Proceedings of ALGORITMY 2016
pp. 44–54

AN UNFITTED DISCONTINUOUS GALERKIN SCHEME FOR CONSERVATION LAWS ON EVOLVING SURFACES*

CHRISTIAN ENGWER†, THOMAS RANNER‡, AND SEBASTIAN WESTERHEIDE†

Abstract. Motivated by considering partial differential equations arising from conservation laws posed on evolving surfaces, a new numerical method for an advection problem is developed and simple numerical tests are performed. The method is based on an unfitted discontinuous Galerkin approach where the surface is not explicitly tracked by the mesh which means the method is extremely flexible with respect to geometry. Furthermore, the discontinuous Galerkin approach is well-suited to capture the advection driven by the evolution of the surface without the need for a space-time formulation, back-tracking trajectories or streamline diffusion. The method is illustrated by a one-dimensional example and numerical results are presented that show good convergence properties for a simple test problem.

Key words. Discontinuous Garlerkin; unfitted finite elements; surface partial differential equations; conservation laws.

AMS subject classifications. 35L02, 35L65, 35Q90, 35R01, 65N12, 65N30

1. Introduction.

Interest in the study of partial differential equations on evolving surfaces has grown in recent years with applications in many areas including materials science (e.g. diffusion of species along grain boundaries [8]), fluid dynamics (e.g. surface active agents along the interface between two fluids [18]) and cell biology (e.g. cell motility involving the processes on the cell membrane [19]), for example. See the review of [11] for a more detailed list. In this work, we derive a new numerical scheme for an essential conservation law on evolving hypersurfaces and present first numerical results.

1.1. Problem.

As a model problem, we consider the evolution of a conserved quantity on an evolving curve or surface. Fix $T > 0$ and let $\{\Gamma(t)\}_{t\in[0,T]}$ be a time-dependent, connected, compact, smooth n-dimensional hypersurface embedded in $\mathbb{R}^{n+1}$ for $n \in \{1, 2\}$, $\vec{\nu}(\cdot, t)\colon \Gamma(t) \to \mathbb{R}^{n+1}$ denote a field of unit normal vectors to $\Gamma(t)$, and write $\Gamma_0 = \Gamma(0)$. We consider two different descriptions of the surface. First, we say $\Gamma(t)$ is defined by a diffeomorphic parametrization $G(\cdot, t)\colon \Gamma_0 \to \mathbb{R}^{n+1}$ so that $\Gamma(t) = G(\Gamma_0, t)$ for $t \in [0, T]$. The map G defines a field $\vec{w}$ which describes the velocity of $\Gamma(t)$, precisely by $\partial_t G(\cdot, t) = \vec{w}(G(\cdot, t), t)$. Second, we describe $\Gamma(t)$ as the zero level set of a smooth function $\Phi(\cdot, t)\colon \mathbb{R}^{n+1} \to \mathbb{R}$ so that $\Gamma(t) = \{x \in \mathbb{R}^{n+1} : \Phi(x, t) = 0\}$. We will use the diffeomorphic parametrization to define the equations we solve and the level set function to define the geometry in our computational scheme.

Consider a control volume $\mathcal{M}(t) \subset \Gamma(t)$ which is the image of a control volume $\mathcal{M}_0 \subset \Gamma_0$ under the flow G, i.e. $\mathcal{M}(t) = G(\mathcal{M}_0, t)$. Let u denote a time-dependent field on $\Gamma(t)$ which satisfies a conservation law of the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\mathcal{M}(t)} u \,\mathrm{d}\sigma = -\int_{\partial\mathcal{M}(t)} \vec{q}\cdot\vec{\mu}\,\mathrm{d}\xi, \tag{1.1}$$

*The work was partially supported by the German Research Foundation (DFG), grant EN 1042/4-1 and by the UK Engineering and Physical Sciences Research Council, grant EPSRC EP/J004057/1.

†Institute for Computational and Applied Mathematics, University of Münster, Germany. (christian.engwer@uni-muenster.de, sebastian.westerheide@uni-muenster.de)

‡School of Computing, EC Stoner Building, University of Leeds, UK. (T.Ranner@leeds.ac.uk)

where $\vec{q}$ is a tangential flux on $\Gamma(t)$ and $\vec{\mu}$ is the co-normal vector to $\partial\mathcal{M}(t)$ ($\vec{\mu}$ is normal to $\partial\mathcal{M}(t)$ and $\vec{\mu}\cdot\vec{\nu}=0$). Applying a transport formula to the left hand side of (1.1) and the divergence theorem for hypersurfaces to the right hand side yields

$$\int_{\mathcal{M}(t)} \partial^\bullet u + u\nabla_\Gamma\cdot\vec{w} + \nabla_\Gamma\cdot\vec{q}\,\mathrm{d}\sigma = 0, \tag{1.2}$$

where $\partial^\bullet u$ is the material derivative of u and $\nabla_\Gamma\cdot$ denotes the tangential divergence operator. See Section 2.1 for details. We describe equation (1.2) as a *local* conservation law for u or a *global* conservation law in the case $\mathcal{M}(t)=\Gamma(t)$.

Since (1.2) holds for any $\mathcal{M}(t)$, we arrive at a pointwise conservation law. We wish to find a time-dependent surface field $u\colon \bigcup_{t\in[0,T]}\Gamma(t)\times\{t\}\to\mathbb{R}$ with

$$\partial^\bullet u + u\nabla_\Gamma\cdot\vec{w} + \nabla_\Gamma\cdot\vec{q} = 0 \qquad \text{on } \bigcup_{t\in[0,T]}\Gamma(t)\times\{t\}. \tag{1.3}$$

For more details on the notation, see [11].

