INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2009; **60**:437–471
Published online 20 August 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1901

The intrinsic XFEM for two-fluid flows

T. P. Fries*,†,‡

RWTH Aachen University, Steinbachstr. 53 B, 52074 Aachen, Germany

SUMMARY

In two-fluid flows, jumps and/or kinks along the interfaces are present in the resulting velocity and pressure fields. Standard methods require mesh manipulations with the aim that either element edges align with the interfaces or that the mesh is sufficiently refined near the interfaces. In contrast, enriched methods, such as the extended finite element method (XFEM), enable the representation of arbitrary jumps and kinks inside elements. Thereby, optimal convergence can be achieved for two-fluid flows with meshes that remain fixed throughout the simulation. In the intrinsic XFEM, in contrast to other enriched methods, no more unknowns are present in the approximation than in a standard finite element approximation. In this work, the intrinsic XFEM is employed for the simulation of incompressible two-fluid flows. Numerical results are shown for a number of test cases and prove the success of the method. Copyright © 2008 John Wiley & Sons, Ltd.

Received 18 December 2007; Revised 26 June 2008; Accepted 14 July 2008

KEY WORDS: XFEM; two-fluid flow; enriched method; partition of unity

1. INTRODUCTION

The interaction of different fluids is frequently observed in the real world. For example, wave, bubble, and drop dynamics are applications of high relevance in engineering sciences. Also, free surface flows can be interpreted as two-phase flows where one fluid has negligible density and viscosity. The numerical simulation of two-fluid flows has a long tradition and remains to be an active research field, see e.g. [1–6]. In incompressible two-fluid flows, the velocity and pressure

*Correspondence to: T. P. Fries, RWTH Aachen University, Steinbachstr. 53 B, 52074 Aachen, Germany.
†E-mail: fries@cats.rwth-aachen.de
‡Chair for Computational Analysis of Technical Systems.

Contract/grant sponsor: German Research Association (DFG)

fields and/or their gradients are discontinuous along the interfaces between the fluids. Standard numerical methods, such as the finite element method (FEM) [7, 8], rely on the approximation properties of polynomials [9, 10]. Therefore, they perform poorly for the approximation of jumps and kinks *within* elements. As a consequence, most applications of the classical FEM for two-fluid flows rely on an appropriate manipulation of the mesh throughout the simulation. The aim is to keep the interfaces between the fluids aligned with element edges or to refine the mesh near the interfaces.

In interface tracking methods [2, 3], the initial mesh aligns with the interface and follows the movement of the interface throughout the simulation. However, large deformations and topological changes in the interface position may render the mesh-update difficult or even impossible. In interface-capturing methods [4, 6, 11–13], the interface is described implicitly on the mesh. The level-set method [14] and the volume of fluid method [4] are frequently used for this purpose. The mesh in the vicinity of the interface is often refined in order to capture the discontinuous behavior of the state variables near the interface. Otherwise, for interfaces going through elements, the accuracy of the classical FEM is poor, which is for example manifested by spurious velocities near the interface, see e.g. [15–17]. The optimal rate of convergence related to the element type employed can no longer be achieved.

Numerical methods for two-fluid flows, which are able to obtain optimal convergence on a *fixed* mesh throughout the simulation, have been proposed recently in the context of the extended finite element method (XFEM). In the XFEM, see e.g. [18, 19], the approximation space is enriched and, as a consequence, able to represent *a priori* known solution properties such as jumps and kinks exactly in element interiors. The XFEM has first been proposed in the context of an extrinsic enrichment of the approximation, i.e. more shape functions and unknowns are used when compared with a classical finite element approximation. This type of XFEM is used by Belytschko [20] and Chessa [21] and Kölke [22] for the simulation of two-fluid flows. A different version of the XFEM has been proposed by Fries and Belytschko in [23]. The method is called 'intrinsic XFEM' and enables the use of approximations of the same form than classical FEM approximations. The method is able to reproduce inner-element jumps and kinks (or other solution characteristics) by using intrinsically enriched shape functions; no more unknowns are introduced in this method. In this work, the intrinsic XFEM is applied to two-fluid flows.

Another method for arbitrary discontinuities is the generalized finite element method (GFEM) by Stroubolis *et al.* [24, 25]. It is closely related to the XFEM and also based on the partition of unity method (PUM) [9, 26] as the underlying concept. An interesting method for arbitrary discontinuities that is not based on the PUM-concept is given by Hansbo and Hansbo [27], though it can be shown to have the same basis functions as the XFEM [28]. Meshfree methods have been successfully used for arbitrary discontinuities, see e.g. the overviews in [29–32], and another approach may be found in [33].

