PREPRINT 419

MARCH 2015

RWTH

igpm

Analysis of Highly Accurate Finite Element Based Algorithms for Computing Distances to Level Sets

Jörg Grande*

Institut für
Geometrie und Praktische Mathematik
Templergraben 55, 52056 Aachen, Germany

* Institut für Geometrie und Praktische Mathematik, RWTH Aachen University, D-52056 Aachen, Germany (grande@igpm.rwth-aachen.de).

ANALYSIS OF HIGHLY ACCURATE FINITE ELEMENT BASED ALGORITHMS FOR COMPUTING DISTANCES TO LEVEL SETS

JÖRG GRANDE*

Abstract. The signed distance function d to an embedded (hyper-) surface Γ is required in the analysis and implementation of some higher order methods for the numerical treatment of partial differential equations on surfaces. Two algorithms for the approximation of d are presented in this paper, which only require a finite element approximation of a (smooth) level set function of Γ. One method is based on a semismooth Newton method; the other method is a nested fixed point iteration. Both are generalizations of known methods. We provide full (local) convergence analyses. Moreover, the methods are compared in two numerical experiments.

Key words. finite elements, level sets, quasi-distance, gradient recovery, semismooth Newton method, convergence analysis

AMS subject classifications. 41A25, 51M04, 65D99, 65G99, 65Y20

1. Introduction. Let $\Gamma \subset \mathbb{R}^N$ be a smooth, oriented (hyper-) surface. The objective of this paper is the convergence analysis of two numerical algorithms to approximate the signed distance function d of Γ. Closely related to d is the base-point function p. For any point x in a (tubular) neighborhood U of Γ, cf. [13], the pair $\big(p(x), d(x)\big) \in \Gamma \times \mathbb{R}$ is a decomposition with the fundamental relation

$$x = p(x) + d(x)n\big(p(x)\big), \quad x \in U, \tag{1.1}$$

where $n(x) = Dd(x)$ is the gradient of d. The coordinate system defined by the tangential directions of p and n is orthogonal. It is used in the theoretical analysis of partial differential equations (PDEs) involving embedded surfaces. It is also required in numerical methods for such problems: In [6], p and d are used in the implementation of an adaptive finite element method for the Laplace-Beltrami PDE. In [5] and [9], higher-order methods for the Laplace-Beltrami PDE are studied which require (approximations of) p and d. The higher-order discretization of interfacial tension in two-phase flows studied in [8] needs approximations of p and d. A central point of the redistancing method for level set functions in [17] is the (higher-order) approximation of d.

For the simplest surfaces, explicit expressions for d are available, e. g. for a hyperplane, a sphere or a torus. The base point p can be computed easily by rearranging (1.1) in this case. In a bigger class of examples, an explicit expression of a smooth level set function ϕ of Γ is known, that is $\Gamma = \{x \in U \mid \phi(x) = 0\}$. This case is considered, for example, in [6], where a straightforward Newton method is used to compute p and d. Further below, we will generalize the Newton method to the setting considered in this paper. In more complex examples like two-phase flow problems, ϕ is not available in the numerical algorithms. Only a finite element approximation ϕ_h of ϕ is given. Similarly, instead of Γ, only the level set $\Gamma_h = \{x \in \mathbb{R}^N \mid \phi_h(x) = 0\}$ is available in the numerical algorithm. Clearly, only approximations p_h and d_h of p and d, respectively, can be computed in this setting.

The approximation of p_h and d_h. The approximation for which numerical algorithms are considered in this paper was introduced in [17]. To generalize (1.1), a

*Institut für Geometrie und Praktische Mathematik, RWTH Aachen University, D-52056 Aachen, Germany (grande@igpm.rwth-aachen.de).

suitable approximation of $n = Dd$ is required. As suggested in [17], one first recovers the gradient $D\phi_h$ as a continuous (vector-valued) finite element function g_h. Simple schemes based on local averaging are sufficient for this. Scaling g_h to unit length yields the quasi-normal field n_h, where "quasi" refers to the fact that n_h is generally not orthogonal to Γ_h, but the angle between n_h and the true normal $\bar{n}_h$ of Γ_h is "small". A main result of [17] is as follows: There is an open neighborhood $\tilde{U}$ of Γ_h such that the decomposition of $x \in \tilde{U}$ into $\big(p_h(x), d_h(x)\big) \in \Gamma_h \times \mathbb{R}$ is well-defined and the generalization of (1.1) holds, more specifically

$$x = p_h(x) + d_h(x) n_h\big(p_h(x)\big), \quad x \in \tilde{U}. \tag{1.2}$$

Further properties of p_h and d_h, in particular, optimal approximation results for p and d, are derived in [17] and [9]; under assumptions of the type

$$\phi \in C^{2,1}(\overline{U}), \quad c^{-1} \le |D\phi| \le c \quad \text{in } U, \tag{1.3}$$

$$\|\phi - \phi_h\|_{L^\infty(U)} \le c h^{k+1}, \tag{1.4}$$

one obtains, for example, $\|d - d_h\|_{L^\infty(\tilde{U})} \le c h^{k+1}$. The main topic of the present paper is the convergence analysis of two algorithms for the computation of p_h and d_h.

Overview of the numerical methods. The first algorithm for p_h and d_h in (1.2) generalizes the Newton method suggested in [6] for the case of a smooth ϕ,

$$\partial F(y, s) = 0, \quad F(y, s) := |y - x| + s\phi(y),$$

where ∂F is the Jacobian of F (with respect to y and s). We modify this by replacing ϕ with ϕ_h and replacing $D\phi$ in the Jacobian by the recovered gradient g_h. The solution of this system of equations requires a non-smooth Newton method, cf. [18, 11]. The error analysis requires a Kantorovich-type convergence theorem for semismooth functions.

The second algorithm for p_h and d_h in (1.2) is a modification of the algorithm given in [17], which is a nested iteration. The outer iteration is

$$y_{k+1} = x - s_k n_h(y_k), \quad k \in \mathbb{N}_0, \tag{1.5}$$

where s_k is determined by a line search from y_k and the condition $\phi_h\big(x - s_k n_h(y_k)\big) = 0$. For the line search, a quasi-Newton algorithm is used in [17]. We modify the inner iteration in order to use a (scalar) non-smooth Newton method, which makes the Kantorovich theorem available in the analysis.

Main results. The main results of this paper are full (local) convergence proofs for both algorithms outlined above, cf. Theorems 4.9 and 5.9: There is a neighborhood $\tilde{U}$ (independent of h) such that both algorithms converge at least linearly for any starting value $x \in \tilde{U}$ to $\big(p_h(x), d_h(x)\big)$. The convergence of the Newton method is locally quadratic. Moreover, the convergence proofs guarantee the existence and uniqueness of the solutions $p_h(x)$ and $d_h(x)$. We do not have to presuppose their existence and the decompsition (1.2). This makes our results independent of the theoretical analysis in [17].

As far as we know, there is no convergence analysis for the Newton method to compute p_h and d_h in the literature if only the approximation ϕ_h is available. For the nested iteration, [17] contains a partial analysis; the convergence and well-posedness of the outer iteration is shown in a modified setting, where the line search is considered as "black box", i. e., an oracle is used for the computation of s_k. Hence, our analysis extends and completes the one in [17].

Two numerical experiments are performed. The methods are applied to a redistancing problem from [17], and a numerical integration over Γ_h is performed. The performance of the methods is compared. In both experiments, the Newton method is slightly more robust. In the second experiment, the nested iteration is slightly faster.

The paper is organized as follows. In Section 2, the semismooth Newton method and some weak concepts of differentiability are reviewed which are required to state the Kantorovich-type convergence theorem, Theorem 2.3. In Section 3, two simple, but non-standard, lemmata on the semismoothness of finite element functions are proved. A simple gradient-recovery method is reviewed and some of its approximation properties are recorded. The semismooth Newton method for p_h and d_h is introduced and analyzed in Section 4. The nested iteration for p_h and d_h is introduced and analyzed in Section 5. In Section 6, some details of the implementation are discussed; a simple damping scheme is introduced, and a caching strategy for some data is described. Section 7 contains the two numerical experiments.

2. Preliminaries on semismooth Newton methods. There is a huge amount of literature on Newton's method. An introduction to semismooth Newton methods is given in [18, 11]. A survey of semismooth and smoothing Newton methods is [15].

REMARK 2.1 (Notation). *As in the introduction, Df denotes the gradient of the scalar function f. Below, the Jacobian ∂f of a scalar function is often used to unify the notation for scalar- and vector-valued mappings.*

The symbol $|\cdot|$ is used for the absolute value, the Euclidean vector norm, and the spectral norm of matrices.

The symbol n usually denotes a (quasi-) normal field. In this section, it also denotes a positive integer. The intended meaning should be clear from the context.

Let $F\colon U \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be a locally Lipschitzian function (with constant L), that is, each point $x \in U$ has a neighborhood V_x such that

$$|F(z) - F(y)| \le L\,|z - y| \quad \text{for all } y, z \in V_x.$$

By Rademacher's theorem, F is differentiable almost everywhere in U. Let D_F denote the exceptional set of measure 0 and, for $x \in U \setminus D_F$, let $\partial F(x)$ denote the Jacobian matrix of F. The generalized Jacobians of Bouligand and Clarke are defined as

$$\partial_{\mathrm{b}} F(x) := \left\{ \lim_{i\to\infty} \partial F(x_i) \;\middle|\; (x_i)_{i\in\mathbb{N}} \subset U \setminus D_F,\ x_i \to x \right\}, \quad x \in U,$$

$$\partial_{\mathrm{c}} F(x) := \operatorname{conv} \partial_{\mathrm{b}} F(x), \quad x \in U, \tag{2.1}$$

cf. [11]. The generalized Jacobians are set valued function. If F is continuously differentiable at x, the reassuring identities $\partial_{\mathrm{b}} F(x) = \partial_{\mathrm{c}} F(x) = \{\partial F(x)\}$ hold. The spectral norm of Clarke's (set-valued) Jacobian matrices is defined as follows,

$$|\partial_{\mathrm{c}} F(x)| = \sup\{|M| \mid M \in \partial_{\mathrm{c}} F(x)\}. \tag{2.2}$$

For some computations below, a chain rule for Clarke's generalized Jacobian is required; generally, it only yields a set inclusion, cf. [12],

$$\partial_{\mathrm{c}}(f \circ g)(x) \subseteq \operatorname{conv}\big(\partial_{\mathrm{c}} f\big(g(x)\big)\partial_{\mathrm{c}} g(x)\big). \tag{2.3}$$

A straightforward generalization of Newton's method is

$$x_{k+1} := x_k - M_k^{-1} F(x_k), \quad \text{for some } M_k \in \partial_{\mathrm{c}} F(x_k), k \in \mathbb{N}_0, \tag{2.4}$$

where x_0 is a suitable initial value. In addition to prescribing an initial value, one must also specify how M_k is chosen in each step to obtain an algorithm. The family of locally Lipschitzian functions is too big for a "good" convergence theory. A Kantorovich-type theorem is known in the class of semismooth functions. As this is not a standard tool in finite element analyses, we give an introduction here.