In this work, we aim to derive a new numerical method using an implicit representation of the surface for which we can show a discrete analogue to the global conservation law. To solve problems of this form, we are required to consider two separate problems. We have to discretize the surface partial differential equation at each given time step and we have to evaluate the transport from the old surface to the new surface. Either these two components are computed in a splitting approach and discretized separately, or computed directly in a coupled manner. In either case, we must understand the effects of the flux $\vec{q}$ and the advection of the moving surface separately. Previously (see the following section), attention has been paid to the stationary case and the treatment of the flux $\vec{q}$. In this work, we are interested in the advection driven by the moving surface. We thus restrict to the case $\vec{q}=0$ and a single time step, which simplifies considerations. Extensions to the truly time-dependent case and $\vec{q}\neq 0$ will be considered in future work.

1.2. Previous approaches. Previous computational approaches lie in two categories. The first is based on a moving mesh approach and the second is based on a static mesh.

For surface PDEs, the basic ideas of the moving mesh approach date back to the ideas of the finite element method for elliptic PDEs on stationary surfaces which was presented in [9]. A smooth surface is approximated by a union of elements (usually simplices but also possibly quadradrilaterals) whose vertices move according to a globally defined smooth velocity. It was applied by [10], using a finite element discretization for an advection-diffusion problem, and by [12], using a finite volume method for conservation laws. Moving mesh schemes have proven to be useful in many practical situations and can be shown analytically to be stable and accurate, or preserve various conservative properties. Their main disadvantage is that the moving meshes may degenerate, leading to frequent remeshing [13]. Extensions have been presented recently by [15, 16] that reduce the need for remeshing by allowing mesh points to move with a different non-physical velocity. However, in many situations preserving a good mesh remains very challenging.

In the static mesh approach, an arbitrary fixed background mesh is used and the evolution of the surface is defined implicitly. One approach is to use a level set function together with an unfitted finite element method where the computational domain consists of partial cut-cell elements [6, 20, 7, 22]. The authors of [21] apply

a space-time formulation to produce a stable, accurate method, but the method is only approximately globally conservative and requires the use of complicated four-dimensional space-time elements which are not available for geometries other than simplices. An alternative method proposed by [7] uses a semi-Lagrangian formulation that employs a non-local right hand side. The resulting scheme recovers a globally mass conservation property and performs well in practice for a test problem on a curve, but lacks analysis and the non-local term is expensive to compute, especially in three space dimensions. A phase field representation of the interface was used by [14, 23]. The authors of [14] use a narrow band formulation and recover a discrete analogue to the continuous level global conservation law but must use stream line diffusion to stabilize the scheme.

A key difficulty to overcome with the static mesh approach is to derive a stable scheme that adequately treats the advection driven by the evolution of the surface. The moving mesh approach implicitly deals with this problem by using moving basis functions. The advective flux usually has a component orthogonal to the surface. Therefore, any other fluxes (e.g. diffusive surface fluxes) can not stabilize the method. Furthermore, the previous ideas based on finite element methods are not well-suited to advection-dominated problems. Extra complications such as space-time formulations, semi-Lagrangian terms or streamline diffusion are required. In this work, we use a more suitable discontinuous Galerkin (DG) method which can naturally handle advection or advection-dominated problems.

2. Computational approach. Our approach is based on reformulating problem (1.3) as a sequence of bulk advection problems with singular source and sink terms and applying the unfitted discontinuous Galerkin (UDG) method [3] on a static bulk mesh. This mesh can be chosen independent of the evolving surface, the surface is not explicitly represented by the mesh. Our formulation guarantees a globally conservative scheme.

2.1. Preliminaries. Let U be a polygonal domain in $\mathbb{R}^{n+1}$ which contains $\Gamma(t)$ for all times $t \in [0,T]$. We suppose that for each time $t \in [0,T]$, $\Gamma(t)$ is defined as the zero level set of a smooth function $\Phi(\cdot,t)\colon U \to \mathbb{R}$ and assume $\nabla\Phi(x,t) \neq 0$ for $x \in U$, $t \in [0,T]$. Let $\vec{\nu}$ be the field of unit normal vectors to $\Gamma(t)$ oriented by $\vec{\nu} = \nabla\Phi/|\nabla\Phi|$. The normal component of the velocity field is given by $\vec{w}\cdot\vec{\nu} = -\Phi_t/|\nabla\Phi|$.

The tangential gradient of a smooth surface field $\eta\colon \Gamma(t) \to \mathbb{R}$ can be defined as

$$\nabla_\Gamma \eta := \nabla\tilde{\eta} - \big(\nabla\tilde{\eta}\cdot\vec{\nu}\big)\vec{\nu},$$

where $\tilde{\eta}$ is a spatially differentiable extension of η to a neighborhood of $\Gamma(t)$ and $\nabla\tilde{\eta}$ is its gradient with respect to the ambient Cartesian coordinates. The tangential gradient $\nabla_\Gamma\eta$ has $n+1$ components $(\underline{\mathrm{D}}_1\eta,\dots,\underline{\mathrm{D}}_{n+1}\eta)$. For a smooth surface vector field $\vec{v}$, the tangential divergence is given by

$$\nabla_\Gamma\cdot\vec{v} = \sum_{i=1}^{n+1} \underline{\mathrm{D}}_i\vec{v}_i.$$

The material derivative of a smooth, time-dependent surface field $\eta(\cdot,t)\colon \Gamma(t) \to \mathbb{R}$ is defined as

$$\partial^\bullet\eta := \partial_t\tilde{\eta} + \vec{w}\cdot\nabla\tilde{\eta},$$

where again $\partial_t\tilde{\eta}$ and $\nabla\tilde{\eta}$ are Cartesian derivatives of a differentiable extension $\tilde{\eta}$ of η to a space-time neighborhood of $\{\Gamma(t)\}_{t\in[0,T]}$. The material derivative describes the

