c(x)$ guarantee that the bilinear form $A(\cdot,\cdot)$ is bounded and coercive, i.e., there exist positive constants m, M so that for $v, w \in H_0^1(\Omega)$,

$$A(v,w) \le M\|v\|_{H^1(\Omega)}\|w\|_{H^1(\Omega)} \text{ and } A(v,v) \ge m\|v\|^2_{H^1(\Omega)}, \tag{2.6}$$

and (2.5) admits a unique solution by the Lax–Milgram lemma. Bound (2.6) also implies that the energy norm $|||v||| := \sqrt{A(v,v)} \sim \|v\|_{H^1(\Omega)}$. In what follows, we use the energy norm $|||.|||$ instead of $\|.\|_{H^1(\Omega)}$ in our adaptive algorithm as well as in the performance analysis.

2.3. Regularization. The regularization of F is based on the approximation of the Dirac delta distribution. To this end, we first define a class of functions ψ satisfying the following assumptions.

Assumption 2.1. Given $k \in \mathbb{N}$, let $\psi(x)$ in $L^\infty(\mathbb{R}^d)$ such that the following hold:

1. Nonnegativity: $\psi(x) \ge 0$.
2. Compact support: $\psi(x)$ is compactly supported, with support $\operatorname{supp}(\psi)$ contained in $B_{r_0}(0)$ (the ball centered in zero with radius r_0) for some $r_0 > 0$.
3. Moments condition: Given $k \in \mathbb{N}$, we say ψ satisfies the kth order moment condition if

$$\int_{\mathbb{R}^d} y_i^\alpha \psi(x-y)\,\mathrm{d}y = x_i^\alpha, \qquad i = 1\ldots d, \quad 0 \le \alpha \le k, \qquad \text{for all } x \in \mathbb{R}^d; \tag{2.7}$$

4. Monotonicity: $\psi(x/r_2) \le \psi(x/r_1)$ if $r_2 < r_1$.

We refer to [25] for some examples of ψ and [28, section 3] for a general discussion. Here we only consider even, nonnegative functions ψ_{1d} that are supported in $[-1,1]$, are nonincreasing in $[0,1]$, and satisfy $\int_{\mathbb{R}} \psi_{1d} = 1$. Then we generate ψ in $\mathbb{R}^d$ by the radially symmetric extension $\psi_{1d}(|x|)$ or the tensor product extension $\prod_{i=1}^d \psi_{1d}(x_i)$. A function ψ defined in this way satisfies Assumption 2.1 with $k = 1$. Using the above ψ, for $r > 0$, we define the Dirac approximation δ^r by

$$\delta^r(x) := \frac{1}{r^d}\psi\left(\frac{x}{r}\right). \tag{2.8}$$

Thus,

$$\lim_{r\to 0} \delta^r(x) = \lim_{r\to 0} \frac{1}{r^d}\psi\left(\frac{x}{r}\right) = \delta(x),$$

where the limit should be understood in the space of Schwarz distributions.

Remark 2.2 (nonnegativity of ψ). We will use the nonnegativity of ψ to analyze the performance of our adaptive algorithm. However, this is not required in the error analysis for finite element discretization of (1.1) using quasi-uniform subdivisions of Ω; see [25] for more details.

Definition 2.3 (regularization). *For a function $v \in L^1(\Omega)$ we define its regularization $v^r(x)$ in the domain Ω through the mollifier ψ by*

$$v^r(x) := \int_\Omega \delta^r(x-y)v(y)\,\mathrm{d}y \qquad \text{for all } x \in \Omega, \tag{2.9}$$

where δ^r is given by (2.8) *and where ψ satisfies Assumption* 2.1 *for some $k \ge 0$.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

For functionals F in negative Sobolev spaces, say, $F \in H^{-s}(\Omega)$, with $s \in (\frac{1}{2}, 1]$, we define their regularization F^r by the action of F on v^r with $v \in H_0^s(\Omega)$, i.e.,

$$\langle F^r, v\rangle_{H^{-s}(\Omega),H_0^s(\Omega)} := \langle F, v^r\rangle_{H^{-s}(\Omega),H_0^s(\Omega)}. \tag{2.10}$$

We note that the definition of F^r is well defined with F given by (2.3). In fact, by [25, Corollary 1], there holds

$$\|v - v^r\|_{H^s(\omega)} \lesssim \|v\|_{H^s(\Omega)}.$$

Therefore, according to the argument in (2.4), we have

$$\langle F^r, v\rangle_{H^{-s}(\Omega),H_0^s(\Omega)} \lesssim \|f\|_{L^2(\gamma)} \|v^r\|_{H^s(\omega)} \lesssim \|f\|_{L^2(\gamma)} \|v\|_{H^s(\Omega)}.$$

Remark 2.4. For F defined by (2.3), applying Fubini's theorem to the right-hand side of (2.10) yields

$$F^r(x) = \int_\gamma f(y)\delta^r(y-x)\,\mathrm{d}y \in L^2(\Omega).$$

If ψ is chosen to be symmetric, the definition of F^r can be interpreted by replacing δ in (2.3) with the Dirac approximation δ^r.

Remark 2.5 (error estimate of the regularization). Lemma 10 of [25] implies that under Assumption 2.1, together with (2.1), the following regularization error estimate holds when $r < 1$:

$$\|F - F^r\|_{H^{-1}(\Omega)} \le C_{\mathtt{reg}} r^{1/2} \|f\|_{L^2(\gamma)}, \tag{2.11}$$

where the constant $C_{\mathtt{reg}}$ depends on ψ in Assumption 2.1 and on ω.

2.4. Regularized problem. A regularized version of problem (2.5) reads, Find $\mathtt{u}^r \in H_0^1(\Omega)$ satisfying

$$A(\mathtt{u}^r, v) = \langle F^r, v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} \qquad \text{for all } v \in H_0^1(\Omega). \tag{2.12}$$

Notice that $\mathtt{u}^r$ exists and is unique. Moreover, (2.6) and Remark 2.5 imply that $\mathtt{u}^r$ converges to u in the energy norm with the rate $O(r^{1/2})$. That is, we have the following proposition:

PROPOSITION 2.6 (see also Theorem 14 of [25]). *When Assumption* 2.1 *holds, let u and $\mathtt{u}^r$ be the solution to* (2.5) *and* (2.12), *respectively. Then there holds*

$$|||u - \mathtt{u}^r||| \le m^{-1/2} C_{\boldsymbol{reg}} r^{1/2} \|f\|_{L^2(\gamma)}. \tag{2.13}$$

3. Numerical algorithm. We approximate the solution to the weak formulation (2.5) by solving the regularized problem (2.12) using AFEMs along with a choice of the regularization parameter r. As the number of degrees of freedom increases, r will tend to zero *with a rate linked to the target tolerance*. Recalling from Remark 2.4, the regularized data F^r is an $L^2(\Omega)$ function so that we can use classical residual error estimators for adaptivity. In this section, we first review AFEMs for elliptic problems with $L^2(\Omega)$ forcing data based on [17, 42, 38]. Then we introduce our adaptive algorithm for (2.5).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

3.1. Finite element approximation.

We additionally assume that Ω is a polytope. Given a data function $g \in L^2(\Omega)$, we consider a finite element approximation of $w_g \in H_0^1(\Omega)$ which uniquely solves

$$A(w_g, v) = (g, v) \quad \text{for all } v \in H_0^1(\Omega). \tag{3.1}$$

Set $\mathcal{T}$ to be a subdivision of Ω made by simplices. We assume that $\mathcal{T}$ is conforming (no hanging nodes) and shape-regular in a sense of [20, 16], i.e., there exists a positive constant $c_{\mathtt{sr}}$ so that for each cell $T \in \mathcal{T}$,

$$\operatorname{diam}(T) \le c_{\mathtt{sr}} \rho_T$$

with $\operatorname{diam}(T)$ and ρ_T denoting the size of T and the diameter of the largest ball inscribed in T, respectively. We also set $h_T = |T|^{1/d}$, with $|T|$ denoting the volume of T. So $h_T \sim \operatorname{diam}(T)$, with the hiding constants depending on $c_{\mathtt{sr}}$. Denote $\mathbb{V}(\mathcal{T}) \subset H_0^1(\Omega)$ the space of continuous piecewise linear functions subordinate to $\mathcal{T}$. So the finite element discretization for (3.1) reads as follows.

Algorithm 3.1 $W_g = \mathtt{GAL}(\mathcal{T}, g)$

```
  Solve $A(W_g, V) = (g, V)$   for all $V \in \mathbb{V}(\mathcal{T})$;
  return $W_g$;
```

3.2. A posteriori error estimates with $L^2(\Omega)$ data.

AFEMs rely on the so-called computable error estimators to evaluate the quality of the finite element approximation on each cell T in the underlying subdivision $\mathcal{T}$. Here we consider the following local jump residual and data indicators: given a conforming subdivision $\mathcal{T}$, a finite element function $V \in \mathbb{V}(\mathcal{T})$, and a data function $g \in L^2(\Omega)$, we denote $\mathcal{F}_T$ the collection of all faces of $T \in \mathcal{T}$ and define

$$j(V, T, \mathcal{T}) := \left(\sum_{F \in \mathcal{F}_T} h_F \| [A \cdot \nabla V] \|_{L^2(F)}^2 \right)^{1/2} \text{ and } d(g, T, \mathcal{T}) := h_T \|g\|_{L^2(T)}, \tag{3.2}$$

where h_F is the size of F and $[.]$ denotes the normal jump across the face F. Their global counterparts are given by

$$\mathcal{J}(V, \mathcal{T}) := \left(\sum_{T \in \mathcal{T}} j(V, T, \mathcal{T})^2 \right)^{1/2} \text{ and } \mathcal{D}(g, \mathcal{T}) := \left(\sum_{T \in \mathcal{T}} d(g, T, \mathcal{T})^2 \right)^{1/2}.$$

Letting $W_g = \mathtt{GAL}(\mathcal{T}, g)$, we define the local error indicator,

$$e(W_g, T, \mathcal{T}) = \left(j(W_g, T, \mathcal{T})^2 + d(g, T, \mathcal{T})^2 \right)^{1/2}$$

as well as the global indicator,

$$\mathcal{E}(W_g, \mathcal{T}) = \left(\sum_{T \in \mathcal{T}} e(W_g, T, \mathcal{T})^2 \right)^{1/2}.$$

The computation of such indicators is usually performed in the stage `ESTIMATE` of AFEM algorithms, as summarized in Algorithm 3.2.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 3.2 $\{j(T), d(T), e(T)\}_{T\in\mathcal{T}} = \texttt{ESTIMATE}(\mathcal{T}, W_g, g)$

```
Given the approximate solution $W_g$ on $\mathcal{T}$;
for $T \in \mathcal{T}$ do
    Compute $j(T) = j(W, T, \mathcal{T})$;
    Compute $d(T) = d(g, T, \mathcal{T})$;
    Compute $e(T) = e(W_g, T, \mathcal{T})$;
end for
return $\{j(T), d(T), e(T)\}_{T\in\mathcal{T}}$;
```

3.3. Marking of cells based on error indicators. The estimated error per cell obtained in the ESTIMATE algorithm are used to perform refinement based on the bulk chasing strategy [18] (or the Dörfler marking strategy), summarized in Algorithm 3.3. Here we set $\operatorname{ind}(T)$ to be a local indicator and the corresponding global indicator is denoted by IND.