The one-sided (Gâteaux) directional derivative of F at $x \in U$ in the direction $v \in \mathbb{R}^n$ is

$$F'(x;v) := \lim_{t\to 0^+} \frac{1}{t}\big(F(x+tv) - F(x)\big), \tag{2.5}$$

if the limit exists. The function F is semismooth at x, if it is locally Lipschitzian and

$$\lim_{M\in\partial_c F(x+tw), w\to v, t\to 0^+} Mw$$

exists for any $v \in \mathbb{R}^n$. The following equivalent characterization is used later,

THEOREM 2.2 ([11, Thm. 2.9]). *The following statements ($x \in U$) are equivalent.*

1. *F is semismooth at x.*
2. *F is locally Lipschitzian at x, $F'(x;\cdot)$ exists, and, for any $M \in \partial_c F(x+v)$, there holds*

$$|Mv - F'(x;v)| = o(|v|) \quad \text{for } v \to 0.$$

The function F is p-semismooth at $x \in U$ for some $0 < p \le 1$, if it is locally Lipschitzian at x, $F'(x;\cdot)$ exists, and, for any $M \in \partial_c F(x+v)$, there holds

$$|Mv - F'(x;v)| = \mathcal{O}(|v|^{1+p}) \quad \text{for } v \to 0. \tag{2.6}$$

Clearly, p-semismoothness implies semismoothness. It is well-known that the products and sums of (p-) semismooth functions are (p-) semismooth. Also, a vector-valued function is (p-) semismooth if and only if all of its component functions are (p-) semismooth.

For semismooth functions, the following generalization of Kantorovich's theorem is available,

THEOREM 2.3 ([16, Thm 3.3]). *Suppose that F is locally Lipschitzian and semismooth on the closed ball $B := \overline{B}(x_0, r)$. Further, suppose that there are real constants β, γ, δ such that*

$$\left|M^{-1}\right| \le \beta \quad \text{for all } M \in \partial_c F(x), x \in B, \tag{2.7}$$

$$|M(y-x) - F'(x;y-x)| \le \gamma\,|y-x| \quad \text{for all } M \in \partial_c F(x), x, y \in B, \tag{2.8}$$

$$|F(y) - F(x) - F'(x, y-x)| \le \delta\,|y-x| \quad \text{for all } x, y \in B. \tag{2.9}$$

If $\alpha := \beta(\gamma+\delta) < 1$ and $\beta|F(x_0)| \le r(1-\alpha)$, there exists a unique solution x_* to $F(x) = 0$ in B, and the Newton sequence (2.4) is well-defined, remains in B, and converges at least linearly to to x_* with rate α. Moreover, the a posteriori error estimate

$$|x_k - x_*| \le \frac{\alpha}{1-\alpha}\,|x_k - x_{k-1}|\,, \quad k \in \mathbb{N}, \tag{2.10}$$

holds.

The requirements of this theorem are similar to that of the classical Kantorovich theorem. Contrary to the classical theorem, the requirement $\alpha < 1$ cannot be forced

by reducing the size of B. Local convergence of higher order is available for p-semismooth functions,

THEOREM 2.4 ([16, Thm. 3.2]). *Suppose that x_* solves $F(x) = 0$, that F is locally Lipschitzian and semismooth at x_*, and that all $M \in \partial_c F(x_*)$ are nonsingular. Then, the Newton sequence* (2.4) *is well-defined and convergent to x_* in a neighborhood of x_*. If in addition F is p-semismooth at x_*, $0 < p \leq 1$, the convergence of* (2.4) *is of order $1 + p$.*

3. Preliminaries. Let $(\mathcal{T}_h)_{h>0}$ be a shape-regular family of triangulations of the domain $\Omega \subset \mathbb{R}^N$, $N \in \mathbb{N}$. The parameter h denotes the mesh width; the maximal mesh width is denoted by $h_0 \geq h > 0$. For any point $x \in \Omega$, let $\mathcal{T}_x = \{S \in \mathcal{T}_h \mid x \in S\}$ be the set of (closed) simplexes which contain x. Due to the shape-regularity, the cardinality $|\mathcal{T}_x|$ is bounded by a constant independent of x and h,

$$|\mathcal{T}_x| \leq c \quad \text{for all } x \in \Omega,\ h > 0. \tag{3.1}$$

Let $(X_h^k)_{h>0}$ be the family of continuous, piecewise polynomial finite element spaces of degree k,

$$X_h^k = \left\{ f \in C(\Omega) \;\middle|\; f|_S \in \mathcal{P}^k, S \in \mathcal{T}_h \right\}. \tag{3.2}$$

The same notation is used for vector- and matrix-valued finite elements. For $f \in X_h^k$, $S \in \mathcal{T}_h$, we write the polynomial $f|_S$ simply as f_S.

LEMMA 3.1. *Let $f \in X_h^k$ be an arbitrary (vector-valued) finite element function. Then, f is locally Lipschitzian and Bouligand's generalized Jacobian is given by*

$$\partial_b f(x) = \{\partial f_S(x) \mid S \in \mathcal{T}_x\} \quad \text{for all } x \in \Omega. \tag{3.3}$$

Moreover, the directional derivative $f'(x; v)$ exists for all $x \in \Omega$, $v \in \mathbb{R}^N$, and there holds

$$f'(x; v) = \partial f_S(x)v \quad \text{for all } S \in \{T \in \mathcal{T}_x \mid x + \epsilon v \in T \text{ for all (suff. small) } \epsilon > 0\}. \tag{3.4}$$

Proof. Let $f \in X_h^k$ and $x \in \Omega$ be arbitrary. For sufficiently small $\epsilon > 0$, the ball $B := B(x, \epsilon)$ satisfies $B \cap \Omega \subset \cup\{S \mid S \in \mathcal{T}_x\}$. For any $y \in B \cap \Omega$, there holds $f(y) = f_S(y)$ for some $S \in \mathcal{T}_x$. By (3.1), the cardinality of $\mathcal{T}_x$ is finite (independent of h). As the f_S are polynomials (which are locally Lipschitzian), f is Lipschitzian on B.

For some $S \in \mathcal{T}_x$, let $(x_i)_{i\in\mathbb{N}} \subset S$ be a sequence that converges to x. Then, $\partial f(x_i) = \partial f_S(x_i)$ exists for all $i \in \mathbb{N}$ and $\{\partial f_S(x) \mid S \in \mathcal{T}_x\} \subseteq \partial_b f(x)$. Conversely, let $M \in \partial_b f(x)$, and let $(x_i)_{i\in\mathbb{N}} \subset \Omega$ be a sequence converging to x such that $\partial f(x_i)$ exists for all $i \in \mathbb{N}$ and $\partial f(x_i) \to M$ for $i \to \infty$. There is a subsequence of (x_i), which we again name (x_i), with $x_i \in B$. Due to $B \cap \Omega \subset \cup\{S \mid S \in \mathcal{T}_x\}$ and $\mathcal{T}_x$ being finite, there exists a simplex $S \in \mathcal{T}_x$ and a further subsequence $(y_i)_{i\in\mathbb{N}}$ of (x_i) with $y_i \in S$, $i \in \mathbb{N}$. Therefore, one has $\partial f(y_i) = \partial f_S(y_i)$ and $M \in \{\partial f_S(x) \mid S \in \mathcal{T}_x\}$.

Let $v \in \mathbb{R}^N$ be arbitrary and $S \in \mathcal{T}_x$ be such that the line segment $\mathrm{conv}\{x, x+\epsilon v\}$ is a subset of S for some $\epsilon > 0$. Then, (3.4) follows immediately from the definition in (2.5) and $f = f_S$ on S. ∎

LEMMA 3.2. *Let $f \in X_h^k$ be an arbitrary (vector-valued) finite element function. Then, f is (p-) semismooth for all $x \in \Omega$ (and for all $0 < p \leq 1$).*

Proof. Let $f \in X_h^k$ and $x \in \Omega$ be arbitrary. From Lemma 3.1, it is known that f is locally Lipschitzian at x and that $f'(x;v)$ exists for all $v \in \mathbb{R}^N$. For sufficiently small $\epsilon > 0$, the ball $B := B(x,\epsilon)$ satisfies $B \cap \Omega \subset \cup\{S \mid S \in \mathcal{T}_x\}$. Let $|v| < \epsilon$ hold. Then, the line segment $\mathrm{conv}\{x,y\}$, $y := x+v$, is a subset of B, and $\mathcal{T}_y \subseteq \mathcal{T}_x$. Let $M \in \partial_{\mathrm{b}} f(y)$ be arbitrary. By Lemma 3.1, there is a $S \in \mathcal{T}_y$ with $M = \partial f_S(y)$. Using $\mathcal{T}_y \subseteq \mathcal{T}_x$, one gets $S \in \mathcal{T}_x$; therefore, $\mathrm{conv}\{x,y\} \in S$. Due to Lemma 3.1, this implies $f'(x;v) = \partial f_S(x)v$. One can write $Mv - f'(x;v) = \big(\partial f_S(y) - \partial f_S(x)\big)v$. The mean value theorem yields

$$|Mv - f'(x;v)| \le c\,|v|^2 \quad \text{for all } M \in \partial_{\mathrm{b}} f(x+v), v \to 0. \tag{3.5}$$

We show that (3.5) also holds if $\partial_{\mathrm{b}} f$ is replaced with $\partial_{\mathrm{c}} f$. By (2.1) and Lemma 3.1, any $\tilde{M} \in \partial_{\mathrm{c}} f(y)$ can be written as a (finite) convex combination $\tilde{M} = \sum_i \lambda_i M_i$ with $M_i = \partial f_{S_i}(y)$ for some $S_i \in \mathcal{T}_x$ and $\sum_i \lambda_i = 1$, $\lambda_i \ge 0$. Thus $\tilde{M}v - f'(x;v) = \sum_i \lambda_i \big(M_i v - f'(x;v)\big)$. One obtains

$$\big|\tilde{M}v - f'(x;v)\big| \le c\,|v|^2 \quad \text{for all } \tilde{M} \in \partial_{\mathrm{c}} f(x+v), v \to 0.$$