In the intrinsic XFEM, the approximation space is built by standard finite element shape functions in those parts of the domain where a polynomial approximation space is adequate. However, in the vicinity of the interfaces between the two fluids, special shape functions are employed that are able to capture the discontinuities in the velocity and pressure fields. These *intrinsically* enriched shape functions are constructed by means of the moving least-squares (MLS) method [34, 35]. This method involves an intrinsic basis vector and the enrichment in the proposed method takes place on the level of this intrinsic basis. The set of MLS functions (used as shape functions near discontinuities) is obtained by minimizing a weighted least-squares error functional. Therefore, locally defined weight functions around each node are involved, and a special mesh-based definition

DOI: 10.1002/fld

of the weight functions is chosen in the intrinsic XFEM. As a result, the enriched MLS shape functions are mesh-based, which is in contrast to standard applications of the MLS in the context of meshfree methods [30, 31, 36].

Important properties of the intrinsic XFEM are summarized as follows [23]: The enrichment is realized locally and intrinsically without introducing additional unknowns. The same enrichment functions than used in the standard XFEM [18, 19] may be employed. The local enrichment is realized without introducing problems in blending elements which are between enriched and non-enriched parts of the domain. These problems are well-known in the standard XFEM, see [37], although this problem has been recently solved in [38]. In the intrinsic XFEM, the increased amount of computational work lies in the evaluation of the MLS shape functions. More integration points than in the classical FEM are needed and the evaluation of the enriched MLS shape functions is more cumbersome than for the classical FEM.

An outline of the paper is as follows: The governing equations of two-fluid flows are given in Section 2. Discontinuities in the velocity and pressure fields at the interfaces are discussed, and the definition of the interfaces by means of the level-set method is described. The construction of the approximation space used in the intrinsic XFEM is worked out in Section 3. The procedure falls into three steps: Decomposition of the domain based on the interface positions into overlapping subdomains (Section 3.1), construction of the finite element or MLS shape functions in the individual subregions (Section 3.2) and coupling such that one set of shape functions is obtained in the whole domain (Section 3.3). The discretization is discussed in Section 4. The discretized weak form is given (Section 4.1), and time integration with simple time-stepping schemes is described (Section 4.2). Special issues related to the intrinsic XFEM such as imposition of boundary conditions and integration of the weak form are mentioned in Sections 4.3 and 4.4. The treatment of the moving interfaces in the context of the level-set method is worked out in Section 4.5. Section 4.6 shows how surface tension is considered in this work. Numerical results are given in Section 5 for several examples of two-fluid flows. For two simplified situations, it is shown that an analytical solution including kinks or jumps can be represented exactly in the intrinsic XFEM, see Section 5.1. Other test cases such as sloshing in a tank, a collapsing water column, and bubble flows at different Eötvös numbers are given in Sections 5.2–5.4. The paper ends in Section 6 with a summary and conclusions.

2. INCOMPRESSIBLE TWO-FLUID FLOWS

2.1. Governing equations

The geometrical situation is as follows: Consider a d-dimensional domain $\Omega \subset \mathbb{R}^d$ with the boundary $\Gamma = \partial\Omega$. The boundary Γ is decomposed into the Dirichlet and Neumann boundary, $\Gamma_{\mathbf{u}}$ and $\Gamma_{\mathbf{h}}$, respectively, such that $\Gamma_{\mathbf{u}} \cup \Gamma_{\mathbf{h}} = \Gamma$ and $\Gamma_{\mathbf{u}} \cap \Gamma_{\mathbf{h}} = \emptyset$. The normal vector on Γ is denoted by $\mathbf{n}$. The domain Ω contains two different, immiscible incompressible Newtonian fluids in Ω_1 and Ω_2, respectively, so that $\Omega = \Omega_1 \cup \Omega_2$. Throughout this work, Ω is considered a time-independent closed container, whereas $\Omega_1(t)$ and $\Omega_2(t)$ change in time. The (moving) interface between the two fluids is denoted by Γ_{d}. The normal vector on Γ_{d} is called $\hat{\mathbf{n}}$ and points from Ω_1 to Ω_2. See Figure 1 for a sketch of the situation.

The governing equations are now given in strong form, see e.g. [5, 11, 16]. Let $\mathbf{u}(\mathbf{x}, t)$ be the velocities and $p(\mathbf{x}, t)$ the pressure; ϱ_i and μ_i with $i = (1, 2)$ are the density and dynamic