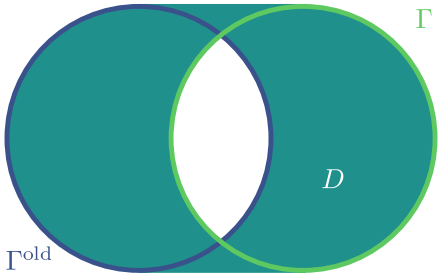

FIG. 2.1. *An example of the geometry configuration. Here, a circle is translated with a horizontal velocity between time steps. Note that a time step with a large time step size τ is illustrated.*

variation of η with respect to the evolution of the surface. It also has a key role in the following transport relation for evolving material surfaces, see e.g. [5, 10]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Gamma(t)} \eta \,\mathrm{d}\sigma = \int_{\Gamma(t)} \partial^\bullet \eta + \eta \nabla_\Gamma \cdot \vec{w} \,\mathrm{d}\sigma. \tag{2.1}$$

2.2. Motivation. Proceeding formally, we multiply problem (1.3) (with $\vec{q} = 0$) by a smooth function $\varphi \colon U \to \mathbb{R}$, integrate over $\Gamma(t)$ for $t \in [0, T]$ and apply transport relation (2.1) to see

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\Gamma(t)} u\varphi \,\mathrm{d}\sigma - \int_{\Gamma(t)} u\partial^\bullet \varphi \,\mathrm{d}\sigma = 0.$$

Since φ does not depend on time, we have $\partial^\bullet \varphi = \vec{w} \cdot \nabla \varphi$.

We fix a time $t^* \in [0, T]$ and $\tau > 0$ small enough that $t^* - \tau \geq 0$. Integrating in time over the time interval $[t^* - \tau, t^*]$ yields

$$\int_{\Gamma(t^*)} u(t^*)\varphi \,\mathrm{d}\sigma - \int_{\Gamma(t^*-\tau)} u(t^* - \tau)\varphi \,\mathrm{d}\sigma - \int_{t^*-\tau}^{t^*} \left(\int_{\Gamma(t)} u\vec{w} \cdot \nabla \varphi \,\mathrm{d}\sigma \right) \mathrm{d}t = 0.$$

We will approximate the space-time integral in the third term by a scaled bulk integral over the spatial-only domain

$$D := \bigcup\nolimits_{t \in [t^*-\tau, t^*]} \Gamma(t)\,,$$

which corresponds to the projection of the space-time domain to spatial-only coordinates. An example of the geometric setup is shown in Figure 2.1.

Denoting an extension of a function $\psi(\cdot, t^*)$ by $\psi^e \colon D \to \mathbb{R}$, we have

$$\begin{aligned} -\int_{t^*-\tau}^{t^*} \left(\int_{\Gamma(t)} u\vec{w} \cdot \nabla \varphi \,\mathrm{d}\sigma \right) \mathrm{d}t &\approx -\tau \int_{\Gamma(t^*)} u\vec{w} \cdot \nabla \varphi \,\mathrm{d}\sigma \\ &\approx -\frac{\tau}{\gamma} \int_{-\gamma^-}^{\gamma^+} \left(\int_{\Gamma_s(t^*)} u^e \vec{w}^e \cdot \nabla \varphi \,\mathrm{d}\sigma \right) \mathrm{d}s = -\frac{\tau}{\gamma} \int_D u^e \vec{w}^e \left| \nabla \Phi(\cdot, t^*) \right| \cdot \nabla \varphi \,\mathrm{d}x, \end{aligned}$$

where we employ the right-hand rectangle method, an approximation of the resulting surface integral using the level sets $\Gamma_s(t^*) := \{x \in D : \Phi(x, t^*) = s\}$ of $\Phi(\cdot, t^*)\big|_D$ and apply the coarea formula, while assuming that the time step size τ as well as

$\gamma := \gamma^- + \gamma^+$ with $\gamma^- := -\min_D \Phi(\cdot, t^*)$ and $\gamma^+ := \max_D \Phi(\cdot, t^*)$ are small. Note that the assumption on γ is reasonable for a small time step size. If $\vec{w}$ is constant and Φ is a signed distance function, i.e. $|\nabla\Phi| \equiv 1$, γ corresponds to the travel distance of an individual point on the surface.

In order to be able to discretize using flux-based numerical schemes, such as unfitted DG schemes, we use integration by parts to reformulate this integral via the divergence of an advective flux tested with φ. As an approximate reformulation of problem (1.3) over the time interval $[t^* - \tau, t^*]$, we therefore obtain the following stationary problem: seek $u^e \colon D \to \mathbb{R}$ with

$$\begin{aligned}(2.2)\quad &\int_\Gamma u^e \varphi \,\mathrm{d}\sigma - \int_{\Gamma^{\mathrm{old}}} u^{\mathrm{old}} \varphi \,\mathrm{d}\sigma \\ &\quad + \frac{\tau}{\gamma} \int_D \nabla \cdot \left(u^e \vec{w}^e \big| \nabla\Phi(\cdot, t^*) \big| \right) \varphi \,\mathrm{d}x - \frac{\tau}{\gamma} \int_{\partial D} u^e \vec{w}^e \big| \nabla\Phi(\cdot, t^*) \big| \cdot \nu_D \varphi \,\mathrm{d}\sigma = 0,\end{aligned}$$

for all smooth functions $\varphi \colon D \to \mathbb{R}$. Here and in the following, we use the notation $\Gamma = \Gamma(t^*)$, $\Gamma^{\mathrm{old}} = \Gamma(t^* - \tau)$ and $u^{\mathrm{old}} = u(t^* - \tau)$, and ν_D denotes the outward pointing unit normal vector field to ∂D.