Algorithm 3.3 $\mathcal{M} = \texttt{MARK}(\{\operatorname{ind}(T)\}_{T\in\mathcal{T}}, \mathcal{T}, \theta)$

```
Given a cell indicator $\{\operatorname{ind}(T)\}_{T\in\mathcal{T}}$ and a bulk parameter $\theta \in (0,1)$;
Find a smallest subset $\mathcal{M}$ of $\mathcal{T}$ satisfying
(3.3)    $\left(\sum_{T\in\mathcal{M}} \operatorname{ind}(T)^2\right)^{1/2} \geq \theta \,\mathrm{IND}.$
return $\mathcal{M}$;
```

3.4. Refinements of subdivisions. Conforming refinement strategies, such as newest vertex bisection [8, 34, 43], can be used to construct a sequence of conforming simplicial subdivisions $\{\mathcal{T}_k\}_{k=0}^{\infty}$ by adaptively bisecting a set of cells $\mathcal{R}_k \subset \mathcal{T}_k$. However, our results hold also for more general nonconforming mesh refinement strategies satisfying conditions 3 (successive subdivisions), 4 (complexity of refinement), and 7 (admissible subdivision) in [13]. For instance, in our numerical illustration in section 5, we use refinements on quad- and hex-meshes where conformity is enforced via hanging node constraints. Irrespective of the strategy used to refine the grid (either conforming or nonconforming with hanging node constraints), we obtain a sequence of uniformly shape-regular subdivisions $\{\mathcal{T}_k\}_{k\geq 0}$ satisfying

$$\#(\mathcal{T}_k) - \#(\mathcal{T}_0) \leq C_{\texttt{com}} \sum_{j=0}^{k-1} \#(\mathcal{R}_j), \tag{3.4}$$

for some universal constant $C_{\texttt{com}} \geq 1$. We write the above refinement process from $\mathcal{T}_k$ to $\mathcal{T}_{k+1}$ as $\mathcal{T}_{k+1} = \texttt{REFINE}(\mathcal{T}_k, \mathcal{R}_k)$, summarized in Algorithm 3.4.

3.4.1. Overlay of two subdivisions. Provided that both $\mathcal{T}_1$ and $\mathcal{T}_2$ are refinements of $\mathcal{T}_0$, we say that $\mathcal{T}$ is the *overlay* of $\mathcal{T}_1$ and $\mathcal{T}_2$ when $\mathcal{T}$ consists of the union of all cells of $\mathcal{T}_1$ that do not contain smaller cells of $\mathcal{T}_2$ and vice versa. Clearly, there holds

$$\#(\mathcal{T}) \leq \#(\mathcal{T}_1) + \#(\mathcal{T}_2) - \#(\mathcal{T}_0). \tag{3.5}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Algorithm 3.4 $\mathcal{T}_{k+1} = \texttt{REFINE}(\mathcal{T}_k, \mathcal{R}_k)$

```
    (i) (for triangular or tetrahedral meshes) Bisect the marked cells $\mathcal{R}_k \subset \mathcal{T}_k$ once;
    Add all extra bisections to produce a conforming subdivision $\mathcal{T}_{k+1}$;
    (ii) (for quadrilateral or hexahedral meshes) Split the marked cells into four
children in two dimensions or eight children in three dimensions;
    Refine all extra cells to produce a nonconforming subdivision $\mathcal{T}_{k+1}$ with at most
    one hanging node per face, and enforce conformity via hanging node constraints;
    return $\mathcal{T}_{k+1}$;
```

3.5. AFEM with control on L^2 data. It is well known (see, e.g., [5, 19, 17]) that one can obtain a global upper and lower bound of the approximation error by the error indicator, i.e., there exist positive constants $C_{\texttt{rel}}$ and $C_{\texttt{eff}}$ so that

$$|||w_g - W_g||| \le C_{\texttt{rel}}\mathcal{E}(W_g, \mathcal{T}) \text{ and } \mathcal{E}(W_g, \mathcal{T}) \le C_{\texttt{eff}} E(w_g, \mathcal{T}) \tag{3.6}$$

with

$$E(w_g, \mathcal{T}) := \left(|||w_g - W_g|||^2 + \mathcal{D}(g, \mathcal{T})^2\right)^{1/2}. \tag{3.7}$$

Remark 3.1 (local lower bound with oscillation). The data indicator $\mathcal{D}(g, \mathcal{T})$ in the lower bound can be replaced by the data oscillation provided that the refinement strategy satisfies the interior node property [12, 17, 38, 36]:

$$\operatorname{osc}(g, \mathcal{T}) = \left(\sum_{T \in \mathcal{T}} h_T^2 \|g - a_T(g)\|_{L^2(T)}^2\right)^{1/2},$$

where $a_T(.)$ denotes the average on T. Note that $\operatorname{osc}(g, \mathcal{T}) \le \mathcal{D}(g, \mathcal{T})$, and the decay of the data oscillation could be faster if g is more regular. However, in our case, we set $g = F^r$ to be as in Definition 2.3, and the smoothness of g depends on the choice of ψ in Assumption 2.1 as well as the regularization parameter r. In order to simplify our analysis, we will treat F^r as an $L^2(\Omega)$ function and the decay rate of oscillation is then the same as the data indicator $\mathcal{D}(g, \mathcal{T})$.

The DATA routine guarantees that the global data indicator $\mathcal{D}$ is below a user defined tolerance. This allows us to control the total error indicator $\mathcal{E}$.

Algorithm 3.5 $\mathcal{T}^* = \texttt{DATA}(\mathcal{T}, g, \tau, \widetilde{\theta})$

```
    $\mathcal{T}^* = \mathcal{T}$;
    while $\mathcal{D}(g, \mathcal{T}^*) > \tau$ do
      $\mathcal{M} = \texttt{MARK}(\{d(g, T, \mathcal{T}^*)\}_{T \in \mathcal{T}}, \mathcal{T}^*, \widetilde{\theta})$;
      $\mathcal{T}^* = \texttt{REFINE}(\mathcal{T}^*, \mathcal{M})$;
    end while
    return $\mathcal{T}^*$;
```

3.6. AFEM algorithm for L^2 data. To summarize the above steps in a complete AFEM algorithm, we follow [17] to solve problem (3.1) by iteratively generating refined subdivisions and the corresponding finite element approximations. For convenience, we denote $W_k \in \mathbb{V}_k := \mathbb{V}(\mathcal{T}_k)$ the finite element approximation of w_g on $\mathcal{T}_k$. Similarly, we denote the local indicators $j_k(T) := j(W_k, T, \mathcal{T}_k)$, $d_k(T) := d(g, T, \mathcal{T}_k)$,

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 3.6 $\{W^*, \mathcal{T}^*\} = \texttt{SOLVE}(\mathcal{T}_0, g, \tau, \theta, \widetilde{\theta}, \lambda)$

```
$W_0 = \texttt{GAL}(\mathcal{T}_0, g)$;
$\{j_0(T), d_0(T), e_0(T)\}_{T\in\mathcal{T}_0} = \texttt{ESTIMATE}(\mathcal{T}_0, W_0, g)$;
$k = 0$;
while $\mathcal{E}_k > \tau$ do
    if $\mathcal{D}_k > \sigma_k := \lambda\theta\mathcal{E}_k$ then
        $\mathcal{T}_{k+1} = \texttt{DATA}(\mathcal{T}_k, g, \frac{\sigma_k}{2}, \widetilde{\theta})$;
    else
        $\mathcal{M}_k = \texttt{MARK}(\{e_k(T)\}_{T\in\mathcal{T}_k}, \mathcal{T}_k, \theta)$;
        $\mathcal{T}_{k+1} = \texttt{REFINE}(\mathcal{T}_k, \mathcal{M}_k)$;
    end if
    $k = k + 1$;
    $W_k = \texttt{GAL}(\mathcal{T}_k, g)$;
    $\{j_k(T), d_k(g,T), e_k(T)\}_{T\in\mathcal{T}_k} = \texttt{ESTIMATE}(\mathcal{T}_k, W_k, g)$;
end while
return $\{W_k, \mathcal{T}_k\}$;
```

$e_k(T) := e(W_k, T, \mathcal{T}_k)$ and global indicators $\mathcal{J}_k := \mathcal{T}(W_k, \mathcal{T}_k)$, $\mathcal{D}_k := \mathcal{D}(g, \mathcal{T}_k)$, $\mathcal{E}_k := \mathcal{E}(W_k, \mathcal{T}_k)$.

Starting from a conforming subdivision $\mathcal{T}_0$ and given a tolerance $\tau > 0$, we choose $\theta, \widetilde{\theta}, \lambda \in (0,1)$ and construct the approximation U_k by the routine SOLVE, defined in Algorithm 3.6.

According to [17], the routine SOLVE guarantees the decay of the error indicator $\mathcal{E}_k$ with some decay factor $\alpha \in (0,1)$ (see also Theorem 4.4) and hence, when this routine terminates, we obtain that

$$\|\!|w_g - W_g|\!\| \le C_{\texttt{reg}}\tau. \tag{3.8}$$

Here we applied the upper bound in (3.6).

Remark 3.2 (an alternative AFEM algorithm). In section 4, we will adapt the approximation theory developed in [17] to investigate the performance of SOLVE for the regularized problem (2.12). On the other hand, we could instead apply the classical AFEM cycle:

$$\texttt{GAL} \to \texttt{ESTIMATE} \to \texttt{MARK} \to \texttt{REFINE}.$$

We note that the same performance in terms of tolerances can be obtained by following the arguments from [38, 12, 14] together with approximation properties developed in section 4 (cf. Corollaries 4.3 and 4.11). However, the classical AFEM cycle would suffer from a higher computational cost related to the higher number of GAL steps that are computed in classical AFEM, and therefore we proceed as in Algorithm 3.6, following the steps of [17].