Applying Theorem 2.2 (and the condition in (2.6) for $0 < p \le 1$) concludes the proof of the lemma. ∎

The following properties of the level set function ϕ and its finite element approximation $\phi_h \in X_h^k$ are assumed for the convergence analyses below:

Assumption 3.3. *There are open sets U, U^e with $\Gamma \subset U \subset U^e \subset \Omega$ such that $\phi \in C^{2,1}(U^e) \cap H^{k+1,\infty}(U^e)$ and*

$$0 < c_0 \le |D\phi(x)| \le c_1 \quad \textit{for all } x \in U^e, \tag{3.6}$$

$$\|\phi_h - \phi\|_{L^\infty(U^e)} + h\,\|D(\phi_h - \phi)\|_{L^\infty(U^e)} + h^2\,\big\|D^2(\phi_h - \phi)\big\|_{L^\infty(U^e)} \le c h^{k+1}, \tag{3.7}$$

$$S \cap U \ne \emptyset \;\Rightarrow\; T \subset U^e \quad \textit{for all } S,T \in \mathcal{T}_h \textit{ with } T \cap S \ne \emptyset, h \le h_0. \tag{3.8}$$

3.1. Gradient recovery and quasi-normal fields. In this section, a simple gradient recovery based on local averaging is described which is used to define the quasi-normal field. The gradient recovery technique is straightforward and also explained in [8, 9].

Let I_h be a (nodal) interpolation operator for X_h^k. Let U_h, $h \le h_0$, be the set $\cup\{S \in \mathcal{T}_h \mid S \cap U \ne \emptyset\}$; it satisfies $U \subset U_h \subset U^e$. A gradient recovery operator is a mapping $\mathbb{G}_h \colon X_h^k \to (X_h^k)^N$ with the following (reasonable) approximation and stability properties:

$$\|\mathbb{G}_h I_h \phi - D\phi\|_{L^\infty(U_h)} \le c h^k, \tag{3.9}$$

$$\|\mathbb{G}_h f_h\|_{L^\infty(U_h)} \le c\,\|f_h\|_{H^1_\infty(U^e)} \quad \text{for all } f_h \in X_h^k. \tag{3.10}$$

A simple method is as follows. Let N_h^k be a set of Lagrangian finite element nodes for X_h^k and let $|\mathcal{T}_\xi|$ denote the cardinality of $\mathcal{T}_\xi$, $\xi \in N_h^k$. One defines $\mathbb{G}_h f_h$, $f_h \in X_h^k$, via

$$(\mathbb{G}_h f_h)(\xi) := \frac{1}{|\mathcal{T}_\xi|} \sum_{S \in \mathcal{T}_\xi} Df_S \quad \text{for all } \xi \in N_h^k. \tag{3.11}$$

Definition 3.4 ([17, Def. 3.3]). *A quasi-normal field is a mapping $n_h \colon \Gamma \to S_{N-1}$, $S_{N-1} = \big\{x \in \mathbb{R}^N \;\big|\; |x| = 1\big\}$, with the following additional properties. For each*

$x \in \Gamma$ there exist real constants $\tilde{\delta}_x < 1$, $r_x > 0$, γ_x, c_x with

$$|n_h(x) - n_h(y)| \leq \gamma_x |x - y| \quad \textit{for all } y \in B(x, r_x) \cap \Gamma, \tag{3.12}$$

$$|(n_h(x), x - y)| \leq \tilde{\delta}_x |x - y| + c_x |x - y|^2 \quad \textit{for all } y \in B(x, r_x) \cap \Gamma, \tag{3.13}$$

$$\sup_{x\in\Gamma} \tilde{\delta}_x =: \tilde{\delta} < 1, \quad \sup_{x\in\Gamma} \gamma_x < \infty, \quad \sup_{x\in\Gamma} c_x < \infty, \quad \inf_{x\in\Gamma} r_x > 0. \tag{3.14}$$

Using the gradient recovery operator $\mathbb{G}_h$ on the finite element level set function ϕ_h, the quasi-normal field $n_h \in (X_h^k)^N$ is defined as

$$n_h(x) := \frac{g_h(x)}{|g_h(x)|}, \quad g_h(x) := \mathbb{G}_h \phi_h(x), \quad \text{for all } x \in \Omega. \tag{3.15}$$

REMARK 3.5. *From* (3.9) *and* (3.10)*, it follows that n_h is a quasi-normal field in the sense of Definition 3.4, cf. [8, La. 3.1]. Moreover, there holds $\tilde{\delta} \leq ch^k$. This fact is only required in the analysis of the nested iteration, cf. Theorem 5.9.*

For the convergence analysis of the Newton method, the following approximation properties of g_h are used.

LEMMA 3.6. *The following inequalities involving the derivatives of ϕ hold,*

$$|v - \partial\phi(x)| \leq ch^k \quad \textit{for all } v \in \partial_c \phi_h(x), x \in U, \tag{3.16}$$

$$|g_h(x) - D\phi(x)| \leq ch^k \quad \textit{for all } x \in U, \tag{3.17}$$

$$\left|M - \partial^2\phi(x)\right| \leq ch^{k-1} \quad \textit{for all } M \in \partial_c g_h(x), x \in U. \tag{3.18}$$

Proof.

First, we show that, for every $S \in \mathcal{T}_h$, $S \cap U \neq \emptyset$, there holds

$$\|D(\phi_h - \phi)\|_{L^\infty(S)} \leq ch^k, \tag{3.19}$$

$$\|g_h - D\phi\|_{L^\infty(S)} \leq ch^k, \tag{3.20}$$

$$\|\partial(g_h - D\phi)\|_{L^\infty(S)} \leq ch^{k-1}. \tag{3.21}$$

The inequality (3.19) follows immediately from (3.7). Writing $g_h - D\phi = g_h - D\phi_h + D\phi_h - D\phi$, the inequality (3.20) follows from (3.9) and (3.19). Finally, writing $\partial g_h - D^2\phi = \partial(g_h - D\phi_h) + D^2(\phi_h - \phi)$, (3.21) is obtained from (3.7), a standard finite element estimate, and (3.20).

The inequality (3.17) follows from (3.20) because g_h and $D\phi$ are continuous. As (3.19) holds on every (closed) simplex $S \in \mathcal{T}_h$, $S \cap U \neq \emptyset$, it holds by definition for Bouligand's generalized Jacobian in all points covered by the simplices S. We show that it also holds for Clarke's generalized Jacobian. Let $v \in \partial_c \phi_h(x)$ be arbitrary, $x \in U$. By (2.1) and Lemma 3.1, $v = \sum_i \lambda_i v_i$ with $v_i = \partial\phi_h|_{S_i}(x)$ for some $S_i \in \mathcal{T}_x$ and $\sum_i \lambda_i = 1$, $\lambda_i \geq 0$. Thus, $v - \partial\phi(x) = \sum_i \lambda_i \big(v_i - \partial\phi(x)\big)$. The inequality (3.16) now follows from (3.19) for the v_i. The inequality (3.18) is proved in the same way using (3.21). ∎

The following elementary estimate for the norm of a block-matrix is used below,

$$\left|\begin{pmatrix} A & B \\ C & D \end{pmatrix}\right| \leq 4 \max\{|A|, |B|, |C|, |D|\}. \tag{3.22}$$

Furthermore, the well-known perturbation formula for matrices is required; let ϵ be a matrix with $|\epsilon| < 1$, then $I + \epsilon$ is invertible and

$$|I + \epsilon| \geq 1 - |\epsilon| > 0. \tag{3.23}$$

4. The Newton method for p_h and d_h. For the remainder of this section, let $x \in U$ denote the point for which $p_h(x)$ and $d_h(x)$ have to be computed. The following problem is solved: Find a pair $(y, s) \in \Gamma_h \times \mathbb{R}$ such that

$$0 = F(y,s) := \big(y + sg_h(y) - x, \phi_h(y)\big)^T. \tag{4.1}$$

The dependence of $F\colon \mathbb{R}^N \times \mathbb{R} \to \mathbb{R}^N \times \mathbb{R}$ on x is not shown explicitly in the notation. It is easy to see that, if a solution $(y, s) \in \mathbb{R}^N \times \mathbb{R}$ of (4.1) exists, then $y = p_h(x)$ and $|g_h(y)|s = d_h(x)$: From (4.1), one immediately obtains $y \in \Gamma_h$; furthermore, by (3.15), $x = y + s|g_h(y)|n_h(y)$.

We solve (4.1) with the (generalized) Newton method (2.4) and the initial value $(x, 0)$. In the remainder of this section, the convergence of this algorithm is proved by applying Theorem 2.3. As Theorem 2.3 is also an *existence* and *uniqueness* theorem for the solution of $F(y, s) = 0$, it is not necessary to presuppose the existence theory of [17] for $p_h(x)$ and $d_h(x)$. The numerical algorithm itself proves the well-definedness of the problem. This way, we avoid the use of Brouwer's theorem on the invariance of the domain which is required in [17].

REMARK 4.1. *Instead of* (4.1)*, one could also consider*

$$0 = \big(y + sn_h(y) - x, \phi_h(y)\big)^T.$$

In this case, Lemma (3.6) *must be extended with more complicated estimates. For example, instead of* (3.17)*, one only has* $|n_h(y) - n(y)| \le ch^k + c|\bar{d}_h(y)|$*,* $y \in U$*, where* $\bar{d}_h$ *is the exact signed distance function to* Γ_h*. The approach ultimately works, but there seems to be little benefit of the additional complexity.*

Due to Lemma 3.2 and the general results in Section 2, F is a (1-) semismooth function. Its generalized Jacobian in Clarke's sense is estimated in the subsequent lemmas.