DOI: 10.1002/fld

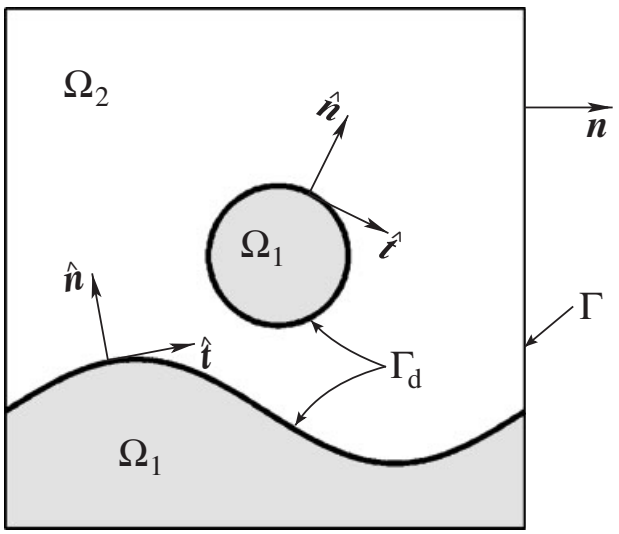

Figure 1. The two fluids in Ω_1 and Ω_2, separated by the interface Γ_d.

viscosity of the two fluids, respectively; $\mathbf{f}$ are volumetric forces such as gravity. The fluids inside $\Omega_i \times (0, t_{\text{end}})$, $i = (1, 2)$, are modeled by the instationary, incompressible Navier–Stokes equations in velocity–pressure formulation

$$\varrho_i \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) - \nabla \cdot \boldsymbol{\sigma} = \varrho_i \mathbf{f} \tag{1}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2}$$

The stress tensor $\boldsymbol{\sigma}$ of the Newtonian fluids is given as

$$\boldsymbol{\sigma}(\mathbf{u}, p) = -p\mathbf{I} + 2\mu_i \boldsymbol{\varepsilon}(\mathbf{u}) \quad \text{with } \boldsymbol{\varepsilon}(\mathbf{u}) = \tfrac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathrm{T}}) \tag{3}$$

where $\mathbf{I}$ is the identity tensor. Dirichlet and Neumann boundary conditions on the outer boundary of Ω are

$$\mathbf{u} = \hat{\mathbf{u}} \quad \text{on } \Gamma_{\mathbf{u}} \times (0, t_{\text{end}}) \tag{4}$$

$$\boldsymbol{\sigma} \cdot \mathbf{n} = \hat{\mathbf{h}} \quad \text{on } \Gamma_{\mathbf{h}} \times (0, t_{\text{end}}) \tag{5}$$

where $\hat{\mathbf{u}}$ and $\hat{\mathbf{h}}$ are prescribed velocities and stresses. The following conditions typically apply at the interface:

$$[\mathbf{u}]_{\Gamma_d} = \mathbf{0} \quad \text{on } \Gamma_d \times (0, t_{\text{end}}) \tag{6}$$

$$-[\boldsymbol{\sigma}]_{\Gamma_d} \cdot \hat{\mathbf{n}} = \gamma \cdot \kappa \cdot \hat{\mathbf{n}} \quad \text{on } \Gamma_d \times (0, t_{\text{end}}) \tag{7}$$

DOI: 10.1002/fld

Here, $[f]_{\Gamma_d}$ is the jump of property f across the interface Γ_d, γ is the surface tension coefficient (material parameter) and κ is the curvature of Γ_d. As an initial condition, a divergence-free velocity field $\hat{\mathbf{u}}_0$ is specified over Ω,

$$\mathbf{u}(\mathbf{x}, 0) = \hat{\mathbf{u}}_0(\mathbf{x}) \quad \text{in } \Omega \text{ at } t = 0 \tag{8}$$

2.2. Discontinuous state variables at the interface

The situation at the interface Γ_d is considered in more detail. The density and viscosity fields

$$\varrho(\mathbf{x}, t) = \begin{cases} \varrho_1 & \forall \mathbf{x} \in \Omega_1(t), \\ \varrho_2 & \forall \mathbf{x} \in \Omega_2(t), \end{cases} \qquad \mu(\mathbf{x}, t) = \begin{cases} \mu_1 & \forall \mathbf{x} \in \Omega_1(t) \\ \mu_2 & \forall \mathbf{x} \in \Omega_2(t) \end{cases} \tag{9}$$

change discontinuously at Γ_d. As a consequence, also the state variables such as the velocities and pressure fields involve discontinuities at the interface. Discontinuities may be classified into strong and weak, see Figure 2. In the case of strong discontinuities, a jump and a change in the gradient are present in the field. For weak discontinuities there is only a kink in the field, i.e. the field is continuous with a discontinuous gradient.

The interface condition (6) states that the velocities are continuous across Γ_d, or, in other words, that the jump in the velocity field is zero. The second interface condition, (7), states that the surface tension balances the jump of the normal stress at the interface. As a consequence of (1)–(2) and (6)–(7), the velocity fields $\mathbf{u}(\mathbf{x}, t)$ are weakly discontinuous across Γ_d, whereas the pressure field $p(\mathbf{x}, t)$ has a strong discontinuity at the interface. In the case that no surface tension is considered, $\gamma = 0$, the jump in the pressure field vanishes and $p(\mathbf{x}, t)$ has a kink at Γ_d. Applications with and without surface tension are considered in this work.

