An accurate and robust Eulerian finite element method for partial differential equations on evolving surfaces

Hauke Sass[*] and Arnold Reusken[†]

[*] Institut für Geometrie und Praktische Mathematik, RWTH-Aachen University, D-52056 Aachen, Germany; email: sass@igpm.rwth-aachen.de
[†] Institut für Geometrie und Praktische Mathematik, RWTH-Aachen University, D-52056 Aachen, Germany; email: reusken@igpm.rwth-aachen.de

AN ACCURATE AND ROBUST EULERIAN FINITE ELEMENT METHOD FOR PARTIAL DIFFERENTIAL EQUATIONS ON EVOLVING SURFACES

HAUKE SASS[*] AND ARNOLD REUSKEN[†]

Abstract. In this paper we present a new Eulerian finite element method for the discretization of scalar partial differential equations on evolving surfaces. In this method we use the restriction of standard space-time finite element spaces on a fixed bulk mesh to the space-time surface. The structure of the method is such that it naturally fits to a level set representation of the evolving surface. The higher order version of the method is based on a space-time variant of a mesh deformation that has been developed in the literature for stationary surfaces. The discretization method that we present is of (optimal) higher order accuracy for smoothly varying surfaces with sufficiently smooth solutions. Without any modifications the method can be used for the discretization of problems with topological singularities. A numerical study demonstrates both the higher order accuracy for smooth cases and the robustness with respect to toplogical singularities.

Key words. surface partial differential equation, trace finite element method, space-time finite element method, unfitted finite element method

1. Introduction. Motivated by applications in, for example, multiphase flow and computational biology, in the past decade there has been a strongly growing interest in the development and analysis of numerical methods for partial differential equations (PDEs) on (evolving) surfaces. For an overview of recent developments in this research field we refer to [6, 2]. The main focus of these two review papers is on numerical methods for scalar PDEs on smooth *stationary* surfaces. There are, however, also many papers in which discretization methods for scalar PDEs on *evolving* surfaces are introduced or analyzed. The present paper contributes to this field of finite element methods for scalar PDEs on evolving surfaces.

In this research field several interesting methods have already been treated. The probably most prominent method is the evolving surface finite element method (ES-FEM) [6, 7, 19]. This elegant method is very popular and a complete error analysis has been developed. A key feature of this method is that it is based on a Lagrangian approach in which a triangulation of the initial surface and a corresponding finite element space are transported along the flow lines of a (given) velocity field. This makes the method very attractive for problems with smoothly and slowly varying surfaces. On the other hand, it is well-known that such Lagrangian techniques have drawbacks if the geometry of the surface is strongly varying on the relevant time scales or if there are topological singularities, e.g., merging or splitting phenomena. We are not aware of any paper in which the ESFEM is applied to a surface PDE with a topological singularity. Another class of finite element methods is based on an Eulerian approach and has been introduced in [24, 23]. These so-called trace (or cut) finite element methods (TraceFEM or CutFEM) are based on restriction of standard space-time finite element spaces on a fixed bulk mesh to the space-time surface. The structure of TraceFEM is such that it naturally fits to a level set representation of the evolving surface. Such level set representations are known to be very attractive for problems with strongly varying surface geometries or with topological singularities. Hybrid ver-

[*]Institut für Geometrie und Praktische Mathematik, RWTH-Aachen University, D-52056 Aachen, Germany; email: `sass@igpm.rwth-aachen.de`

[†]Institut für Geometrie und Praktische Mathematik, RWTH-Aachen University, D-52056 Aachen, Germany; email: `reusken@igpm.rwth-aachen.de`

sions of this Eulerian approach have been introduced in [15, 22]. The method treated in [15] is based on a characteristic-Galerkin, i.e., Lagrangian, formulation combined with a piecewise linear CutFEM in space. The fully Eulerian method introduced in [22] is based on a combination of TraceFEM in space and finite differences for the time discretization. In the numerical experiments presented in these papers the methods are applied to problems with smoothly and slowly varying surfaces. In [22] not only linear finite elements but also higher order ones are considered.

In the present paper we return to the Eulerian space-time technique introduced in [24, 23]. We extend the method treated in those papers in the following two directions. Firstly, we will introduce *a higher order* space-time TraceFEM. In [24, 23] only piecewise linears (in space and time) are used and the issue of geometry approximation is not considered, The development of the higher order technique is based on the parametric TraceFEM introduced for stationary surfaces in [20] and analyzed in [12]. The unfitted space-time finite element technique that we propose in this paper is based on very similar ideas as used for scalar PDEs on a moving domain in [17]. Secondly, in the method presented in this paper we include a space-time variant of the so-called *normal derivative volume stabilization* that has been developed for TraceFEM applied to PDEs on stationary surfaces, cf. [3, 12]. This stabilization is important, in particular for higher order finite elements, to control the conditioning of the stiffness matrix.

Using these extensions results in a method with the following features. The method is of Eulerian type, using a fixed bulk triangulation and corresponding standard space-time (tensor product) finite element spaces. As input for the method one needs (only) a level set representation of the evolving surface. If the level set function approximation is sufficiently accurate (made precise further on in the paper) and higher order bulk space-time finite element spaces are used, the resulting surface PDE discretization method is also of (optimal) higher order accuracy for smoothly varying surfaces with sufficiently smooth solutions. Without any modifications the method can be used for the discretization of problems with topological singularities. We are not aware of any paper in which this combination of properties, i.e., higher order for smooth cases and robustness w.r.t. topological singularities, is demonstrated for a discretization method for surface PDEs.

In this paper we introduce the method, discuss important properties of the method and present an extensive numerical study that demonstrates the features mentioned above. We do not include a stability or error analysis. An error analysis for the lowest order case (piecewise planar surface approximation and P_1 finite elements) is presented in [27] and will be further treated in a forthcoming paper. Implementation of the method is not straightforward due to the use of a space-time framework and a particular space-time mesh transformation (for the higher order case). An implementation, however, is available in the open source software `ngsxfem` [21], an add-on package to the finite element library `NGSolve\Netgen`, see [28, 29].

The remainder of the paper is organized as follows. In Section 2 we explain the parabolic model problem that we consider and a well-posed space-time variational formulation of this problem. The space-time TraceFEM is explained in Section 3. Several aspects of the method are treated. We start with the level set representation of the evolving surface, needed as input for the discretization method. We then outline the space-time mesh deformation that we use for the higher order variant of our discretization method. Several implementation aspects such as, for example, the efficient quadrature applied to the space-time integrals are addressed. In Section 4 we

present results of numerical experiments, both for examples with smoothly evolving surfaces and surfaces with topological singularities. These results illustrate the higher order convergence for the smooth case and the robustness of the method with respect to topological singularities.

2. The model problem. Let $\Omega \subset \mathbb{R}^3$, $T > 0$ and $\mathbf{w} : \Omega \times [0, T] \to \mathbb{R}^3$ a given smooth velocity field. We consider a closed, connected and orientable C^2-hypersurface $\Gamma(t) \subset \Omega$, $t \in [0, T]$, that is is advected $\mathbf{w}$. By $\mathbf{n}$ we denote the outward pointing unit outer normal on $\Gamma(t)$. The smooth space-time manifold is denoted by

$$ S := \bigcup_{t \in [0,T]} \big(\Gamma(t) \times \{t\} \big) \subset \mathbb{R}^4. $$

We recall standard surface differential operators. For a smooth scalar function g on $\Gamma(t)$, with t fixed, a smooth extension to a tubular neighborhood of $\Gamma(t)$ is denoted by g^e. Its spatial and space-time gradients are denoted by ∇ and $\nabla_{(\mathbf{x},t)}$, respectively. The tangential gradient is defined as $\nabla_\Gamma g := (I - \mathbf{n}\mathbf{n}^\intercal)\nabla g^e$. For a vector valued function $\mathbf{g}$ we define the surface divergence $\operatorname{div}_\Gamma \mathbf{g} := \operatorname{tr}\big((I - \mathbf{n}\mathbf{n}^\intercal)\nabla \mathbf{g}^e\big)$. The surface Laplace-Beltrami operator is $\Delta_\Gamma := \operatorname{div}_\Gamma \nabla_\Gamma$. We consider a basic model for convection and molecular diffusion of a surface species. The conservation of mass principle combined with Fick's law for the diffusive flux leads to the parabolic surface partial differential equation

$$
\begin{aligned}
\dot{u} + u \operatorname{div}_\Gamma \mathbf{w} - \mu_d \Delta_\Gamma u &= f \quad \text{on } \Gamma(t), \quad t \in (0, T], \\
u(\cdot, 0) &= 0 \quad \text{on } \Gamma(0),
\end{aligned}
\tag{2.1}
$$

for the scalar unknown function $u = u(\mathbf{x}, t)$. Here $\mu_d > 0$ denotes the constant diffusion coefficient and $\dot{u}$ the material derivative along the velocity field $\mathbf{w}$, which can be expressed as $\dot{u} = \frac{\partial u^e}{\partial t} + \mathbf{w} \cdot \nabla u^e$. The right-hand side f must satisfy the consistency condition $\int_{\Gamma(t)} f \, ds = 0$, $t \in [0, T]$. In the remainder we assume that this condition is satisfied. For details on the derivation of (2.1) we refer to [18, 6, 14].

We outline a weak variational space-time formulation of (2.1), introduced in [24], which is the basis for the finite element method treated in Section 3. For this we first define suitable Hilbert spaces that can be used for a discontinuous in time Galerkin method. For $N \in \mathbb{N}$, let the time interval $[0, T]$ be partitioned into smaller time intervals $I_n := [t_{n-1}, t_n]$, $n \in \{1, \ldots, N\}$, where $0 = t_0 < t_1 < \cdots < t_N = T$. The corresponding space-time surface is denoted by $S^n := \bigcup_{t \in I_n}(\Gamma(t) \times \{t\})$. We define the Hilbert spaces

$$ H := \left\{ v \in L^2(S) : \|\nabla_\Gamma v\|_{L^2(S)} < \infty \right\}, \qquad \|v\|_H^2 := \int_0^T \|v\|_{H^1(\Gamma(t))}^2 \, dt $$

$$ H_n := \left\{ v \in L^2(S^n) : \|\nabla_\Gamma v\|_{L^2(S^n)} < \infty \right\}, \quad \|v\|_{H_n}^2 := \int_{t_{n-1}}^{t_n} \|v\|_{H^1(\Gamma(t))}^2 \, dt, \ 1 \le n \le N. $$

The weak material derivative is based on the linear functional

$$ \langle \dot{u}, \psi \rangle_n := -\int_{t_{n-1}}^{t_n} \int_{\Gamma(t)} u\dot{\psi} + v\psi \operatorname{div}_\Gamma \mathbf{w} \, ds \, dt \quad \text{for all } \psi \in C_0^1(S^n). \tag{2.2} $$

All functions $u \in H_n$ whose weak material derivative $\dot{u}$ define a bounded linear functional on H_n form the subspace $W_n \subset H_n$, i.e.

$$ W_n := \left\{ u \in H_n : \dot{u} \in H_n' \right\}, \quad \|u\|_{W_n}^2 := \|v\|_{H_n}^2 + \|\dot{u}\|_{H_n'}^2. $$

3

For $v \in H$ we define $v_n \in H_n$ as $v_n := v|_{S^n}$, $n = 1, \ldots, N$. The space that we use for the space-time variational formulation is

$$W^b := \left\{ v \in H : v|_{S^n} \in W_n, n = 1, \ldots, N \right\}, \quad \|v\|_{W^b}^2 := \sum_{n=1}^{N} \|v_n\|_{W_n}^2. \qquad (2.3)$$

This space is a Hilbert space. For further properties of these spaces we refer to [24]. For the discontinuous Galerkin method we need the usual jump operator to relate values between subsequent time slabs. For $v \in W^b$ we define

$$v_+^n := v_+(\cdot, t_n) = \lim_{\eta \searrow 0} v(\cdot, t_n + \eta) \in L^2(\Gamma(t_n)), \quad n = 0, \ldots, N-1,$$

$$v_-^n := v_-(\cdot, t_n) = \lim_{\eta \searrow 0} v(\cdot, t_n - \eta) \in L^2(\Gamma(t_n)), \quad n = 1, \ldots, N, \quad v_-^0 := 0,$$

$$[v]^n := v_+^n - v_-^n \in L^2(\Gamma(t_n)), \qquad\qquad\qquad n = 0, \ldots, N-1.$$

Note that $[v]^0 = v_+^0$ since we have by definition $v_-^0 = 0$. On $\Gamma(t)$, $t \in [0, T]$, the L^2 scalar product is denoted by $(\cdot, \cdot)_{\Gamma(t)}$. Space-time integrals and the surface integral on S are related by

$$\int_0^T \int_{\Gamma(t)} g \, \mathrm{d}s \, \mathrm{d}t = \int_S \frac{g}{\alpha} \, \mathrm{d}\sigma, \quad \alpha := \sqrt{1 + (\mathbf{w} \cdot \mathbf{n})^2}. \qquad (2.4)$$

With these preparations we can introduce a space-time variational problem consistent with (2.1): For a given $f \in L^2(S)$ determine $u \in W^b$ such that

$$\sum_{n=1}^{N} B_n(u, v) := \sum_{n=1}^{N} \left[\langle \dot{u}_n, v_n \rangle_n + \int_{S^n} \frac{1}{\alpha} \left(uv \operatorname{div}_\Gamma \mathbf{w} + \mu_d \nabla_\Gamma u \cdot \nabla_\Gamma v \right) \mathrm{d}\sigma \right.$$

$$\left. + \left([u]^{n-1}, v_+^{n-1} \right)_{\Gamma(t_{n-1})} \right] = \int_S \frac{1}{\alpha} fv \, \mathrm{d}s \, \mathrm{d}t \quad \text{for all } v \in W^b. \qquad (2.5)$$

Well-posedness of this problem is shown in [24, Theorem 5.3]. In the formulation (2.5) the material derivative is applied to the trial function u_n. By partial integration this derivative can be shifted to the test function v_n. As is well-known from the literature on DG methods this then leads do a different discretization. We will also address this issue for the space-time finite element method that we propose. Hence, besides the formulation in (2.5) we also introduce two other natural formulations, namely one with the material derivative on the test function and an antisymmetric variant. For deriving these we use the partial integration rule

$$\int_{S^n} \frac{1}{\alpha} \dot{u} v \, \mathrm{d}\sigma$$

$$= - \int_{S^n} \frac{1}{\alpha} \left(u\dot{v} + uv \operatorname{div}_\Gamma \mathbf{w} \right) \mathrm{d}\sigma + \left(u_-^n, v_-^n \right)_{\Gamma(t_n)} - \left(u_+^{n-1}, v_+^{n-1} \right)_{\Gamma(t_{n-1})}, \qquad (2.6)$$

for $u, v \in C^1(S^n)$, which directly follows from the Leibniz rule and (2.4). Note that for $u, v \in H^1(S^n)$ the weak material derivative can be expressed as $\langle \dot{u}, v \rangle_n = \int_{S^n} \frac{1}{\alpha} \dot{u} v \, \mathrm{d}\sigma$. For arbitrary $\beta \in \mathbb{R}$ and $u, v \in H^1(S^n)$ the following holds:

$$B_n(u, v) = \int_{S^n} \frac{1}{\alpha} \left((1 - \beta)\dot{u}v - \beta u\dot{v} + (1 - \beta)uv \operatorname{div}_\Gamma \mathbf{w} + \mu_d \nabla_\Gamma u \cdot \nabla_\Gamma v \right) \mathrm{d}\sigma$$

$$+ \beta \left(u_-^n, v_-^n \right)_{\Gamma(t_n)} + (1 - \beta) \left(u_+^{n-1}, v_+^{n-1} \right)_{\Gamma(t_{n-1})} - \left(u_-^{n-1}, v_+^{n-1} \right)_{\Gamma(t_{n-1})}. \qquad (2.7)$$

4

This follows from (2.6) using the following argument. For fixed $u, v \in H^1(S^n)$ we take the derivative of the right-hand side (2.7) with respect to β, which yields

$$-\int_{S^n} \frac{1}{\alpha} \left(\dot{u}v + u\dot{v} + uv \operatorname{div}_\Gamma \mathbf{w} \right) d\sigma + \left(u_-^n, v_-^n \right)_{\Gamma(t_n)} - \left(u_+^{n-1}, v_+^{n-1} \right)_{\Gamma(t_{n-1})}, \qquad (2.8)$$

which equals zero due to (2.6). Hence the right-hand side in (2.7) does not depend on β. For $\beta = 0$ formula (2.7) is correct, which follows by comparing with the definition in (2.5). Particularly interesting, in view of finite element discretization methods, are the choices $\beta = 0$, $\beta = \frac{1}{2}$ and $\beta = 1$. When one replaces the smooth surface S_n by an approximate Lipschitz surface, as we will do in our finite element discretization below, the resulting three formulations are no longer equivalent. The reason for this is that the identities (2.4) and (2.6) do not hold on Lipschitz surfaces.

3. Space-time TraceFEM. In this section we explain the space-time finite element method that we use for discretization of (2.5). Key ingredients of the method are a level set representation of the evolving surface and a parametric mapping based on the level set function (approximation) that is used to approximate the surface and to define the (higher order) space-time finite element spaces. In this approach it is essential that the evolving surface $\Gamma(t)$ is represented as the zero level of an evolving level set function, denoted by $\phi(\mathbf{x}, t)$. This level set function is approximated by a (higher order) finite element function, denoted by ϕ_h, on a fixed tetrahedral mesh, cf. Section 3.1. Evaluating integrals on the zero level of a higher order polynomial is computationally (very) expensive. To avoid this, we use an approach similar to the classical parametric finite element technique. First a spatial mesh deformation, that is based on $\phi_h(\cdot, t)$ for fixed t, is constructed, cf. Section 3.2. Using a tensor product approach this then results in a mapping that deforms of the space-time mesh, cf. Section 3.3. This space-time mesh deformation mapping, that can be efficiently evaluated, has a twofold purpose. It is used to construct a parametrization of a higher order accurate space-time surface approximation that can be efficiently evaluated. This allows efficient quadrature for integrals over higher order space-time surface approximations, cf. Section 3.7. The mapping is also used for constructing parametric finite element spaces that have higher order approximation accuracy, cf. Section 3.6.