LEMMA 4.2. *There holds*

$$\partial_c F(y,s) \subseteq \begin{pmatrix} I + s\partial_c g_h(y) & g_h(y) \\ \partial_c \phi_h(y) & 0 \end{pmatrix}. \tag{4.2}$$

Proof. The standard approach to the result would be the repeated application of the chain rule (2.3). A quick alternative is to use the explicit characterization of $\partial_{\mathrm{b}} F$ in Lemma 3.1,

$$\begin{aligned} \partial_{\mathrm{c}} F(y,s) &= \operatorname{conv} \partial_{\mathrm{b}} F(y,s) = \operatorname{conv} \{\partial F_S(y,s) \mid y \in S, s \in \mathbb{R}\} \\ &= \operatorname{conv} \left\{ \begin{pmatrix} I + s\partial g_h|_S(y) & g_h(y) \\ \partial \phi_h|_S(y) & 0 \end{pmatrix} \;\middle|\; y \in S, s \in \mathbb{R} \right\} \\ &\subseteq \operatorname{conv} \begin{pmatrix} I + s\partial_{\mathrm{b}} g_h|_S(y) & g_h(y) \\ \partial_{\mathrm{b}} \phi_h|_S(y) & 0 \end{pmatrix} \end{aligned}$$

If $A \subseteq \mathbb{R}^m$ and $B \subseteq \mathbb{R}^n$ are convex sets, the direct product $A \times B \subseteq \mathbb{R}^{n+m}$ is convex. Thus, if one extends $\partial_{\mathrm{b}} g_h$ and $\partial_{\mathrm{b}} \phi_h$ to $\partial_{\mathrm{c}} g_h$ and $\partial_{\mathrm{c}} \phi_h$ in the above formula, the outer convex hull operation becomes redundant. ∎

REMARK 4.3 (Implementation). *The Newton method* (2.4) *requires the evaluation of* $\partial_c F(y, s)$*. Formula* (4.2) *cannot be used directly for this as the right-hand side might be a strict super-set of the left-hand side. However, in the implementation of the method, one tracks a simplex* $S \in \mathcal{T}_h$ *with* $y \in S$*. The proof of Lemma 4.2 shows*

that using the classical Jacobian $\partial F_S(y,s)$ always yields an element of $\partial_b F(y,s) \subseteq \partial_c F(y,s)$.

From the triangle inequality and (3.18), one gets

$$|\partial_{\mathrm{c}} g_h(y)| \le \left|D^2\phi(y)\right| + ch^{k-1} \le c\,\|\phi\|_{C^2(U)} + ch_0^{k-1} \le c_2 \quad \text{for all } y \in U \tag{4.3}$$

for a constant c_2 independent of h.

LEMMA 4.4. *Let $c_3 := 4c_0^{-2}\max\{c_0^2 + c_1^2, c_1, 1\}$. If $8c_3\max\{c_2|s|, ch^k\} \le 1$, there holds*

$$\left|\partial_c F(y,s)^{-1}\right| \le 2c_3 \quad \textit{for all } y \in U.$$

Proof. Let $y \in U$ and $s \in \mathbb{R}$ with $8c_3\max\{c_2|s|, ch^k\} \le 1$ be arbitrary. Let $q := |D\phi(y)|^{-2}$. Owing to (3.6), one has $q \le c_0^{-2}$. Let

$$M := \begin{pmatrix} I & D\phi(y) \\ D\phi(y)^T & 0 \end{pmatrix}.$$

It is easy to check that

$$M^{-1} = \begin{pmatrix} I - qD\phi(y)D\phi(y)^T & qD\phi(y) \\ qD\phi(y)^T & -q \end{pmatrix}.$$

Using (3.22) and (3.6), one gets $|M^{-1}| \le 4c_0^{-2}\max\{c_0^2 + c_1^2, c_1, 1\} = c_3$. By Lemma 4.2, there holds

$$\delta := \partial_{\mathrm{c}} F(y,s) - M \subseteq \begin{pmatrix} s\partial_{\mathrm{c}} g_h(y) & (g_h - D\phi)(y) \\ (\partial_{\mathrm{c}}\phi_h - D\phi)(y)^T & 0 \end{pmatrix}.$$

Using (3.22), (4.3), and Lemma 3.6, one obtains $|\delta| \le 4\max\{|s|c_2, ch^k\}$. Hence, $|\delta| \le (2c_3)^{-1}$. Combining this with the bound on $|M^{-1}|$, one gets $|M^{-1}||\delta| \le \frac{1}{2}$. One writes $\partial_{\mathrm{c}} F(y,s)^{-1} = (M+\delta)^{-1} = M^{-1}(I + M^{-1}\delta)^{-1}$. The conclusion follows from (3.23). ∎

A smooth approximation of F is required below. This will be the function

$$\tilde{F}(y,s) := \big(y + sD\phi(y) - x, \phi(y)\big)^T. \tag{4.4}$$

REMARK 4.5. *The solution of $\tilde{F}(y,s) = 0$ is given by $\big(p(y), |D\phi(y)|d(y)\big) \in \Gamma \times \mathbb{R}$, where $p(y)$ and $d(y)$ are the smooth, orthogonal coordinates defined in* (1.1)*. However, this fact is not used in the analysis below.*

LEMMA 4.6. *There is a positive constant c such that the function*

$$b(h,s) = ch^{k-1}\max\{h, |s|\} \tag{4.5}$$

provides the upper bound

$$\left|\partial_c(F - \tilde{F})(y,s)\right| \le b(h,s) \quad \textit{for all } (y,s) \in U \times \mathbb{R}. \tag{4.6}$$

Proof. Clarke's generalized Jacobian of $F - \tilde{F}$ at $(y,s) \in U \times \mathbb{R}$ satisfies

$$\partial_{\mathrm{c}}(F - \tilde{F})(y,s) \subseteq \begin{pmatrix} s\partial_{\mathrm{c}}(g_h - D\phi)(y) & (g_h - D\phi)(y) \\ \partial_{\mathrm{c}}(\phi_h - \phi)(y) & 0 \end{pmatrix}.$$

From Lemma 3.6 and (3.22), one immediately gets (4.6). ∎

LEMMA 4.7. *For any $M \in \partial_c F(\hat{y})$ there holds*

$$|M(\hat{z}-\hat{y}) - F'(\hat{y};\hat{z}-\hat{y})| \le 2b(h,s)\,|\hat{z}-\hat{y}| \quad \textit{for all } \hat{y}=(y,s), \hat{z} \in U\times\mathbb{R}.$$

Proof. Let $\hat{y},\hat{z} \in U\times\mathbb{R}$ be given. By (3.4), there is a matrix $\tilde{M} \in \partial_c F(\hat{y})$ with $F'(\hat{y};\hat{z}-\hat{y}) = \tilde{M}(\hat{z}-\hat{y})$. Therefore, one obtains $M(\hat{z}-\hat{y})-F'(\hat{y};\hat{z}-\hat{y}) = (M-\tilde{M})(\hat{z}-\hat{y})$ and the result follows from Lemma 4.6. ∎

LEMMA 4.8. *For all $\hat{y} := (y,s) \in U\times\mathbb{R}$, $\hat{z} := (z,t) \in U\times\mathbb{R}$, $\hat{z} \ne \hat{y}$, such that the line segment between them satisfies* $\operatorname{conv}\{\hat{y},\hat{z}\} \subset U\times\mathbb{R}$*, there holds*

$$|F(\hat{z}) - F(\hat{y}) - F'(\hat{y};\hat{z}-\hat{y})| \le \Big(2b\big(h,\max\{s,t\}\big) + c_4\,\|\phi\|_{C^{2,1}(U)}\,|\hat{z}-\hat{y}|\Big)\,|\hat{z}-\hat{y}|$$

with a constant c_4 independent of h.

Proof. Let $e := (\hat{z}-\hat{y})/|\hat{z}-\hat{y}|$. The line segment $\operatorname{conv}\{\hat{y},\hat{z}\}$ is parameterized by $l(r) = \hat{y} + re$, $r \in I := [0, |\hat{z}-\hat{y}|]$. One writes

$$\begin{aligned} F(\hat{z}) - F(\hat{y}) - F'(\hat{y};\hat{z}-\hat{y}) &= \int_I F'(l(r);e)\,dr - F'(\hat{y};e)\,|\hat{z}-\hat{y}| \\ &= \int_I F'(l(r);e) - F'(\hat{y};e)\,dr. \end{aligned}$$

The integrand is rearranged as a sum of three differences,

$$\begin{aligned} F'(l(r);e) - F'(\hat{y};e) &= F'(l(r);e) - \tilde{F}'(l(r);e) + \tilde{F}'(l(r);e) - \tilde{F}'(\hat{y};e) \\ &\quad + \tilde{F}'(\hat{y};e) - F'(\hat{y};e) \\ &=: A + B + C. \end{aligned}$$

The term B is smooth; hence, the mean value theorem implies $B = \tilde{F}''(l(\rho);e)(l(\rho)-\hat{y})$ for some $\rho \in I$. Therefore, $|B| \le c|\hat{z}-\hat{y}|$.

For the term C, one writes $\tilde{F}'(\hat{y};e) = \partial\tilde{F}(\hat{y})e$ and $F'(\hat{y};e) = Me$ for some $M \in \partial_c F(\hat{y})$. Using Lemma 4.6, one gets $|C| \le b(h,s)$. Similarly, one obtains $|A| \le b(h,\max\{s,t\})$. ∎

4.1. The convergence theorem for the Newton method. Consider the tubular neighborhood $U_\epsilon = \big\{x \in \Omega \;\big|\; |\bar{d}_h(x)| \le \epsilon\big\}$,$\epsilon > 0$, of Γ_h, where $\bar{d}_h$ is the (exact) signed distance function of Γ_h. It will be shown that there is a (small) positive value ϵ such that the generalized Newton iteration (2.4) for F converges for all initial values $\hat{x} = (x,0)$, $x \in U_\epsilon$. Due to (3.6), there holds

$$|F(\hat{x})| = \Big|\big(0,\phi_h(x)\big)^T\Big| \le c_1\epsilon \quad \text{for all } x \in U_\epsilon. \tag{4.7}$$

The auxiliary set $\hat{U}_{s_0} = U_{s_0}\times(-s_0,s_0)$ with parameter $s_0 > 0$ is used to obtain the bounds for the parameters α, β, γ, δ in Theorem 2.3 by choosing s_0 sufficiently small. Moreover, there is an auxiliary parameter $0 < r \le s_0$, also chosen below, which occurs in the 'existence part' of Theorem 2.3.