3.7. AFEM algorithm for regularized H^{-1} data. Let us first provide an assumption on the initial subdivision $\mathcal{T}_0$ related to the interface γ. We denote with

$$\mathcal{G} := \mathcal{G}(\gamma, \mathcal{T}) := \{T \in \mathcal{T} : T \cap \gamma \ne \emptyset\} \text{ and } \operatorname{diam}(\mathcal{G}) = \max_{T\in\mathcal{G}} h_T,$$

and we assume that the initial subdivision is sufficiently refined to capture the characteristics of γ, that is, $\mathcal{G}(\gamma, \mathcal{T}_0)$ is quasi-uniform and for any uniform refinement $\mathcal{T}_i$ of level i of $\mathcal{T}_0$, we have that

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

$$\sum_{T\in\mathcal{G}(\gamma,\mathcal{T}_i)} |T| \sim q^{-i/d} \sum_{T\in\mathcal{G}(\gamma,\mathcal{T}_0)} |T| \sim q^{-i/d}|\gamma|, \tag{3.9}$$

where $q > 1$ is the volume ratio between a cell and its children. In two-dimensional space, for instance, $q = 2$ for the newest vertex bisection and $q = 4$ for the quad-refinement. The above assumption shows that there exists a positive constant c depending on $c_{\mathtt{sr}}$ such that the tubular neighborhood of γ with width $cq^{-i/d}$ covers $\mathcal{G}(\gamma,\mathcal{T}_i)$.

Remark 3.3. Condition 3.9 is a way to ask that the initial subdivision $\mathcal{T}_0$ properly resolves γ. This is possible for Lipschitz curves and surfaces and requires that the initial subdivision $\mathcal{T}$ is sufficiently refined around γ, with a local grid size that generally depends on the Lipschitz constant of γ.

Given a target tolerance $\tau > 0$, we shall determine the regularization parameter r and approximate problem (3.1) with $g = F^r$ via SOLVE so that the output approximation U satisfies

$$|||u - U||| \le |||u - \mathtt{u}^r||| + |||\mathtt{u}^r - U||| \le C_{\mathtt{reg}}\tau.$$

To control $|||u - \mathtt{u}^r|||$, in view of Proposition 2.6, we can set

$$m^{-1/2}C_{\mathtt{reg}}r^{1/2}\|f\|_{L^2(\gamma)} \le \frac{C_{\mathtt{rel}}\tau}{2}.$$

Hence we choose the regularization parameter

$$r =: r(\tau) := m\left(\frac{C_{\mathtt{rel}}\tau}{2C_{\mathtt{reg}}\|f\|_{L^2(\gamma)}}\right)^2. \tag{3.10}$$

Remark 3.4 (values of the constants in (3.10)). Since it is nontrivial to compute the constants that appear in (3.10), in the simulations presented in section 5 we select $r = \tau^2$.

From the computational point of view, if $r \ll h_T$ for $T \in \mathcal{G}(\gamma,\mathcal{T})$ and if $V \in \mathbb{V}(\mathcal{T})$ is nonzero in T, it is possible that $\delta^r(q_1 - q_2) = 0$ when q_1 is a quadrature point on γ and q_2 is a quadrature point in T. In such a case, we would approximate $\int_T F^r V$ by zero using the quadrature scheme, resulting in a "transparent" γ, implying a total loss of accuracy. In order to avoid such a situation, we also refine the subdivision before controlling the error $|||\mathtt{u}^r - U|||$ from SOLVE. Our goal is to find a refinement $\mathcal{T}^*$ of $\mathcal{T}$ so that

$$2\ \mathrm{diam}(\mathcal{G}(\gamma,\mathcal{T}^*)) \le r.$$

To this end, we introduce the routine INTERFACE in Algorithm 3.7.

Algorithm 3.7 $\mathcal{T}^* = \mathtt{INTERFACE}(\mathcal{T}, r)$

```
$\mathcal{T}^* = \mathcal{T}$;
while $\mathrm{diam}(\mathcal{G}) > \frac{r}{2}$ do
    Find the set $\mathcal{M} := \{T \in \mathcal{G}(\gamma,\mathcal{T})$ s.t. $h_T > \frac{r}{2}\}$;
    $\mathcal{T}^* = \mathtt{REFINE}(\mathcal{T}^*, \mathcal{M})$;
end while
return $\mathcal{T}^*$;
```

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 3.8 $\{U,\mathcal{T}\} = \texttt{REGSOLVE}(g,\mathcal{T}_0,j_{\max},\tau_0,\beta,\theta,\widetilde{\theta},\lambda,\widetilde{\mu})$

```
for $j = 0 : j_{\max}$ do
    $r_j = r(\tau_j)$;
    $\widetilde{\mathcal{T}}_j = \texttt{INTERFACE}(\mathcal{T}_j, r_j)$;
    $\{U_{j+1},\mathcal{T}_{j+1}\} = \texttt{SOLVE}(\widetilde{\mathcal{T}}_j, F^{r_j}, \widetilde{\mu}\tau_j, \theta, \widetilde{\theta}, \lambda)$;
    $\tau_{j+1} = \beta\tau_j$;
    $j = j+1$;
end for
return $\{U_{j_{\max}}, \mathcal{T}_{j_{\max}}\}$;
```

Given an initial conforming subdivision $\mathcal{T}_0$ satisfying assumption (3.9), an initial tolerance τ_0, and $\beta,\theta,\widetilde{\theta},\lambda,\widetilde{\mu} \in (0,1)$, the solver routine REGSOLVE for (2.5) reads as in Algorithm 3.8.

Here $\widetilde{\mu}$ is a constant whose choice will be explained later in Lemma 4.16. Note that the subroutine SOLVE in REGSOLVE guarantees that the energy error between U and $\mathtt{u}^r$ is bounded by $\widetilde{\mu}\tau_j$. Therefore we have the following proposition:

PROPOSITION 3.5. *Let* u *and* U_j *be defined as in* (2.5) *and* REGSOLVE*, respectively. Then for each nonnegative integer* j*,*

$$\begin{aligned}|||u - U_{j+1}||| &\leq |||u - u^{r_j}||| + |||u^{r_j} - U_{j+1}||| \\ &\lesssim |||u - u^{r_j}||| + \mathcal{E}(u^{r_j}, \mathcal{T}_{j+1}) \lesssim \tau_j.\end{aligned}$$

Remark 3.6 (another algorithm). Since INTERFACE is an a priori process, we can also solve (2.5) with only one iteration in REGSOLVE. That is,

$$\{U,\mathcal{T}\} = \texttt{REGSOLVE}(g,\mathcal{T}_0,1,\tau,\cdot,\theta,\widetilde{\theta},\lambda,\widetilde{\mu})$$

with $\tau = \tau_0\beta^{j_{\max}}$.

4. Measuring the performance.

In this section we measure the performance of REGSOLVE, i.e., we analyze the subroutines INTERFACE and SOLVE, respectively. We use the notation :: to connect a routine and its subroutine. For instance, the routine SOLVE in REGSOLVE is denoted by REGSOLVE::SOLVE.

4.1. Performance of INTERFACE.

The following proposition provides the performance of INTERFACE.

PROPOSITION 4.1 (performance of INTERFACE). *Under assumption* (3.9) *for the initial subdivision* $\mathcal{T}_0$*, given a refinement* $\mathcal{T}$ *of* $\mathcal{T}_0$*, let* $\widetilde{\mathcal{T}} = \texttt{INTERFACE}(\mathcal{T}, r)$ *with* $r < 2\,\mathrm{diam}(\mathcal{G}(\gamma,\mathcal{T}_0))$*. Then there exists a positive constant* $\widetilde{I}_0 := \widetilde{I}_0(c_{\boldsymbol{sr}},\gamma,C_{\boldsymbol{com}})$ *so that*

$$\#(\widetilde{\mathcal{T}}) - \#(\mathcal{T}) \leq \widetilde{I}_0 r^{1-d}. \tag{4.1}$$

The proof is based on counting the number of bisections of $T \in \mathcal{G}(\gamma,\mathcal{T}_0)$. Here we skip the proof and refer to the appendix of [26] for more details.

Remark 4.2. The above estimate holds provided that the initial refinement $\mathcal{T}_0$ is capable of capturing the shape of γ, i.e., that the assumption provided in (3.9) is valid.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

A direct application of Proposition 4.1 is to bound the cardinality of refined cells from `INTERFACE` in `REGSOLVE`.

Corollary 4.3 (performance of `REGSOLVE::INTERFACE`). *Let $\{\widetilde{\mathcal{T}}_j\}$ be the sequence of subdivisions generated by* `INTERFACE` *in* `REGSOLVE`*. Then at the jth iterate, there exists a positive constant $I_0 := \widetilde{I}_0(c_{\boldsymbol{sr}}, \gamma, f, C_{\boldsymbol{com}}, C_{\boldsymbol{rel}}, m)$ satisfying*

$$\#(\widetilde{\mathcal{T}}_j) - \#(\mathcal{T}_j) \le I_0 \tau_j^{-2(d-1)}. \tag{4.2}$$

Proof. The target estimate directly follows from (4.1) as well as $r \sim \tau^2$ according to (3.10). ∎

4.2. Performance of `SOLVE`. Let us review some estimates for the complexity of `SOLVE` following the analysis from [17].

Contraction property. One instrumental tool to evaluate the performance of `SOLVE` is the following contraction property (cf. [17, Theorem 4.3]).

Theorem 4.4 (contraction of `SOLVE`). *There exist two constants $\alpha \in (0,1)$ and $\widetilde{\alpha} > 0$ depending on $c_{\boldsymbol{sr}}$, m, M and on the bulk parameter θ in* `SOLVE` *such that for all $k \ge 0$,*

$$|||w_g - W_{k+1}|||^2 + \widetilde{\alpha}\mathcal{E}(W_{k+1}, \mathcal{T}_{k+1})^2 \le \alpha^2 \Big(|||w_g - W_k|||^2 + \widetilde{\alpha}\mathcal{E}(W_k, \mathcal{T}_k)^2 \Big).$$

Approximation classes. We denote $\mathscr{T}_n$ the set of all conforming subdivisions generated from $\mathcal{T}_0$ satisfying $\#(\mathcal{T}) \le n$. Define the best error obtained in $\mathscr{T}_n$,

$$\sigma_n(u)_{H_0^1(\Omega)} := \inf_{\mathcal{T} \in \mathscr{T}_n} |||u - U_{\mathcal{T}}|||,$$

with $U_{\mathcal{T}} \in \mathbb{V}(\mathcal{T})$ denoting the Galerkin projection of u, i.e.,

$$A(U_{\mathcal{T}}, V) = \langle F, V \rangle_{H^{-1}(\Omega), H_0^1(\Omega)} \qquad \text{for all } V \in \mathbb{V}(\mathcal{T}),$$

and it also satisfies that

$$|||u - U_{\mathcal{T}}||| = \inf_{V \in \mathbb{V}(\mathcal{T})} |||u - V|||.$$

Define the approximation class $\mathcal{A}^s$ with $s \in (0, \frac{1}{d}]$ to be the set of all $v \in H_0^1(\Omega)$ such that the quasi-seminorm

$$|v|_{\mathcal{A}^s} := \sup_{n \ge 1} \Big(n^s \sigma_n(v)_{H_0^1(\Omega)} \Big)$$

is finite. Due to the nonzero jump of the normal derivative of u on γ and according to the discussion from section 10 of [6], the best possible convergence rate is given by $s = \frac{1}{2(d-1)}$.