REMARK 1. *In the special case $\partial D = \Gamma^{old} \dot\cup \Gamma$ (e.g. a geometrical setup as in Figure 2.2), problem* (2.2) *is consistent with the following bulk advection problem in strong form:*

$$\begin{aligned} \nabla \cdot \left(u^e \vec{w}^e \big| \nabla\Phi(\cdot, t^*) \big| \right) &= 0 && \text{in} \quad D, \\ u^e \vec{w}^e \big| \nabla\Phi(\cdot, t^*) \big| \cdot \nu_D &= -\frac{\gamma}{\tau} u^{old} && \text{on} \quad \Gamma^{old}, \\ u^e \vec{w}^e \big| \nabla\Phi(\cdot, t^*) \big| \cdot \nu_D &= \frac{\gamma}{\tau} u^e && \text{on} \quad \Gamma. \end{aligned}$$

2.3. Unfitted discrete geometry. In order to discretize problem (2.2), we start by discretizing the geometry. Let $\tilde{\mathcal{T}}_h$ be a shape regular decomposition of U into closed elements, either tetrahedra or hexahedra for $n = 2$, triangles or quadrilaterals for $n = 1$, and denote by h the maximum element size. Let X_h be the space of piecewise linear (for simplices), bilinear (for quadrilaterals) or trilinear (for hexahedra) continuous functions over $\tilde{\mathcal{T}}_h$ and denote by I_h interpolation of functions in $C(U)$ into X_h. We will write $\Phi_h(\cdot, t) = I_h \Phi(\cdot, t)$ for a discrete level set function and set

$$\begin{aligned} \Gamma_h &:= \{x \in U : \Phi_h(x, t^*) = 0\}, \quad \Gamma_h^{\mathrm{old}} := \{x \in U : \Phi_h(x, t^* - \tau) = 0\} \\ D_h &:= \{x \in U : \text{there exists } t \in [t^* - \tau, t^*] \text{ with } \Phi_h(x, t) = 0\}. \end{aligned}$$

Note that $\Gamma_h^{\mathrm{old}}, \Gamma_h \subset D_h$ since D_h is a closed set. We also use the notation

$$\mathcal{T}_h := \left\{ K \in \tilde{\mathcal{T}}_h : \mathrm{meas}(K \cap D_h) > 0 \right\}.$$

An example is shown in Figure 2.2. It will be useful to consider the restriction of the decomposition $\mathcal{T}_h$ to the computational domain D_h:

$$\hat{\mathcal{T}}_h := \left\{ \hat{K} = K \cap D_h : K \in \mathcal{T}_h \right\}.$$

We note that the $\hat{K} \in \hat{\mathcal{T}}_h$ are arbitrary shaped elements and call those elements cut cells. In general, they are not either shape regular or even convex. We will also

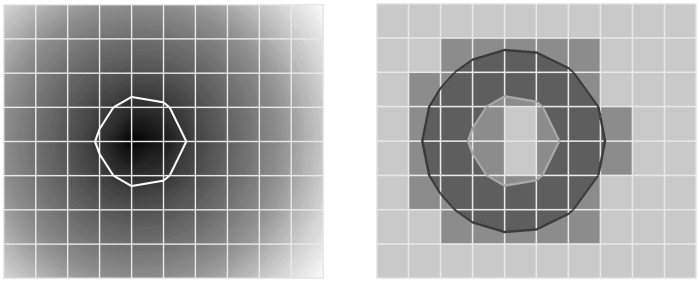

FIG. 2.2. *Example of an unfitted discrete geometry. The left image shows a discrete level set function Φ_h and its zero level set in white when Γ is a circle. The right image shows an example of $\hat{\mathcal{T}}_h$, Γ_h, $\mathcal{T}_h$, the cut cell mesh $\hat{\mathcal{T}}_h$ for computational domain D_h, and Γ_h^{old} (from light gray to dark gray, where pixels which represent meshes have the color of the most specialized mesh) when the initial curve Γ^{old} is shrunk to a smaller circle Γ.*

consider the internal faces of the cut cell mesh $\hat{\mathcal{T}}_h$, denoted by $\hat{\mathcal{E}}_h$. The set $\hat{\mathcal{E}}_h$ is often called the skeleton of the mesh. To each internal face $\hat{E} \in \hat{\mathcal{E}}_h$, which is the intersection of two elements $\hat{K}^+, \hat{K}^- \in \hat{\mathcal{T}}_h$, we assign a unit normal vector field $\vec{\nu}_{\hat{K}^+} = -\vec{\nu}_{\hat{K}^-}$ and arbitrarily choose $\vec{\nu}_{\hat{E}} = \vec{\nu}_{\hat{K}^+}$.

REMARK 2. *Our solution will be defined over the union of elements in $\mathcal{T}_h$, but we are only interested in the values of our solution variable over the sharp interface approximation Γ_h. The integrals in the method will be computed over either the sharp interface approximation Γ_h or the unfitted cut cell mesh $\hat{\mathcal{T}}_h$. This is a similar approach to [7], but different to [20], thus avoiding difficulties in defining our discrete spaces and constructing a natural basis.*

2.4. Method. We introduce the discrete spaces V_h given by

$$V_h := \big\{\varphi_h = (\varphi_K)_{K\in\mathcal{T}_h} : \varphi_K \in P^k(K) \text{ for all } K \in \mathcal{T}_h\big\},$$

where $P^k(K)$ denotes the space of piecewise polynomials of degree k over the volumetric domain K. Note that functions in V_h are discontinuous and do not take a unique value along the faces $\hat{\mathcal{E}}_h$, in general. On each face $\hat{E} \in \hat{\mathcal{E}}_h$ with adjacent elements $\hat{K}^+, \hat{K}^- \in \hat{\mathcal{T}}_h$ as defined above, we define the jump of a function $\varphi_h \in V_h$ as $\big[\!\big[\varphi_h\big]\!\big] := \varphi_h|_{\hat{K}^+} - \varphi_h|_{\hat{K}^-}$ and its average as $\big\{\varphi_h\big\} := \frac{1}{2}\big(\varphi_h|_{\hat{K}^+} + \varphi_h|_{\hat{K}^-}\big)$.