3.1. Level set representation. We assume that the evolving surface $\Gamma(t)$ is represented as the zero level of a smooth level set function $\phi : \Omega \times [0, T] \to \mathbb{R}$, i.e.

$$\Gamma(t) = \left\{ \mathbf{x} \in \Omega : \phi(\mathbf{x}, t) = 0 \right\}, \quad t \in [0, T].$$

We assume standard properties of a level set function, i.e. for all $(\mathbf{x}, t)$ in a neighborhood U of S,

$$\left\| \nabla \phi(\mathbf{x}, t) \right\| \sim 1, \quad \left\| D^2 \phi(\mathbf{x}, t) \right\| \lesssim 1 \qquad (3.1)$$

hold. We define the space-time cylinders

$$Q := \Omega \times [0, T] \subset \mathbb{R}^4, \quad Q_n := \Omega \times I_n, \quad n = 1, \dots, N.$$

Let $\mathcal{T}$ be an element of a family $\{\mathcal{T}_h\}_{h>0}$ of shape regular tetrahedral triangulations of Ω, The triangulation $Q_{h,n} := \mathcal{T} \times I_n$ divides the time slab Q_n into space-time prismatic elements. The space-time triangulation of Q is denoted by $Q_h := \bigcup_{n=1}^N Q_{h,n}$. Let V_h^m,

$m \in \mathbb{N}$, be the standard $H^1(\Omega)$-conforming finite element space on the triangulation $\mathcal{T}$. For $m_s, m_q \in \mathbb{N}$ a space-time finite element product space is given by

$$V_h^{m_s,m_q} := \left\{ v : Q \to \mathbb{R} : v(\mathbf{x}, t) = \sum_{i=0}^{m_q} t^i v_i(\mathbf{x}), v_i \in V_h^{m_s}, (\mathbf{x}, t) \in Q_n, n = 1, \ldots, N \right\}.$$

(3.2)

Note that functions from $V_h^{m_s,m_q}$ are continuous is space but may be discontinuous in time at the time interval boundaries t_n. For the geometry approximation we use a function from this space that is continuous both in space and time and has degree $k_{g,s}$ in $\mathbf{x}$ and $k_{g,q}$ in t. More specifically, we assume an approximation $\phi_h \in V_h^{k_{g,s},k_{g,q}}$ of the level set function ϕ with $\phi_h(\mathbf{x}, \cdot) \in C^0([0, T])$, $\mathbf{x} \in \Omega$ that satisfies

$$\max_{\substack{K \in \mathcal{T} \\ n \in \{1, \ldots, N\}}} |\phi_h - \phi|_{W^{m,\infty}((K \times I_n) \cap U)} \lesssim h^{k_g + 1 - m}, \quad 0 \le m \le k_g + 1, \qquad (3.3)$$

with $k_g := \min\{k_{g,s}, k_{g,q}\}$. Such a ϕ_h can be obtained by interpolation of ϕ, if the latter is available. In applications, ϕ may be determined implicitly by a level set equation. To satisfy (3.3) one has to solve this equation sufficiently accurate. The continuity in time of ϕ_h later ensures that the piecewise planar reference space-time surface is a connected Lipschitz manifold.

3.2. Spatial mesh deformation. We briefly explain the main idea of the parametric mapping introduced in [20] to construct higher order *unfitted* finite element approximations. It is used to define a corresponding space-time mapping in the next section. By $\mathcal{I}_m$ we denote the spatial nodal $\mathcal{P}^m$-interpolation operator

$$\mathcal{I}_m : C^0(\Omega) \to V_h^m, \quad m \in \mathbb{N}. \qquad (3.4)$$

Let the spatially piecewise linear nodal interpolation $\hat{\phi}_h \in V_h^{1,k_{g,t}}$ be defined by

$$\hat{\phi}_h(\cdot, t) := \mathcal{I}_1 \phi_h(\cdot, t) \in V_h^1 \quad \text{for all } t \in [0, T], \qquad (3.5)$$

and its corresponding piecewise planar zero level at time t, which is easy to compute:

$$\Gamma_{\text{lin}}(t) := \left\{ \mathbf{x} \in \Omega : \hat{\phi}_h(\mathbf{x}, t) = 0 \right\}.$$

By construction, $\Gamma_{\text{lin}}(t)$ is only a second order approximation to $\Gamma(t)$. We consider all elements that are cut by the piecewise linear surface at *any point in time within one time slab*, i.e.,

$$\mathcal{T}_n^\Gamma := \left\{ K \in \mathcal{T} : \text{meas}_2((K \times I_n) \cap \Gamma_{\text{lin}}(t)) > 0 \text{ for any } t \in I_n \right\}, \quad n \in \{1, \ldots, N\}.$$

The corresponding domain is denoted by $\Omega_n^\Gamma := \{\mathbf{x} \in K : K \in \mathcal{T}_n^\Gamma\}$. On this domain a mapping $\Theta_{h,t}^n \in (V_h^{k_{g,s}})^3$ is defined that *depends (only) on* ϕ_h and deforms all elements of the triangulation $\mathcal{T}_n^\Gamma$. The image of this mapping restricted to $\Gamma_{\text{lin}}(t)$ defines a higher order approximation of $\Gamma(t)$ such that $\text{dist}_3(\Gamma(t), \Theta_{h,t}^n(\Gamma_{\text{lin}}(t))) \lesssim h^{k_{g,s}+1}$ for all $n = 1, \ldots, N$, $t \in I_n$. We refer to [20, 12] for precise definitions and an analysis of approximation properties of this mapping. In these papers this mapping only deforms elements that are cut by $\Gamma_{\text{lin}}(t)$, which, however, is not essential for the general construction. In [17, 26] the extended deformation, i.e., the one applied to all tetrahedra in $\mathcal{T}_n^\Gamma$ is explained.

6

3.3. Space-time mesh deformation. We explain the space-time mesh deformation on an arbitrary time slab Q_n, $n = 1, \ldots, N$, also used in [17, 26]. Let $\tau_m^n \in I_n$ be discrete points and $\mathcal{X}_{\tau_m^n} \in \mathcal{P}^{k_{g,q}}$, $m = 0, \ldots, k_{g,q}$, the corresponding finite element (e.g. nodal) basis functions. On the time slab Q_n the finite space-time finite element function $\phi_h \in V_h^{k_{g,s}, k_{g,q}}$ can be represented as

$$\phi_h(\mathbf{x}, t) = \sum_{m=0}^{k_{g,q}} \mathcal{X}_{\tau_m^n}(t) \phi_m(\mathbf{x}), \quad \phi_m \in V_h^{k_{g,s}},$$

We define the triangulation of space-time prisms that are intersected by $\Gamma_{\mathrm{lin}}(t)$ at any point in time $t \in I_n$ as $Q_{h,n}^S := \mathcal{T}_n^\Gamma \times I_n$, $n = 1, \ldots, N$, and $Q_h^S := \cup_{n=1}^N Q_{h,n}^S$. The subdomains formed by these triangulations are denoted by Q_n^S and Q^S. The space-time mesh transformation $\Theta_h^n \in (V_h^{k_{g,s}, k_{g,q}}|_{Q_{h,n}^S})^4$ is defined by

$$\Theta_h^n(\mathbf{x}, t) := \Big(\sum_{m=0}^{k_{g,q}} \mathcal{X}_{\tau_m^n}(t) \Theta_{h,\tau_m^n}^n(\mathbf{x}), \; t \Big) =: \big(\Theta_{h,s}^n(\mathbf{x}, t), t \big), \quad (\mathbf{x}, t) \in Q_n^S. \qquad (3.6)$$

The mapping $\Theta_{h,s}^n$ denotes the spatial part of Θ_h^n. For $n \in \{1, \ldots, N\}$ we define the discrete space-time manifolds

$$S_{\mathrm{lin}}^n := \Big\{ (\mathbf{x}, t) \in Q_n : \hat{\phi}_h(\mathbf{x}, t) = 0 \Big\}, \qquad S_{\mathrm{lin}} := \bigcup_{n=1}^N S_{\mathrm{lin}}^n, \qquad (3.7)$$

$$S_h^n := \big\{ (\mathbf{x}, t) \in Q_n : (\mathbf{x}, t) \in \Theta_h^n(S_{\mathrm{lin}}^n) \big\}, \qquad S_h := \bigcup_{n=1}^N S_h^n \qquad (3.8)$$

and the corresponding time slices of S_h^n are denoted by

$$\Gamma_h^n(t) := \Big\{ \mathbf{x} \in \mathbb{R}^3 : (\mathbf{x}, t) \in S_h^n \Big\}, \quad t \in I_n, \quad n = 1, \ldots, N. \qquad (3.9)$$

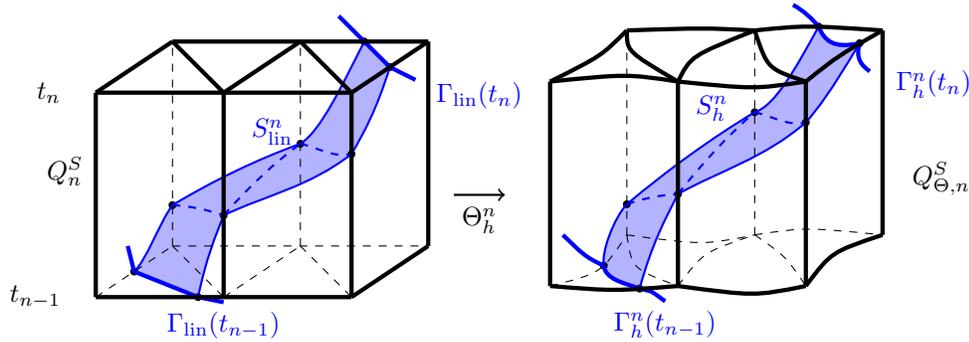We introduce notation for the curved space-time surface triangulation. Let the space-

Fig. 3.1: The mapping Θ_h^n, defined on $Q_{h,n}^S$, deforms the surface S_{lin}. Note that S_{lin} can be higher order in time.

time surface triangulation $\mathcal{T}_{S_h^n}$ be the set of smooth three-dimensional manifolds that

7

S_h^n consists of, i.e.

$$\mathcal{T}_{S_h^n} := \{\Theta_h^n(P) \cap S_h^n : P \in Q_{h,n}\}.$$

Then, we have $S_h^n = \bigcup_{K_S \in \mathcal{T}_{S_h^n}} K_S$. Let $\mathcal{T}_{S_h} := \bigcup_{n=1}^N \mathcal{T}_{S_h^n}$. Note that $K_S \in \mathcal{T}_{S_h}$ is a three-dimensional curved polytope that can be partitioned into curved tetrahedra. This deformed triangulation $\mathcal{T}_{S_h}$ is not necessarily shape regular, in the sense that internal angles can be very large and the size of neighboring elements may vary strongly. This is due to the fact that S_{lin} can have arbitrary cuts with the background bulk mesh Q_h. An illustration of the two-dimensional analogon of $\mathcal{T}_{S_h^n}$ is given in Figure 3.1.

REMARK 1. In general we have $\Omega_n^\Gamma \neq \Omega_{n+1}^\Gamma$, and therefore $\Theta_h^n \neq \Theta_h^{n+1}$. Take a fixed $t = t_n$ and let $K \in \mathcal{T}$ be a tetrahedron that is cut by $\Gamma_{\mathrm{lin}}(t_n)$. By construction we then have $K \in \mathcal{T}_n^\Gamma \cap \mathcal{T}_{n+1}^\Gamma$. The sets of neighboring tetrahedra of K in $\mathcal{T}_n^\Gamma$ and in $\mathcal{T}_{n+1}^\Gamma$, however, are not necessarily the same. Due to an averaging at the finite element nodes used in the construction of $\Theta_{h,t}^n$, cf. [20, 12], the mappings Θ_{h,t_n}^n and Θ_{h,t_n}^{n+1} are in general not the same on K. Thus in general we have that for certain $\mathbf{x} \in \Omega_n^\Gamma \cap \Omega_{n+1}^\Gamma$

$$\Theta_h^{n+1}(\mathbf{x}, t_n) = \Theta_{h,t_n}^{n+1}(\mathbf{x}) \neq \Theta_{h,t_n}^n(\mathbf{x}) = \Theta_h^n(\mathbf{x}, t_n),$$

which in particular implies $\Gamma_h^{n+1}(t_n) \neq \Gamma^n(t_n)$, cf. Figure 3.2 for an illustration. Due to the time stepping procedure in the finite element method, we weakly pass the discrete solution at the end point of a time interval I_n, i.e. on $\Gamma^n(t_n)$, to the next time interval, i.e. to $\Gamma_h^{n+1}(t_n)$. As these surfaces are not equal we need a suitable projection when defining the corresponding discrete bilinear form below.
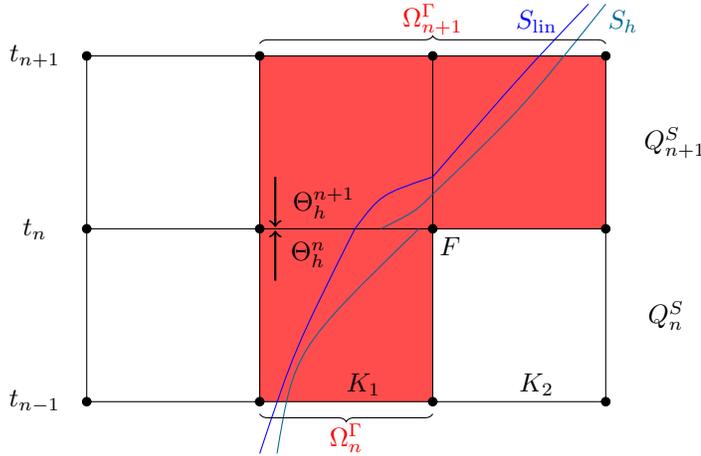

Fig. 3.2: Sketches of the discrete manifolds S_{lin} and S_h. Other than S_{lin}, the surface S_h is not necessarily continuous. In the calculation of $\Theta_h^n(K_1 \times I_n)$ the degrees of freedom on the face F only depend on K_1. In the calculation of $\Theta_h^{n+1}(K_1 \times I_{n+1})$ the same degrees of freedoms depend on $K_2 \in \mathcal{T}_{n+1}^\Gamma$ as well.

Below we will define and compute surface differential operators using discrete normal vectors on S_{lin} and S_h, both in the spatial and in the space-time sense. For a

given n and almost every $(\mathbf{x}, t) \in Q_n^S$ we define (with $D\Theta$ the spatial Jacobian of Θ)

$$\mathbf{n}_{\mathrm{lin}}(\mathbf{x}, t) := \frac{\nabla \hat{\phi}_h(\mathbf{x}, t)}{\left\| \nabla \hat{\phi}_h(\mathbf{x}, t) \right\|}, \quad \mathbf{n}_h(\Theta_h^n(\mathbf{x}, t)) := \frac{D^{-\mathsf{T}} \Theta_{h,s}^n(\mathbf{x}, t) \mathbf{n}_{\mathrm{lin}}(\mathbf{x}, t)}{\left\| D^{-\mathsf{T}} \Theta_{h,s}^n(\mathbf{x}, t) \mathbf{n}_{\mathrm{lin}}(\mathbf{x}, t) \right\|}, \qquad (3.10)$$

with $\hat{\phi}_h$ as defined in (3.5) and $\Theta_{h,s}^n$ as in (3.6). Restricted to $(\mathbf{x}, t)$ with $\mathbf{x} \in \Gamma_{\mathrm{lin}}(t)$ these vectors are indeed the unit (spatial) normal vectors to these surface approximations. We also introduce space-time variants, for almost every $(\mathbf{x}, t) \in Q_n^S$:

$$\mathbf{n}_{S_{\mathrm{lin}}}(\mathbf{x}, t) := \frac{\nabla_{(\mathbf{x},t)} \hat{\phi}_h(\mathbf{x}, t)}{\left\| \nabla_{(\mathbf{x},t)} \hat{\phi}_h(\mathbf{x}, t) \right\|} \qquad (3.11)$$

$$\mathbf{n}_{S_h}(\Theta_h^n(\mathbf{x}, t)) := \frac{D_{(\mathbf{x},t)}^{-\mathsf{T}} \Theta_h^n(\mathbf{x}, t) \mathbf{n}_{S_{\mathrm{lin}}}(\mathbf{x}, t)}{\left\| D_{(\mathbf{x},t)}^{-\mathsf{T}} \Theta_h^n(\mathbf{x}, t) \mathbf{n}_{S_{\mathrm{lin}}}(\mathbf{x}, t) \right\|}. \qquad (3.12)$$

For $(\mathbf{x}, t)$ restricted to S_{lin}^n these normals are the unit space-time normals to the respective space-time surfaces, cf. [27, Subsection 4.2.4] for more discussion.