Performance of `DATA`. The approximation class $\mathcal{A}^s$ provides the rate of convergence for the energy error $|||u - U_{\mathcal{T}}|||$. Recalling that given $g \in L^2(\Omega)$, the total error $E(w_g, \mathcal{T})$ defined in (3.7) consists of both the energy error and the data indicator. So we are also concerned with the rate of convergence for the data indicator $\mathcal{D}(g, \mathcal{T})$. Here we assume as follows.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Assumption 4.5. For $\tau > 0$ and a fixed bulk parameter $\widetilde{\theta} \in (0,1)$, set $\mathcal{T}^* = \mathtt{DATA}(\mathcal{T}, g, \tau, \widetilde{\theta})$. Then for $s \in (0, \frac{1}{d}]$, there exists a positive constant G_s (depending on g and $\widetilde{\theta}$) satisfying

$$\#(\mathcal{T}^*) - \#(\mathcal{T}) \leq G_s \tau^{-1/s}.$$

Cardinality of refined cells in SOLVE. In the routine SOLVE, we need to estimate the cardinalities of $\mathcal{M}_k$ as well as the cells refined from DATA. The latter comes from Assumption 4.5. The estimate of the former requires the following bulk property (cf. [17, Lemma 5.2]).

LEMMA 4.6. *Assume that the bulk parameter $\theta \in (0, \theta_*)$ with*

$$\theta_* = \frac{1}{C_{\boldsymbol{eff}} \sqrt{1 + C_L^2}}. \tag{4.3}$$

Let $\mathcal{T}^*$ be a refinement of $\mathcal{T}$ and denote $\mathcal{R}_{\mathcal{T} \to \mathcal{T}^*}$ the set all refined cells from $\mathcal{T}$ to $\mathcal{T}^*$. If $E(w_g, \mathcal{T}^*) \leq \xi E(w_g, \mathcal{T})$ with

$$\xi := \sqrt{1 - \frac{\theta^2}{\theta_*^2}}, \tag{4.4}$$

there holds $\mathcal{E}(W_g, \mathcal{R}_{\mathcal{T} \to \mathcal{T}^*}) \geq \theta \mathcal{E}(W_g, \mathcal{T})$.

Using Assumption 4.5 and the above lemma, Lemma 5.3 of [17] implies that for each iterate k in SOLVE, we have

$$\#(\mathcal{M}_k) \lesssim (|w_g|_{\mathcal{A}^s} + G_s)^{1/s} E(w_g, \mathcal{T}_k)^{-1/s}. \tag{4.5}$$

4.3. Performance of REGSOLVE. In this section, we shall adapt the results in the previous subsection to REGSOLVE.

4.3.1. Performance of DATA using $\boldsymbol{F^r}$. To show that Assumption 4.5 holds for $g \in L^2(\Omega)$ with $s = \frac{1}{d}$, starting from a conforming initial subdivision $\mathcal{T}_0$ and using a greedy algorithm (see Algorithm 4.1), we can find a refinement $\mathcal{T}$ of $\mathcal{T}_0$ so that the data indicator $\mathcal{D}(g, \mathcal{T})$ is smaller than a target tolerance τ.

Algorithm 4.1 $\mathcal{T} = \mathtt{GREEDY}(\mathcal{T}_0, g, \tau)$

```
$\mathcal{T} = \mathcal{T}_0$
while $\mathcal{D}(g, \mathcal{T}) > \tau$ do
    $T = \operatorname{argmax}\{d(g, T, \mathcal{T})\}$;
    $\mathcal{T} = \mathtt{REFINE}(\mathcal{T}, \{T\})$;
end while
return $\mathcal{T}$;
```

According to [17, Theorem 7.3], there exists a positive constant K depending only on the shape regularity constant $c_{\mathtt{sr}}$ such that

$$\#(\mathcal{T}) - \#(\mathcal{T}_0) \leq K \|g\|_{L^2(\Omega)}^2 \tau^{-d}.$$

The above result can be extended by replacing $\mathcal{T}_0$ with its refinement $\mathcal{T}$, i.e., $\mathcal{T}^* = \mathtt{GREEDY}(\mathcal{T}, g, \tau)$, and there holds

$$\#(\mathcal{T}^*) - \#(\mathcal{T}) \leq K \|g\|_{L^2(\Omega)}^2 \tau^{-d}. \tag{4.6}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

This is because the marked cells in $\texttt{GREEDY}(\mathcal{T}, g, \tau)$ are contained in those generated by $\texttt{GREEDY}(\mathcal{T}_0, g, \tau)$; see [12, Proposition 2] for a detailed discussion. Hence, any $L^2(\Omega)$ function g satisfies Assumption 4.5 with $s = \frac{1}{d}$ and $\|g\|^2_{L^2(\Omega)} \sim G_{1/d}$. When $g = F^r$ as defined in Remark 2.4, the constant $G_{1/d}$ may still depend on r in an arbitrary refinement of $\mathcal{T}_0$. However, the refinement process in DATA is based on the subdivisions generated by INTERFACE. So cells marked in GREEDY should be located in a neighborhood of a tubular extension of γ, whose width can be controlled by the regularization parameter r. In order to see the dependence of (4.6) on r, we modify the argument of Lemma 7.3 of [17].

LEMMA 4.7 (approximation class for F^r). *Assume that $f \in L^\infty(\gamma)$ and F^r is defined as in Remark 2.4 for any $r > 0$. Letting the initial subdivision $\mathcal{T}_0$ satisfy (3.9), we define $\widetilde{\mathcal{T}} = \texttt{INTERFACE}(\mathcal{T}_0, r)$ with $r < c_\gamma$. For any $\tau > 0$, the cardinality of refined cells in $\mathcal{T}^* = \texttt{GREEDY}(\widetilde{\mathcal{T}}, F^r, \tau)$ can be bounded by*

$$\#(\mathcal{T}^*) - \#(\widetilde{\mathcal{T}}) \le K_0 r^{1-d/2} \|f\|^d_{L^\infty(\gamma)} \tau^{-d},$$

where the constant K_0 is independent of r and τ. This implies that Assumption 4.5 holds for F^r with $s = \frac{1}{2}$ and $G_{1/2} \sim r^{1-d/2} \|f\|^d_{L^\infty(\gamma)}$ when $\mathcal{T} = \widetilde{\mathcal{T}}$.

Proof. Here we sketch the proof and refer to the appendix of [26] for a complete version. Suppose that there are totally N iterations executed in the while loop when $\texttt{GREEDY}(\widetilde{\mathcal{T}}, F^r, \tau)$ terminates. We denote with $\{T^i\}_{i=0}^N$ the marked cells in the sequence and set $\mathcal{T}^i = \texttt{REFINE}(\mathcal{T}^{i-1}, \{T^{i-1}\})$ for $i = 1, \dots, N$ and $\mathcal{T}^0 = \widetilde{\mathcal{T}}$. Let

$$\delta := d(F^r, T^{N-1}, \mathcal{T}^{N-1}) = \operatorname{argmax}\{d(F^r, T, \mathcal{T}^{N-1}) : T \in \mathcal{T}^{N-1}\}.$$

Clearly, by the above setting there hold for $0 \le i \le N-1$

$$d(F^r, T^i, \mathcal{T}^i) \ge \delta \quad \text{and} \quad \tau \le \mathcal{D}(F^r, \mathcal{T}^{N-1}) \le \delta\sqrt{\#(T^{N-1})}. \tag{4.7}$$

Since F^r is supported in U_r, $T^i \subset U_{cr}$ for some constant $c \ge 1$ depending on $c_{\texttt{sr}}$. Let $\mathcal{B}_j \subset \{T^i\}$ be the set satisfying

$$2^{-(j+1)}|U_{cr}| < |T^i| \le 2^{-j}|U_{cr}|, \quad j \ge 0. \tag{4.8}$$

Since $\{T^i\}$ are distinct from each other, the left inequality above implies that $\#(\mathcal{B}_j) < 2^{j+1}$, while the right inequality as well as (4.7) imply that

$$\delta \le d(F^r, T^i, \mathcal{T}^i) = |T^i|^{1/d} \|F^r\|_{L^2(T^i)} \le 2^{-j/d} |U_{cr}|^{1/d} \|F^r\|_{L^2(T^i)}.$$

By summing up for all $T^i \in \mathcal{B}_j$, there holds

$$\delta^2 \#(\mathcal{B}_j) \le 2^{-2j/d} |U_{cr}|^{2/d} \|F^r\|^2_{L^2(U_{cr})},$$

whence

$$N = \sum_{j \ge 0} \#(\mathcal{B}_j) \le \sum_{j < j_0} 2^{j+1} + \delta^{-2} |U_{cr}|^{2/d} \|F^r\|^2_{L^2(U_{cr})} \sum_{j \ge j_0} 2^{-2j/d}, \tag{4.9}$$

where j_0 is the smallest integer such that $2^{j_0+1} > 2^{-2j_0/d} \delta^{-2} |U_{cr}|^{2/d} \|F^r\|^2_{L^2(U_{cr})}$. Using the definition of j_0 together with $|U_{cr}| \sim r$ and $\|F^r\|_{L^2(U_{cr})} \lesssim r^{1/2} \|f\|_{L^\infty(\gamma)}$, if

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

$j_0 > 0$, we can estimate 2^{j_0} (with respect to δ, r, and F^r) to bound the right-hand side of (4.9) with

$$N \lesssim \delta^{-2d/(2+d)} r^{(2-d)/(2+d)} \|f\|_{L^\infty(\gamma)}^{2d/(2+d)}. \tag{4.10}$$

A similar estimate can be obtained when $j_0 = 0$. Now we invoke (4.7), (3.4), and (4.10) to conclude that

$$\tau \lesssim \frac{\sqrt{\#(\mathcal{T}^0) + C_{\texttt{com}} N}}{N^{(2+d)/(2d)}} r^{1/d-1/2} \|f\|_{L^\infty(\gamma)} \lesssim N^{-1/d} r^{1/d-1/2} \|f\|_{L^\infty(\gamma)},$$

and the proof is complete. □

Remark 4.8. Following the proof of [12, Proposition 2], we can extend the results in Lemma 4.7 by replacing $\widetilde{\mathcal{T}}$ with any of its refinements. More precisely speaking, let $\mathcal{T}^+$ be any refinement of $\widetilde{\mathcal{T}}$, and $\mathcal{T}^* = \texttt{GREEDY}(F^r, \mathcal{T}^+, \tau)$. Then,

$$\#(\mathcal{T}^*) - \#(\mathcal{T}^+) \lesssim r^{1-d/2} \|f\|_{L^\infty(\gamma)}^d \tau^{-d}.$$

Remark 4.9. An estimate similar to the one in Lemma 4.7 could also be obtained when the local data indicator in GREEDY is replaced by the surrogate $L^p(\Omega)$ data indicator defined by (7.1) of [17]. Here $p = \frac{2d}{d+2}$ so that $L^p(\Omega)$ is on the same nonlinear Sobolev scale of $H^{-1}(\Omega)$. Note that $\|F^r\|_{L^p(\Omega)} \lesssim \|f\|_{L^\infty(\gamma)} r^{1/p-1} = \|f\|_{L^\infty(\gamma)} r^{1/d-1/2}$. Applying [17, Lemma 7.3] directly we get

$$\#(\mathcal{T}^*) - \#(\widetilde{\mathcal{T}}) \lesssim \|F^r\|_{L^p(\Omega)}^d \tau^{-d} \lesssim r^{1-d/2} \|f\|_{L^\infty(\gamma)}^d \tau^{-d}.$$

Remark 4.10. We note that by treating F^r as an $L^2(\Omega)$ data, Lemma 4.7 also reveals the dependency of r for for the decay of the oscillation $\operatorname{osc}(F^r, \mathcal{T})$.