One chooses $h_0 \le s_0$ and s_0 so small that (cf. Lemma 4.4)

$$8c_3\max\{c_2, ch_0^{k-1}\}s_0 \le 1 \quad \text{and} \quad U_{s_0} \subset U.$$

By Lemma 4.4,

$$\left|\partial_{\mathrm{c}} F(\hat{y})^{-1}\right| \le 2c_3 =: \beta \quad \text{for all } \hat{y} \in \hat{U}_{s_0}. \tag{4.8}$$

Possibly decreasing h_0 and s_0 further, one obtains from (4.5) that

$$0 < b(h,s) \le \frac{1}{10}\beta^{-1} \quad \text{for all } |s| \le s_0. \tag{4.9}$$

From this and Lemma 4.7, one obtains

$$|M(\hat{z}-\hat{y}) - F'(\hat{y};\hat{z}-\hat{y})| \le \gamma\,|\hat{z}-\hat{y}|\,, \quad \gamma := \frac{1}{5}\beta^{-1}, \\ \text{for all } \hat{y}=(y,s), \hat{z} \in \hat{U}_{s_0}, M \in \partial_{\mathrm{c}} F(\hat{y}). \tag{4.10}$$

The bound in Lemma 4.8 is considered. The parameter $r > 0$ is chosen such that

$$2c_4 \left\|\phi\right\|_{C^{2,1}(\Omega)} r \le \tfrac{1}{10}\beta^{-1}, \quad 2r \le s_0. \tag{4.11}$$

Let $\hat{y}, \hat{z} \in \hat{U}_{s_0}$, $|\hat{z}-\hat{y}| \le 2r$, be arbitrary. Using (4.9) and (4.11), one gets

$$\begin{aligned} |F(\hat{z}) - F(\hat{y}) - F'(\hat{y};\hat{z}-\hat{y})| &\le \left(\frac{1}{5}\beta^{-1} + c_4 \left\|\phi\right\|_{C^{2,1}(\Omega)} 2r\right) |\hat{z}-\hat{y}| \\ &\le \delta\,|\hat{z}-\hat{y}|\,, \quad \delta := \frac{3}{10}\beta^{-1}. \end{aligned} \tag{4.12}$$

From (4.8), (4.10), (4.12), one obtains $\alpha := \beta(\gamma+\delta) = \frac{1}{2} < 1$. Now the (last) free parameter ϵ is chosen such that

$$0 < \epsilon \le r, \quad \beta c_1 \epsilon \le \frac{1}{2} r. \tag{4.13}$$

Using (4.11) and (4.13), one finds $\epsilon + r \le 2r \le s_0$. Hence, for any starting value $x \in U_\epsilon$, the closed ball $B := \overline{B}(\hat{x}, r)$ satisfies $B \subset \hat{U}_{s_0}$. Therefore, (4.8), (4.10), and (4.12) hold on B. Finally, by (4.7) and (4.13), $\beta|F(\hat{x})| \le \beta c_1 \epsilon \le \frac{1}{2} r = r(1-\alpha)$. Applying Theorem 2.3 yields

THEOREM 4.9. *With the positive constants $\epsilon \le r < s_0$ as above and $h_0 \le s_0$, the following holds for all initial values $x \in U_\epsilon$: There is a unique solution $(y,s) \in \overline{B}\big((x,0),r\big) \subset \hat{U}_{s_0}$ to (4.1). In particular, $x = y + s|g_h(y)|n_h(y)$ and $y \in \Gamma_h$. The Newton method (2.4) for F with initial value $\hat{y}_0 := (x,0)$ converges to (y,s) at least linearly with the rate $\alpha = \frac{1}{2}$. Moreover, the a posteriori error estimate*

$$\left|\hat{y}_k - (y,s)^T\right| \le |\hat{y}_k - \hat{y}_{k-1}|\,, \quad k \in \mathbb{N}, \tag{4.14}$$

holds.

COROLLARY 4.10. *The convergence in Theorem 4.9 is quadratic locally around the solution of (4.1).*

Proof. This follows immediately from Theorem 2.4, Lemma 3.2, and Theorem 4.9. □

REMARK 4.11. *The parameter h_0 is not used explicitly in the determination of the value of the constants β, γ, δ, and r. This is due to the special case $k = 1$ (linear finite elements for ϕ_h). In this case, $b(s,h)$ in (4.9) can only be controlled via s_0, not*

h_0. Essentially, this is a consequence of linear finite elements being unsuitable for the (pointwise) approximation of the second derivatives of ϕ.

For $k \geq 2$, the situation is different. The condition (4.9) *can be satisfied by choosing h_0 sufficiently small. Also, the second summand in* (4.3) *can be controlled by h_0. For h_0 sufficiently small, the only requirement on s_0 is essentially $8c_0^{-2}\max\{c_0^2 + c_1^2, c_1, 1\}c\,\|\phi\|_{C^2(\Omega)}\,s_0 \leq 1$. This inequality depends only on (the slope and curvature of) ϕ and on constants which are independent of the finite element interpolation estimates in Lemma 3.6. Consequently, the convergence radius r is only constrained by such quantities, cf.* (4.11)*. Thus, for $k \geq 2$ and sufficiently fine meshes, the radius of convergence of the method only depends on the continuous level set function ϕ.*

5. The nested iteration for p_h and d_h. As in Section 4, the (arbitrary) point $x \in U$ is fixed, for which $p_h(x)$ and $d_h(x)$ have to be computed. The nested iteration is composed of an inner and an outer problem which are both solved iteratively. The inner problem reads: Given $y \in U$, find an $s = s(y) \in \mathbb{R}$ such that

$$0 = \phi_h\big(G_s(y)\big) =: \phi_y(s) \quad \text{with } G_s(y) := x - s g_h(y). \tag{5.1}$$

It is shown below that this problem is well-defined and uniquely solvable, if y is "close enough" to x and x "close enough" to Γ_h. To solve (5.1), the (scalar) semismooth Newton method (2.4) is used with $F(s) = \phi_y(s)$ and the initial value $s = 0$.

Remark 5.1. *Instead of* (5.1)*, one could also consider $0 = \phi_h\big(x - \tilde{s} n_h(y)\big)$. The solution differs from that of* (5.1) *only by the scaling $\tilde{s} = s|g_h(y)|$.*

The semismooth Newton method is slightly different from the ad hoc quasi-Newton method used in [17] to solve the inner problem. In the latter, the recovered gradient $g_h\big(x - \tilde{s} n_h(y)\big)$ is used instead of $\partial_c \phi_h\big(x - \tilde{s} n_h(y)\big)$ in the computation of $\partial_c \phi_y(s)$. There is no convergence analysis for the inner problem in [17].

The outer problem depends on the solution $s(y) \in \mathbb{R}$ of (5.1) for a given $y \in U$. The outer problem reads: Find $y \in U$ such that

$$y = G(y) \quad \text{with } G(y) := G_{s(y)}(y) = x - s(y) g_h(y). \tag{5.2}$$

The outer problem is thus a fixed point problem for G which is solved iteratively by

$$y_{k+1} := G(y_k), \quad y_0 := x. \tag{5.3}$$

Clearly, a solution y of (5.2) satisfies $p_h(x) = y$ and $d_h(x) = s(y)|g_h(y)|$.

Remark 5.2. *In [6], an iterative method is proposed which is similar to the nested iteration. Instead of solving an inner problem like* (5.1)*, only a single step in the direction of an approximate normal is performed in each iteration. The method is used for numerical experiments in [6], but no convergence analysis is given. We do not consider the method below.*

5.1. Analysis of the inner iteration. For $y \in U$, one can write $g_h(y) = D\phi(y) + \big(g_h(y) - D\phi(y)\big)$. Using (3.17) and (3.6), this yields

$$|g_h(y)| \leq c_1 + ch^k \quad \text{for all } y \in U. \tag{5.4}$$

Lemma 5.3. *Let $y, z \in U$ be such that $\operatorname{conv}\{y, z\} \subset U$. If h and $|z - y|$ are so small that $ch^k + c_1\|D^2\phi\|_{C(U)}|z - y| \leq \frac{1}{2}c_0^2$, there holds*

$$v g_h(y) \geq \frac{1}{2}c_0^2 \quad \textit{for all } v \in \partial_c \phi_h(z).$$

Proof. Let $v \in \partial_c \phi_h(z) \subset \mathbb{R}^{1\times N}$ be arbitrary. One writes

$$vg_h(y) - D\phi(z)^T D\phi(z) = \big(v - D\phi(z)^T\big) g_h(y) + D\phi(z)^T \big(g_h(y) - D\phi(y) + D\phi(y) - D\phi(z)\big). \tag{5.5}$$

Using (3.16), (3.17), and (3.6), one gets

$$\left|vg_h(y) - D\phi(z)^T D\phi(z)\right| \le ch^k \left|g_h(y)\right| + c_1 ch^k + c_1 \left|D\phi(y) - D\phi(z)\right|.$$

For $D\phi(y) - D\phi(z)$, one uses the mean value theorem to find $|D\phi(y) - D\phi(z)| \le \|D^2\phi\|_{C(U)}|z-y|$. With this and (5.4), one obtains

$$\left|vg_h(y) - D\phi(z)^T D\phi(z)\right| \le ch^k + c_1 \left\|D^2\phi\right\|_{C(U)} |z-y| \le \frac{1}{2} c_0^2.$$

The conclusion follows from (5.5) with $D\phi(z)^T D\phi(z) \ge c_0^2$ and the triangle inequality. ∎

LEMMA 5.4. *For all $y \in U$ $s,t \in \mathbb{R}$ with $G_s(y) \in U$ there holds*

$$\left|M(t-s) - \phi_y'(s; t-s)\right| \le ch^k \left|t-s\right| \quad \textit{for all } M \in \partial_c \phi_y(s).$$

Proof. As ϕ_y is continuous and piecewise polynomial (as a function of s), the results of Lemmas 3.1 and 3.2 hold for ϕ_y. By (2.3), one has the inclusion $\partial_c \phi_y(s) \subseteq -\partial_c \phi_h\big(G_s(y)\big) g_h(y)$. By Lemma 3.1, one can write $\phi_y'(s; t-s) = M_1 g_h(y)(t-s)$ for some $M_1 \in \partial_c \phi_h(z)$ with $z := G_s(y)$. Let $M = M_2 g_h(y) \in \partial_c \phi_h(z) g_h(y)$ be arbitrary. Then,

$$\begin{aligned} M(t-s) - \phi_y'(s; t-s) &= (M_2 - M_1) g_h(y)(t-s) \\ &= \Big(\big(M_2 - D\phi(z)\big) + \big(D\phi(z) - M_1\big)\Big) g_h(y)(t-s). \end{aligned}$$

Using (5.4) with the triangle inequality and (3.16) concludes the proof. ∎

A smooth approximation of ϕ_y, $y \in U$, (with respect to s) is given by

$$\tilde{\phi}_y(s) := \phi\big(G_s(y)\big) = \phi\big(x - sg_h(y)\big). \tag{5.6}$$

LEMMA 5.5. *For all $y \in U$ and $s,t \in \mathbb{R}$ with $\operatorname{conv}\{G_s(y), G_t(y)\} \subset U$ there holds*

$$\left|\phi_y(t) - \phi_y(s) - \phi_y'(s; t-s)\right| \le \left(ch^k + c\left\|\phi\right\|_{C^{1,1}(U)} \left|t-s\right|\right) \left|t-s\right|.$$

Proof. With $e := t - s$, one has

$$\begin{aligned} \phi_y(t) - \phi_y(s) - \phi_y'(s; t-s) &= \int_{(s,t)} \phi_y'(r; e)\, dr - \phi_y'(s; e)\, |e| \\ &= \int_{(s,t)} \phi_y'(r; e) - \phi_y'(s; e)\, dr. \end{aligned}$$

The integrand is rearranged as a sum of three differences,

$$\begin{aligned} \phi_y'(r; e) - \phi_y'(s; e) &= \phi_y'(r; e) - \tilde{\phi}_y'(r; e) + \tilde{\phi}_y'(r; e) - \tilde{\phi}_y'(s; e) \\ &\quad + \tilde{\phi}_y'(s; e) - \phi_y'(s; e) \\ &=: A + B + C. \end{aligned}$$

The term B is smooth; hence, the mean value theorem implies $B = \tilde{\phi}''_y(\rho; e)(\rho - s)$ for some $\rho \in (s,t)$. Therefore, $|B| \le c|t-s|$.