3.4. Discrete surface differential operators. We consider the prisms that are cut by S_{lin}, i.e. prisms in Q_h^S, and apply the mesh deformation to obtain the sets:

$$Q_{\Theta,n}^S := \Theta_h^n(Q_n^S), \ n = 1, \dots, N, \quad Q_\Theta^S := \bigcup_{n=1}^N Q_{\Theta,n}^S.$$

Almost everywhere on the domain Q_Θ^S we define the projections $\mathbf{P}_h := \mathbf{I} - \mathbf{n}_h \mathbf{n}_h^\mathsf{T}$ and $\mathbf{P}_{S_h} := \mathbf{I} - \mathbf{n}_{S_h} \mathbf{n}_{S_h}^\mathsf{T}$. For a fixed t and smooth extension u^e ($\mathbf{u}^e$) of u ($\mathbf{u}$) defined on $\Gamma_h(t)$ we define

$$\nabla_{\Gamma_h} u := \mathbf{P}_h \nabla u^e, \quad \mathrm{div}_{\Gamma_h} \mathbf{u} := \mathrm{tr}\left(\mathbf{P}_h \nabla \mathbf{u}^e\right).$$

Similarly, for sufficiently smooth u ($\mathbf{u}$) defined on $K_S \in \mathcal{T}_{S_h}$ we define

$$\nabla_{S_h} u := \mathbf{P}_{S_h} \nabla_{(\mathbf{x},t)} u^e, \quad \mathrm{div}_{S_h} \mathbf{u} := \mathrm{tr}\left(\mathbf{P}_{S_h} \nabla_{(\mathbf{x},t)} \mathbf{u}^e\right).$$

It is convenient to introduce broken spaces of "sufficiently smooth" functions on S_h (similar to (2.3)) and Q_Θ^S:

$$W^{h,b} := \left\{ v \in L^2(S_h): \ v|_{S_h^n} \in H^1(S_h^n), \quad n = 1, \dots, N \right\},$$

$$V_{\mathrm{reg},h} := \left\{ v \in L^2(Q_\Theta^S) : v|_{Q_{\Theta,n}^S} \in H^1(Q_{\Theta,n}^S), n = 1, \dots, N, v|_{S_h} \in W^{h,b} \right\}.$$

For more information concerning Sobolev spaces on Lipschitz manifolds we refer to the literature, e.g. [8]

3.5. Integration over Lipschitz space-time surfaces. The approximate surface S_h^n defined in (3.8) in general has only Lipschitz smoothness. This loss of smoothness, compared to the smooth space-time surface S, has consequences for (partial) integration formulas, which play a key role in the derivation and analysis of the

9

discretization method. In this section, we introduce discrete variants of the transformation formula (2.4) and the partial integration identity (2.6). These will be used in the derivation of the discretization method in Section 3.6 and in the discussion of implementation aspects in Section 3.7. Almost everywhere on S_h^n, $n \in \{1, \ldots, N\}$, we define

$$V_h := \frac{-\partial \left(\hat{\phi}_h \circ (\Theta_h^n)^{-1} \right)}{\partial t} \left\| \nabla \left(\hat{\phi}_h \circ (\Theta_h^n)^{-1} \right) \right\|^{-1}. \qquad (3.13)$$

REMARK 2. Note that the discrete space-time surface S_h^n is defined via the surface S_{lin}^n, which is the zero level of $\hat{\phi}_h$, and the parametric mapping Θ_h^n, cf. (3.8). The function V_h is a discrete analogue to the normal velocity $\mathbf{w} \cdot \mathbf{n}$ on S in the following sense. In the continuous setting we use the level set equation $\frac{\partial \phi}{\partial t} + \mathbf{w} \cdot \nabla \phi = 0$ to see that on S we have the relations

$$\mathbf{w} \cdot \mathbf{n} = \mathbf{w} \cdot \frac{\nabla \phi}{\|\nabla \phi\|} = \frac{-1}{\|\nabla \phi\|} \frac{\partial \phi}{\partial t}. \qquad (3.14)$$

The function V_h is a discrete approximation of the function on the right-hand side in (3.14).

We introduce a discrete version of the integral transformation (2.4).

THEOREM 3.1. *For $g_h \in L^2(S_h^n)$ we have*

$$\int_{t_{n-1}}^{t_n} \int_{\Gamma_h^n(t)} g_h \, \mathrm{d}s_h \, \mathrm{d}t = \int_{S_h^n} \frac{g_h}{\sqrt{1 + V_h^2}} \, \mathrm{d}\sigma_h, \quad n \in \{1, \ldots, N\}, \qquad (3.15)$$

where V_h, S_h^n and $\Gamma_h^n(t)$ are defined in (3.13), (3.8) and (3.9), respectively.

A proof is given in [27, Theorem 4.27]. Below, in Theorem 3.2, we derive an integration by parts identity on the discrete space-time manifold S_h^n. For this we need further definitions. First we define a discrete version of the material derivative in (2.2). Let $\mathbf{w}_S^{\mathsf{T}} := (\mathbf{w}^{\mathsf{T}}, 1)$. For $v \in H^1(S_h^n)$, $n \in \{1, \ldots, N\}$, we define the discrete material derivative as

$$\mathring{v} := \mathbf{w}_S \cdot \nabla_{S_h} v = (\mathbf{P}_{S_h} \mathbf{w}_S) \cdot \nabla_{S_h} v. \qquad (3.16)$$

This derivative is tangential to the corresponding discrete space-time surface. For $n = 1, \ldots, N$, let $\mathcal{F}_I^n := \{\partial K_S^1 \cap \partial K_S^2 : K_S^1, K_S^2 \in \mathcal{T}_{S_h^n}, K_S^1 \neq K_S^2\}$ be the set of interior boundary faces of the elements in $\mathcal{T}_{S_h^n}$ and $\mathcal{F}_T^n := \{\partial K_S \cap \Gamma_h^n(t_n) : K_S \in \mathcal{T}_{S_h^n}\}$ the set of element boundaries of $\mathcal{T}_{S_h^n}$ that lie on the top time slab boundary. Similarly, the bottom time slab boundary $\mathcal{F}_B^n := \{\partial K_S \cap \Gamma_h^n(t_{n-1}) : K_S \in \mathcal{T}_{S_h^n}\}$. Hence, $\partial S_h^n = \mathcal{F}_T^n \cup \mathcal{F}_B^n$. Almost everywhere on ∂S_h^n the vector

$$\boldsymbol{\nu}_\partial := \frac{1}{\sqrt{1 + V_h^2}} \begin{pmatrix} V_h \mathbf{n}_h \\ 1 \end{pmatrix} \qquad (3.17)$$

is the unit normal of ∂S_h^n in positive t-direction. Hence, $\boldsymbol{\nu}_\partial$ is the unit outer normal of the top boundary of S_h^n and $-\boldsymbol{\nu}_\partial$ is the unit outer normal of the bottom boundary of S_h^n. Note that $\boldsymbol{\nu}_\partial$ jumps between time slabs. For $K_S \in \mathcal{T}_{S_h}$ the conormals are denoted by $\boldsymbol{\nu}_h|_{K_S}$, i.e., the unit outer normal on the faces of K_S that is tangential to K_S. For $K_S \in \mathcal{T}_{S_h^n}$ with $K_S \cap \Gamma_h^n(t_n) \neq \emptyset$ the unit normal $\boldsymbol{\nu}_\partial$ is the conormal vector

of K_S at the top boundary of S_h^n. Analogously, if $K_S \cap \Gamma_h^n(t_{n-1}) \neq \emptyset$, the vector $-\boldsymbol{\nu}_\partial$ is the conormal vector of K_S at the bottom boundary of S_h^n, cf. Figure 3.3 for an illustration. Consider an interior boundary face $F \in \mathcal{F}_I^n$ with $F = K_S^1 \cap K_S^2$, $K_S^1, K_S^2 \in \mathcal{T}_{S_h^n}$. For a function v defined on $K_S^1 \cup K_S^2$ we define the (vector valued) conormal jump on F:

$$[v]_{\boldsymbol{\nu}}\big|_F := \big(v|_{K_S^1} \boldsymbol{\nu}_h|_{K_S^1} + v|_{K_S^2} \boldsymbol{\nu}_h|_{K_S^2}\big)\big|_F. \tag{3.18}$$

Note that in general for the Lipschitz surface S_h^n we do not have C^1 smoothness across F and thus $\boldsymbol{\nu}_h|_{K_S^1} \neq -\boldsymbol{\nu}_h|_{K_S^2}$ on F. If the function v is continuous across F we have $[v]_{\boldsymbol{\nu}}\big|_F = v|_F [1]_{\boldsymbol{\nu}}\big|_F$. For one-sided values we use the standard notation
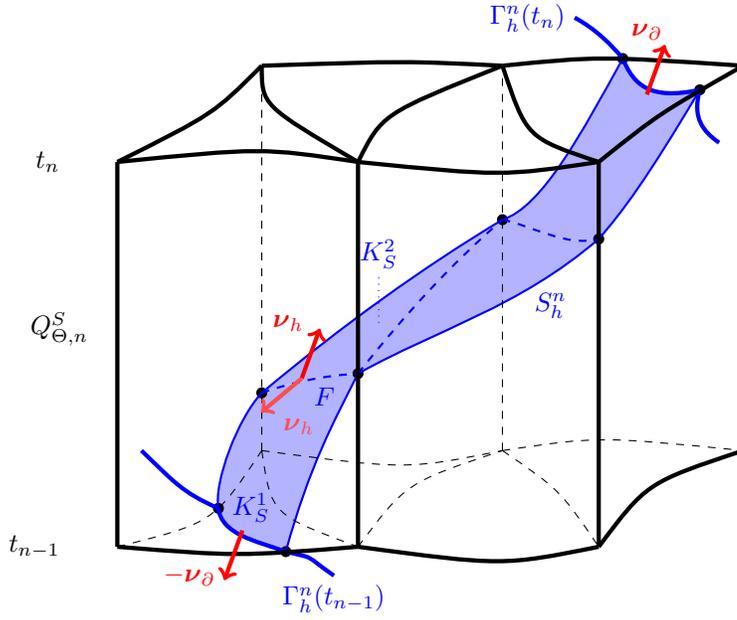

Fig. 3.3: Illustration of the conormals. At the time slab boundaries, the conormals coincide with $\pm\boldsymbol{\nu}_\partial$. Due to the non-smoothness of S_h, $(\boldsymbol{\nu}_h|_{K_S^1})|_F \neq -(\boldsymbol{\nu}_h|_{K_S^2})|_F$ at a common face F is possible.

$$u_+^n := u_+(\cdot, t_n) = \lim_{\eta \searrow 0} u(\cdot, t_n + \eta), \quad u_-^n := u_-(\cdot, t_n) = \lim_{\eta \searrow 0} u(\cdot, t_n - \eta).$$

Below we use a generic approximation of (the extension of) $\alpha = \sqrt{1 + (\mathbf{w} \cdot \mathbf{n})}$ on the discrete surface space-time surface S_h, denoted by α_h. One specific possibility is $\alpha_h = \sqrt{1 + V_h^2}$. However, as we will see below, other choices may be better.

THEOREM 3.2. *On S_h, let $R := \frac{1}{\alpha_h} \mathbf{w}_S \cdot \boldsymbol{\nu}_\partial$. For $u, v \in H^1(S_h^n)$, $n = 1, \dots, N$,*

11

the following identity holds:

$$\int_{S_h^n} \frac{1}{\alpha_h} \mathring{u} v \, \mathrm{d}\sigma_h = - \int_{S_h^n} \frac{1}{\alpha_h} u \mathring{v} \, \mathrm{d}\sigma_h + \int_{\Gamma_h^n(t_n)} u_-^n v_-^n R_-^n \, \mathrm{d}s_h$$

$$- \int_{\Gamma_h^n(t_{n-1})} u_+^{n-1} v_+^{n-1} R_+^{n-1} \, \mathrm{d}s_h + \sum_{F \in \mathcal{F}_I^n} \int_F uv \mathbf{w}_S \cdot \left[\frac{1}{\alpha_h} \right]_{\boldsymbol{\nu}} \mathrm{d}F \tag{3.19}$$

$$- \sum_{K_S \in \mathcal{T}_{S_h^n}} \int_{K_S} uv \, \mathrm{div}_{S_h} \left(\frac{1}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \mathrm{d}\sigma_h.$$

A proof is given in Appendix A. The partial integration (3.19) identity contains several geometric perturbation terms that do not arise for the smooth surface S. These terms play a role in the definition and error analysis of the space time finite element method, cf. the next section. The proof of Theorem 3.2 relies on the surface triangulation of S_h^n. Using (3.15) the surface integrals $\int_{S_h^n} \cdot \; \mathrm{d}\sigma_h$ can be reformulated as product space-time integrals $\int_{t_{n-1}}^{t_n} \int_{\Gamma_h^n(t)} \cdot \; \mathrm{d}s_h \, \mathrm{d}t$. However, a reformulation of the integrals $\int_F \cdot \; \mathrm{d}F$ and $\int_{K_S} \cdot \; \mathrm{d}\sigma_h$ in Theorem 3.2 as product space-time integrals is not natural.

3.6. Space-time trace finite element discretization. In this section we introduce a fully discrete higher order space-time discretization of (2.5). This method essentially uses the same approach as in [24, 23], but there are important differences, as explained in the Introduction, cf. Section 1. The method is a space-time Eulerian method that uses standard tensor product finite element spaces, combined with a parametric mapping to obtain a feasible method with higher order accuracy. More precisely, for $k_s, k_q \in \mathbb{N}$ we consider the same product space $V_h^{k_s, k_q}$, cf. (3.2), as used for the level set approximation and combine it with the space-time parametric mapping (3.6), which depends on the (higher order accurate) level set approximation ϕ_h. We define the parametric finite element space

$$V_{h,\Theta}^{k_s,k_q} := \left\{ v_h : Q_\Theta^S \to \mathbb{R} : (\mathbf{x}, t) \mapsto v_h(\Theta_h^n(\mathbf{x}, t)) \in V_h^{k_s,k_q}|_{Q_n^S}, n = 1, \dots, N \right\}. \tag{3.20}$$

Note that $V_{h,\Theta}^{k_s,k_q} \subset V_{\mathrm{reg},h}$. For $\beta \in [0,1]$ and $u, v \in V_{\mathrm{reg},h}$ the following discrete bilinear form on one time slab is defined (with $u_-^0 := 0$)

$$B_{h,n}^\beta(u, v) := \int_{S_h^n} \frac{1}{\alpha_h} \left((1-\beta)\mathring{u}v - \beta u\mathring{v} + (1-\beta)uv \, \mathrm{div}_{\Gamma_h} \mathbf{w} + \mu_d \nabla_{\Gamma_h} u \cdot \nabla_{\Gamma_h} v \right) \mathrm{d}\sigma_h$$

$$+ \beta \left(R_-^n u_-^n, v_-^n \right)_{\Gamma_h^n(t_n)} + (1 - \beta) \left(R_+^{n-1} u_+^{n-1}, v_+^{n-1} \right)_{\Gamma_h^n(t_{n-1})}$$

$$- \left(R_+^{n-1} u_-^{n-1} \circ \Theta_h^{n-1} \circ (\Theta_h^n)^{-1}, v_+^{n-1} \right)_{\Gamma_h^n(t_{n-1})}, \tag{3.21}$$

which is the discrete analogon of (2.7) , and its sum

$$B_h^\beta(u, v) := \sum_{n=1}^N B_{h,n}^\beta(u, v). \tag{3.22}$$

12

REMARK 3. Opposite to the bilinear form (2.7) the one in (3.21) is in general *not* independent of β. If for fixed arguments (u, v) we take the derivative with respect to β and use (3.19) we obtain

$$\frac{\partial}{\partial \beta} B_{h,n}^{\beta}(u, v) = -\int_{S_h^n} \frac{1}{\alpha_h} \left(\mathring{u}v + u\mathring{v} + uv \operatorname{div}_{\Gamma_h} \mathbf{w} \right) \mathrm{d}\sigma_h$$

$$+ \left(R_-^n u_-^n, v_-^n \right)_{\Gamma_h^n(t_n)} - \left(R_+^{n-1} u_+^{n-1}, v_+^{n-1} \right)_{\Gamma_h^n(t_{n-1})} \tag{3.23}$$

$$= \int_{S_h^n} \frac{1}{\alpha_h} uv \operatorname{div}_{\Gamma_h} \mathbf{w} \, \mathrm{d}\sigma_h - \sum_{K_S \in \mathcal{T}_{S_h^n}} \int_{K_S} u_h v_h \operatorname{div}_{S_h} \left(\frac{1}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \mathrm{d}\sigma_h \tag{3.24}$$

$$+ \sum_{F \in \mathcal{F}_I^n} \int_F u_h v_h \mathbf{w}_S \cdot \left[\frac{1}{\alpha_h} \right]_{\boldsymbol{\nu}} \mathrm{d}F. \tag{3.25}$$

Opposite to (2.8) this expression is in general not zero. Note that in (3.24)-(3.25) there are no integral terms $(\cdot, \cdot)_{\Gamma_h^n(t_n)}$. This is due to the fact that in the definition of the method (3.21) we use an appropriate weighting with R such that in the partial integration step, cf. (3.19), these terms vanish. The perturbation terms (3.24)-(3.25) are due to geometric errors and can be controlled by the accuracy of the discrete space-time surface approximation S_h^n.

In our space-time discretization we use a variant of *volume normal derivative stabilization* [12, 4], namely:

$$s(u, v) := \xi \int_{Q_{\Theta}^S} (\mathbf{n}_h \cdot \nabla u)(\mathbf{n}_h \cdot \nabla v) \, \mathrm{d}(\mathbf{x}, t) . \tag{3.26}$$

Based on the literature we take the parameter range

$$h \lesssim \xi \lesssim h^{-1}. \tag{3.27}$$

We add this stabilization to the bilinear form:

$$B_h^{\beta, \mathrm{stab}}(u, v) := B_h^{\beta}(u, v) + s(u, v). \tag{3.28}$$

Before defining the fully discrete problem we need a suitable approximation of the right hand side f. For $t \in I_n$ we denote by $\mu_h = \mu_h(t)$ the quotient of the surface measures of $\Gamma(t)$ and $\Gamma_h^n(t)$, i.e. $\mu_h \, \mathrm{d}s_h = \mathrm{d}s$. We assume $f_h \in L^2(S_h)$ to be an approximation of the exact data f satisfying

$$\|f_h - \mu_h f^e\|_{L^2(S_h)} \lesssim h^{k+1}, \tag{3.29}$$

where $k := \min\{k_s, k_q, k_{g,s}, k_{g,q}\}$. From the literature it is known how such a data approximation can be determined, see e.g. [27, Remark 4.43] or [12, Remark 5] in similar settings. We now define the *space-time discrete variational problem*: Given a right-hand side $f_h \in L^2(S_h)$ that satisfies (3.29), determine $u_h \in V_{h,\Theta}^{k_s, k_q}$ such that

$$B_h^{\beta, \mathrm{stab}}(u_h, v_h) = \int_{S_h} \frac{f_h v_h}{\sqrt{1 + V_h^2}} \, \mathrm{d}\sigma_h \quad \text{for all } v_h \in V_{h,\Theta}^{k_s, k_q}. \tag{3.30}$$

We are particularly interested in the case $\beta = 0$ (material derivative on u), $\beta = 1$ (material derivative on v) and $\beta = \frac{1}{2}$ (antisymmetric form). Implementation aspects

13

of this method are discussed in Section 3.7 below. In this paper we do not address rigorous discretization error analysis. This topic will be treated in a forthcoming paper. Rigorous error bounds for the case of bi-linear finite element approximations are presented in [27]. In the remarks below we discuss a few aspects of this discretization method that we consider to be relevant for a better understanding of the method.