Now we are in a position to verify Assumption 4.5 when $g = F^r$. The proof follows [17, Theorem 7.5] using a contraction property of $\mathcal{D}(F^r, \mathcal{T})$, a bulk property, and Lemma 4.7. Here we again omit the proof.

COROLLARY 4.11 (performance of DATA). *Under the settings in Lemma* 4.7, *Assumption* 4.5 *holds with* $s = \frac{1}{d}$ *and* $g = F^r$ *starting from* $\widetilde{\mathcal{T}} = \texttt{INTERFACE}(\mathcal{T}_0, r)$. *Precisely speaking, given a refinement* $\mathcal{T}$ *of* $\widetilde{\mathcal{T}}$, *let* $\mathcal{T}^*$ *be the output of* $\texttt{DATA}(\mathcal{T}, F^r, \tau, \widetilde{\theta})$ *with a fixed* $\widetilde{\theta} \in (0,1)$. *Then, there exists a constant* $K_0 > 0$ *not depending on* r *or* τ *(but depending on* $\widetilde{\theta}$*) satisfying*

$$\#(\mathcal{T}^*) - \#(\mathcal{T}) \le K_0 r^{1-d/2} \tau^{-d}.$$

4.3.2. Quasi-monotoniciy of the data indicator.

The following lemma provides a quasi-monotonicity of $\mathcal{D}(F^r, \mathcal{T})$ with respect to r. We note that this property relies on some additional hypothesis on the forcing data f and on the nonnegativity of δ^r.

LEMMA 4.12. *Given* $r_2 < r_1$, *let* $\mathcal{T}$ *be a refinement of* $\texttt{INTERFACE}(\mathcal{T}_0, r_2)$. *Then there holds that*

$$\mathcal{D}(F^{r_2}, \mathcal{T}) \lesssim \widetilde{\beta}^d \mathcal{D}(F^{r_1}, \mathcal{T}) + r_2,$$

where $\widetilde{\beta} = \frac{r_2}{r_1} < 1$.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Proof. We investigate the local data indicator for F^{r_2} when (i) T is away from the tubular neighborhood of γ with radius r_2, (ii) T intersects the tubular neighborhood and f changes sign in T, and (iii) T intersects with the tubular neighborhood and f is nonnegative/nonpositive. Clearly, $d(F^{r_2},T,\mathcal{T})=0$ when $\operatorname{dist}(T,\gamma)>r_2$. We shall focus on the other cases.

We recall from the configuration of f in section 2 that the set I defined in (2.2) separates the sign of f in γ. Define

$$\mathcal{B}:=\{T\in\mathcal{T}:T\cap B_{r_2}(x_0)\neq\emptyset \text{ for some } x_0\in I\}.$$

Since $h_T\lesssim r_2$ for $T\in\mathcal{B}$, there holds

$$\sum_{T\in\mathcal{B}}|T|\lesssim r_2^2. \tag{4.11}$$

Here the hidden constant above depends on the measure of I in co-dimension 2. Now we bound $d(F^{r_2},T,\mathcal{T})$. If $T\notin\mathcal{B}$, since δ^r is nonnegative, and thanks to Assumption 2.1, item 4, we have $\delta^{r_2}\le\widetilde{\beta}^d\delta^{r_1}$. Hence,

$$d(F^{r_2},T,\mathcal{T})\le\widetilde{\beta}^d d(F^{r_1},T,\mathcal{T}).$$

If $T\in\mathcal{B}$, there holds

$$d(F^{r_2},T,\mathcal{T})^2\lesssim\frac{h_T^2}{r_2^{2d}}\int_T|B_{r_2}(x)\cap\gamma|^2\,\mathrm{d}x\lesssim\frac{h_T^2}{r_2^{2d}}r_2^{2(d-1)}|T|\lesssim|T|.$$

By summing up all contributions above and invoking (4.11), we arrive at

$$\begin{aligned}\mathcal{D}(F^{r_2},\mathcal{T})^2&=\sum_{T\in\mathcal{B}}d(F^{r_2},T,\mathcal{T})^2+\sum_{T\notin\mathcal{B}}d(F^{r_2},T,\mathcal{T})^2\\&\lesssim\sum_{T\in\mathcal{B}}|T|+\sum_{T\notin\mathcal{B}}\widetilde{\beta}^{2d}d(F^{r_1},T,\mathcal{T})^2\lesssim r_2^2+\widetilde{\beta}^{2d}\mathcal{D}(F^{r_1},\mathcal{T}),\end{aligned}$$

which concludes the proof. □

Remark 4.13. If f is nonnegative or nonpositive along γ, according to the proof of Lemma 4.12, we immediately get $\mathcal{D}(F^{r_2},\mathcal{T})\le\widetilde{\beta}^d\mathcal{D}(F^{r_1},\mathcal{T})$.

4.3.3. Performace of each subroutine in REGSOLVE.

In terms of the approximation class for $\mathtt{u}^r$, Lemma 3.2 of [12] enlightens us to exploit the fact that $\mathtt{u}^r$ is an approximation of u and then to characterize approximation properties of $\mathtt{u}^r$ with the approximation class of u, i.e., using the quasi-seminorm $|u|_{\mathcal{A}^s}$ for some $s\in(0,\frac{1}{d})$.

LEMMA 4.14 (Lemma 3.2 of [12]). *If $|||u-\mathtt{u}^r|||<\varepsilon$ for some $\varepsilon>0$, then $\mathtt{u}^r$ is a 2ε-approximation to u of order s: for all $\delta>2\varepsilon$, there exists a positive integer n such that*

$$\sigma_n(\mathtt{u}^r)_{H_0^1(\Omega)}\le\delta\quad\textit{and}\quad n\lesssim|u|_{\mathcal{A}^s}^{1/s}\delta^{-1/s}.$$

LEMMA 4.15 (a priori asymptotic decay of the total error; see Lemma 5.1 of [17]). *Under the settings in Lemma 4.7, we set $r=r(\tau)$ according to* (3.10) *so that $|||u-\mathtt{u}^r|||\le C_{\boldsymbol{rel}}\tau/2$ for some $\tau>0$. Then for any $1>\delta\ge\sqrt{2}C_{\boldsymbol{rel}}\tau$, there is a refinement $\mathcal{T}$ of $\widetilde{\mathcal{T}}=\mathtt{INTERFACE}(\mathcal{T}_0,r)$ such that*

$$E(\mathtt{u}^r,\mathcal{T})\le\delta\quad\textit{and}\quad\#(\mathcal{T})-\#(\widetilde{\mathcal{T}})\lesssim(K_0r^{1-d/2}+|u|_{\mathcal{A}^s}^{1/s})\delta^{-1/s}.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Proof. A desired refinement $\mathcal{T}$ of $\widetilde{\mathcal{T}}$ is the overlay of $\mathcal{T}_f^r = \mathtt{DATA}(\widetilde{\mathcal{T}}, F^r, \delta/\sqrt{2})$ and $\mathcal{T}_{\mathtt{u}^r}$ from Lemma 4.14 by replacing δ with $\frac{\delta}{\sqrt{2}}$. □

The next lemma provides the estimate of marked cells in SOLVE. The proof follows from [17, Lemma 5.3], together with Lemma 4.15, as well as the minimal assumption of MARK.

LEMMA 4.16 (cardinality of REGSOLVE::SOLVE::MARK). *Under the settings given by Lemma 4.7, let the bulk parameter θ defined in* SOLVE *satisfy the condition $\theta < \theta_*$, with θ_* provided by* (4.3)*. For a fixed $\tau > 0$, set $r = r(\tau)$ in* (3.10) *and $\widetilde{\mathcal{T}} =$* $\mathtt{INTERFACE}(\mathcal{T}_0, r)$*. We also let $\{\mathcal{T}_k\}$ be defined in* $\mathtt{SOLVE}(\widetilde{\mathcal{T}}, F^r, \widetilde{\mu}\tau)$ *with $\widetilde{\mu} \geq \sqrt{2}C_{\boldsymbol{rel}}/(\xi C_{\boldsymbol{eff}})$ and $\{\mathcal{M}_k\}$ be the set of marked cells generated from* SOLVE::MARK *at $\mathcal{T}_k$. Then there holds*

$$\#(\mathcal{M}_k) \lesssim (K_0 r^{1-d/2} + U_s) E(\mathtt{u}^r, \mathcal{T}_k)^{-1/s},$$

where $U_s := |u|_{\mathcal{A}^s}^{1/s}$.