It is relevant to consider the presence of strong and weak discontinuities in the state variables when simulating two-fluid flows. This has been realized by many different approaches: In interface tracking methods, weak discontinuities are accounted for by the standard FEM automatically. The strong discontinuity in the pressure field can be realized by using completely decoupled

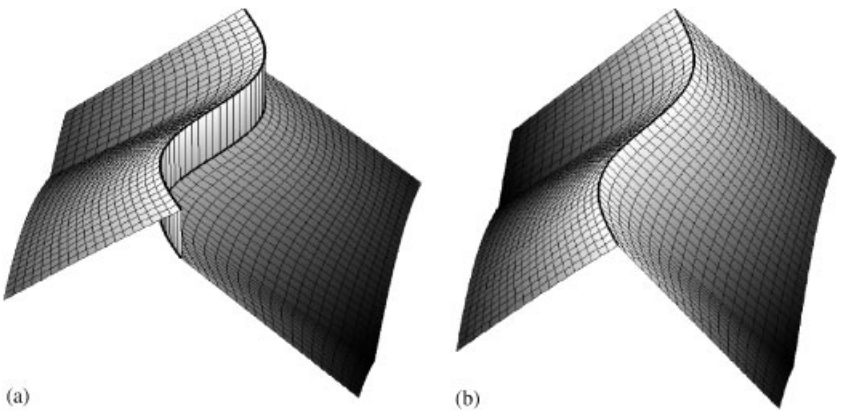

Figure 2. Examples of (a) a strong and (b) a weak discontinuity.

DOI: 10.1002/fld

meshes for the two-fluid regions. It is again noted that these methods do not, in general, allow topological changes of the two-fluid domains. In interface-capturing methods, the strong and weak discontinuities may be considered by local h-refinement in the vicinity of the interface. Also, under-integration in the elements cut by the interface is often applied. For interface-capturing methods, especially strong discontinuities often pose a serious problem for the accuracy of the overall simulation. It is, therefore, our belief that enriched methods such as the XFEM are ideally suited for two-fluid flow simulations as they consider both, strong and weak discontinuities within elements, by special enrichments of the approximation space.

2.3. Description of the interfaces by the level-set method

The level-set method [14] is used for the description of interfaces $\Gamma^{\pm}=\Gamma_{\mathrm{d}}$ between $\Omega^{-}=\Omega_1$ and $\Omega^{+}=\Omega_2$. It is a numerical technique for the implicit tracking of moving interfaces, and has been applied in the context of two-fluid flows e.g. in [6, 13, 20, 22, 39]. A level-set function $\phi(\mathbf{x})$ has the properties that it is positive in Ω^{+}, negative in Ω^{-}, and zero on the interface. In this work, the signed-distance function [14] is used as a particular level-set function,

$$\phi(\mathbf{x})=\pm \min_{\mathbf{x}^{\star}\in\Gamma^{\pm}} \|\mathbf{x}-\mathbf{x}^{\star}\| \quad \forall \mathbf{x}\in\Omega \tag{10}$$

where the sign is different on the two sides of the interface and $\|\cdot\|$ denotes the Euclidean norm, see Figure 3. It follows directly from (10) that the zero-level of this scalar function is a representation of the discontinuity, i.e.

$$\phi(\mathbf{x})=0 \quad \forall \mathbf{x}\in\Gamma^{\pm} \tag{11}$$

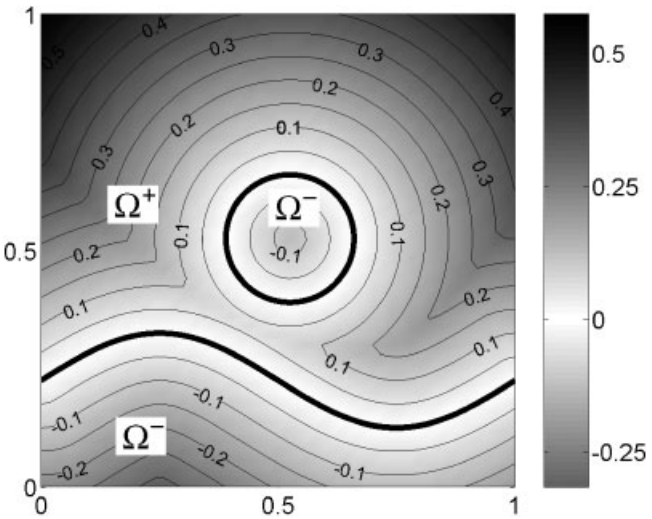

Figure 3. Two-fluid domain Ω with level-set function $\phi(\mathbf{x})$. The zero-level of $\phi(\mathbf{x})$ describes the interface position between Ω^{+} and Ω^{-}.

DOI: 10.1002/fld

In what follows, we use the notation of $\Gamma^{\pm}$, Ω^{-}, and Ω^{+} rather than Γ_{d}, Ω_1 and Ω_2 (as in Section 2.1) due to its more intuitive linkage to the level-set function used for the partitioning of the domain.