REMARK 4. We comment on the role of the *volume normal derivative stabilization* $s(\cdot, \cdot)$. In the setting of a space-time Eulerian method that is based on a trace technique it is very natural to use the standard nodal basis of the outer space $V_{h,\Theta}^{k_s,k_q}$. It is easy to see that in general (in particular for higher order discretizations) the traces of these outer nodal basis functions on the (space-time) surface become linearly dependent and thus do not form a basis of the trace space. Without the stabilization term $s(\cdot, \cdot)$ the bilinear form (3.28) only contains surface integrals on S_h. In the stabilization term we have an integral over the *volumetric* domain Q_Θ^S, consisting of all (deformed) space-time prisms that are cut by the piecewise linear in space surface approximation S_{lin}. By adding the volumetric normal derivative we get *control of the variation of the discrete solution in normal direction*. This has several advantages, the most important one being that the stiffness matrix (w.r.t. the outer nodal basis) is nonsingular and there is no blow up of the condition number of this matrix. It is well-known that without stabilization, very small cuts of S_{lin} with the outer triangulation can lead to a very large condition number of the discretization matrix. A further relevant property is that the stabilization is *consistent* in the following sense. If one neglects geometric errors then the stabilization term vanishes if we insert $u = u^e$, i.e., the continuous solution constantly extended in spatial direction. This property is important for optimal order discretization accuracy of the method with stabilization. A detailed analysis of these conditioning and discretization accuracy propertes for the case of a stationary surface Γ is given in [22]. In the present paper we do not address the conditioning of the discretization matrix. In experiments (not presented here) we observe that there is no blow up of condition numbers and that these have a similar behaviour as for the space-time finite element method applied to a parabolic PDE in the volumetric domain $\Omega \times I$. Results that illustrate the discretization accuracy of the method are given in Section 4.

REMARK 5. In the discrete bilinear form (3.21) we use a generic function α_h, which is a discrete analogon of the function α appearing in the continuous bilinear form in (2.5). The formula (3.15) suggests that $\alpha_h = \sqrt{1 + V_h^2}$ may be a good choice. In numerical experiments in Section 4 we observe that this choice leads to suboptimal results in certain cases. In [27, Subsection 5.2.2] a theoretical analysis is given that explains these suboptimal results for the case $k_{g,s} = k_{g,q} = 1$. It is better to use an α_h that is a more accurate approximation of α than the one suggested above. One feasible possibility is the following. Let $\tilde{\phi}_h \in V_h^{k_{g,s}+1,k_{g,q}+1}$ be a one order higher order approximation of the level set function ϕ that satisfies

$$\left| \tilde{\phi}_h - \phi \right|_{W^{m,\infty}(Q^S)} \lesssim h^{k_g+2-m}, \quad 0 \le m \le k_g + 1, \tag{3.31}$$

with $k_g := \min\{k_{g,s}, k_{g,q}\}$, cf. (3.3). As an (improved) alternative to (3.13) we define

$$\tilde{V}_h := -\frac{\partial \tilde{\phi}_h}{\partial t} \left\| \nabla \tilde{\phi}_h \right\|^{-1}. \tag{3.32}$$

The function $\tilde{V}_h$ is an approximation of order $k_g + 1$ of the normal velocity $\mathbf{w} \cdot \mathbf{n}$ that does not use a mesh deformation. Thus we get the candidate $\alpha_h = \sqrt{1 + \tilde{V}_h^2}$, which

14

is smooth on the space-time cylinders in Q_h^S but in general discontinuous across the faces of these cylinders. We therefore apply a straightforward Oswald type projection as follows. Let $C(\mathcal{T}_n^\Gamma)$ and $C(Q_{h,n}^S)$ be the space of piecewise continuous functions on the triangulation $\mathcal{T}_n^\Gamma$ and $Q_{h,n}^S$, respectively. By $P_{h,n} : C(\mathcal{T}_n^\Gamma) \to V_h^{k_{g,s}}|_{\Omega_n^\Gamma}$ we denote an Oswald-type averaging operator [25, 20]. We introduce the corresponding space-time projection $\hat{P}_{h,n} : C(Q_{h,n}^S) \to V_h^{k_{g,s},k_{g,q}}$, $n = 1, \ldots, N$:

$$\hat{P}_{h,n} v(\mathbf{x}, t) := \sum_{m=0}^{k_{g,q}} \mathcal{X}_{\tau_m^n}^{\mathrm{nod}}(t) P_{h,n} v(\mathbf{x}, \tau_m^n),$$

where $\mathcal{X}_{\tau_m^n}^{\mathrm{nod}}$ denote the nodal basis functions of $\mathcal{P}^{k_{g,q}}$ with respect to the discrete points $\tau_m^n \in I_n$. The choice $\alpha_h = \hat{P}_{h,n}(\sqrt{1 + \tilde{V}_h^2})$, $n = 1, \ldots, N$, is a finite element function that is a better approximation of α^e than $\sqrt{1 + V_h^2}$. This choice is used in the numerical experiments in Section 4.1.

REMARK 6. We comment on the function R used in the boundary terms in (3.21), which serves as a weighting of the time slab boundary integrals. These weights can be included without significant additional computational costs. The difference $|R - 1|$ is controlled by geometric errors. In case of an exact surface we have $R = 1$. Adding the R-weighting leads to a more consistent formulation in a sense as discussed in Remark 3. If we include these weights the boundary terms (3.23) vanish in the partial integration on the discrete surface. In the error analysis in [27, Chapter 5] this cancellation plays a key role in the derivation of optimal discretization error bounds. The analysis does not yield optimal bounds if we replace R by 1. On the other hand, in numerical experiments (not presented here) we obtain optimal order convergence also if we replace R by 1. Below in Section 4 we use the method with the R-weighting in the boundary terms.

REMARK 7. As mentioned above, cf. Remark 3, the discretization method depends on β. Natural choices are $\beta = 0$, $\beta = 1$, $\beta = \frac{1}{2}$. The form with $\beta = 0$ is "closest" to the original strong formulation (2.1). In the error analysis, cf. [27, Chapter 5], this leads to a relatively simple consistency error analysis. The method for $\beta = \frac{1}{2}$ is antisymmetric with respect to the material derivative, which almost immediately leads to a stable method, without using partial integration. The method for $\beta = 1$ and with R replaced by 1 is exactly mass conserving for $k_{g,s} = 1$ on the discrete level cf. Remark 10 below. We will consider different β values in the numerical experiments in Section 4.1 below.

REMARK 8. We explain the factor $\Theta_h^{n-1} \circ (\Theta_h^n)^{-1}$ that appears in the last term in (3.21). This term is included to (weakly) transfer the solution from the previous time step to an initial condition in the current one. As explained in Remark 1, the space-time manifold S_h is not necessarily continuous between time slabs. To evaluate u_-^{n-1} on the triangulation deformed by Θ_h^n, the points on $\Gamma_h^n(t_{n-1})$ are mapped to $\Gamma_{\mathrm{lin}}(t_{n-1})$ using $(\Theta_h^n)^{-1}$. Then, using Θ_h^{n-1}, these points are mapped to $\Gamma_h^{n-1}(t_{n-1})$ where u_-^{n-1} is defined on.

REMARK 9. In [23] another stabilization, namely of the form

$$s_1(u, v) := \xi_1 \sum_{n=1}^N \int_{t_{n-1}}^{t_n} \int_{\Gamma_h^n(t)} u \, ds_h \int_{\Gamma_h^n(t)} v \, ds_h \, dt, \quad \xi_1 \geq 0, \qquad (3.33)$$

is used. This stabilization is needed in the stability analysis presented in that paper. Based on numerical experiments it seems that this term is not essential for stability

15

of the method. In this paper, in particular also in the numerical experiments below, we do not include this stabilization term.

3.7. Implementation aspects. We discuss several implementation aspects of the discrete problem (3.30). We first note that all components needed for the implementation of this discretization are available in the open source package [21], an add-on to the finite element library `NGSolve\Netgen` [28, 29]. The test functions in (3.30) are in the temporal discontinuous finite element space $V_{h,\Theta}^{k_s,k_q}$, which allows us to implement the method as a *time stepping algorithm*, solving (3.30) time slab after time slab. The solution of the $(n-1)$-th time slab is weakly transferred to n-th time slab by the term $(R_+^{n-1} u_-^{n-1} \circ \Theta_h^{n-1} \circ (\Theta_h^n)^{-1}, v_+^{n-1})_{\Gamma_h^n(t_{n-1})}$, $n \in \{1, \ldots, N\}$, see Remark 8. We consider the bilinear form in one fixed time slab n. By N_n we denote the number of degrees of freedom in that time slab, i.e. the dimension of the finite element space $V_{h,\Theta}^{k_s,k_q}|_{Q_{\Theta,n}^S}$. Let the corresponding basis functions be denoted by φ_i^n, $i = 1, \ldots, N_n$. The implementation of the higher order spatial and space-time surface integrals in the discrete bilinear form (3.21) is done using quadrature on $\Gamma_{\mathrm{lin}}(\tau)$ at appropriate temporal quadrature points $\tau \in [0, T]$. We illustrate this for the term

$$\int_{S_h^n} \frac{1}{\alpha_h} \mathring{\varphi}_i^n \varphi_j^n \, \mathrm{d}\sigma_h = \int_{S_h^n} \frac{1}{\alpha_h} \left(\mathbf{w}_S \cdot \nabla_{S_h} \varphi_i^n \right) \varphi_j^n \, \mathrm{d}\sigma_h. \tag{3.34}$$

Other integrals in (3.30) can be treated in a similar way. The quadrature of the volume integral in the stabilization bilinear form (3.26) is simpler because it involves integrals over (deformed) prisms. We use the integrals transformation (3.15) and the chain rule to transform (3.34) to an integral over the tensor product surface $\Gamma_{\mathrm{lin}}(t) \times I_n$.

$$\int_{S_h^n} \frac{1}{\alpha_h} \left(\mathbf{w}_S \cdot \nabla_{S_h} \varphi_i^n \right) \varphi_j^n \, \mathrm{d}\sigma_h = \int_{t_{n-1}}^{t_n} \int_{\Gamma_h^n(t)} \frac{\sqrt{1 + V_h^2}}{\alpha_h} \left(\mathbf{w}_S \cdot \nabla_{S_h} \varphi_i^n \right) \varphi_j^n \, \mathrm{d}s_h \, \mathrm{d}t$$

$$= \int_{t_{n-1}}^{t_n} \int_{\Gamma_{\mathrm{lin}}(t)} J_\alpha \left((\mathbf{P}_{S_h} \mathbf{w}_S) \circ \Theta_h^n \cdot \nabla_{(\mathbf{x},t)}(\varphi_i^n) \circ \Theta_h^n \right) (\varphi_j^n \circ \Theta_h^n) \, \mathrm{d}s_{\mathrm{lin}} \, \mathrm{d}t$$

$$= \int_{t_{n-1}}^{t_n} \int_{\Gamma_{\mathrm{lin}}(t)} J_\alpha \left((\mathbf{P}_{S_h} \mathbf{w}_S) \circ \Theta_h^n \cdot D_{(\mathbf{x},t)}^{-\intercal} \Theta_h^n \nabla_{(\mathbf{x},t)}(\varphi_i^n \circ \Theta_h^n) \right) (\varphi_j^n \circ \Theta_h) \, \mathrm{d}s_{\mathrm{lin}} \, \mathrm{d}t$$

$$=: \int_{t_{n-1}}^{t_n} \int_{\Gamma_{\mathrm{lin}}(t)} g \, \mathrm{d}s_{\mathrm{lin}} \, \mathrm{d}t, \tag{3.35}$$

where

$$J_\alpha := \frac{\left| \det(D\Theta_{h,s}^n) \right| \left\| D^{-\intercal} \Theta_{h,s}^n \mathbf{n}_{\mathrm{lin}} \right\| \sqrt{1 + \left(V_h \circ \Theta_h^n \right)^2}}{\alpha_h \circ \Theta_h^n} \quad \text{on } S_{\mathrm{lin}}^n.$$

We briefly comment on computational aspects of the functions appearing in (3.35). We need the basis functions φ_i^n, φ_j^n composed with the mesh deformation Θ_h^n. These compositions are finite element functions on the undeformed mesh, as seen in the definition of the parametric finite element space $V_{h,\Theta}^{k_s,k_q}$ in (3.20). This means that $\varphi_i^n \circ \Theta_h^n \in V_h^{k_s,k_q}|_{Q_n^S}$. Hence, we only need basis functions of $V_h^{k_s,k_q}$, cf. (3.2), to calculate the discretization matrix. The projection $\mathbf{P}_{S_h}$ depends on the normal $\mathbf{n}_{S_h}$. Formula for this normal and for $V_h \circ \Theta_h^n$ are given (3.12) and (3.13) and are suitable for an efficient implementation. The functions J_α and $D_{(\mathbf{x},t)}^{-\intercal} \Theta_h^n$ are defined on $\Gamma_{\mathrm{lin}}(t)$

16

as well. The sufficiently smooth given functions α_h and $\mathbf{w}$ are evaluated using Θ_h^n explicitly. For an efficient and easy implementation of Θ_h^n it is important that this is a finite element vector function. We refer to [26, Chapter 5], where implementation aspects of the parametric mapping $\Theta_h^n \in (V_{h,\Theta}^{k_s,k_q}|_{Q_{\Theta,n}^S})^4$ are discussed. Hence, for a given $t \in I_n$, all functions in the integrand of (3.35) can be efficiently computed on $\Gamma_{\text{lin}}(t)$. We outline how space-time quadrature for the double integral in (3.35) can be implemented. We write

$$\int_{t_{n-1}}^{t_n} \int_{\Gamma_{\text{lin}}(t)} g \, \mathrm{d}s_{\text{lin}} \, \mathrm{d}t = \sum_{K \in \mathcal{T}_n^\Gamma} \int_{t_{n-1}}^{t_n} \int_{K \cap \Gamma_{\text{lin}}(t)} g \, \mathrm{d}s_{\text{lin}} \, \mathrm{d}t. \qquad (3.36)$$

Similar to [17, Section 5] and [26, Chapter 5] we split the temporal integral into parts, where the cut topology does not change. Let K_V the set of vertices of any $K \in \mathcal{T}$. We define the set of temporal nodes where the surface cuts a vertex of a tetrahedron K, enriched with the initial and final times t_{n-1} and t_n, i.e.

$$\Sigma_K := \left\{ t \in [t_{n-1}, t_n] : \hat{\phi}_h(\mathbf{x}_V, t) = 0, \mathbf{x}_V \in K_V \right\} \cup \{t_{n-1}, t_n\}.$$

The roots of the one-dimensional polynomial $t \mapsto \hat{\phi}_h(\mathbf{x}_V, t)$ can be found by using e.g. the bisection method. We sort the temporal nodes in Σ_K. Let $|\Sigma_K|$ denote the cardinality of Σ_K. We define $t_j^* \in \Sigma_K$, $j \in \{0, \dots, |\Sigma_K| - 1\}$, such that

$$t_{n-1} = t_0^* \leq \cdots \leq t_{|\Sigma_K|-1}^* = t_n.$$

Within $[t_j^*, t_{j+1}^*]$ the cut topology does not change, i.e. the surface $\Gamma_{\text{lin}}(t)$ cuts the same edges and faces of K. As a consequence, the integrand g is smooth on $K \cap \Gamma_{\text{lin}}(t) \times [t_j^*, t_{j+1}^*]$. We refer to Figure 3.4 for an illustration. Hence, the mapping

$$t \mapsto \int_{K \cap \Gamma_{\text{lin}}(t)} g(\mathbf{x}, t) \, \mathrm{d}s_{\text{lin}}, \quad t \in [t_j^*, t_{j+1}^*] \qquad (3.37)$$

is smooth. If we do not take care of the cut topology, the mapping (3.37) is in general not smooth, which may lead to a significantly larger quadrature error. Such a simpler approach in which topology changes are ignored is used in [16, 31]. In [17, Section 6] numerical investigations with these two approaches are presented in a moving domain setting, showing that the naive quadrature approach performs significantly worse. Motivated by these results we take the cut topology into account here. We write (3.36) as

$$\int_{t_{n-1}}^{t_n} \int_{\Gamma_{\text{lin}}(t)} g \, \mathrm{d}s_{\text{lin}} \, \mathrm{d}t = \sum_{K \in \mathcal{T}_n^\Gamma} \sum_{j=0}^{|\Sigma_K|-2} \int_{t_j^*}^{t_{j+1}^*} \int_{K \cap \Gamma_{\text{lin}}(t)} g \, \mathrm{d}s_{\text{lin}} \, \mathrm{d}t.$$

Depending on k_q and $k_{g,q}$ we take quadrature nodes $t_l \in [t_j^*, t_{j+1}^*]$, $l = 0, \dots, L-1$, $L \in \mathbb{N}$, with corresponding weights ω_l. The number L is chosen sufficiently large, such that the numerical integration does not influence the order of optimal discretization errors. The choice of L is not straightforward, see [27, Subsection 6.3.1.1]. We thus get a quadrature rule for the time integral of the following form

$$\int_{t_{n-1}}^{t_n} \int_{\Gamma_{\text{lin}}(t)} g \, \mathrm{d}s_{\text{lin}} \, \mathrm{d}t \approx \sum_{K \in \mathcal{T}_n^\Gamma} \sum_{j=0}^{|\Sigma_K|-2} \sum_{l=0}^{L} \omega_l \int_{K \cap \Gamma_{\text{lin}}(t_l)} g \, \mathrm{d}s_{\text{lin}}. \qquad (3.38)$$
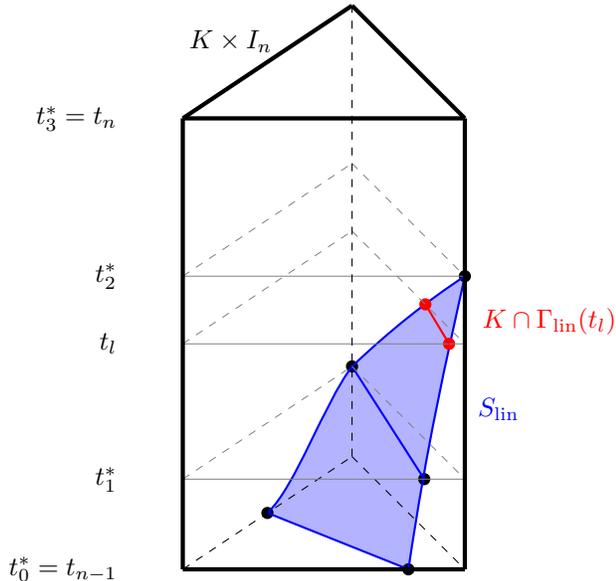
17

Fig. 3.4: We calculate integrals on S_h by using quadrature rules on sufficient spatial planar intersections $K \cap \Gamma_{\text{lin}}(t_l)$.