We discretize problem (2.2) by approximating u^e by $u_h \in V_h$, the test functions φ by $\varphi_h \in V_h$, Φ by Φ_h and $\vec{w}^e$ by a discrete approximation $\vec{w}_h$ (which is given in more detail in the sequel). Integrals over Γ and Γ^{old} are approximated by integrals over Γ_h and Γ_h^{old}, respectively, and integrals over D are approximated by integrals over D_h. Finally, we split the bulk integral into a sum of integrals over the cut cells $\hat{K} \in \hat{\mathcal{T}}_h$ and integrate by parts on each $\hat{K}$. Using classical upwind stabilization (as in [4]), we obtain the following unfitted DG scheme:

SCHEME 1. *Given $u_h^{old} \in L^2(\Gamma_h^{old})$, find $u_h \in V_h$, such that*

$$\begin{aligned}(2.3)\quad &\int_{\Gamma_h} u_h\varphi_h \,\mathrm{d}\sigma_h + \frac{\tau}{\gamma}\Big(\sum_{\hat{E}\in\hat{\mathcal{E}}_h}\int_{\hat{E}} u_h^{\uparrow}\big[\!\big[\varphi_h\big]\!\big]\big\{\vec{w}_h|\nabla\Phi_h(\cdot,t^*)|\big\}\cdot\vec{\nu}_{\hat{E}}\,\mathrm{d}\sigma_h\\ &- \sum_{\hat{K}\in\hat{\mathcal{T}}_h}\int_{\hat{K}} u_h\vec{w}_h|\nabla\Phi_h(\cdot,t^*)|\cdot\nabla\varphi_h\,\mathrm{d}x_h\Big) = \int_{\Gamma_h^{old}} u_h^{old}\varphi_h\,\mathrm{d}\sigma_h \quad \textit{for all } \varphi_h \in V_h,\end{aligned}$$

where $u_h^\uparrow$ denotes the upwind solution of u_h on $\hat{\mathcal{E}}_h$ which is given by

$$u_h^\uparrow\big|_{\hat{E}} = \begin{cases} u_h|_{\hat{K}^+} & \text{if } \left\{\vec{w}_h|\nabla\Phi_h(\cdot,t^*)|\right\}\cdot\vec{\nu}_{\hat{E}} \geq 0 \\ u_h|_{\hat{K}^-} & \text{if } \left\{\vec{w}_h|\nabla\Phi_h(\cdot,t^*)|\right\}\cdot\vec{\nu}_{\hat{E}} < 0 \end{cases} \qquad \text{for each } \hat{E}\in\hat{\mathcal{E}}_h.$$

Note that if a section of Γ_h or Γ_h^{old} is part of a face $\hat{E}\in\hat{\mathcal{E}}_h$, we choose u_h as $u_h^\uparrow$ and u_h^{old} as $u_h^{old,\uparrow}$, respectively.

REMARK 3. *We note that, since this construction implies that $\varphi_h \equiv 1$ is an admissible test function, a global mass conservation law is recovered:*

$$\int_{\Gamma_h} u_h \,\mathrm{d}\sigma_h = \int_{\Gamma_h^{old}} u_h^{old}\,\mathrm{d}\sigma_h.$$

REMARK 4. *There are many different options for the choice of $\vec{w}_h$. In this work, we will simply use the continuous normal velocity defined by the level set function and assume there is no tangential component to the velocity field. It is possible to use a backward difference to extract a normal velocity from the two discrete level set functions $\Phi_h(\cdot,t^*)$ and $\Phi_h(\cdot,t^*-\tau)$ by*

$$\vec{w}_h = -\frac{1}{\tau}\frac{\Phi_h(\cdot,t^*)-\Phi_h(\cdot,t^*-\tau)}{|\nabla\Phi_h(\cdot,t^*)|}\frac{\nabla\Phi_h(\cdot,t^*)}{|\nabla\Phi_h(\cdot,t^*)|}.$$

3. Understanding the method in one dimension. In this section, we want to give a better interpretation of the method and its limitations, using a simple 1D example. We consider a point moving along a one-dimensional axis with positive speed w, i.e. $\Phi(x,t) = x - wt$, $x\in\mathbb{R}, t\in[0,T]$. Furthermore, we consider a time step with $t^* = \tau = T$.