LEMMA 4.17 (performance of REGSOLVE::SOLVE; cf. Theorem 4.1 of [12]). *Denote $\{(\mathcal{T}_j, U_j)\}_{j=0}^{j_{\max}}$ to be the sequence of subdivisions and approximations of u generated by* REGSOLVE*, respectively. Set $\widetilde{\mathcal{T}}_j = \mathtt{INTERFACE}(\mathcal{T}_j, r_j)$ with $r_j = r(\tau_j)$. Under the assumptions provided by Lemmas* 4.7 *and* 4.16*, there holds that for $j \geq 1$,*

$$\#(\mathcal{T}_{j+1}) - \#(\widetilde{\mathcal{T}}_j) \lesssim (K_0 r^{1-d/2} + U_s)\tau_j^{-1/s}.$$

Proof. For each $j \geq 1$, we let $k_{\max}$ be the number of iterations executed in SOLVE. Let us first show that $k_{\max}$ is uniform bound with respect to j. Let $\widehat{\tau}_j$ be the error indicator for $\widetilde{U}_j = \mathtt{GAL}(\widetilde{\mathcal{T}}_j, F^{r_j})$ with $r_j = r(\tau_j)$ in REGSOLVE. In view of (3.6) and Lemma 4.12, we have

$$\begin{aligned}\widehat{\tau}_j \lesssim E(u^{r_j}, \widetilde{\mathcal{T}}_j) &\lesssim \left|\!\left|\!\left| u^{r_j} - \widetilde{U}_j \right|\!\right|\!\right| + \mathcal{D}(F^{r_j}, \widetilde{\mathcal{T}}_j) \lesssim |||u^{r_j} - U_j||| + \mathcal{D}(F^{r_{j-1}}, \widetilde{\mathcal{T}}_j) + r_j \\ &\lesssim |||u - u^{r_j}||| + |||u - U_j||| + \mathcal{E}(u^{r_{j-1}}, \mathcal{T}_j) + r_j.\end{aligned}$$

Now we invoke Propositions 3.5 and 2.6 to deduce

$$\widehat{\tau}_j \lesssim r_j^{1/2} + \tau_{j-1} + r_j \lesssim \tau_j + \tau_{j-1} \lesssim \tau_j. \tag{4.12}$$

In the above estimates we also used the relations $r_j \lesssim \tau_{j+1}^2$ and $\tau_j = \beta\tau_{j-1}$. The contraction property (4.4) together with (4.12) yields the uniform boundedness of $k_{\max}$.

At each iteration $k = 0, 1, \ldots, k_{\max}$ in SOLVE, Lemma 4.16 controls the number of marked cells in REFINE. For the cardinality of the marked cells in DATA, we set $\mathcal{T}_k^+$ to be the corresponding output and apply Corollary 4.11 to get

$$\#(\mathcal{T}_k^+) - \#(\mathcal{T}_k) \lesssim K_0 r^{1-d/2} (\lambda\theta\mathcal{E}_k)^{-1/s} \lesssim K_0 r^{1-d/2} E(\mathtt{u}^r, \mathcal{T}_k)^{-1/s}.$$

Combining the above estimate together with Lemma 4.16, we obtain that

$$\begin{aligned}\#(\mathcal{T}_{j+1}) - \#(\widetilde{\mathcal{T}}_j) &\lesssim \sum_{k=0}^{k_{\max}} \big(\#(\mathcal{M}_k) + \#(\mathcal{T}_k^+) - \#(\mathcal{T}_k)\big) \\ &\lesssim (K_0 r^{1-d/2} + U_s) E(\mathtt{u}^r, \mathcal{T}_{k_{\max}})^{-1/s} \sum_{k=0}^{k_{\max}} \alpha^{(k_{\max}-k)/s} \\ &\lesssim (K_0 r^{1-d/2} + U_s)\tau^{-1/s}, \qquad (4.13)\end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

where for the last two inequalities above we applied Theorem 4.4, $\tau \lesssim E(\mathtt{u}^r, \mathcal{T}_{k_{\max}})$, and $\sum_{k=0}^{k_{\max}} \alpha^{(k_{\max}-k)/s} \le \sum_{k=0}^{\infty} \alpha^{k/s} \lesssim 1$. The proof is complete. □

4.3.4. Performace of REGSOLVE. We are now in a position to show our main result.

THEOREM 4.18 (performance of REGSOLVE). *Denote* $\{(\mathcal{T}_j, U_j)\}_{j=0}^{j_{\max}}$ *to be the sequence of subdivisions and approximations of* u *generated by* REGSOLVE*, respectively. Under the assumptions provided by Lemmas* 4.7 *and* 4.16*, there holds that*

$$\#(\mathcal{T}_{j_{\max}}) - \#(\mathcal{T}_0) \lesssim (K_0 + I_0 + U_s)\tau_{j_{\max}}^{2-d-1/s}.$$

Proof. Denote $\mathcal{M}_j$ the collections of cells marked for refinement in the jth iteration of solve. Invoking Corollary 4.3 and Lemma 4.17, we have

$$\begin{aligned}\#(\mathcal{M}_j) &= (\#(\widetilde{\mathcal{T}}_j) - \#(\mathcal{T}_j)) + (\#(\mathcal{T}_{j+1}) - \#(\widetilde{\mathcal{T}}_j))\\ &\lesssim (I_0\tau_j^{2-d} + K_0\tau_j^{2-d} + U_s)\tau_j^{-1/s} \lesssim (I_0 + K_0 + U_s)\tau_j^{2-d-1/s},\end{aligned}$$

where we used the setting $r \sim \tau_j^2$ according to (3.10). Summing up the above estimate for $j = 0, \ldots, j_{\max} - 1$ together with the relation $\tau_{j_{\max}} = \beta^{j_{\max}-j}\tau_j$ implies the target estimate. □

Remark 4.19 (convergence rates). As mentioned in section 4.2, the best possible rate is $s = \frac{1}{2(d-1)}$. So Theorem 4.18 implies that

$$|||u - U_j||| \lesssim \tau_j \lesssim (\#(\mathcal{T}_j) - \#(\mathcal{T}_0))^{-1/(3d-4)}.$$

Hence, in two-dimensional space, we guarantee that the adaptive method is quasi-optimal. However, in three-dimensional space, we have

$$|||u - U_j||| \lesssim (\#(\mathcal{T}_j) - \#(\mathcal{T}_0))^{-1/5},$$

which turns out to be suboptimal compared with the optimal rate $\frac{1}{4}$.

5. Numerical illustration.

In this section, we test our numerical algorithm proposed in section 3 for the following interface problem: letting γ be defined as in (2.1), we want to find u satisfying

$$\begin{aligned} -\Delta u &= 0 && \text{in } \Omega\backslash\gamma,\\ [u] &= 0 && \text{on } \gamma,\\ [\nabla u \cdot \nu_\gamma] &= f && \text{on } \gamma,\\ u &= g && \text{on } \partial\Omega, \end{aligned} \tag{5.1}$$

where [.] denotes the jump of the function across the interface γ and ν_γ is the outward normal direction along γ. So u satisfies the weak formulation (2.5) with the forcing data F defined by (2.3) and a nonhomogeneous boundary condition.

As we mentioned in section 3.4, our numerical implementation relies on the deal.II finite element library [3, 4] and we use quadrilateral subdivisions in two dimensions and hexahedral subdivisions in three dimensions. For the computation of the right-hand side of the discrete system, we refer to Remark 22 of [25] for more details. In the following numerical simulations, we use a radially symmetric C^1 approximation of the Dirac delta approximation, i.e., $\psi(x) = c_d(1+\cos(|\pi x|))\chi(x)$, where

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

$\chi(x)$ is the characteristic function on the unit ball and c_d is a normalization constant so that $\int_{\mathbb{R}^d} \psi = 1$.

In REGSOLVE, we fix $\widetilde{\mu} = \frac{1}{2}$. The parameters $\mathcal{T}_0$ (initial subdivision), τ_0 (initial tolerance), β (tolerance reduction), $j_{\max}$ (number of iterations), the bulk parameters θ and $\widetilde{\theta}$, and λ (ratio between $\mathcal{E}$ and $\theta\mathcal{D}$) will be provided for each numerical test. For the regularization parameter, we simply set $r(\tau_j) = \tau_j^2$ in REGSOLVE::INTERFACE to avoid the estimate of the constants $C_{\mathtt{reg}}$, $C_{\mathtt{rel}}$ and $\|f\|_{L^2(\gamma)}$ in (3.10). Furthermore, after the last iteration of REGSOLVE, we perform the following extra steps:

$r_{j+1} = r(\tau_{j_{\max}+1})$;
$\widetilde{\mathcal{T}}_{j_{\max}+1} = \mathtt{INTERFACE}(\mathcal{T}_{j_{\max}+1}, r_{j+1})$;
$\mathtt{GAL}(\widetilde{\mathcal{T}}_{j_{\max}+1}, F^{r_{j+1}})$.

5.1. Convergence tests on an L-shaped domain. Following test cases similar to those presented in [27], we set $\Omega = (-1,1)^2 \backslash [0,1]^2$, $\gamma = \partial B_R(c)$ with $R = 0.2$ and $c = (0.5, -0.5)^{\mathtt{T}}$, $f = \frac{1}{R}$, and $g = \ln(|x - c|)$. The analytic solution is given by

$$u(x) = r(x)^{2/3} \sin(\tfrac{2}{3}(\theta(x) - \tfrac{\pi}{2})) + \begin{cases} -\ln(|x-c|) & \text{if } |x-c| > R, \\ -\ln(R) & \text{if } |x-c| \leq R, \end{cases}$$

with (r, θ) denoting the polar coordinates. We start with an initial uniform grid $\mathcal{T}_0$ with the mesh size $\sqrt{2}/4$. Note that we also approximate the interface γ with a uniform subdivision whose vertices lie on γ. The corresponding mesh size is fixed as $2\pi R/2^{14}$ so that the geometric error will not dominate the total error. For the parameters showing the numerical algorithm, we set $j_{\max} = 6$, $\tau_0 = 0.6$, $\beta = 0.8$, $\lambda = \frac{1}{3}$, and $\theta = \widetilde{\theta} = 0.7$ in SOLVE and DATA, respectively. The left plot in Figure 1 reports the $H^1(\Omega)$-error versus the number of degrees of freedom (#DoFs) when GAL is executed. We note that the error goes down almost vertically when we update the regularization radius after INTERFACE. In order to verify Theorem 4.18 (or Remark 4.19), we extract the sampling points only for U_j (i.e., the last Galerkin approximation in each iteration of REGSOLVE) in red. Based on the observation we confirm the first order rate of convergence. We also present our approximated solution U_3 and its underlying subdivision in Figure 2 using the tensor product extension of the C^1 function.