For a detailed discussion on the implementation of the spatial parametric mappings Θ^n_{h,τ^n_m} we refer to [20]. The domain of integration of the remaining spatial integral is the zero level of the piecewise linear level set function $\hat{\phi}_h(\cdot, t_l)$. The whole space-time integral is broken down to spatial integrals with respect to the low order geometry $\Gamma_{\text{lin}}(t_l)$. Every set $K \cap \Gamma_{\text{lin}}(t_l)$ is either a quadrilateral or a triangle. Quadrilaterals are divided into triangles before we use a quadrature rule of sufficient exactness degree ($\approx 2k_s$) on each triangle to approximate integrals on $K \cap \Gamma_{\text{lin}}(t_l)$. For more details concerning the quadrature method outlined above we refer to [17, 27].

4. Numerical experiments. In this section we present and discuss several numerical experiments. All methods are implemented in `ngsxfem` [21], an add-on package to the finite element library `NGSolve\Netgen`, see [28, 29]. To solve the arising linear systems, we use the direct solver `Pardiso` which is part of the `Intel MKL` library [30]. To visualize our results we use the software `ParaView`, see [1]. In Section 4.1 we illustrate the influence of the factor α_h on convergence results by considering a piecewise linear and piecewise higher order variant. In Section 4.2 we present results for two different smoothly evolving surfaces. We then show results of numerical experiments for an evolving surface with topological singularities in Section 4.3. Both spatially two- and three-dimensional settings are considered.

We explain general settings used in the experiments. We solve the discrete problem (3.30), and set the diffusion coefficient to $\mu_d = 1$. We consider $\beta \in \{0, \frac{1}{2}, 1\}$ and the stabilization parameter is taken as $\xi \in \{h, h^{-1}\}$. All test cases employ velocity fields that are not divergence-free, i.e. $\text{div}_\Gamma \mathbf{w} \neq 0$. As input for the mesh deformation we use the nodal space-time interpolation $\phi_h = I_h(\phi) \in V_h^{k_{g,s}, k_{g,q}}$ of a given level set function ϕ, resulting in the discrete space-time manifold S_h, cf. (3.8). In Section 4.2 we prescribe a smooth solution u in the neighborhood U and calculate the correspond-

ing right hand side f. Different from (2.5) we allow a non-zero initial condition u_0. In Section 4.3 we are not able to compute a right-hand side to a prescribed solution u, since there are geometric singularities. Instead, we take a nonzero initial solution u_0 on $\Gamma(0)$ and a source term $f = 0$. As initial condition for the discrete problem (3.30) we take the parametric interpolation

$$(u_h)^0_-(\mathbf{x}) := \mathcal{I}_{k_s}(u(\Theta_{h,0}(\cdot),0))(\Theta_{h,0}^{-1}(\mathbf{x})), \quad \mathbf{x} \in \Gamma_h(0).$$

The initial condition is weakly enforced via the integral over the deformed $\Gamma_h(0)$ contained in the bilinear form defined in (3.21). The right-hand side is taken as $f_h = f$ on S_h, which can be shown to be sufficiently accurate, see e.g. [27, Remark 4.43] or [10, Remark 5] in similar settings. We now explain the error measures that we use. For this we introduce the jump brackets on the discrete time slab boundaries

$$[v]^n_h := v^n_+ - v^n_- \circ \Theta^n_h \circ (\Theta^{n+1}_h)^{-1} \in L^2(\Gamma^{n+1}_h(t_n)), \quad n \in \{0, \dots, N-1\}, \qquad (4.1)$$

and define the energy norm

$$\|v\|^2_h := \max_{n \in \{1,\dots,N\}} \left\|v^n_-\right\|^2_{L^2(\Gamma^n_h(t_n))} + \sum_{n=1}^{N} \left\|[v]^{n-1}_h\right\|^2_{L^2(\Gamma^n_h(t_{n-1}))} + \|v\|^2_{L^2(S_h)} \qquad (4.2)$$
$$+ \left\|\nabla_{\Gamma_h} v\right\|^2_{L^2(S_h)} + \xi \|\mathbf{n}_h \cdot \nabla v\|^2_{L^2(Q^S_\ominus)}.$$

For measuring the discretization accuracy we will use the energy norm error $\|u^e - u_h\|_h$. For this measure to be useful it is important that u^e is chosen as the constant extension of u in spatial normal direction. Otherwise the bulk error $\xi^{\frac{1}{2}} \|\mathbf{n}_h \cdot \nabla(u^e - u_h)\|_{L^2(Q^S_\ominus)}$ does not necessarily converge with optimal order. In Sections 4.1 and 4.2 such a constant extension is algebraically computable and u^e is chosen accordingly. In Section 4.2.2 we consider a more complicated geometry for which this constant normal extension is not readily available. In that case we consider the modified energy norm

$$\|u^e - u_h\|^2_{h,*} := \|u^e - u_h\|^2_h - \xi \|\mathbf{n}_h \cdot \nabla(u^e - u_h)\|^2_{L^2(Q^S_\ominus)} \qquad (4.3)$$

which requires u^e values only on S_h. In this case it is not necessery to use a constant u extension in normal direction. Instead we use a natural, smooth extension u^e. We also measure the error in a discrete $L^\infty L^2$-norm

$$\|u^e - u_h\|_{L^\infty L^2} := \max_{n \in \{1,\dots,N\}} \|u^e - u_h\|_{L^2(\Gamma^n_h(t_n))}.$$

We use the following simple spatial mesh refinement strategy. We start with a coarse unstructured triangular/tetrahedral quasi-uniform triangulation of Ω, with an initial mesh size denoted by h_{init}. We refine the mesh uniformly to obtain the spatial refinement levels $l_s \in \{0, \dots, 8\}$, such that $h = h_{\text{init}} 2^{-l_s}$. Using a marked-edge bisection method, we only refine simplices that are intersected by the surface. Taking an initial uniform time step size Δt_{init}, we halve this value for each temporal refinement level $l_q \in \{0, \dots, 7\}$, i.e. $\Delta t = \Delta t_{\text{init}} 2^{-l_q}$. To assess the order of convergence in space, time and space-time, we calculate the spatial, temporal and space-time experimental order of convergence, which we denote as eoc_s, eoc_q and eoc_{qs} respectively. To do so we take two errors e_l and e_{l+1}, $l \in \{l_q, l_s\}$, of subsequent refinement levels l and $l+1$ and calculate $\text{eoc} = \frac{1}{\log(2)} \log\left(\frac{e_l}{e_{l+1}}\right)$. For the calculation of eoc_{qs} we consider one refinement in both space and time as subsequent refinement. To compute eoc_s and eoc_q we consider the errors of the finest temporal and the finest spatial discretizations used in the specific experiment, respectively.

4.1. On the factor α_h in the discrete bilinear form. We consider a shrinking circle that is moving through the background mesh with a time- and space-dependent velocity field. The center of the circle moves along a semicircle in the mesh. As outer domain we take $\Omega = (-1, 1)^2$ and as time interval we choose $[0, 1]$. On $Q = \Omega \times [0, 1]$ the evolution of the surface $\Gamma(t)$ is described by the zero level of

$$\phi(\mathbf{x}, t) = \sqrt{\left(x - \frac{\cos(\pi t)}{2}\right)^2 + \left(y - \frac{\sin(\pi t)}{2}\right)^2} - \frac{9e^{-\frac{t}{4}}}{20}, \quad (\mathbf{x}, t) = (x, y, t) \in Q.$$

On Q we construct a normal velocity field that transports $\Gamma(t)$ by

$$\mathbf{w}_n = -\frac{\partial \phi}{\partial t} \nabla \phi.$$

Note that $\|\nabla \phi(\mathbf{x}, t)\| = 1$ for all $(\mathbf{x}, t) \in Q$, since $\phi(\cdot, t)$ is a signed distance function to $\Gamma(t)$. We do not want to restrict to the special case of a problem with a strictly normal velocity field $\mathbf{w}$. Therefore we add a tangential part $\mathbf{w}_{\text{tang}} = \frac{1}{2}\left((\nabla\phi)_2, -(\nabla\phi)_1\right)^{\mathsf{T}}$ to $\mathbf{w}_n$, where $(\nabla\phi)_i$ denotes the ith component of $\nabla\phi$. Since $\mathbf{w}_n \cdot \nabla\phi = -\frac{\partial\phi}{\partial t}$ and $\mathbf{w}_{\text{tang}} \cdot \nabla\phi = 0$ we conclude that $\mathbf{w} = \mathbf{w}_n + \mathbf{w}_{\text{tang}}$ satisfies the level set equation $\frac{\partial\phi}{\partial t} + \mathbf{w} \cdot \nabla\phi = 0$. We use this $\mathbf{w}$ for the experiments and prescribe the exact solution as

$$u(\mathbf{x}, t) = y(x^2 + 1)e^{-t}, \quad (\mathbf{x}, t) \in S.$$

We take an initial mesh size of $h_{\text{init}} = 2^{-3}$ and an initial time step size as $\Delta t_{\text{init}} = 2^{-3}$. In Figure 4.1 we illustrate the space-time manifold S_h in the space-time prismatic background mesh for relatively coarse discretization parameters. In Figure 4.2a we present numerical results for $k = 1$. Note that for $k = 1$ the mesh-transformation reduces to the identity: $\Theta_h^n = \text{id}$. A seemingly natural choice is the approximation $\alpha_h = \sqrt{1 + V_h^2}$, where V_h is a piecewise bi-linear approximation of the normal velocity field $\mathbf{w} \cdot \mathbf{n}$, cf. (3.13). It is natural in the sense that $\sqrt{1 + V_h^2}$ is the transformation factor between an integral over S_h^n, i.e. $\int_{S_h^n} \cdot \, d\sigma_h$, and the corresponding iterated integral $\int_{t_{n-1}}^{t_n} \int_{\Gamma_h(t)} \cdot \, ds_h \, dt$, cf. Theorem 3.1. In the continuous setting this factor is α, which we approximate by α_h. In Figure 4.2b results for the piecewise bi-quadratic variant $\alpha_h = \hat{P}_{h,n}(\sqrt{1 + \tilde{V}_h^2})$, cf. Remark 5, are shown. In Figure 4.3 we present corresponding results for $k = 3$. In Figures 4.2a and 4.3a we observe optimal order convergence for the methods with $\beta = 0$ and one order less for the methods with $\beta \in \{\frac{1}{2}, 1\}$. A heuristic explanation for this difference in order of convergence is as follows. Note that the methods with $\beta \in \{\frac{1}{2}, 1\}$ contain a time derivative of the test function and due to this one needs partial integration in the consistency error analysis. Hence, jumps of α_h occur in the consistency error, see Theorem 3.2. In the method with $\beta = 0$ partial integration is not needed in the consistency error analysis (only in the stability analysis). The relatively poor α^e approximation $\sqrt{1 + V_h^2}$, which is discontinuous across surface element faces, causes large jumps of α_h that lead to a suboptimal order term in the consistency error. This does not happen for the case $\beta = 0$. For the more accurate approximation $\alpha_h = \hat{P}_{h,n}(\sqrt{1 + \tilde{V}_h^2})$ the jump terms on interior boundary faces of surface elements, i.e. on $\mathcal{F}_I^n$, vanish. Furthermore, corresponding jumps between time slabs are sufficiently small to allow an optimal order consistency error.

If not stated differently, we restrict to $\beta = 0$, $\xi = h$ and $\alpha_h = \sqrt{1 + V_h^2}$ in the experiments below.
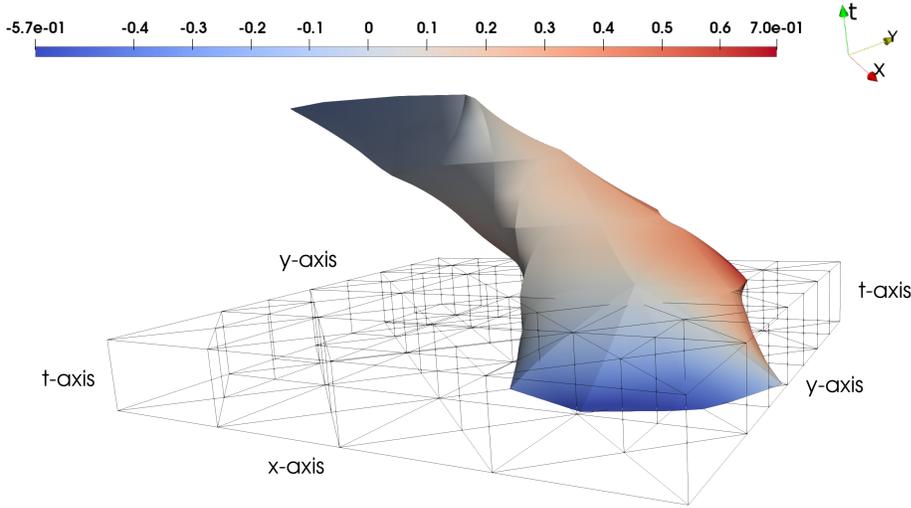
Fig. 4.1: Moving circle: a sketch of the space-time manifold S_h in the space-time prismatic triangulation with $N = 4$ time slabs. Here, we have $h = 2^{-2}$ and $k_{g_s} = k_{g,q} = 1$. We see the discrete solution u_h on a circle moving through the triangular background mesh.
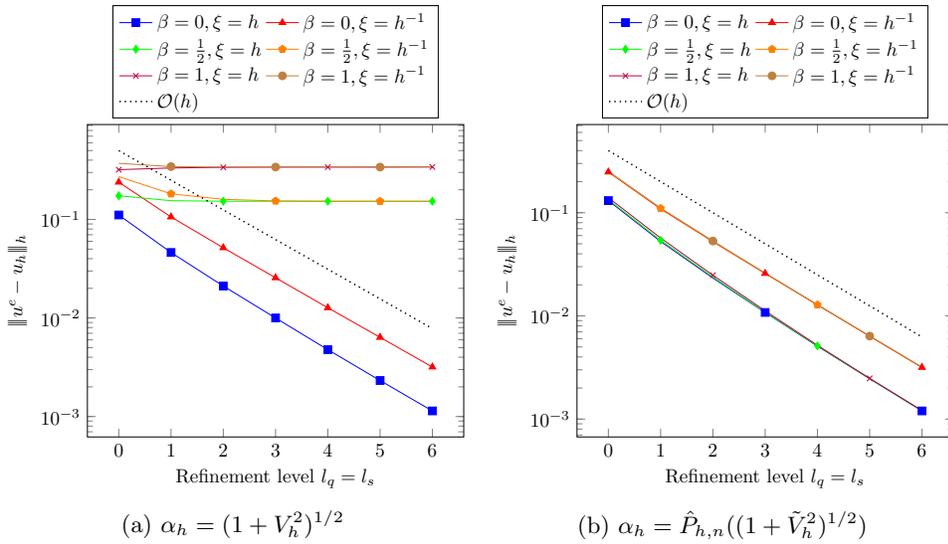

(a) $\alpha_h = (1 + V_h^2)^{1/2}$

(b) $\alpha_h = \hat{P}_{h,n}((1 + \tilde{V}_h^2)^{1/2})$

Fig. 4.2: Moving circle: convergence results varying α_h, β and ξ in the bilinear forms for $k = 1$.