Let U be the interval $[x_0, x_N]$ and $\tilde{\mathcal{T}}_h$ be its decomposition into N sub-intervals $\{e_j\}_{j=1}^N$ of width $h = \frac{wT}{N}$, with $e_j = [x_{j-1}, x_j]$. Let $\Gamma_h^{\text{old}} = x_0$ and $\Gamma_h = x_N$, such that $D_h = U$ and $\hat{\mathcal{T}}_h = \mathcal{T}_h = \tilde{\mathcal{T}}_h$. Discretizing using Scheme 1 and a piecewise constant discrete space V_h, i.e. polynomial degree $k=0$, we obtain a finite volume type scheme and equation (2.3) simplifies to

$$\big[u_h\varphi_h\big](x_N) + \frac{\tau}{\gamma}\left(\sum_{i=1}^{N-1} u_h|_{e_i}\,[\![\varphi_h]\!](x_i)\,w\right) = u_h^{\text{old}}\,\varphi_h(x_0) \quad \text{ for all } \varphi_h\in V_h.$$

By fixing the basis of V_h which consists of characteristic functions $\varphi_j = \chi_{e_j}$, $j=1,\dots,N$, we obtain a discrete system. Denoting the vector of unkowns by u and supposing that u_h^{old} is a given scalar u^{old} at x_0, we wish to find $u_h = \sum_{j=1}^N \mathsf{u}_j\varphi_j$ with

$$\begin{aligned} w\gamma^{-1}\mathsf{u}_1 &= \tau^{-1}\mathsf{u}^{\text{old}} \\ w\gamma^{-1}\left(\mathsf{u}_j - \mathsf{u}_{j-1}\right) &= 0 \qquad\qquad \text{for } j = 2,\dots,N-1 \\ \tau^{-1}\mathsf{u}_N + w\gamma^{-1}\left(-\mathsf{u}_{N-1}\right) &= 0. \end{aligned}$$

This system is uniquely solvable, yielding a piecewise constant solution u_h with

$$\begin{aligned} \mathsf{u}_j &= \gamma(\tau w)^{-1}\mathsf{u}^{\text{old}} \qquad\qquad \text{for } j = 1,\dots,N-1 \\ u_h|_{\Gamma_h} &= \mathsf{u}_N = \mathsf{u}^{\text{old}}. \end{aligned}$$

For implementation reasons one might want to compute on a domain larger than D_h, say a computational domain $\Omega \supset D_h$. For the domain D in Figure 2.2, for example, D_h is not easily reconstructed without using knowledge of all intermediate curves. As we will illustrate now, the construction of the scheme unfortunately does not immediately carry over to this case. Unique solvability requires that Ω resolves $\Gamma_h \cap D_h$ as a domain boundary. Furthermore, if Ω does not meet this requirement, we cannot guarantee mass conservation any more.

We take a slightly larger domain with the same mesh size and extend the domain by $3h$ — $\frac{3}{2}h$ to the left and $\frac{3}{2}h$ to the right. This means that $x_{0+\frac{3}{2}} = \Gamma_h^{\text{old}}$ and $x_{N-\frac{3}{2}} = \Gamma_h$, i.e. Γ_h^{old} and Γ_h lie in the inner part of e_2 and e_{N-1}, respectively. The system now changes to

$$\begin{aligned} w\gamma^{-1}\mathsf{u}_1 &= 0 \\ w\gamma^{-1}\mathsf{u}_2 &= -\tau^{-1}\mathsf{u}^{\text{old}} \\ w\gamma^{-1}\left(\mathsf{u}_j - \mathsf{u}_{j-1}\right) &= 0 \qquad\qquad \text{for } j = 3, \dots, N-2 \\ \tau^{-1}\mathsf{u}_{N-1} + w\gamma^{-1}\left(\mathsf{u}_{N-1} - \mathsf{u}_{N-2}\right) &= 0 \\ -w\gamma^{-1}\mathsf{u}_{N-1} &= 0, \end{aligned}$$

which is underdetermined, due to the pure Neumann boundary conditions. Analogous to pseudo time stepping, we regularize the system with an additional mass term of order $\epsilon > 0$, the limit $\epsilon \to 0$ yields the solution

$$\begin{aligned} \mathsf{u}_1 &= 0 \\ \mathsf{u}_j &= \gamma(\tau w)^{-1}\mathsf{u}^{\text{old}} \qquad\qquad \text{for } j = 2, \dots, N-2 \\ u_h|_{\Gamma_h} = \mathsf{u}_{N-1} &= \gamma(\tau w + \gamma)^{-1}\mathsf{u}^{\text{old}} \\ \mathsf{u}_N &\to \infty. \end{aligned}$$

For u_N, the solution is not well defined and diverges in the ϵ–limit. For u_{N-1}, we observe that mass conservation is not fulfilled anymore and for the special case $\gamma = \tau w$ we obtain $u_h|_{\Gamma_h} = \mathsf{u}_{N-1} = \frac{1}{2}\mathsf{u}^{\text{old}}$. Thus it is necessary to resolve ∂D with the computational domain Ω.

4. Numerical results. The presented scheme has been implemented in the DUNE framework [1, 2] using the dune-UDG library [17]. Besides providing unfitted DG spaces like the discrete space V_h from Section 2.4, dune-UDG enables the evaluation of integrals over cut cells $\hat{K} \in \hat{\mathcal{T}}_h$ and their faces $\hat{E} \in \hat{\mathcal{E}}_h$, which is needed for the assembly of system matrices in unfitted DG schemes.

The fundamental mesh $\tilde{\mathcal{T}}_h$ (see Section 2.3) used in our code is a structured, Cartesian, quadradrilateral mesh over a freely choosable domain U. In the following numerical experiments, we use the two-dimensional domain $U = [-1.5, 2] \times [-1.5, 1.5]$, decomposed into an equal number of quadrilaterals in x- and y-direction. Furthermore, we employ spaces V_h of polynomial degree $k = 0$. With respect to an a priori known analytical solution u on $\Gamma(t^*)$, we compute errors $\|u(t^*) - u_h\|_{L^1(\Gamma_h)}$, $\|u(t^*) - u_h\|_{L^2(\Gamma_h)}$ and $\|u(t^*) - u_h\|_{L^\infty(\Gamma_h)}$.