For discretized domains, the values of the level-set function are typically stored at nodes $\phi_i = \phi(\mathbf{x}_i)$, and the level-set function is interpolated by

$$\phi^h(\mathbf{x}) = \sum_{i \in I} N_i^{\mathrm{FEM}}(\mathbf{x}) \phi_i \tag{12}$$

using standard FE shape functions N_i^{FEM} as interpolation functions. I is the set of all nodes in Ω. The representation of the discontinuity as the zero-level of $\phi^h(\mathbf{x})$ is only an approximation of the real position, which improves with mesh refinement. In this work we restrict ourselves to two-dimensional applications. Bi-linear interpolation functions are used in (12) for all elements that are not cut by the interface. In cut elements, however, we subdivide the quadrilateral element into two triangulars and employ linear interpolation functions. Thereby, the zero-level of $\phi^h(\mathbf{x})$, i.e. the representation of the interface, remains piecewise linear.

It is noted that the level-set method is ideally suited for the description of interfaces in closed domains, i.e. without inflow and outflow boundaries, as considered throughout this work. The situation may become problematic in special situations where inflow boundaries are present and—as an example—bubbles (Ω^{-}) enter the surrounding fluid (Ω^{+}).

3. THE INTRINSIC XFEM FOR TWO-FLUID FLOWS

The intrinsic XFEM falls into three steps. In the first step the domain is decomposed into subdomains that overlap in one element layer. The second step is the construction of shape functions for all nodes of each subdomain. They build partition of unities (PUs) with certain properties. Standard finite element or special enriched MLS functions are employed here. In the third step, the shape functions in the overlapping element layers are coupled such that only one shape function per node results.

3.1. Decomposition of the domain

It is assumed that the domain $\Omega \in \mathbb{R}^d$ is subdivided into n^{el} elements; the set of all elements is labeled $\mathcal{Q} = \{1, \ldots, n^{\mathrm{el}}\}$. The element area of element k is Ω_k^{el}, $k \in \mathcal{Q}$. The set of m element nodes corresponding to each element is denoted by $I_k^{\mathrm{el}} \in (\mathbb{N}^+)^m$, $k \in \mathcal{Q}$. The union of all element nodes is the nodal set $I = \bigcup_{i \in \mathcal{Q}} I_i^{\mathrm{el}} = \{1, \ldots, n^{\mathrm{no}}\}$, where n^{no} is the total number of nodes in the entire domain Ω.

The set of elements that are cut by the interface $\Gamma^{\pm}$ between Ω^{+} and Ω^{-} is determined by means of the level-set function $\phi(\mathbf{x})$ as

$$\mathcal{Q}_{\mathrm{cut}} = \left\{ k \in \mathcal{Q} : \min_{i \in I_k^{\mathrm{el}}} (\phi(\mathbf{x}_i)) \cdot \max_{i \in I_k^{\mathrm{el}}} (\phi(\mathbf{x}_i)) < 0 \right\} \tag{13}$$

The union of the element nodes of elements in $\mathcal{Q}_{\mathrm{cut}}$ composes the nodal subset I_{cut}, see Figure 4(a),

$$I_{\mathrm{cut}} = \bigcup_{i \in \mathcal{Q}_{\mathrm{cut}}} I_i^{\mathrm{el}} \tag{14}$$

DOI: 10.1002/fld

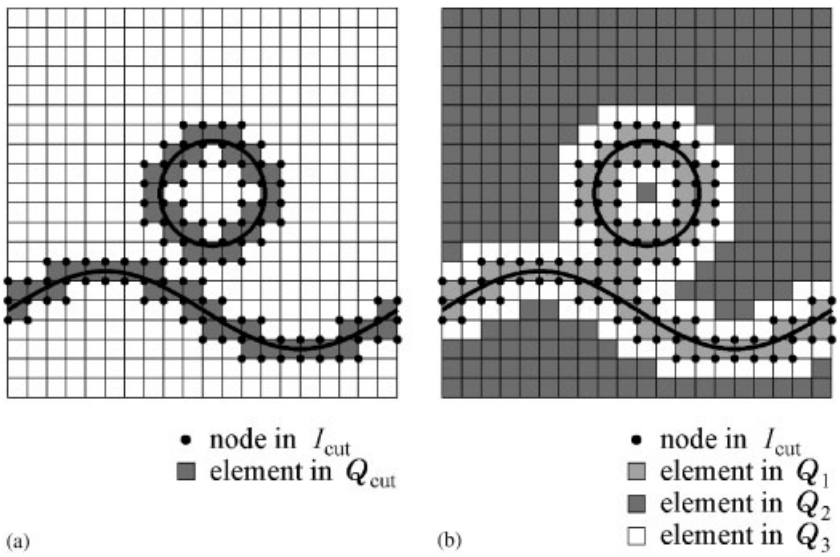

Figure 4. (a) Elements in $\mathcal{Q}_{cut}$ and nodal set I_{cut}, (b) elements in $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_3$ based on the fact whether all, none, or some of their element nodes are in I_{cut}.