We test the algorithm in Remark 3.6 (i.e., we make one single iteration, and set the initial target tolerance to $\tau_0\beta^{j_{\max}}$), and report the energy error for the final approximation against #DoFs in the right plot of Figure 1. Here we use the same parameters

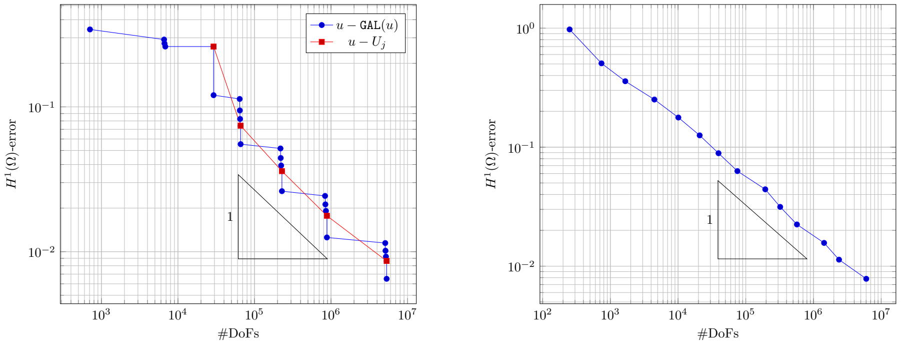

FIG. 1. *Test on an L-shaped domain: (left) $H^1(\Omega)$-error decay between the solution u and every Galerkin approximation ($\mathtt{GAL}(u)$) in* REGSOLVE *and between u and U_j defined in* REGSOLVE*; (right) $H^1(\Omega)$-error decay between u and $U_{j_{\max}}$ defined from Remark* 3.6*. We set $j_{\max} = 6$, and $\tau_0 = .6$.*

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

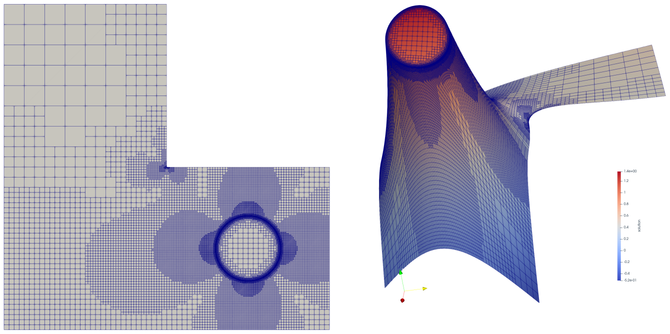

FIG. 2. *Test on an L-shaped domain: (left) the subdivisions of* U_3 *in* `REGSOLVE` *and (right) the corresponding Galerkin approximation using* tensor product C^1*.*

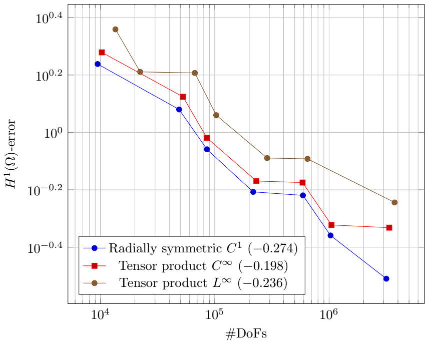

FIG. 3. *Tests in the unit cube:* $H^1(\Omega)$*-error decay between the solution* u *and* U_j *for* $j = 1,\dots,7$ *and for different choices of* $\delta^r(x)$*. In terms of each plot, the slope of the linear regression of the last five sampling points is reported in the legend.*

except that $j_{\max} = 14$, in order to reach a similar true error. Comparing with the left plot of Figure 1, we note that although both algorithms guarantee the quasi-optimal convergence rate, the energy error $|||U_{j_{\max}} - u|||$ using the algorithm in Remark 3.6 is much larger than that computed from `REGSOLVE` with multiple iterations.

5.2. Convergence tests in the unit cube. We test our numerical algorithm in three dimensions by setting $\Omega = (-1, 1)^3$ and $\gamma = \partial B_R(c)$ with $R = 0.2$ and $c = (0.3, 0.3, 0.3)^{\intercal}$. We also set the data function $f = \frac{1}{R^2}$ on γ and $g = 1/|x - c|$ so that the analytic solution is given by

$$u(x) = \begin{cases} 1/|x-c| & \text{if } |x-c| > R, \\ 1/R & \text{if } |x-c| \le R. \end{cases}$$

We start with an initial uniform grid $\mathcal{T}_0$ with the mesh size $\sqrt{3}/16$. To approximate the interface γ, we start with initial quasi-uniform coarse mesh and refine it globally 7 times so that the geometric error is small enough. For the other approximation parameters, we set $j_{\max} = 5$, $\tau_0 = 1.5$, $\beta = 0.8$, $\lambda = 1$, $\theta = 0.5$ in `SOLVE` and $\widetilde{\theta} = 0.8$ in `DATA`. In Figure 3, we report the $H^1(\Omega)$-error against #DoFs for the following three different types of δ^r: the *radially symmetric* C^1 type, the *tensor product* C^∞ type generated by $\psi_{1d}(x) = \exp(1 - 1/(1 - x^2))\chi_{(-1,1)}(x)$, and the *tensor product* L^∞ type generated by $\psi_{1d}(x) = \frac{1}{2}\chi_{(-1,1)}(x)$. For each error plot, we also report the slope of the linear regression of the last five sampling points. For the choice of *tensor product* C^∞, the performance is suboptimal and close to the predicted rate $\frac{1}{5}$. When using

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

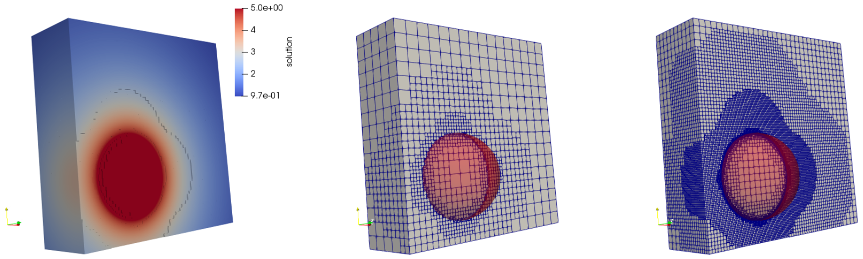

FIG. 4. *Tests in the unit cube: the crinkle clip ($x_1 \leq 0.3$) of the approximation U_7 (3741904 DoFs) (left) as well as the subdivisions for U_5 (mid) and U_7 (right) using radially symmetric C^1. The interface γ is marked in red.*

radially symmetric C^1, the observed convergence rate is better than the best possible rate $\frac{1}{4}$. As for *tensor product* L^∞, the performance is between $\frac{1}{4}$ and $\frac{1}{5}$. We also report the coarse grid and the grids for U_5, U_7 as well as the approximation U_7 using *radially symmetric* C^1 in Figure 4.

5.3. Performance tests in the unit square. Consider $\Omega = (0,1)^2$, $\gamma = \partial B_R(c)$ with $R = 0.2$ and $c = (0.3, 0.3)^\intercal$, $f = \frac{1}{R}$, and $g = \ln(|x-c|)$. Similar to the previous section, we can obtain the following exact solution:

$$u(x) = \begin{cases} -\ln(|x-c|) & \text{if } |x-c| > R, \\ -\ln(R) & \text{if } |x-c| \leq R. \end{cases}$$

We shall compare the performance of our numerical algorithms both in Algorithm 3.8 and Remark 3.6 with the algorithm without regularization; see the numerical algorithm from section 7.2 of [17]. To be more precise, the algorithm without using the regularization is based on SOLVE by replacing the data indicator $\mathcal{D}$ with the following surrogate data indicator:

$$\widetilde{\mathcal{D}}(f, T, \mathcal{T}) := h_T^{1/2} \|f\|_{L^2(T \cap \gamma)}.$$

Using the exact solution u, after the jth iterate of REGSOLVE in Algorithm 3.8 using *tensor product* L^∞, we compute the $H^1(\Omega)$-error between u and U_j, denoted by e_j. For the parameters we set $\tau_0 = 0.3$, $\beta = 0.7$, $\lambda = \frac{1}{3}$, $\theta = \widetilde{\theta} = 0.55$. Then we run the nonregularized program with the same parameters and terminate it when the energy error is smaller than e_j, denoting $\widetilde{e}_j$ the energy error for the corresponding output approximation, for $j = 8, 9, \ldots, 12$. We also compute e_j using the algorithm provided by Remark 3.6 with $j = 11, \ldots, 15$. Now we report those errors and the CPU times for each program against #DoFs in Figure 6. We observe that all algorithms are quasi-optimal but the algorithm from Remark 3.6 requires more DoFs.

In terms of the computation time, it turns out that Algorithm 3.8 needs more time when the discrete system is small (less than 10^7) and becomes more efficient when the size of the system is increasing. Since the computational cost associated to the regularized version is comparable to the one required by the nonregularized version *when computed on the same grid* (see [25, Figure 7]), a fair comparison between the different AFEM algorithms should take into account the computational cost in terms of the attained accuracy.

The regularized case reaches lower errors, for the same number of degrees of freedom, but it is more expensive (due to a larger number of refined elements around

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

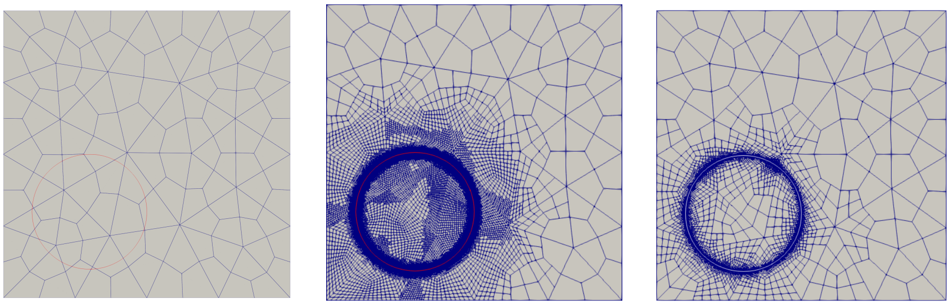

FIG. 5. *Tests on a square: (left) the unstructured coarse mesh $\mathcal{T}_0$, (center) the subdivision for U_5, and (right) the corresponding subdivision using the nonregularized algorithm.*

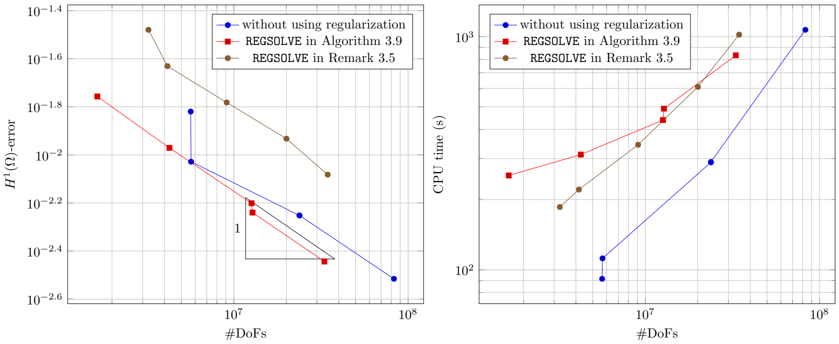

FIG. 6. *Tests on a square: (left)* $\|U_j - u\|_{H^1(\Omega)}$ *using* `REGSOLVE` *with* tensor product L^∞ *for* $j = 8, \ldots$ *and the corresponding* $H^1(\Omega)$*-error decay without using the regularization; (right) computational time against #nDoFs for two adaptive algorithms. We note that the sampling points for the nonregularized algorithm at* $j = 10$ *and* 11 *are so close that they overlap with each other.*

the interface γ required by our algorithms). The computational cost is compensated for by the higher accuracy in the largest scale computations, where the computational cost per degree of freedom is comparable, making the regularized approach roughly comparable to the nonregularized one also in the AFEM context. In Figure 5, we finally report the grid for U_5 using Algorithm 3.8 and corresponding grid for the nonregularized algorithm.