4.2. Smoothly evolving surfaces. In this section we consider spatially two- and three-dimensional examples, both with smooth geometries.
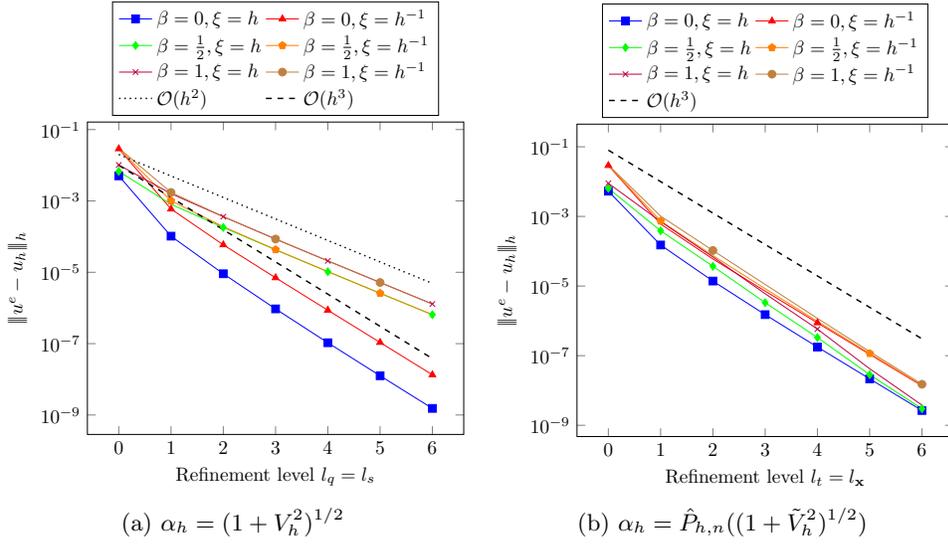
Fig. 4.3: Moving circle: convergence results varying α_h, β and ξ in the bilinear forms for $k = 3$.

4.2.1. Moving circle. We take the same Ω, ϕ, $\mathbf{w}$, T and u as in Section 4.1. We choose the initial mesh size $h_{\text{init}} = 2^{-2}$ and the initial time step size $\Delta t_{\text{init}} = 2^{-2}$. In Figure 4.4 and Tables 4.1 and 4.2 we depict numerical results in the energy norm and in the $L^\infty L^2$-norm for 7 levels of refinement. For $k \in \{1, 2, 3, 4, 5\}$ we observe in
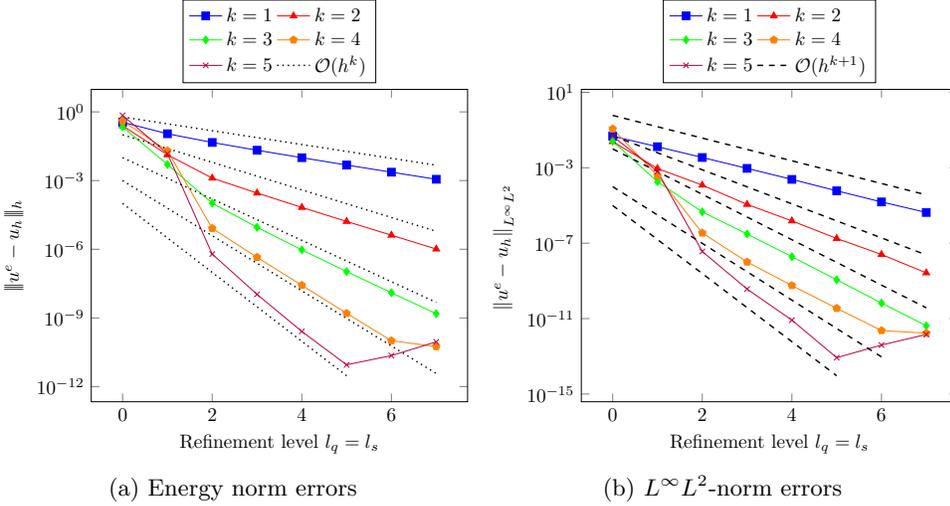
Fig. 4.4: Moving circle: convergence results in different norms.

Figure 4.4 optimal order convergence for both the energy norm error and the $L^\infty L^2$-

$l_q \downarrow \backslash l_s \rightarrow$	0	1	2	3	4	5	6	7	eoc_q
0	2.20E-1	6.48E-2	2.45E-2	1.37E-2	1.24E-2	1.12E-2	5.97E-3	6.79E-3	-
1	2.72E-1	5.05E-3	4.33E-4	2.73E-4	1.98E-4	1.52E-4	1.25E-4	1.10E-4	5.95
2	1.20E+0	1.40E-3	1.03E-4	3.24E-5	2.07E-5	1.43E-5	1.07E-5	8.67E-6	3.67
3	1.54E+0	1.19E-3	7.53E-5	9.09E-6	2.69E-6	1.64E-6	1.09E-6	7.88E-7	3.46
4	1.12E+0	1.11E-3	6.52E-5	6.32E-6	9.42E-7	2.35E-7	1.31E-7	8.73E-8	3.17
5	8.10E-1	1.09E-3	6.18E-5	4.98E-6	6.23E-7	1.06E-7	2.20E-8	1.08E-8	3.01
6	1.66E+0	1.08E-3	6.06E-5	4.37E-6	4.30E-7	6.95E-8	1.26E-8	2.28E-9	2.24
7	2.06E+2	1.07E-3	6.02E-5	4.15E-6	3.38E-7	4.37E-8	8.24E-9	1.53E-9	0.576
eoc_s	-	17.6	4.15	3.86	3.62	2.95	2.41	2.43	
eoc_{qs}	-	5.45	5.62	3.5	3.27	3.15	3.07	3.04	

Table 4.1: Moving circle: the errors $\|u^e - u_h\|_h$ for $k = 3$.

$l_q \downarrow \backslash l_s \rightarrow$	0	1	2	3	4	5	6	7	eoc_q
0	2.57E-2	5.93E-3	5.81E-4	4.28E-4	3.68E-4	3.53E-4	3.46E-4	3.42E-4	-
1	3.20E-2	1.83E-4	2.30E-5	1.62E-5	1.44E-5	1.38E-5	1.37E-5	1.37E-5	4.64
2	2.95E-1	1.64E-4	4.56E-6	1.01E-6	6.43E-7	5.88E-7	5.84E-7	5.86E-7	4.55
3	2.27E-1	1.62E-4	4.46E-6	3.08E-7	5.89E-8	3.38E-8	2.28E-8	2.12E-8	4.79
4	7.46E-1	1.62E-4	4.45E-6	3.12E-7	1.92E-8	3.65E-9	2.25E-9	1.16E-9	4.19
5	4.77E-1	1.62E-4	4.45E-6	3.11E-7	1.92E-8	1.15E-9	2.12E-10	1.44E-10	3.01
6	1.04E+0	1.62E-4	4.45E-6	3.16E-7	1.93E-8	1.15E-9	6.68E-11	1.23E-11	3.55
7	9.49E+1	1.62E-4	4.45E-6	3.16E-7	1.93E-8	1.16E-9	6.67E-11	4.13E-12	1.57
eoc_s	-	19.2	5.19	3.82	4.03	4.06	4.12	4.01	
eoc_{qs}	-	7.13	5.33	3.89	4.0	4.06	4.11	4.02	

Table 4.2: Moving circle: the errors $\|u^e - u_h\|_{L^\infty L^2}$ for $k = 3$.

error until error values small than 10^{-10} are reached. For $k = 4$, the error in the $L^\infty L^2$-norm shows a suboptimal rate of convergence with order approximately 4 (instead of the optimal order 5). In further experiments (results not shown here) we observed that an optimal order convergence can be obtained by increasing the order of the geometry approximation, i.e. taking $k_{g,s} = k_{g,t} = 5$. Taking only a higher order approximation of the geometric quantity α_h is not sufficient here, cf. Section 4.1. To further study the higher order methods, we present more detailed space-time convergence results for the case $k = 3$ in Tables 4.1 and 4.2. We observe optimal order convergence on the space-time diagonal. In these tables the first column gives somewhat unstable results. This is caused by the mesh deformation. When deforming a too coarse mesh, the distance between an undeformed and the corresponding deformed tetrahedron might become so large that undesirable effects occur. For example, faces of the deformed tetrahedra might touch each other. To avoid such effects, one uses a strategy in which certain tetrahedra are not allowed to deform, leading to an only piecewise linear approximation of the surface for the under-resolved case.

4.2.2. Twisting torus. We consider a torus, formed with the y-axis as axis of rotation. The torus is twisted over time. We take the background domain $\Omega = (-1, 1)^3$ and the time interval $[0, T]$, $T \in \{\frac{5}{2}, 5\}$. On $Q = \Omega \times [0, 5]$ the evolution of the surface

$\Gamma(t)$ is described by the zero level of

$$\phi(\mathbf{x}, t) = \left(\omega_1^2(\mathbf{x}, t) + \omega_2^2(\mathbf{x}, t) + \omega_3^2(\mathbf{x}, t) + \frac{9}{50}\right)^2 - \frac{121}{100}\left(\omega_1^2(\mathbf{x}, t) + \omega_3^2(\mathbf{x}, t)\right),$$

$$(\mathbf{x}, t) = (x, y, z, t) \in Q,$$

where $\omega_1, \omega_2, \omega_2 : Q \to \mathbb{R}$ are defined as

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} (\mathbf{x}, t) := \begin{pmatrix} \cos(ty) & 0 & -\sin(ty) \\ \sin(ty) & 0 & \cos(ty) \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix}.$$

On Q we use a normal velocity field that transports $\Gamma(t)$ by

$$\mathbf{w} = -\frac{\partial \phi}{\partial t} \frac{\nabla \phi}{\|\nabla \phi\|^2}.$$

We prescribe the exact solution as

$$u(\mathbf{x}, t) = u^e(\mathbf{x}, t) = x \sin\left(\frac{\pi t}{2}\right), \quad (\mathbf{x}, t) = (x, y, z, t) \in Q.$$

We take the initial mesh size as $h_{\text{init}} = 2^{-1}$ and the initial time step size as $\Delta t_{\text{init}} = 2^{-3}$. In Figure 4.5 we give a sketch of the geometry and the discrete solution on the torus. In Figures 4.6 and 4.7 we present errors in the surface norm $\|\cdot\|_{h,*}$ and in the L^2L^∞-norm. In Figure 4.6 we observe optimal order convergence for $k \in \{1, 2, 3\}$ and $T = \frac{5}{2}$, while for $k = 4$ we see faster convergence as expected. This is probably because, for runtime reasons, we have not refined further than $h = 2^{-5}$ (level $l = 4$) and the asymptotic range of the space-time convergence order has not been reached, yet. In Figure 4.7 we chose a larger temporal end point $T = 5$ which leads to much larger curvatures, cf. Figure 4.5. For $k \in \{1, 2\}$ the results show optimal convergence behavior, with in some cases even larger orders. This may caused by the fact that we use a larger geometry order here ($k_g = k + 1$) and due to the very large curvatures the geometric error still dominates on the levels considered. The experiments above indicate that the higher order methods perform well also in the spatially three-dimensional case.

REMARK 10. We comment on a difference with respect to mass conservation between the methods with $\beta = 1$ and $\beta \neq 1$. The method for $\beta = 1$ and with R replaced by 1 is exactly mass conserving for $k_{g,s} = 1$ in the following sense. Let u_h be the solution of (3.30) with $R = 1$ and $f_h = 0$, and define $i^n_{\text{mass}} := \int_{\Gamma_{\text{lin}}(t_n)} u^n_{h,-}$, $n = 0, \ldots, N$. Using a piecewise (in time) constant finite element test function $v_h = 1$ for $t \leq t_n$ and $v_h = 0$ for $t > t_n$ one easily checks that $i^n_{\text{mass}} = i^0_{\text{mass}}$ for $n = 0, \ldots, N$. Note that we consider $k_{g,s} = 1$ which implies $\Theta^n_h = \text{id}$ and $\Gamma^n_h(t_n) = \Gamma^{n-1}_h(t_n) = \Gamma_{\text{lin}}(t_n)$. In Figure 4.8, for the twisting torus problem with initial condition $u_0(x, y, z) = 5 + xyz$ and $f_h = 0$ we show results for $e_{\text{mass}} := \max_{n=0,\ldots,N} |i^n_{\text{mass}} - i^0_{\text{mass}}|$. These confirm the exact mass conservation property. The results also show that the method with $\beta = 1$ leads to significantly better results than the method with $\beta = 0$, also for the $R \neq 1$ case used in (3.21). We present results only for $k = 1$. For the higher order case with $\beta = 1$ we observe that the discretization errors in the $L^\infty L^2$ norm are of optimal order for $k = 3$ but suboptimal for $k = 2$. The reason for this (unexpected) behaviour is unclear and has to be investigated further.
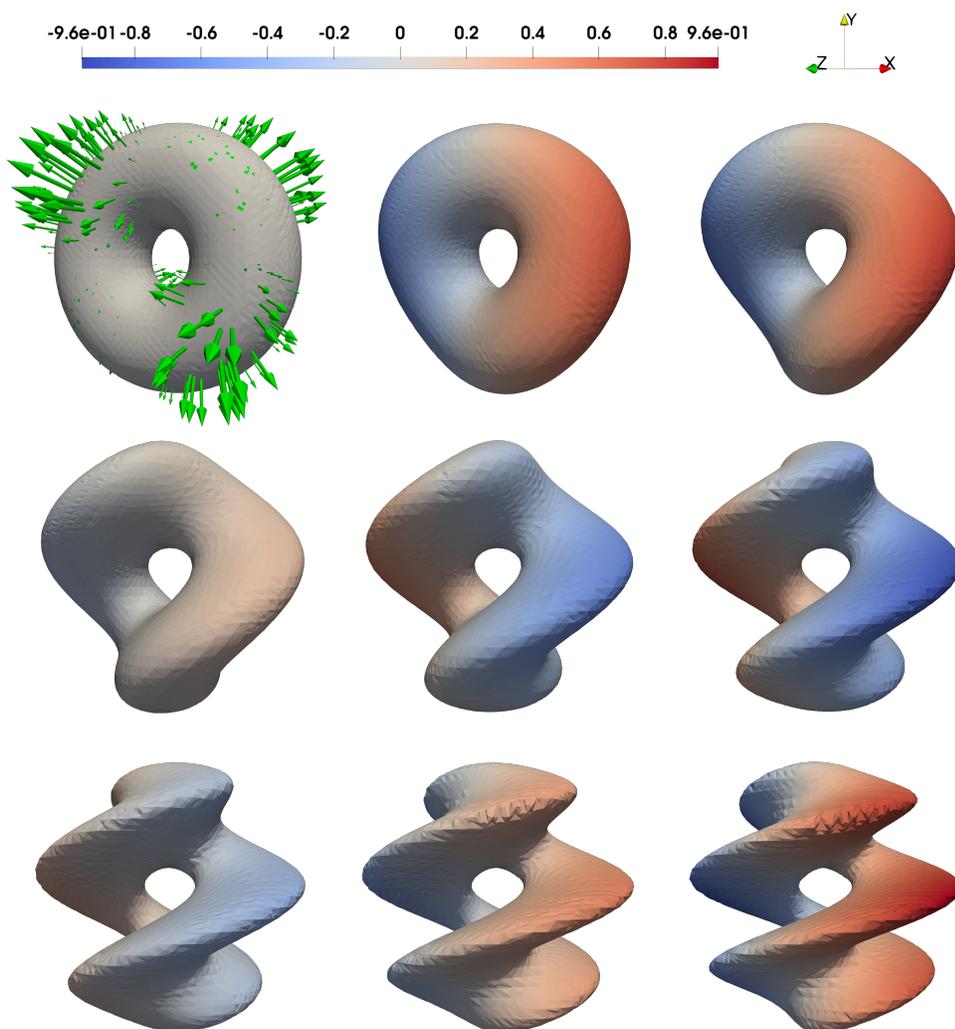
Fig. 4.5: A sketch of the evolution of the twisted torus for $k = 1$. The discrete solution u_h at times $t = \frac{iT}{8}$, $i = 0, \ldots, 8$, is depicted in the images from top left to bottom right. Here, we have $\Delta t = 5 \cdot 2^{-5}$, $h = 2^{-5}$ and $T = 5$. The green arrows represent the velocity field $\mathbf{w}$. A full animation can be found here.

4.3. Space-time surfaces with topological singularities. A unique feature of the technique presented in this paper is that it is also applicable to problems in which evolving surfaces with topological singularities occur. In the two examples below we illustrate this robustness property. Due to the singularities, higher order convergence can not be expected and therefore we restrict to $k = 1$.

4.3.1. Merging spheres. We consider two spheres that merge, similar to [13]. We take the background domain $\Omega = (-3, 3)^3$ and the time interval $[0, 1]$. On

(a) Surface energy norm errors

(b) $L^\infty L^2$-norm errors
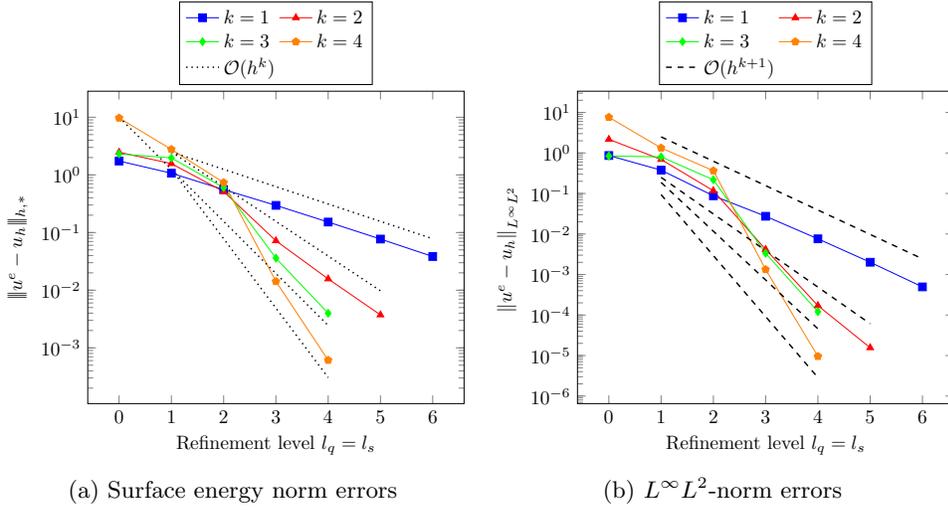
Fig. 4.6: Twisting torus: convergence results for $T = \frac{5}{2}$ in different norms.