Based on I_{cut}, the set of all elements $\mathcal{Q}$ is partitioned into three disjoint element sets. The sets $\mathcal{Q}_1$, $\mathcal{Q}_2$, $\mathcal{Q}_3$ contain elements with all, none, or some of their element nodes belonging to I_{cut}, respectively,

$$\mathcal{Q}_1 = \{k \in \mathcal{Q} : I_k^{el} \subseteq I_{cut}\} \tag{15}$$

$$\mathcal{Q}_2 = \{k \in \mathcal{Q} : I_k^{el} \cap I_{cut} = \emptyset\} \tag{16}$$

$$\mathcal{Q}_3 = \{k \in \mathcal{Q} : k \notin \mathcal{Q}_1 \text{ and } k \notin \mathcal{Q}_2\} \tag{17}$$

See Figure 4(b) for a sketch of the situation. It is noted that some elements have all their element nodes in I_{cut} but are not cut themselves, so that $\mathcal{Q}_{cut} \subseteq \mathcal{Q}_1$. This can be seen by comparing $\mathcal{Q}_{cut}$ and $\mathcal{Q}_1$ in Figure 4(a) and (b).

Furthermore, we define the element sets $\mathcal{Q}_{FEM}$ and $\mathcal{Q}_{MLS}$ as

$$\mathcal{Q}_{FEM} = \mathcal{Q}_2 \cup \mathcal{Q}_3 \tag{18}$$

$$\mathcal{Q}_{MLS} = \mathcal{Q}_1 \cup \mathcal{Q}_3 \tag{19}$$

Obviously, $\mathcal{Q}_{MLS} \cap \mathcal{Q}_{FEM} = \mathcal{Q}_3$. The union of the element areas of elements in $\mathcal{Q}_{FEM}$ composes the subdomain Ω_{FEM}, and I_{FEM} is the set of all element nodes of the elements in $\mathcal{Q}_{FEM}$, see Figure 5(a). Analogously, Ω_{MLS} and I_{MLS} result from $\mathcal{Q}_{MLS}$, see Figure 5(b):

$$\Omega_{FEM} = \bigcup_{i \in \mathcal{Q}_{FEM}} \Omega_i^{el}, \quad I_{FEM} = \bigcup_{i \in \mathcal{Q}_{FEM}} I_i^{el} \tag{20}$$

$$\Omega_{MLS} = \bigcup_{i \in \mathcal{Q}_{MLS}} \Omega_i^{el}, \quad I_{MLS} = \bigcup_{i \in \mathcal{Q}_{MLS}} I_i^{el} \tag{21}$$

DOI: 10.1002/fld

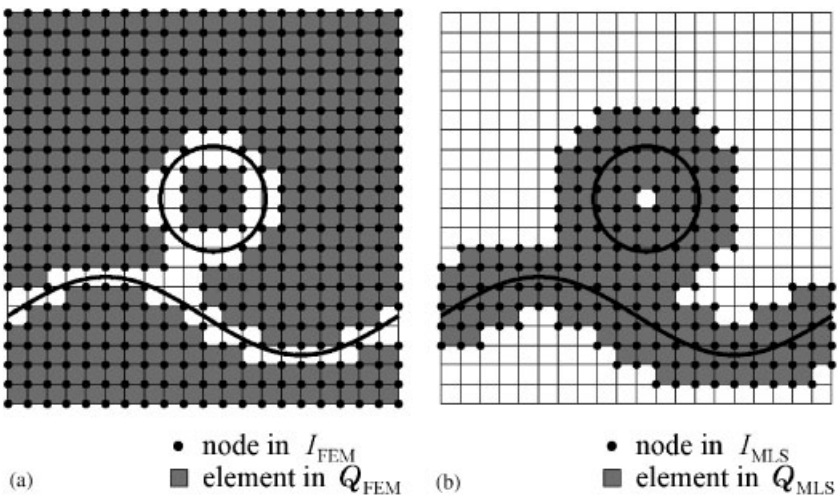

Figure 5. (a) Subdomain Ω_{FEM} with nodes I_{FEM}, (b) subdomain Ω_{MLS} with nodes I_{MLS}.

The overlap of the domains Ω_{FEM} and Ω_{MLS} is labeled transition area $\Omega_{\mathrm{trans}} = \Omega_{\mathrm{FEM}} \setminus \Omega_{\mathrm{MLS}}$, it contains the elements in $\mathcal{Q}_3$. The nodes that appear in I_{FEM} and I_{MLS} are called $I_{\mathrm{trans}} = I_{\mathrm{FEM}} \setminus I_{\mathrm{MLS}}$; these are the element nodes of elements in the transition area.