6. Conclusion and outlook. We have proposed an adaptive finite element algorithm to approximate the solutions of elliptic problems with rough data approximated by regularization. Such problems are relevant in many applications ranging from fluid-structure interaction to the modeling of biomedical applications with complex embedded domains or networks.

Our approach builds on classical results for adaptive finite element theory for H^{-1} data, and for L^2 data. In particular, we analyze the regularization of line Dirac delta distributions via convolutions with compactly supported approximated Dirac delta functions, with radius r. What characterizes the regularization process is that even if the resulting forcing term is as regular as desired—at fixed r—its regularity does not hold uniformly with respect to r.

This observation suggests that one could exploit the knowledge of the asymptotic behavior of the data regularity with respect to r to construct an algorithm that a

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

priori refines around the rough part of the forcing term, in a way that guarantees quasi optimal convergence, at least in the two-dimensional case.

The resulting approximation error is split into a regularization error for u and the finite element approximation error for the regularized u^r. In this work we show how to control the dependencies between these two errors and provide an algorithm in which the error decay in the energy norm is quasi-optimal in two-dimensional space and suboptimal in three-dimensional space.

Our findings are specific for the co-dimension one case but could be easily extended to the co-dimension zero case, where the dependency of the regularity on r disappears naturally, due to the intrinsic L^2 nature of the resulting forcing term.

Acknowledgment. The author would like to thank the anonymous reviewers who provided valuable comments and remarks on the earlier version of the manuscript.

REFERENCES

[1] A. Allendes, E. Otárola, and A. J. Salgado, *A posteriori error estimates for the Stokes problem with singular sources*, Comput. Methods Appl. Mech. Engrg., 345 (2019), pp. 1007–1032.

[2] G. Alzetta and L. Heltai, *Multiscale modeling of fiber reinforced materials via non-matching immersed methods*, Comput. Structures, 239 (2020), 106334.

[3] D. Arndt, W. Bangerth, B. Blais, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, U. Köcher, M. Kronbichler, M. Maier, et al., *The deal.*II *library, version* 9.3, J. Numer. Math., 29 (2021), pp. 171–186.

[4] D. Arndt, W. Bangerth, D. Davydov, T. Heister, L. Heltai, M. Kronbichler, M. Maier, J.-P. Pelteret, B. Turcksin, and D. Wells, *The deal.*II *finite element library: Design, features, and insights*, Comput. Math. Appl., 81 (2021), pp. 407–422.

[5] R. E. Bank and A. Weiser, *Some a posteriori error estimators for elliptic partial differential equations*, Math. Comp., 44 (1985), pp. 283–301.

[6] S. Berrone, A. Bonito, R. Stevenson, and M. Verani, *An optimal adaptive fictitious domain method*, Math. Comp., 88 (2019), pp. 2101–2134.

[7] S. Bertoluzza, A. Decoene, L. Lacouture, and S. Martin, *Local error estimates of the finite element method for an elliptic problem with a Dirac source term*, Numer. Methods Partial Differential Equations, 34 (2017), pp. 97–120.

[8] P. Binev, W. Dahmen, and R. DeVore, *Adaptive finite element methods with convergence rates*, Numer. Math., 97 (2004), pp. 219–268.

[9] D. Boffi, F. Credali, and L. Gastaldi, *On the interface matrix for fluid-structure interaction problems with fictitious domain approach*, Comput. Methods Appl. Mech. Engrg., 401, (2022), 115650.

[10] D. Boffi and L. Gastaldi, *A finite element approach for the immersed boundary method*, Comput. Structures, 81 (2003), pp. 491–501.

[11] D. Boffi and L. Gastaldi, *On the existence and the uniqueness of the solution to a fluid-structure interaction problem*, J. Differential Equations, 279 (2021), pp. 136–161.

[12] A. Bonito, R. A. DeVore, and R. H. Nochetto, *Adaptive finite element methods for elliptic problems with discontinuous coefficients*, SIAM J. Numer. Anal., 51 (2013), pp. 3106–3134.

[13] A. Bonito and R. H. Nochetto, *Quasi-optimal convergence rate of an adaptive discontinuous Galerkin method*, SIAM J. Numer. Anal., 48 (2010), pp. 734–771.

[14] J. M. Cascon, C. Kreuzer, R. H. Nochetto, and K. G. Siebert, *Quasi-optimal convergence rate for an adaptive finite element method*, SIAM J. Numer. Anal., 46 (2008), pp. 2524–2550.

[15] D. Cerroni, F. Laurino, and P. Zunino, *Mathematical analysis, finite element approximation and numerical solvers for the interaction of* 3*D reservoirs with* 1*D wells*, GEM Int. J. Geomath., 10 (2019).

[16] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, Classics in Appl. Math. 40, SIAM, Philadelphia, PA, 2002.

[17] A. Cohen, R. DeVore, and R. H. Nochetto, *Convergence rates of AFEM with H^{-1} data*, Found. Comput. Math., 12 (2012), pp. 671–718.

[18] W. Dörfler, *A convergent adaptive algorithm for Poisson's equation*, SIAM J. Numer. Anal., 33 (1996), pp. 1106–1124.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 03/15/23 to 109.171.145.23 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

[19] W. Dörfler and R. H. Nochetto, *Small data oscillation implies the saturation assumption*, Numer. Math., 91 (2002), pp. 1–12.

[20] A. Ern and J.-L. Guermond, *Theory and Practice of Finite Elements*, Appl. Math. Sci. 159, Springer-Verlag, New York, 2004.

[21] P. Grisvard, *Elliptic Problems in Nonsmooth Domains*, Monographs and Studies in Mathematics 24, Pitman, Boston, MA, 1985.

[22] L. Heltai and A. Caiazzo, *Multiscale modeling of vascularized tissues via nonmatching immersed methods*, Int. J. Numer. Methods Biomed. Eng., 35 (2019), e3264.

[23] L. Heltai, A. Caiazzo, and L. O. Müller, *Multiscale coupling of one-dimensional vascular models and elastic tissues*, Ann. Biomed. Eng., 49 (2021), pp. 3243–3254.

[24] L. Heltai and F. Costanzo, *Variational implementation of immersed finite element methods*, Comput. Methods Appl. Mech. Engrg., 229 (2012), pp. 110–127.

[25] L. Heltai and W. Lei, *A priori error estimates of regularized elliptic problems*, Numer. Math., 146 (2020), pp. 571–596.

[26] L. Heltai and W. Lei, *Adaptive Finite Element Approximations for Elliptic Problems Using Regularized Forcing Data*, preprint, https://arxiv.org/abs/2110.15029, 2021.

[27] L. Heltai and N. Rotundo, *Error estimates in weighted sobolev norms for finite element immersed interface methods*, Comput. Math. Appl., 78 (2019), pp. 3586–3604.

[28] B. Hosseini, N. Nigam, and J. M. Stockie, *On regularizations of the Dirac delta distribution*, J. Comput. Phys., 305 (2016), pp. 423–447.

[29] P. Houston and T. P. Wihler, *Discontinuous galerkin methods for problems with Dirac delta source*, ESAIM Math. Model. Numer. Anal., 46 (2012), pp. 1467–1483.

[30] R. Krause and P. Zulian, *A parallel approach to the variational transfer of discrete fields between arbitrarily distributed unstructured finite element meshes*, SIAM J. Sci. Comput., 38 (2016), pp. C307–C333.

[31] C. Kreuzer and A. Veeser, *Oscillation in a posteriori error estimation*, Numer. Math., 148 (2021), pp. 43–78.

[32] H. Li, X. Wan, P. Yin, and L. Zhao, *Regularity and finite element approximation for two-dimensional elliptic equations with line dirac sources*, J. Comput. Appl. Math., 393 (2021), 113518.

[33] F. Millar, I. Muga, S. Rojas, and K. G. V. der Zee, *Projection in Negative Norms and the Regularization of Rough Linear Functionals*, https://arxiv.org/abs/2101.03044, 2021.

[34] W. F. Mitchell, *A comparison of adaptive refinement techniques for elliptic problems*, ACM Trans. Math. Software, 15 (1989), pp. 326–347, 1990.

[35] R. Mittal and G. Iaccarino, *Immersed boundary methods*, in Annual Review of Fluid Mechanics, Annu. Rev. Fluid Mech. 37, Annual Reviews, Palo Alto, CA, 2005, pp. 239–261.

[36] P. Morin, R. H. Nochetto, and K. G. Siebert, *Data oscillation and convergence of adaptive fem*, SIAM J. Numer. Anal., 38 (2000), pp. 466–488.

[37] R. H. Nochetto, *Pointwise a posteriori error estimates for elliptic problems on highly graded meshes*, Math. Comp., 64 (1995), pp. 1–22.

[38] R. H. Nochetto, K. G. Siebert, and A. Veeser, *Theory of adaptive finite element methods: An introduction*, in Multiscale, Nonlinear and Adaptive Approximation, Springer, Berlin, 2009, pp. 409–542.

[39] C. S. Peskin, *The immersed boundary method*, Acta Numer., 11 (2002), pp. 479–517.

[40] D. D. Silva, F. Ferrari, and S. Salsa, *Perron's solutions for two-phase free boundary problems with distributed sources*, Nonlinear Anal. Theory Methods Appl., 121 (2015), pp. 382–402.

[41] R. Stevenson, *An optimal adaptive finite element method*, SIAM J. Numer. Anal., 42 (2005), pp. 2188–2217.

[42] R. Stevenson, *Optimality of a standard adaptive finite element method*, Found. Comput. Math., 7 (2007), pp. 245–269.

[43] R. Stevenson, *The completion of locally refined simplicial partitions created by bisection*, Math. Comp., 77 (2008), pp. 227–241.

[44] J.-P. Suarez, G. B. Jacobs, and W.-S. Don, *A high-order Dirac-delta regularization with optimal scaling in the spectral solution of one-dimensional singular hyperbolic conservation laws*, SIAM J. Sci. Comput., 36 (2014), pp. A1831–A1849.

[45] A.-K. Tornberg, *Multi-dimensional quadrature of singular and discontinuous functions*, BIT, 42 (2002), pp. 644–669.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.