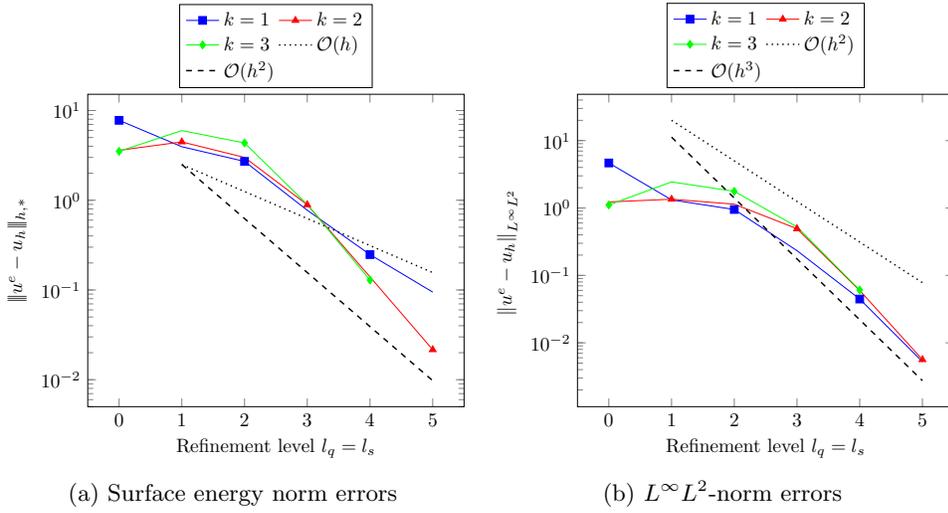(a) Surface energy norm errors

(b) $L^\infty L^2$-norm errors

Fig. 4.7: Twisting torus: convergence results for $T = 5$ in different norms. We use a higher order geometry: $k_{g,s} = k_{g,q} = k_s + 1 = k_q + 1 = k + 1$.

$Q = \Omega \times [0, 1]$ the evolution of the surface $\Gamma(t)$ is described by the zero level of

$$\phi(\mathbf{x}, t) = 1 - \frac{1}{\left\| \mathbf{x} - c_+(t) \right\|^3} - \frac{1}{\left\| \mathbf{x} - c_-(t) \right\|^3}, \quad (\mathbf{x}, t) \in Q,$$
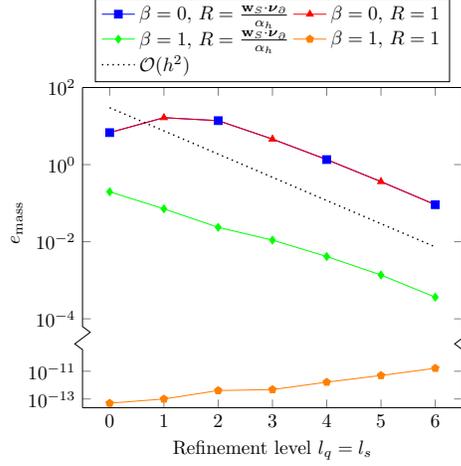
Fig. 4.8: Mass conservation results; $k = 1$.

where $c_\pm = \pm\frac{3}{2}(t-1,0,0)^\intercal$. On Q we construct a normal velocity field that transports $\Gamma(t)$ by

$$\mathbf{w} = -\frac{\partial\phi}{\partial t}\frac{\nabla\phi}{\|\nabla\phi\|^2}.$$

We consider the initial condition

$$u_0(\mathbf{x},t) = yz + x + 15, \quad (\mathbf{x},t) = (x,y,z,t) \in Q$$

and $f = 0$ for the right-hand side. We take the initial mesh size as $h_{\text{init}} = 2^{-1}$ and the initial time step size as $\Delta t_{\text{init}} = 2^{-3}$. In Figure 4.9 we illustrate the discrete solution at different points in time.

To evaluate the robustness of the method, we measure the mass evolution over time. We run numerical experiments to assess if and how accurate the discrete solution satisfies a mass conservation property. To illustrate the change of total mass and area over time, we measure these quantities at the discrete points t_n, i.e.

$$i_{\text{mass}}^n := \int_{\Gamma_h^n(t_n)} u_{h,-}^n \, ds_h, \quad i_{\text{surf}}^n := \int_{\Gamma_h^n(t)} 1 \, ds_h, \quad n = 0, \dots, N.$$

In the figures below these quantities are linearly interpolated in each time interval, resulting in the continuous time dependent functions i_{mass} and i_{surf}. We measure the maximum numerical mass change using the quantity

$$e_{\text{mass}} := \max_{n=0,\dots,N} \left| i_{\text{mass}}^n - i_{\text{mass}}^0 \right|. \tag{4.4}$$

In Figure 4.10 we illustrate the change of mass and surface area over time. Recall that l_q and l_s denote the refinement level in time and space, respectively. In Figure 4.11 we depict the maximal mass error quantity e_{mass}.

Since the velocity field $\mathbf{w}$ is not divergence free, the surface area changes over time. The surface area increases until the spheres collide at $t = 1 - \frac{2}{3}2^{\frac{1}{3}} \approx 0.16$. After this

27
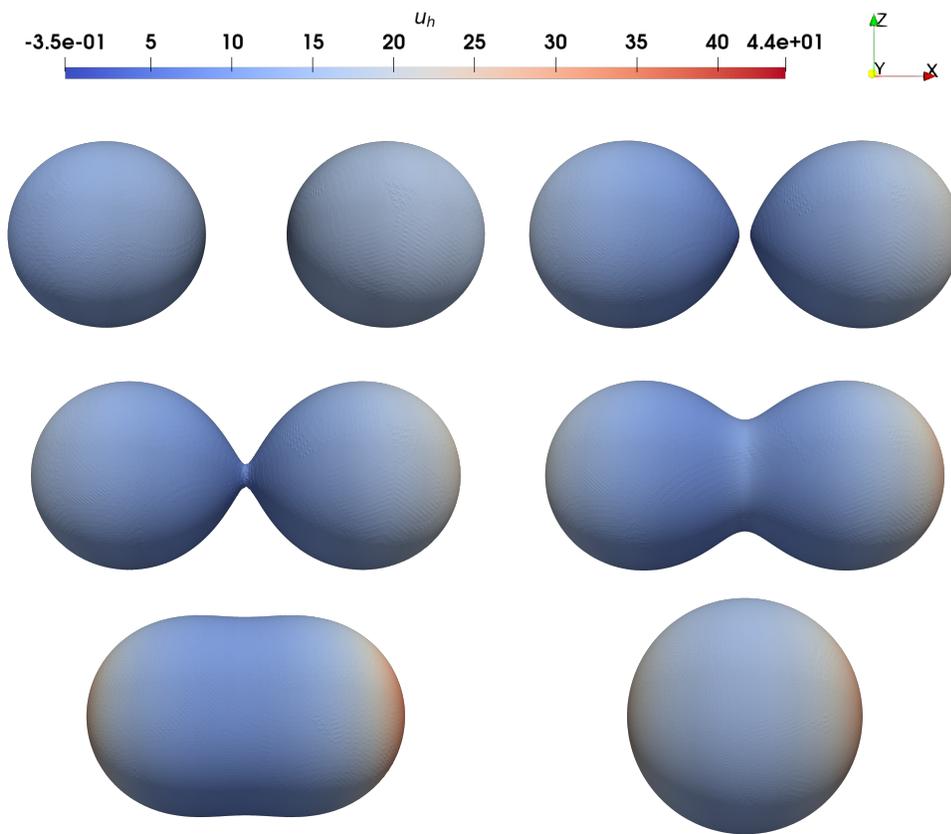
Fig. 4.9: A sketch of the evolution of the smooth sphere for $k = 1$. The discrete solution u_h at times $t \in \{0, \frac{20}{128}, \frac{21}{128}, \frac{34}{128}, \frac{71}{128}, 1\}$, is depicted in the images from top left to bottom right. Here, we have $\Delta t = 2^{-7}$ and $h = 2^{-5}$. The full animation can be found here.

collision the surface evolves to a sphere and the surface area decreases as a function of time, cf. Figure 4.10b. We find that the topological singularity at $t \approx 0.16$ has no significant influence on the approximate mass conservation. Under space and time mesh refinement the discrete mass $i_{\mathrm{mass}}(t)$ converges towards a constant function, cf. Figure 4.10a. The maximum mass error e_{mass} decreases with (optimal) second order, see Figure 4.11.

4.3.2. Disintegrating sphere. We consider a sphere that gets perforated and splits into eight smaller closed manifolds that are not connected. The level set equation of the perforated manifold is similar to the *decocube* considered in [5, 11]. As outer domain we consider $\Omega = (-3, 3)^3$ and as time interval we choose $[0, \frac{9}{4}]$. On $Q = \Omega \times [0, \frac{9}{4}]$ the evolution of the surface $\Gamma(t)$ is described by the zero level of the

28

(a) Total mass over time.
(b) Surface area over time.
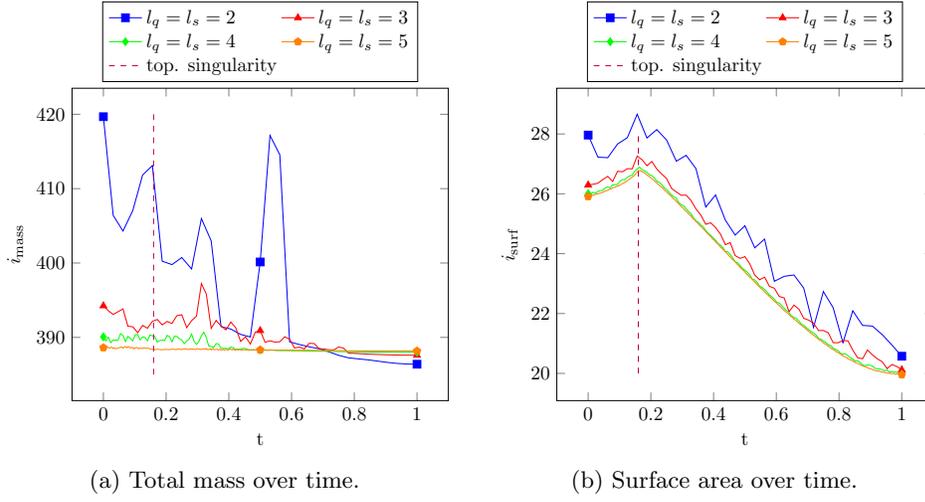
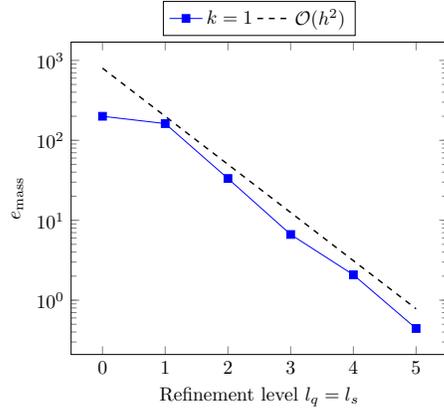Fig. 4.10: Merging spheres: the evolution of mass and surface area for $k = 1$.

Fig. 4.11: The maximum mass error (4.4). Note that due to the large values of i_{mass}, cf. Figure 4.10, the relative errors have reasonable sizes.

fifth order polynomial

$$\frac{\left(12x^4 + \left(8y^2 + 8z^2 - 76\right)x^2 + 12y^4 + \left(8z^2 - 76\right)y^2 + 12z^4 - 76z^2 + 205\right)t}{4}$$
$$+ 5x^2 + 5y^2 + 5z^2 - \frac{125}{4} =: \phi(\mathbf{x}, t), \quad (\mathbf{x}, t) = (x, y, z, t) \in Q.$$

On Q we construct a normal velocity field that transports $\Gamma(t)$ by

$$\mathbf{w} = -\frac{\partial \phi}{\partial t} \frac{\nabla \phi}{\|\nabla \phi\|^2}.$$
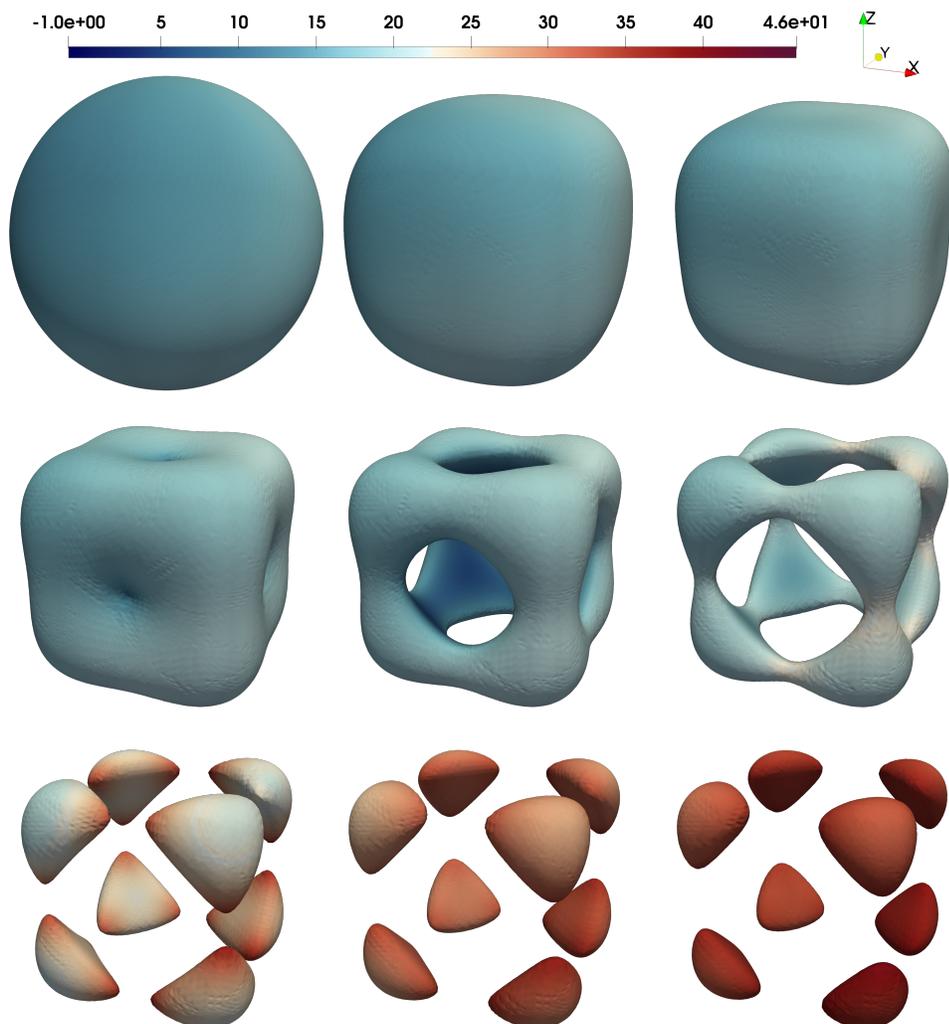
Fig. 4.12: A sketch of the evolution of the disintegrating sphere. The discrete solution u_h at times $t = \frac{iT}{8}$, $i \in \{0, \ldots, 8\}$, is depicted in the images from top left to bottom right. Here, we have $\Delta t = T \cdot 2^{-6}$ and $h = 2^{-4}$. The full animation can be found here.

We take the initial condition

$$u_0(\mathbf{x}) \coloneqq x + yz + 15, \quad \mathbf{x} \in \Gamma_h(0)$$

and $f = 0$ for the right hand side. We choose $k = 1$, $h_{\text{init}} = 2^{-1}$ and $\Delta t_{\text{init}} = 2^{-3}$. As in Section 4.3.1 we do not measure the discretization error $u^e - u_h$ because u is not known. In Figure 4.12 we show results of numerical simulations. At $t = 0$ we have a sphere which at first deforms to a rounded cube with small bumps. At $t \approx 0.61$ an additional sphere originates from a point singularity in the center of the outer surface. This sphere gets larger over time and merges with the outer surface at $t \approx 0.87$. The

two disjoint surfaces with genus 0 merge to a connected surface with genus 6 which we call *decocube*. At $t \approx 1.55$ this decocube splits into eight smaller surfaces with genus 0. Hence, in this example there are three points in time at which a topological singularity occurs. In Figures 4.13 and 4.14 close-ups of the surface near two different topological singularities are shown.
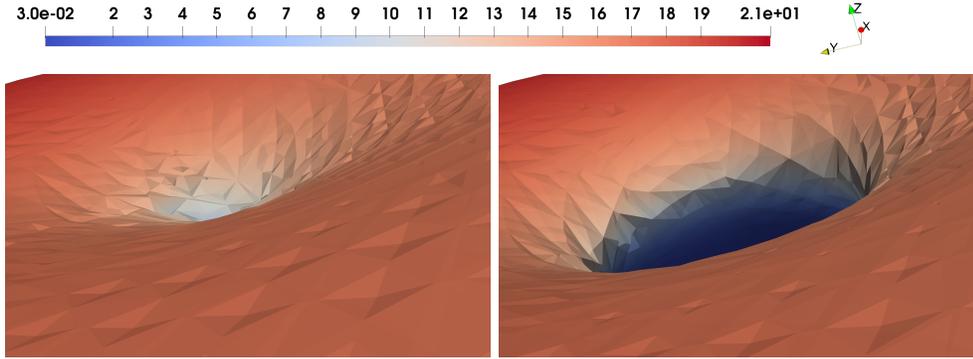

Fig. 4.13: Disintegrating sphere: on the left side we illustrate the solution on the surface $\Gamma_h(t)$, zoomed in near the spot where a hole emerges later. On the right side we have the same perspective one time step later after the hole arose. The images are colored with the discrete solution u_h at times $t = \frac{24}{64}T$ (left) and $t = \frac{25}{64}T$ (right). Here, we have $\Delta t = T \cdot 2^{-6}$ and $h = 2^{-4}$.
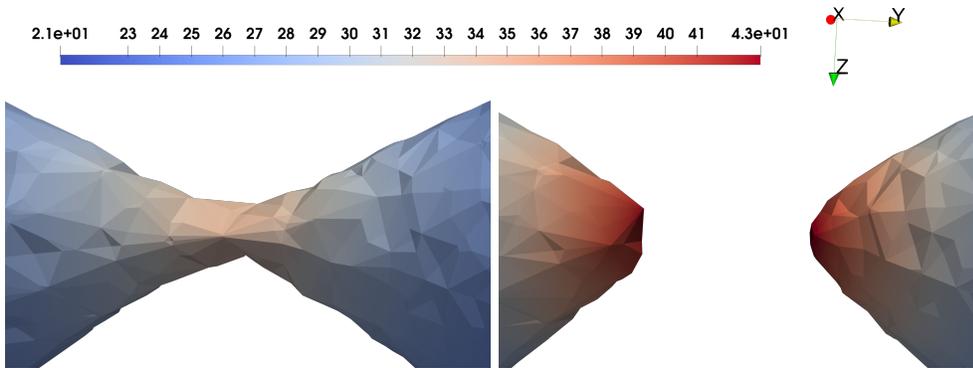

Fig. 4.14: Disintegrating sphere: on the left side we illustrate the solution on the surface $\Gamma_h(t)$, zoomed in near one of the edges of the decocube, where the surface splits later. On the right side we have the same perspective one time step later after the split happened. The images are colored with the discrete solution u_h at times $t = \frac{44}{64}T$ (left) and $t = \frac{45}{64}T$ (right). Here, we have $\Delta t = T \cdot 2^{-6}$ and $h = 2^{-4}$.