4.1. A circle shrinking with constant normal speed. We consider a circle $\Gamma(t)$ of initial radius 1, centered at the origin, which is shrinking with constant normal velocity $\vec{w} \cdot \vec{\nu} = -1$. Using the description as a level set function, $\Gamma(t)$ can be described as the zero level set of $\Phi(x, t) = |x| - (1 - t)$, $x \in \mathbb{R}^2$, $t \in [0, T]$, $T < 1$. Note that Φ is a signed distance function, i.e. $|\nabla\Phi| \equiv 1$. Considering a time step with $t^* = \tau = 0.5$, the

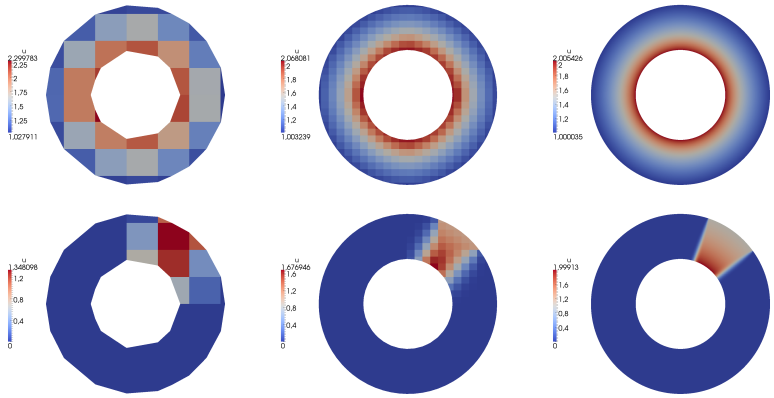

FIG. 4.1. *Computational domain D_h (discrete level set based reconstruction of D) and numerical solution u_h under h-refinement, using $t^* = \tau = \gamma = 0.5$. The depicted sequence of numerical solutions in the top row corresponds to the cases $\#\text{ncells} = 10^2, 40^2, 640^2$ in Table 4.1. The bottom row considers the same cases for a binary distributed u_h^{old}, showing the effect of numerical diffusion.*

TABLE 4.1
Convergence results under h-refinement, using $t^* = \tau = \gamma = 0.5$.

#cells	h	L^1-error	(eoc)	L^2-error	(eoc)	L^∞-error	(eoc)	mass
5^2	$9.220 \cdot 10^{-1}$	1.147	—	$9.078 \cdot 10^{-1}$	—	1.002	—	6.10369
10^2	$4.610 \cdot 10^{-1}$	$3.165 \cdot 10^{-1}$	1.86	$2.199 \cdot 10^{-1}$	2.05	$2.998 \cdot 10^{-1}$	1.74	6.24228
20^2	$2.305 \cdot 10^{-1}$	$1.951 \cdot 10^{-1}$	0.70	$1.203 \cdot 10^{-1}$	0.87	$1.041 \cdot 10^{-1}$	1.53	6.27299
40^2	$1.152 \cdot 10^{-1}$	$1.065 \cdot 10^{-1}$	0.87	$7.175 \cdot 10^{-2}$	0.75	$8.250 \cdot 10^{-2}$	0.34	6.28064
80^2	$5.762 \cdot 10^{-2}$	$5.733 \cdot 10^{-2}$	0.89	$3.925 \cdot 10^{-2}$	0.87	$6.628 \cdot 10^{-2}$	0.32	6.28255
160^2	$2.881 \cdot 10^{-2}$	$3.075 \cdot 10^{-2}$	0.90	$2.133 \cdot 10^{-2}$	0.88	$3.716 \cdot 10^{-2}$	0.83	6.28303
320^2	$1.441 \cdot 10^{-2}$	$1.592 \cdot 10^{-2}$	0.95	$1.081 \cdot 10^{-2}$	0.98	$2.021 \cdot 10^{-2}$	0.88	6.28315
640^2	$7.203 \cdot 10^{-3}$	$8.063 \cdot 10^{-3}$	0.98	$5.489 \cdot 10^{-3}$	0.98	$1.005 \cdot 10^{-2}$	1.01	6.28318

spatial-only domain D from Section 2.2 takes the form $D = \{x \in \mathbb{R}^2 : 0.5 \leq |x| \leq 1\}$. Its discrete reconstruction D_h for different values of h can be seen in Figure 4.1.

Results for h-refinement using $t^* = \tau = \gamma = 0.5$, $u_h^{\text{old}} \equiv 1$ on Γ_h^{old} and a corresponding analytical solution $u(t^*) \equiv 2$ on $\Gamma(t^*)$ are shown in the top row of Figure 4.1 and in Table 4.1. Note that #ncells is the number of cells in the fundamental mesh $\hat{\mathcal{T}}_h$. It does not refer to the number of cut cells $\hat{K} \in \hat{\mathcal{T}}_h$. Furthermore, since the scheme is globally mass conservative up to machine precision, the mass written in Table 4.1 is both the mass of u_h^{old} on Γ_h^{old} and the mass of u_h on Γ_h. Note that it is an indicator for the geometrical error coming from the reconstruction of Γ_h^{old}, which determines the amount of mass entering the discrete system. For the continuous problem and $u^{\text{old}} \equiv 1$, the total mass in the system equals $2\pi \approx 6.283185$. We observe convergence of order 1 in the L^1-norm and the L^2-norm, limited by geometrical errors for coarse meshes. For the L^∞-norm, the convergence rate is not that clear but also seems to be approaching order 1 for small values of h.

4.2. Non-constant initial concentration. To illustrate the effect of a non-constant u_h^{old}, we compute the same setup as in Section 4.1 but use a u_h^{old} with a binary distribution. As the worst case scenario, we consider value 1 for $0.2\pi < \text{angle} < 0.4\pi$ and 0 else. In this scenario, the numerical diffusion is expected to have the largest impact and for small values of h the jump sharpens, see bottom row of Figure 4.1. The numerical diffusion can be further reduced by using higher-order methods with flux-limiters.