For the term C, one writes $\tilde{\phi}'_y(s;e) = \partial\phi\big(G_s(y)\big)g_h(y)e$ and $\phi'_y(s;e) = Mg_h(y)e$ for some $M \in \partial_c\phi_h\big(G_s(y)\big)$. Using (5.6), (3.16), and (5.4), one gets $|C| \le ch^k$. Similarly, one obtains $|A| \le ch^k$. □

5.2. The convergence theorem for the inner iteration. Consider the tubular neighborhood $U_\epsilon = \big\{x \in \Omega \;\big|\; |\bar{d}_h(x)| \le \epsilon\big\}$, $\epsilon > 0$, of Γ_h. It will be shown that there is a (small) positive value ϵ and a (small) positive radius r_0 such that the generalized Newton iteration (2.4) for (5.1) with initial value $s = 0$ converges for all points $x \in U_\epsilon$ and all $y \in B(x, r_0)$. There holds

$$|\phi_y(0)| = |\phi_h(x)| \le c_1\epsilon \quad \text{for all } x \in U_\epsilon, y \in U. \tag{5.7}$$

The auxiliary parameter $r > 0$ is used to obtain the bounds for the parameters α, β, γ, δ in Theorem 2.3 by choosing r sufficiently small.

Choose $r_0 > 0$ and h_0 so small that

$$ch_0^k + c_1\|D^2\phi\|_{C(U)}2r_0 \le \tfrac{1}{2}c_0^2 \quad \text{and} \quad \{x+\delta \mid x \in U_{r_0}, |\delta| < r_0\} \subset U. \tag{5.8}$$

From (5.4), one gets $|G_s(y) - x)| = |-sg_h(y)| \le |s|\max\{1, c_1 + ch^k\}$. Choose r and h_0 such that $h_0 \le r$ and

$$r\max\{1, c_1 + ch_0^k\} \le r_0. \tag{5.9}$$

Owing to (5.8) and (5.9), there holds

$$G_s(y) \in B(x, r_0) \subset U \quad \text{for all } x \in U_{r_0}, y \in B(x, r_0), s \in B(0, r). \tag{5.10}$$

By Lemma 5.3 and (5.8), one has

$$\big|\partial_c\phi_y(s)^{-1}\big| \le \beta := 2c_0^{-2} \quad \text{for all } x \in U_{r_0}, y \in B(x, r_0), s \in B(0, r). \tag{5.11}$$

Possibly reducing h_0 and r, one can ensure

$$ch_0^k \le \frac{1}{5}\beta^{-1} =: \gamma, \quad ch_0^k + c\,\|\phi\|_{C^{1,1}(U)}\,2r \le \frac{3}{10}\beta^{-1} =: \delta. \tag{5.12}$$

From this and the Lemmas 5.4 and 5.5, one obtains for all $x \in U_{r_0}$, $y \in B(x, r_0)$, and $s, t \in B(0, r)$ that

$$\begin{aligned} \big|M(t-s) - \phi'_y(s; t-s)\big| &\le \gamma\,|t-s| \quad \text{for all } M \in \partial_c\phi_y(s), \\ \big|\phi_y(t) - \phi_y(s) - \phi'_y(s; t-s)\big| &\le \delta\,|t-s|\,. \end{aligned}$$

One checks $\alpha := \beta(\gamma + \delta) = \frac{1}{2} < 1$. Now the (last) free parameter ϵ is chosen such that

$$0 < \epsilon \le r, \quad \beta c_1\epsilon \le \frac{1}{2}r. \tag{5.13}$$

By (5.7) and (5.13), $\beta|\phi_y(0)| \le \beta c_1\epsilon \le \frac{1}{2}r = r(1-\alpha)$. Applying Theorem 2.3 yields

THEOREM 5.6. *With the positive constants $\epsilon \le r < r_0$ as above, the following holds for all all points $x \in U_\epsilon$ and all $y \in B(x, r_0)$: There is a unique solution $s(y) \in \overline{B}\big(0, r\big)$ to (5.1). In particular, $x - s(y)|g_h(y)|n_h(y) \in \Gamma_h$. The Newton method*

(2.4) *for ϕ_y with initial value $s = 0$ converges to $s(y)$ at least linearly with the rate $\alpha = \frac{1}{2}$. Moreover, one has the a posteriori error estimate*

$$|s_k - s(y)| \le |s_k - s_{k-1}|, \quad k \in \mathbb{N}. \tag{5.14}$$

COROLLARY 5.7. *The convergence in Theorem 5.6 is quadratic locally around the solution of* (5.1).

Proof. This follows immediately from Theorem 2.4, Lemma 3.2, and Theorem 5.6. ∎

5.3. The convergence theorem for the nested iteration. The proof of convergence is a straightforward application of Banach's contraction mapping principle. It is similar to the proof of [17, Thm. 5.1]. We extend the latter by considering the inner iteration (5.1) instead of an "oracle" that yields $s(y)$. Thus, Theorem 5.6 can be used, which allows us to avoid an implicit function theorem for Lipschitz functions in the proof (which is required in [17]).

LEMMA 5.8. *Let $\epsilon \le r < r_0$ be as in Theorem 5.6 and $x \in U_\epsilon$ be arbitrary. Then, either $s(y)\phi_h(x) > 0$ or $s(y) = 0 = \phi_h(x)$ for each $y \in B(x, r_0)$.*

Proof. Let $l(t) := (1-t)G(y) + tx$, $t \in \mathbb{R}$. Due to (5.8) and (5.10), there holds $l(t) \in B(x, r_0)$, $t \in [0,1]$. As $\phi_h(l(0)) = 0$ and $\phi_h(l(1)) = \phi_h(x)$, one gets

$$\phi_h(x) = \int_0^1 D\phi_h\big(l(\tau)\big)^T \big(x - G(y)\big)\, d\tau = s(y) \int_0^1 D\phi_h\big(l(\tau)\big) g_h(y)\, d\tau.$$

The conclusion follows from Lemma 5.3 because the integrand is bounded from below by $\frac{1}{2}c_0^2 > 0$. ∎

THEOREM 5.9. *Let $\epsilon < r \le r_0$ be as in Theorem 5.6. In addition to* (5.8)*, let r_0 be so small that $2cr_0 \le 1 - \tilde{\delta} - cr_0$, where $\tilde{\delta} < 1$ is defined in* (3.14)*; c represents constants from the proof below. Then, for any $x \in U_\epsilon$, the iterates of the fixed point iteration* (5.3) *are well-defined, remain in $B(x, r_0)$, and converge linearly to the unique solution of* (5.2) *in $B(x, r_0)$.*

Proof. Let $x \in U_\epsilon$ and $y, z \in B(x, r_0)$ be arbitrary. By Theorem 5.6, there are unique solutions $s(y)$ and $s(z)$ of (5.1) in $B(0, r)$, which makes G in (5.2) well-defined on $B(x, r_0)$. By (5.10), one has $G(y), G(z) \in B(x, r_0)$—hence, G maps $B(x, r_0)$ to itself. Let $\delta := G(z) - G(y) = s(y)g_h(y) - s(z)g_h(z)$. By Lemma 5.8, $s(y)$ and $s(z)$ have the same sign; the triangle inequality gives

$$\big|s(y)\,|g_h(y)| - s(z)\,|g_h(z)|\big| \le |s(y)g_h(y) - s(z)g_h(z)| = |\delta|. \tag{5.15}$$

From $\delta = s(y)|g_h(y)|\big(n_h(y) - n_h(z)\big) + \big(s(y)|g_h(y)| - s(z)|g_h(z)|\big)n_h(z)$ and (5.15), one obtains

$$|\delta|^2 \le |s(y)|\,|g_h(y)|\,|n_h(y) - n_h(z)|\,|\delta| + |\delta|\,|(n_h(z), \delta)|.$$

Using $|s(y)| \le r$, (5.4), and (3.12), this implies

$$|\delta| \le cr\,|z - y| + |(n_h(z), \delta)|.$$

Using (3.12), (3.13), and $|G(z) - z| \le 2r_0$,

$$\begin{aligned} |(n_h(z), \delta)| &\le |(n_h(z) - n_h(G(z)), \delta)| + |(n_h(G(z)), \delta)| \\ &\le c2r_0\,|\delta| + \tilde{\delta}_{G(z)}\,|\delta| + c_{G(z)}\,|\delta|^2 \\ &\le (cr_0 + \tilde{\delta}_{G(z)})\,|\delta|. \end{aligned}$$

Altogether, one obtains

$$|\delta| \leq \frac{cr}{1-\tilde{\delta}-cr_0}\,|z-y| \leq \frac{1}{2}\,|z-y|\,.$$

The conclusion follows from Banach's contraction mapping principle. □

6. Implementation details. Both the Newton method for (4.1) and the Newton method for the inner iteration (5.1) can be made more robust by damping big Newton steps. This is important on coarse meshes and in the context of redistancing, where one considers distorted level set functions (as opposed to distance-like level set functions). The following simple scheme based on the Armijo rule (with backtracking), cf. [14], is used in the implementation: Let $f(x)$ be one of the functions $F(\hat{x})$, $\phi_y(s)$. Using the current iterate x_n, the (undamped) Newton step δ_n, and $\alpha_i = 2^{-i}$, $i \in \mathbb{N}_0$, find the smallest i satisfying

$$|f(x_n+\alpha_i\delta_n)|^2 \leq |f(x_n)|^2 + 0.02 f(x_n)^T \partial_{\mathrm{c}} f(x_n)\delta_n.$$

The next Newton iterate is defined as $x_{n+1} = x_n+\alpha_i\delta_n$. This requires a few additional evaluations of f, but not of $\partial_{\mathrm{c}} f$. Hence, this damping strategy is computationally cheap. As it always considers $\alpha_0 = 1$ first, the quadratic convergence of the Newton method close to the solution is not compromised.