3.2. Construction of PUs in each subdomain

In each subdomain Ω_{FEM} and Ω_{MLS}, a set of shape functions $\{N_i^{\mathrm{FEM}}\}$ and $\{N_i^{\mathrm{MLS}}\}$ is constructed for each of the nodes in I_{FEM} and I_{MLS}, respectively. Standard bi-linear finite element shape functions are used in Ω_{FEM}. It is noted that these shape functions are not constructed in cut elements, see Figure 5(a), because they are not suited for capturing the strong or weak discontinuity across the interface. In Ω_{MLS}, which is in the vicinity of the interface, see Figure 5(b), special enriched MLS functions are constructed. The enrichment enables the shape functions to represent jumps and kinks in the solution of a field variable.

3.2.1. Moving least-squares (MLS) method. MLS shape functions have been discussed in a number of publications [31, 35, 40], we briefly define them as follows. MLS functions N_i^{MLS} are constructed in Ω_{MLS} for the nodes in I_{MLS}. They are defined as

$$N_i^{\mathrm{MLS}}(\mathbf{x}) = \mathbf{p}^{\mathrm{T}}(\mathbf{x})[\mathbf{M}(\mathbf{x})]^{-1} w_i(\mathbf{x})\mathbf{p}(\mathbf{x}_i) \tag{22}$$

$$\text{with} \quad \mathbf{M}(\mathbf{x}) = \sum_{i \in I_{\mathrm{MLS}}} w_i(\mathbf{x})\mathbf{p}(\mathbf{x}_i)\mathbf{p}^{\mathrm{T}}(\mathbf{x}_i) \tag{23}$$

The functions $w_i(\mathbf{x})$ are called MLS weight functions and $\mathbf{p}(\mathbf{x})$ is the intrinsic basis consisting of k components. The set of MLS functions $\{N_i^{\mathrm{MLS}}(\mathbf{x})\}$ builds a PU over the domain Ω_{MLS}, i.e. for any function in the basis $\mathbf{p}(\mathbf{x})$:

$$\sum_{i \in I_{\mathrm{MLS}}} N_i^{\mathrm{MLS}}(\mathbf{x})\mathbf{p}(\mathbf{x}_i) = \mathbf{p}(\mathbf{x}), \quad \mathbf{x} \in \Omega_{\mathrm{MLS}} \tag{24}$$

DOI: 10.1002/fld

We use the term PU in the generalized sense of (24), the original expression 'PU' only refers to the property that $\sum_i N_i^{\mathrm{MLS}}(\mathbf{x}) = 1$.

It is important to note that the basis vector chosen sets up requirements for the weight functions. To maintain the regularity of the matrices $\mathbf{M}(\mathbf{x})$, it is necessary that the weight functions $w_i(\mathbf{x})$ in the MLS procedure, see Equations (22) and (23), overlap sufficiently at any point $\mathbf{x} \in \Omega_{\mathrm{MLS}}$. More precisely, the relevant condition is that every point $\mathbf{x} \in \Omega_{\mathrm{MLS}}$ must lie in the area of influence of at least $k = \dim(\mathbf{M}) = \dim(\mathbf{p})$ nodes [41], hence

$$\mathrm{card}\{\mathbf{x} | w_i(\mathbf{x}) \neq 0 \ \forall i \in I_{\mathrm{MLS}}\} \geqslant k = \dim(\mathbf{M}) \quad \forall \mathbf{x} \in \Omega_{\mathrm{MLS}} \tag{25}$$

For the evaluation of MLS shape functions, which are defined through Equations (22) and (23), the weight functions $w_i(\mathbf{x})$ and the basis vector $\mathbf{p}(\mathbf{x})$ have to be specified.

3.2.2. Special weight functions. The weight functions must be constructed so as to ensure a sufficient overlap such that the MLS moment matrices are invertible, see Equation (25). They also determine some important properties of the resulting shape functions. The support and the continuity of the shape functions are identical to the weight functions, that is, $\forall i \in I_{\mathrm{MLS}}$: $N_i^{\mathrm{MLS}} = 0$ where $w_i = 0$, and $N_i^{\mathrm{MLS}} \in C^s(\Omega)$ if $w_i \in C^s(\Omega)$ (assuming that $\mathbf{p}(\mathbf{x})$ is sufficiently smooth). For the new weight functions, the supports consist in the elements contiguous to a node and their neighboring elements. This is shown in Figure 6 for quadrilateral elements.