As in Section 4.2 we run numerical experiments to assess if and how accurate the discrete solution satisfies a mass conservation property. In Figure 4.15 we illustrate the evolution of mass and surface area over time. In Figure 4.15b we see that after the first singularity at $t \approx 0.61$ there is a strong increase of the surface area. The merging of the inner and outer surfaces, i.e. the formation of the decocube at $t \approx 0.87$, leads to a decrease of the surface area and the splitting at $t \approx 1.55$ does not change
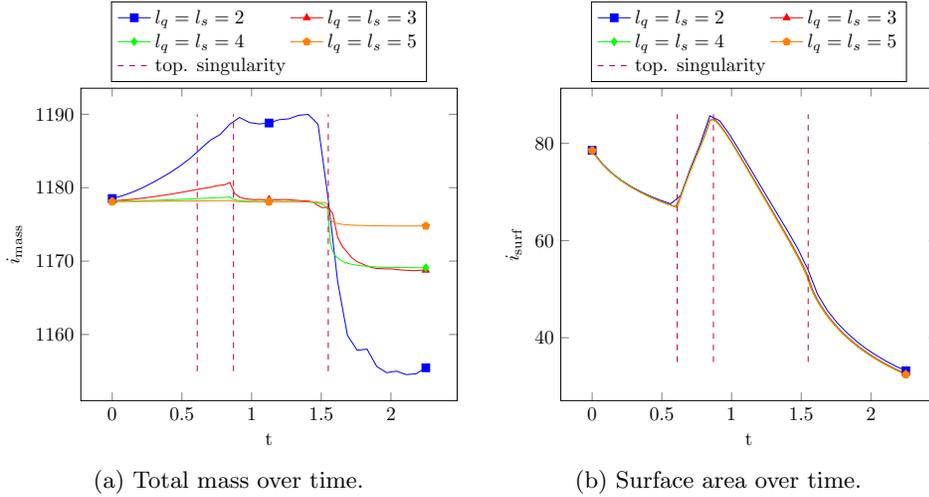
(a) Total mass over time.

(b) Surface area over time.

Fig. 4.15: Disintegrating sphere: the change of mass and surface area for $k = 1$.

this decreasing surface area behavior significantly. The mass on the inner sphere, which arises from a point singularity at $t \approx 0.61$, remains zero until the merging with the outer surface at $t \approx 0.87$. When the two surfaces merge we observe an abrupt change of mass, which gets significantly smaller for finer refinement levels. The third topological singularity, the splitting into eight smaller connected surfaces at $t \approx 1.55$, yields a (large) mass loss, even though the surface area does not change significantly at that time. This mass loss becomes smaller on finer levels. The mass function $i_{\mathrm{mass}}(t)$ seems to converge to a constant function, which is the expected behavior.

Finally we note one further robustness aspect. The results above show that both in the example of the merging spheres and of the disintegrating sphere one obtains stable and reasonably accurate results on very coarse meshes, e.g., level $l = 2$. This indicates that with this discretization method one preserves a main favourable property of the level set technique, namely that it can handle topological singularities in a very stable way.

5. Conclusions. In this paper we studied a Eulerian finite element method for the full discretization of scalar partial differential equations on evolving surfaces. The method uses standard space-time finite element spaces on a bulk mesh in combination with the trace technique known from the literature. The method is applied to the model problem (2.1) which describes diffusive transport on an evolving surface. For higher order accuracy we use a space-time parametric mapping. This leads to the discrete problems (3.30), parameterized with $\beta \in [0, 1]$. Key properties and implementation issues of this discretization method have been addressed. The performance of the method is illustrated in examples with one and two-dimensional surfaces. The results show that the method has (optimal) higher order convergence (in space and time) in cases with smoothly evolving surfaces and can handle topological singularities in a robust way.

REFERENCES

[1] J. Ahrens, B. Geveci, and C. Law, *Paraview: An end-user tool for large data visualization*, The visualization handbook, 717 (2005).

[2] A. Bonito, A. Demlow, and R. H. Nochetto, *Finite element methods for the Laplace-Beltrami operator*, 21 (2020), pp. 1–103.

[3] E. Burman, P. Hansbo, M. G. Larson, and A. Massing, *A cut discontinuous Galerkin method for the Laplace-Beltrami operator*, IMA J. Numer. Anal., 37 (2017), pp. 138–169.

[4] ——, *Cut finite element methods for partial differential equations on embedded manifolds of arbitrary codimensions*, ESAIM Math. Model. Numer. Anal., 52 (2018), pp. 2247–2282.

[5] K. Deckelnick, G. Dziuk, C. M. Elliott, and C.-J. Heine, *An h-narrow band finite-element method for elliptic equations on implicit surfaces*, IMA J. Numer. Anal., 30 (2010), pp. 351–376.

[6] G. Dziuk and C. M. Elliott, *Finite element methods for surface PDEs*, Acta Numer., 22 (2013), pp. 289–396.

[7] ——, *L^2-estimates for the evolving surface finite element method*, Math. Comp., 82 (2013), pp. 1–24.

[8] C. M. Elliott and T. Ranner, *A unified theory for continuous-in-time evolving finite element space approximations to partial differential equations in evolving domains*, IMA J. Numer. Anal., 41 (2021), pp. 1696–1845.

[9] D. Gilbarg and N. S. Trudinger, *Elliptic partial differential equations of second order*, Classics in Mathematics, Springer-Verlag, Berlin, 2001. Reprint of the 1998 edition.

[10] J. Grande, *Eulerian finite element methods for parabolic equations on moving surfaces*, SIAM J. Sci. Comput., 36 (2014), pp. B248–B271.

[11] ——, *Analysis of highly accurate finite element based algorithms for computing distances to level sets*, SIAM J. Numer. Anal., 55 (2017), pp. 376–399.

[12] J. Grande, C. Lehrenfeld, and A. Reusken, *Analysis of a high-order trace finite element method for PDEs on level set surfaces*, SIAM J. Numer. Anal., 56 (2018), pp. 228–255.

[13] J. Grande, M. Olshanskii, and A. Reusken, *A space-time fem for pdes on evolving surfaces*, in 11th World Congress on Computational Mechanics, A. H. Eugenio Oñate, Javier Oliver, ed., Barcelona, Spain, 2014, Eccomas.

[14] S. Gross and A. Reusken, *Numerical methods for two-phase incompressible flows*, vol. 40 of Springer Series in Computational Mathematics, Springer-Verlag, Berlin, 2011.

[15] P. Hansbo, M. G. Larson, and S. Zahedi, *Characteristic cut finite element methods for convection-diffusion problems on time dependent surfaces*, Comput. Methods Appl. Mech. Engrg., 293 (2015), pp. 431–461.

[16] ——, *A cut finite element method for coupled bulk-surface problems on time-dependent domains*, Comput. Methods Appl. Mech. Engrg., 307 (2016), pp. 96–116.

[17] F. Heimann, C. Lehrenfeld, and J. Preuss, *Geometrically higher order unfitted space-time methods for pdes on moving domains*, arXiv preprint arXiv:2202.02216, (2022).

[18] A. J. James and J. Lowengrub, *A surfactant-conserving volume-of-fluid method for interfacial flows with insoluble surfactant*, J. Comput. Phys., 201 (2004), pp. 685–722.

[19] B. Kovács, *High-order evolving surface finite element method for parabolic problems on evolving surfaces*, IMA J. Numer. Anal., 38 (2018), pp. 430–459.

[20] C. Lehrenfeld, *High order unfitted finite element methods on level set domains using isoparametric mappings*, Comput. Methods Appl. Mech. Engrg., 300 (2016), pp. 716–733.

[21] C. Lehrenfeld, F. Heimann, J. Preuss, and H. von Wahl, *ngsxfem: Add-on to ngsolve for geometrically unfitted finite element discretizations*, J. Open Source Softw., 6 (2021), p. 3237.

[22] C. Lehrenfeld, M. A. Olshanskii, and X. Xu, *A stabilized trace finite element method for partial differential equations on evolving surfaces*, SIAM J. Numer. Anal., 56 (2018), pp. 1643–1672.

[23] M. A. Olshanskii and A. Reusken, *Error analysis of a space-time finite element method for solving PDEs on evolving surfaces*, SIAM J. Numer. Anal., 52 (2014), pp. 2092–2120.

[24] M. A. Olshanskii, A. Reusken, and X. Xu, *An Eulerian space-time finite element method for diffusion problems on evolving surfaces*, SIAM J. Numer. Anal., 52 (2014), pp. 1354–1377.

[25] P. Oswald, *On a BPX-preconditioner for P1 elements*, Computing, 51 (1993), pp. 125–133.

[26] J. Preuss, *Higher order unfitted isoparametric space-time FEM on moving domains*, master's

thesis, NAM, University of Göttingen, 2018.

[27] H. SASS, *Space-time Trace Finite Element Methods for Partial Differential Equations on Evolving Surfaces*, phd thesis, RWTH Aachen University, 2022.

[28] J. SCHÖBERL, *NETGEN an advancing front 2d/3d-mesh generator based on abstract rules*, Comput. Vis. Sci., 1 (1997), pp. 41–52.

[29] ———, *C++11 implementation of finite elements in ngsolve*, asc report no. 30/2014. tech. rep., Institute for Analysis and Scientific Computing, TU Wien, 2014.

[30] E. WANG, Q. ZHANG, B. SHEN, G. ZHANG, X. LU, Q. WU, AND Y. WANG, *Intel math kernel library*, in High-Performance Computing on the Intel® Xeon Phi™, Springer, 2014, pp. 167–188.

[31] S. ZAHEDI, *A space-time cut finite element method with quadrature in time*, in Geometrically unfitted finite element methods and applications, vol. 121 of Lect. Notes Comput. Sci. Eng., Springer, Cham, 2017, pp. 281–306.

Appendix A. Proof of Theorem 3.2.

Proof. By e.g. [9] we know the partial integration rule for a smooth function g defined on an arbitrary $K_S \in \mathcal{T}_{S_h^n}$

$$\int_{K_S} \nabla_{S_h,i} g \, \mathrm{d}\sigma_h = \int_{K_S} g \kappa_{K_S} (\mathbf{n}_{S_h})_i \, \mathrm{d}\sigma_h + \int_{\partial K_S} g (\boldsymbol{\nu}_h)_i \, \mathrm{d}F, \quad i \in \{1, \ldots, 4\}.$$

Here, $\nabla_{S_h,i} g$ denotes the i-th Cartesian component of $\nabla_{S_h} g$. The function κ_{K_S} denotes the mean curvature on K_S. Replacing g by $\frac{uv}{\alpha_h} \left(\mathbf{P}_{S_h} \mathbf{w}_S \right)_i$, summing over i and using the product rule we get

$$\int_{K_S} u \, \mathrm{div}_{S_h} \left(\frac{v}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) + \left(\frac{v}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \cdot \nabla_{S_h} u \, \mathrm{d}\sigma_h$$
$$= \int_{K_S} \kappa_{K_S} \frac{uv}{\alpha_h} \left(\mathbf{P}_{S_h} \mathbf{w}_S \right) \cdot \mathbf{n}_{S_h} \, \mathrm{d}\sigma_h + \int_{\partial K_S} \frac{hv}{\alpha_h} \left(\mathbf{P}_{S_h} \mathbf{w}_S \right) \cdot \boldsymbol{\nu}_h \, \mathrm{d}F. \tag{A.1}$$

The first term on the right hand side vanishes, since the vectors in the dot product are orthogonal. We calculate using (A.1) and the definition (3.16) of the discrete material derivative

$$\int_{S_h^n} \frac{1}{\alpha_h} \mathring{u} v \, \mathrm{d}\sigma_h = \sum_{K_S \in \mathcal{T}_{S_h^n}} \int_{K_S} \frac{v}{\alpha_h} \left(\mathbf{P}_{S_h} \mathbf{w}_S \right) \cdot \nabla_{S_h} u \, \mathrm{d}\sigma_h$$

$$= \sum_{K_S \in \mathcal{T}_{S_h^n}} \left[-\int_{K_S} u \, \mathrm{div}_{S_h} \left(\frac{v}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \mathrm{d}\sigma_h + \int_{\partial K_S} \frac{uv}{\alpha_h} \left(\mathbf{P}_{S_h} \mathbf{w}_S \right) \cdot \boldsymbol{\nu}_h \, \mathrm{d}F \right].$$

We continue using the product rule and the tangency of $\boldsymbol{\nu}_h|_{K_S}$ to K_S

$$\int_{S_h^n} \frac{1}{\alpha_h} \mathring{u} v \, \mathrm{d}\sigma_h = \sum_{K_S \in \mathcal{T}_{S_h^n}} \left[-\int_{K_S} uv \, \mathrm{div}_{S_h} \left(\frac{1}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \right.$$

$$\left. -\frac{u}{\alpha_h} \left(\mathbf{P}_{S_h} \mathbf{w}_S \right) \cdot \nabla_{S_h} v \, \mathrm{d}\sigma_h + \int_{\partial K_S} \frac{uv}{\alpha_h} \mathbf{w}_S \cdot \boldsymbol{\nu}_h \, \mathrm{d}F \right].$$

For the second term on the right hand side we use the tangency of $\mathbf{P}_{S_h} \mathbf{w}_S$ to S_h and the definition of the weak material derivative (3.16). We split the sum in the third term on the right-hand side. In case $\partial K_S \subset \partial S_h^n$ we know $\boldsymbol{\nu}_h = \boldsymbol{\nu}_\partial$ at the top boundary of the time slab and $\boldsymbol{\nu}_h = -\boldsymbol{\nu}_\partial$ at the bottom boundary of the time slab.

We obtain

$$
\int_{S_h^n} \frac{1}{\alpha_h} \mathring{u} v \, \mathrm{d}\sigma_h = -\int_{S_h^n} \frac{1}{\alpha_h} u \mathring{v} \, \mathrm{d}\sigma_h + \sum_{F \in \mathcal{F}_T^n} \int_F \frac{u_-^n v_-^n}{\alpha_{h,-}^n} \mathbf{w}_S \cdot (\boldsymbol{\nu}_\partial)_-^n \, \mathrm{d}F
$$

$$
- \sum_{F \in \mathcal{F}_B^n} \int_F \frac{u_+^{n-1} v_+^{n-1}}{\alpha_{h,+}^{n-1}} \mathbf{w}_S \cdot (\boldsymbol{\nu}_\partial)_+^{n-1} \, \mathrm{d}F + \sum_{F \in \mathcal{F}_I^n} \int_F uv \mathbf{w}_S \cdot \left[\frac{1}{\alpha_h} \right]_{\boldsymbol{\nu}} \, \mathrm{d}F
$$

$$
- \sum_{K_S \in \mathcal{T}_{S_h^n}} \int_{K_S} uv \, \mathrm{div}_{S_h} \left(\frac{1}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \, \mathrm{d}\sigma_h.
$$

Due to the definition of $\mathcal{F}_T^n$ and $\mathcal{F}_B^n$ we get

$$
\int_{S_h^n} \frac{1}{\alpha_h} \mathring{u} v \, \mathrm{d}\sigma_h = -\int_{S_h^n} \frac{1}{\alpha_h} u \mathring{v} \, \mathrm{d}\sigma_h + \int_{\Gamma_h^n(t_n)} \frac{u_-^n v_-^n}{\alpha_{h,-}^n} \mathbf{w}_S \cdot (\boldsymbol{\nu}_\partial)_-^n \, \mathrm{d}s_h
$$

$$
- \int_{\Gamma_h^n(t_{n-1})} \frac{u_+^{n-1} v_+^{n-1}}{\alpha_{h,+}^{n-1}} \mathbf{w}_S \cdot (\boldsymbol{\nu}_\partial)_+^{n-1} \, \mathrm{d}s_h
$$

$$
+ \sum_{F \in \mathcal{F}_I^n} \int_F uv \mathbf{w}_S \cdot \left[\frac{1}{\alpha_h} \right]_{\boldsymbol{\nu}} \, \mathrm{d}F - \sum_{K_S \in \mathcal{T}_{S_h^n}} \int_{K_S} uv \, \mathrm{div}_{S_h} \left(\frac{1}{\alpha_h} \mathbf{P}_{S_h} \mathbf{w}_S \right) \, \mathrm{d}\sigma_h.
$$

Finally, using the definition $R = \frac{1}{\alpha_h} \mathbf{w}_S \cdot \boldsymbol{\nu}_\partial$ completes the proof. $\square$