A common sub-problem of (4.1) and (5.1) is the computation of a simplex $S_{n+1} \in \mathcal{T}$ which contains the next (tentative) iterate x_{n+1}, cf. Remark 4.3. Several approaches to this problem are possible, which differ in speed and memory consumption. For example, as such a simplex S_n is known for x_n, one could search x_{n+1} in increasingly larger neighborhoods of S_n in $\mathcal{T}$. The size of such neighborhoods grows rapidly, at least for $N \geq 3$. On the other hand, additional damping may be necessary if the size of the neighborhoods is insufficient.
In the experiments below, the coarse meshes have a regular block structure. A cuboid C containing x_{n+1} can be located easily, i. e. with a small, constant number of flops. All cuboids of the coarse mesh are partitioned into six tetrahedra using the Kuhn triangulation ($N = 3$). The tetrahedra in C are searched sequentially for x_{n+1}. On average, this requires three containment tests to locate a tetrahedron S^0_{n+1} in the coarse mesh. Then, given a tetrahedron S^i_{n+1} on refinement level i, which contains x_{n+1}, its children are searched, and S^{i+1}_{n+1} is a child containing x_{n+1}. As the number of children is at most eight, on average $3+4l$ containment tests are required to find the desired tetrahedron S_{n+1} with $x_{n+1} \in S_{n+1}$, where l is the number of refinement levels of the mesh. Thus, the number of containment tests grows as $\mathcal{O}(l)$.
The search strategy can also be applied if the coarsest mesh is not block structured. In this case, the linear search on the coarsest refinement level must be replaced by the search in a space-partitioning data structure like an octree or a kd-tree, cf. [3, 1]. This salvages the average number of $\mathcal{O}(l)$ containment tests per search point.

As a further optimization, before starting the recursive search of S_{n+1}, one tests, whether $x_{n+1} \in S_n$, that is $S_{n+1} = S_n$. This occurs frequently, particularly on fine meshes. Related to this, the gradients of the shape functions and the values of ϕ_h on S_n are cached. Both, the Newton method and the nested iteration, are accelerated by this caching. The benefit is higher for the Newton method than for the nested iteration.

l	e_∞	factor	# calls
4	0.01378*	–	24688
	0.002708*		
5	4.778e-4	28.8	99644
	4.778e-4*	5.66	
6	6.692e-05	7.13	403620
7	8.697e-06	7.69	1606268
8	1.169e-06	7.43	6411644
9	1.449e-07	8.06	25671768

TABLE 7.1

Redistancing: Error in the degrees of freedom at Γ_h. (No convergence for some points, cf. Table 7.2 and Table 7.3; the first error is for the Newton method, the second for the nested iteration.)*

7. Numerical experiments. The Newton method for (4.1) and the nested iteration for (5.2) are compared in two experiments. In Section 7.1, the redistancing problem of [17, 7] is reconsidered; a perturbed level set function for a torus is given, and the value of d_h is required in all degrees of freedom close to Γ_h. In Section 7.2, higher order numerical integration is performed on Γ_h. For example, such integrals are required for the methods in [6, 8, 9]. In both setups, the run time of the two algorithms is considered as a measure of the overall performance. Furthermore, the number of (outer) iterations is given, and the time for searching a simplex S_{n+1} containing the next iterate x_{n+1}, cf. Section 6.

In both experiments, the domain Ω is a cube. The coarse mesh has a regular block structure. Each cuboid is partitioned into six tetrahedra by inserting a space diagonal and its projections on the faces of the cube. The resulting Kuhn triangulation is refined using a red-green refinement algorithm, cf. [2, 10]. Refinement is applied to all tetrahedra, which are intersected by Γ_h until the required refinement level l is reached. As the discrete level set functions $\phi_h \in X_h^2$, the nodal interpolants of the level set functions ϕ given below are used. The recovered gradient $g_h \in X_h^2$ of ϕ_h is computed with (3.11). The stopping criterion for the Newton method is

$$|F(x_n)| \leq \text{1e-8}.$$

The stopping criterion for the nested iteration is

$$|\phi_h(x_n)| \leq \text{5e-9} \quad \text{and} \quad |x_n - x_{n-1}| \leq \text{1e-8}.$$

7.1. Redistancing. Let $\Omega = (-1,1)^3$ and Γ be the torus which is the level set of the perturbed distance function

$$\phi(x) = \left(\sqrt{x_3^2 + \left(\sqrt{x_1^2 + x_2^2} - R \right)^2} - r \right) \left(9 + 4\cos(\alpha x_1 x_2 / |x|) \right)$$

with $R = 0.4$, $r = 0.2$, and $\alpha = 50$. This is the setup with the strongest perturbation considered in [17]; the perturbation preserves the torus as zero level, but introduces a large, oscillating gradient. The mesh width at Γ_h on refinement level l is $h(l) = \frac{2}{3} \cdot 2^{-l}$.

Let $N(\mathcal{T}_h^\Gamma)$ be the set of all vertices and edge-barycenters of the tetrahedra which are cut by Γ_h. The error measure $e_\infty = \max_{v \in N(\mathcal{T}_h^\Gamma)} |d_h(x) - d(x)|$ which is also used

	time [s]		iterations				
l	t_d	t_{loc}	average	min	max	no conv.	#damping
4	0.683	0.495	4.195	1	50	134	51390
5	1.14	0.755	3.395	1	7	0	16
6	5.99	4.20	3.132	1	6	0	0
7	19.5	14.2	2.824	1	6	0	0
8	78.8	59.3	2.539	1	5	0	0
9	334	260	2.452	1	5	0	0

TABLE 7.2
Redistancing: Performance of the Newton method.

	time [s]		outer iterations				inner iterations	
l	t_d	t_{loc}	average	min	max	no conv.	average	#damping
4	1.80	1.25	9.189	2	50	335	2.396	117542
5	1.79	1.18	6.073	2	50	28	1.502	11050
6	6.33	4.49	4.836	2	11	0	1.207	9168
7	19.1	14.2	4.110	2	7	0	1.034	3136
8	78.2	60.5	3.634	2	5	0	0.9435	524
9	331	263	3.279	1	4	0	0.9672	0

TABLE 7.3
Redistancing: Performance of the nested iteration.

in [17] is reported in Table 7.1. The Newton method and the nested iteration produce the same result with a relative error of less than 1e-4 for $l \geq 5$. The error is close to the one reported in [17, Tab. 8.3]; it shows a $\mathcal{O}(h^3)$ behavior on the finer meshes. We cannot expect to reproduce the results of [17] exactly because a different (simpler) gradient recovery method is used in our paper. As in [17], there are convergence problems for the nested iteration for $l \in \{4, 5\}$. The Newton method converges in all points on the levels $l \geq 5$. For $l = 4$, the Newton method has convergence problems as well.

In Table 7.2 and Table 7.3, the following performance data are shown:

l the refinement level,

t_d the total time (in seconds) spent in the method,

t_{loc} the part of t_d (in seconds) spent locating the tetrahedron S_{n+1} on level l containing x_{n+1},

(outer) iterations the average/min./max. number of (outer) iterations per degree of freedom (dof) and the number of dof without convergence after 50 iterations,

inner iterations the average number of inner iterations per outer iteration (only in Table 7.3),

#damping the total number of rejected steps-sizes of the Newton method (in the inner iteration).

The number of distance computations ('#calls' in Table 7.1) scales with a factor 4 when l is incremented as there is one computation per dof close to Γ_h. The time required by the Newton iteration and the nested iteration is comparable for $l > 6$. It

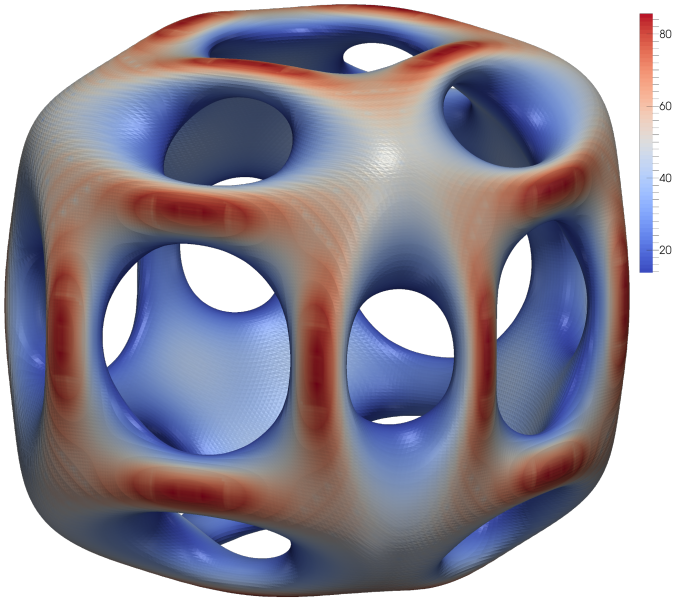

FIG. 7.1. *Area computation:* Γ_h*; the grayscale (color in the online version) indicates* $|D\phi_h|$.

is dominated by t_{loc} and also scales with a factor of about 4. Both methods require fewer iteration to converge on the finer levels, but the effect of this on t_d is dominated by the computation of S_{n+1} as well. The Newton method requires no damping steps for $l \geq 6$, whereas the inner iteration of the nested iteration requires some damping up to $l = 8$. The Newton method converges in all dof for $l \geq 5$, but the nested iteration converges only for $l \geq 6$. In this sense, the Newton method is more robust. However, the error e_∞ of the nested iteration is hardly affected by this for $l \in \{4, 5\}$, whereas e_∞ is bigger for the Newton method on level 4, comparable to $h(4) \approx 0.04$. For $l \in \{4, 5\}$, the nested iteration is slower than the Newton method. This is explained by the large number of damping steps, each of which requires a search for S_{n+1}. This is visible in the larger values of t_{loc}.