5. Conclusion. We presented a new approach to solve the advection problem driven by the evolution of an evolving surface. The method shows attractive properties which justify further investigation. It is mass conservative and relatively easy to implement. We avoid constructing space-time meshes or following characteristics by reformulating the original problem as a classical transport problem on an unfitted domain.

In future work, we plan to extend the method to allow computations on $\Omega \supset D$ and high-order shape functions. Furthermore, we will apply the method to truly time-dependent problems and will also consider more general equations with $\vec{q} \neq 0$.

REFERENCES

[1] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn , M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. Part I: Abstract framework. *Computing*, 82(2–3):103–119, 7 2008.

[2] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn , R. Kornhuber, M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. Part II: Implementation and tests in DUNE. *Computing*, 82(2–3):121–138, 7 2008.

[3] P. Bastian and C. Engwer. An unfitted finite element method using discontinuous galerkin. *International Journal for Numerical Methods in Engineering*, 79(12):1557–1576, 2009.

[4] Peter Bastian, Christian Engwer, Jorrit Fahlke, and Olaf Ippisch. An unfitted discontinuous galerkin method for pore-scale simulations of solute transport. *Mathematics and Computers in Simulation*, 81(10):2051 – 2061, 2011. MAMERN 2009: 3rd International Conference on Approximation Methods and Numerical Modeling in Environment and Natural Resources.

[5] Paolo Cermelli, Eliot Fried, and Morton E. Gurtin. Transport relations for surface integrals arising in the formulation of balance laws for evolving fluid interfaces. *Journal of Fluid Mechanics*, 544:339–351, 12 2005.

[6] Klaus Deckelnick, Gerhard Dziuk, Charles M. Elliott, and Claus-Justus Heine. An h-narrow band finite-element method for elliptic equations on implicit surfaces. *IMA Journal of Numerical Analysis*, 30(2):351–376, 2010.

[7] Klaus Deckelnick, Charles M. Elliott, and Thomas Ranner. Unfitted finite element methods using bulk meshes for surface partial differential equations. *SIAM Journal on Numerical Analysis*, 52(4):2137–2162, 2014.

[8] Klaus Deckelnick, Charles M Elliott, and Vanessa Styles. Numerical diffusion-induced grain boundary motion. *Interfaces and Free Boundaries*, 3(4):393–414, 2001.

[9] Gerhard Dziuk. Finite elements for the Beltrami operator on arbitrary surfaces. In Stefan Hildebrandt and Rolf Leis, editors, *Partial Differential Equations and Calculus of Variations*, volume 1357 of *Lecture Notes in Mathematics*, pages 142–155. 1988.

[10] Gerhard Dziuk and Charles M. Elliott. Finite elements on evolving surfaces. *IMA Journal of Numerical Analysis*, 27(2):262–292, 2007.

[11] Gerhard Dziuk and Charles M. Elliott. Finite element methods for surface PDEs. *Acta Numerica*, 22:289–396, 2013.

[12] Gerhard Dziuk, Dietmar Kröner, and Thomas Müller. Scalar conservation laws on moving hypersurfaces. *Interfaces and Free Boundaries*, 15(2):203–236, 2013.

[13] Carston Eilks and Charles M. Elliott. Numerical simulation of dealloying by surface dissolution via the evolving surface finite element method. *Journal of Computational Physics*, 227(23):9727–9741, 2008.

[14] Charles M. Elliott, Björn Stinner, Vanessa Styles, and Richard Welford. Numerical computation of advection and diffusion on evolving diffuse interfaces. *IMA Journal of Numerical Analysis*, 31(3):786–812, 2011.

[15] Charles M. Elliott and Vanessa Styles. An ALE ESFEM for solving PDEs on evolving surfaces. *Milan Journal of Mathematics*, 80(2):469–501, 2012.

[16] Charles M Elliott and Chandrasekhar Venkataraman. Error analysis for an ale evolving surface finite element method. *Numerical Methods for Partial Differential Equations*, 31(2):459–499, 2015.

[17] C. Engwer and F. Heimann. Dune-udg: A cut-cell framework for unfitted discontinuous galerkin methods. In *Proceedings of the DUNE User Meeting 2010*, 2011.

[18] Ashley J. James and John Lowengrub. A surfactant-conserving volume-of-fluid method for interfacial flows with insoluble surfactant. *Journal of Computational Physics*, 201(2):685–

722, December 2004.

[19] Matthew P. Neilson, John A. Mackenzie, Steven D. Webb, and Robert H. Insall. Modeling Cell Movement and Chemotaxis Using Pseduopod-Based Feedback. *SIAM Journal on Scientific Computing*, 33:1035–1057, 2011.

[20] Maxim A. Olshanskii, Arnold Reusken, and Jörg Grande. A finite element method for elliptic equations on surfaces. *SIAM Journal on Numerical Analysis*, 47(5):3339–3358, 2009.

[21] Maxim A. Olshanskii, Arnold Reusken, and Xianmin Xu. An eulerian space-time finite element method for diffusion problems on evolving surfaces. *SIAM Journal on Numerical Analysis*, 52(3):1354–1377, 2014.

[22] Maxim A. Olshanskii, Arnold Reusken, and Xianmin Xu. A stabilized finite element method for advection–diffusion equations on surfaces. *IMA Journal of Numerical Analysis*, 34(2):732–758, 2014.

[23] Knut Erik Teigen, Xiangrong Li, John Lowengrub, Fan Wang, and Axel Voigt. A diffuse-interface approach for modeling transport, diffusion and adsorption/desorption of material quantities on a deformable interface. *Communications in mathematical sciences*, 4(7):1009, 2009.