7.2. Area computation. Let $\Omega = (-3, 3)^3$ and Γ be the zero level of the following polynomial of degree 12,

$$f(x) = p(x_1, x_2, x_3) \cdot p(x_2, x_3, x_1) \cdot p(x_3, x_2, x_1) - 3 \quad \text{with}$$
$$p(x) = (x_1^2 + x_2^2 - 1.8^2)^2 + (x_3^2 - 1)^2,$$

which is taken from [4].[1] The mesh width at Γ_h on refinement level l is $h(l) = 6 \cdot 2^{-l}$. The discrete level set function ϕ_h is the nodal interpolant of ϕ in X_h^2. On the global regular refinement $\tilde{\mathcal{T}}_h$ of $\mathcal{T}_h$, the function ϕ_h can be interpreted as a finite element in $X_h^1(\tilde{\mathcal{T}}_h)$. This function and its zero level are denoted as $\hat{\phi}_h$ and $\hat{\Gamma}_h$. For any $T \in \mathcal{T}_h$, both $\hat{\phi}_h|_T$ and $\hat{\Gamma}_h \cap T$ can be evaluated on a tetrahedron by tetrahedron basis without actually computing (and storing) $\tilde{\mathcal{T}}_h$ as a whole.

[1] We correct a typographical error in [4, Example 5.3] which caused some multiplication signs to be printed as plus signs. The resulting zero level, Fig. 7.1, is then in agreement with the one shown in [4, Fig. 5].

	Γ_h		$\hat{\Gamma}_h$		$\lvert D\phi_h\rvert$		curvature	
l	area	error	area	error	min	max	max	#calls
6	74.6729	9.83e-2	74.6117	3.70e-2	13.4	83.9	44.4	2117304
7	74.5803	5.62e-3	74.5826	8.00e-3	13.8	89.5	28.0	8462496
8	74.5750	3.73e-4	74.5764	1.73e-3	13.8	89.8	23.1	33839400
9	74.5747	1.84e-5	74.5751	4.84e-4	13.8	89.3	21.7	135454704
10	74.5746	—	74.5748	1.24e-4	13.8	89.3	20.9	541725408

TABLE 7.4
Area computation.

The piecewise linear approximation $\hat{\Gamma}_h$ of Γ is used as the domain of a parametrization of Γ_h via $p_h|_{\hat{\Gamma}_h}\colon \hat{\Gamma}_h \to \Gamma_h$. With the transformation rule for integrals, the integral of a function f over Γ_h can be represented as

$$\int_{\Gamma_h} f(x)\,d\sigma(x) = \int_{\hat{\Gamma}_h} f\circ p_h(x) J(x)\,d\sigma(x), \quad J(x) = \sqrt{\big(\partial p_h(x)U(x)\big)^T \partial p_h(x)U(x)},$$

where $U(x) \in \mathbb{R}^{3\times 2}$ is an orthogonal matrix that spans the tangential space of $\hat{\Gamma}_h$ at x; $\partial p_h(x)$ is the Jacobian of p_h, which can be computed in closed form from (1.1) as a function of $p_h(x)$, $n_h\big(p_h(x)\big)$, and $\partial g_h\big(p_h(x)\big)$. The right-hand side is accumulated tetrahedron by tetrahedron using a fifth order accurate quadrature rule with positive weights and seven quadrature points in the interior of the reference triangle. As the focus of the experiment is on p_h, the simple function $f(x) = 1$ is used, which gives the area of Γ_h up to quadrature and approximation errors.

Table 7.4 shows the area of Γ_h and of $\hat{\Gamma}_h$. The errors are computed with respect to $|\Gamma_h|$ for $l = 10$. The difference between the Newton method and the nested iteration is less than 1e-7 for $l \geq 6$. The error of $|\hat{\Gamma}_h|$ shows an $\mathcal{O}(h^2)$-behavior. The error of $|\Gamma_h|$ shows an $\mathcal{O}(h^4)$-behavior, which is better than the expected $\mathcal{O}(h^3)$-behavior. This is probably due to the very simple integrand.
The minimum and maximum of $|D\phi_h|$ is sampled in the quadrature points on Γ_h and shown in column six and seven. They show that the level set function is steep. The maximum of the curvature radius of Γ_h (sampled over the quadrature points) is shown in column eight of Table 7.4. Together with the fact that the domain has a side-length of 6, the observed maximum curvature of circa 20 explains the relatively high refinement level $l \geq 6$ used in the computations: For the mesh width, there holds $h(5) \approx 0.19$, $h(6) \approx 0.09$. Both values are bigger than the minimum curvature radius of about 0.05. The methods converge nevertheless because the points to which they are applied lie on $\hat{\Gamma}_h$ such that their distance to Γ_h is $\mathcal{O}(h^2)$, that is 0.035 and 0.0088. Neither the Newton method nor the nested iteration produce useful results for $l = 5$. Compared to the redistancing experiment, the number of distance computations is an order of magnitude larger. There are seven quadrature points per surface triangle; this makes the caching mechanism for S_{n+1} explained in Section 6 particularly efficient. This manifests itself in a smaller value of t_{loc}/t_d in Table 7.2 compared to Table 7.5 and in Table 7.3 compared to Table 7.6.

The tables 7.5 and 7.6 show the same performance data as the tables 7.2 and 7.3. The columns 'no conv.' and '#damping' are absent from Table 7.5 as the Newton method converges in all points and no step size is rejected in this experiment.

The time required by both methods scales with a factor between 3 and 4 if l

	time [s]		iterations		
l	t_d	t_{loc}	average	min	max
6	4.946	1.46	2.459	1	5
7	16.36	3.50	2.019	0	3
8	45.90	5.87	1.835	0	3
9	148.4	12.2	1.549	0	2
10	459.1	30.8	1.166	0	2

TABLE 7.5
Area computation: Performance of the Newton method.

	time [s]		outer iterations				inner iterations	
l	t_d	t_{loc}	average	min	max	no conv.	average	#damping
6	8.152	2.24	3.367	1	50	42	2.993	79242
7	16.19	4.00	2.707	1	4	0	1.477	1296
8	39.03	6.74	2.228	1	3	0	1.130	132
9	120.2	14.1	1.998	1	3	0	0.9651	0
10	434.7	39.5	1.990	1	2	0	0.8026	0

TABLE 7.6
Area computation: Performance of the nested iteration.

is incremented by 1, whereas the number of calls scales with the factor 4. This is explained by the decrease of the average iteration number on finer meshes. Although this effect is also present in the redistancing experiment, it is dominated there by t_{loc}. In the present experiment, the effect of t_{loc} is reduced by the higher efficiency of the caching mechanism for S_{n+1}. This makes the nested iteration about 10 per cent faster than the Newton iteration on the finer meshes.
The Newton method is slightly more robust in the sense that it converges in all points for $l \geq 6$. The nested iteration fails in 42 points for $l = 6$. However, the effect on the area computation is negligible. This could be due to the fact that the inner iteration converged in the final step of these 42 computations, returning some point on Γ_h "close" to the true base point.

8. Acknowledgement. I thank Arnold Reusken for reading and discussing the initial version of this paper. His feedback led to an improved presentation.

REFERENCES

[1] Jon Louis Bentley, *Multidimensional binary search trees used for associative searching*, Commun. ACM, 18 (1975), pp. 509–517.
[2] Jürgen Bey, *Finite-Volumen- und Mehrgitter-Verfahren für elliptische Randwertprobleme*, Advances in Numerical Mathematics, Teubner Verlag, 1998.
[3] Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf, *Computational Geometry*, Springer-Verlag, Berlin, revised ed., 2000. Algorithms and applications.
[4] Klaus Deckelnick, Gerhard Dziuk, Charles M. Elliott, and Claus-Justus Heine, *An h-narrow band finite-element method for elliptic equations on implicit surfaces*, IMA Journal of Numerical Analysis, 30 (2010), pp. 351–376.
[5] Alan Demlow, *Higher-order finite element methods and pointwise error estimates for elliptic problems on surfaces*, SIAM J. Numer. Anal., 47 (2009), pp. 805–827.

[6] Alan Demlow and Gerhard Dziuk, *An adaptive finite element method for the Laplace-Beltrami operator on implicitly defined surfaces*, SIAM J. Numer. Anal., 45 (2007), pp. 421–442.

[7] Patrick Esser and Jörg Grande, *An accurate and robust finite element level set redistancing method*, IMA Journal of Numerical Analysis, advance online publication 2014, doi: 10.1093/imanum/dru052.

[8] Jörg Grande, *Finite element discretization error analysis of a general interfacial stress functional*, IGPM-Report 380, IGPM, RWTH Aachen University, 2014, accepted for publication in SIAM J. Numer. Anal.

[9] Jörg Grande and Arnold Reusken, *A higher order finite element method for partial differential equations on surfaces*, IGPM-Report 403, IGPM, RWTH Aachen University, 2014, submitted.

[10] Sven Gross, Jörg Peters, Volker Reichelt, and Arnold Reusken, *The DROPS package for numerical simulations of incompressible flows using parallel adaptive multigrid techniques*, IGPM-Report 211, IGPM, RWTH Aachen University, 2002.

[11] Michael Hintermüller, *Semismooth Newton methods and application*, Oberwolfach-Seminar on "Mathematics of PDE-Constrained Optimization", (2010), pp. 1–72.

[12] Vaithilingam Jeyakumar and Dinh The Luc, *Nonsmooth Vector Functions and Continuous Optimization*, vol. 10 of Springer Optimization and Its Applications, Springer, 2008.

[13] John M. Lee, *Introduction to Smooth Manifolds*, Graduate texts in mathematics, Springer, 2012.

[14] Jorge Nocedal and Stephen J. Wright, *Numerical Optimization*, Springer Series in Operations Research and Financial Engineering, Springer, New York, 2nd ed., 2006.

[15] Li Qun Qi and Defeng Sun, *A survey of some nonsmooth equations and smoothing Newton methods*, in Progress in optimization, vol. 30 of Appl. Optim., Kluwer Acad. Publ., Dordrecht, 1999, pp. 121–146.

[16] Li Qun Qi and Jie Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–367.

[17] Arnold Reusken, *A finite element level set redistancing method based on gradient recovery*, SIAM J. Numer. Anal., 51 (2013), pp. 2723–2745.

[18] Michael Ulbrich, *Semismooth Newton Methods for Variational Inequalities and Constrained Optimization Problems in Function Spaces*, Society for Industrial and Applied Mathematics, 2011.