The following definition of the weight functions has been found useful, see [23]: Let $I_\beta^\star$ be the nodes in I_{MLS} which belong to the same element(s) as node $\beta \in I_{\mathrm{MLS}}$. This set of nodes is called neighboring nodes and can be described as

$$I_\beta^\star = \{i \in I_{\mathrm{MLS}} \setminus \beta : (i, \beta) \subset I_k^{\mathrm{el}}, \ k \in \mathcal{2}\} \tag{26}$$

The weight function of node β in the subdomain Ω_{MLS} is then defined by

$$w_\beta(\mathbf{x}) = 2 \cdot N_\beta^{\mathrm{FEM}}(\mathbf{x}) + \sum_{i \in I_\beta^\star} N_i^{\mathrm{FEM}}(\mathbf{x}) \tag{27}$$

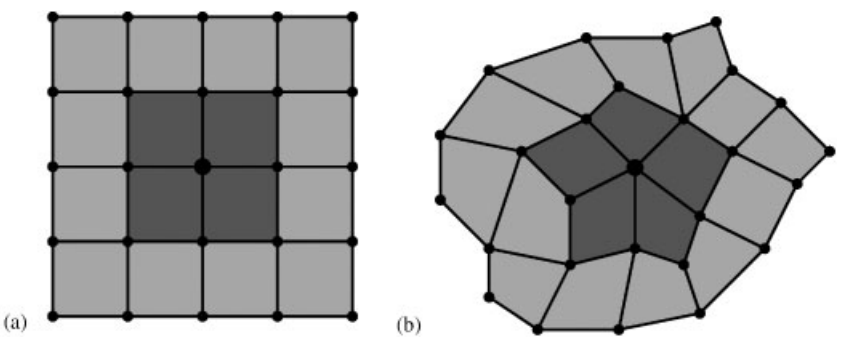

Figure 6. The weight function corresponding to the center node has a support which includes the neighboring elements of that node (dark-gray area) and the next-neighboring elements (light-gray area).

DOI: 10.1002/fld

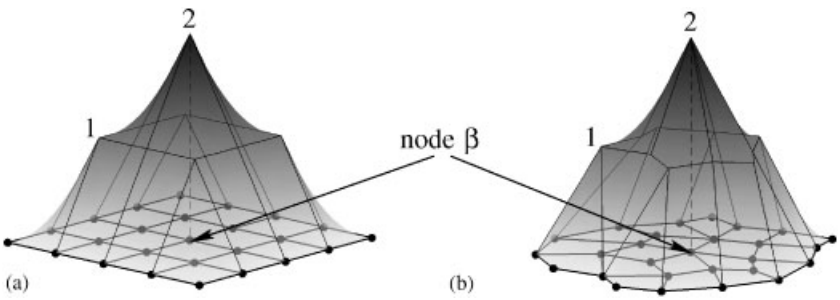

Figure 7. The proposed weight function of node β in a structured and unstructured element situation.

where $N_i^{\mathrm{FEM}}(\mathbf{x})$ is a standard finite element shape function. This weight function is depicted in Figure 7 for a node in a structured and unstructured quadrilateral element setting. One may use the new weight functions (27) for either triangular and quadrilateral elements. However, in this work, without loss of generality, only quadrilateral elements with corresponding bi-linear shape functions are considered.

3.2.3. Basis functions for discontinuities. For smooth solutions, the basis vector $\mathbf{p}(\mathbf{x})$ of the MLS often consists of monomials depending on the desired order of accuracy. For example, $\mathbf{p}^{\mathrm{T}}(x)=[1, x, x^2]$ serves as a quadratic basis in one dimension, $\mathbf{p}^{\mathrm{T}}(\mathbf{x})=[1, x, y]$ as a linear basis in two dimensions. The approximation properties of these bases are similar to those of the corresponding finite element shape functions. However, in contrast to finite element shape functions, the basis in the MLS method can be easily enriched by any desirable terms which enable the resulting MLS shape functions to improve their approximation properties in the presence of non-smooth solutions.

For two-fluid flows, as discussed in Section 2.2, strong and weak discontinuities are present across the interface of the two fluids. Then, the following basis-vectors, depending on the level-set function $\phi(\mathbf{x})$, are suited for the construction of the MLS shape functions. For weak discontinuities we define

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x})=[1, x, y, \mathrm{abs}(\phi(\mathbf{x}))] \tag{28}$$

and for strong discontinuities one may choose

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x})=[1, x, y, \mathrm{abs}(\phi(\mathbf{x})), \mathrm{sign}(\phi(\mathbf{x}))] \tag{29}$$

However, we found that instead of using (29) for strong discontinuities, it is more efficient (and intuitive) to use a standard basis

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x})=[1, x, y] \tag{30}$$

and modify the weight functions $w_i(\mathbf{x})$ according to the visibility method [42]. In this method, the interface is considered opaque, and $w_\beta(\mathbf{x})$ is set to zero in those part of the support which is not visible from the node $\mathbf{x}_\beta$. Shortly, the support of the weight function is truncated on the other side of the discontinuity, see Figure 8. This modification of the support of the weight functions is

DOI: 10.1002/fld

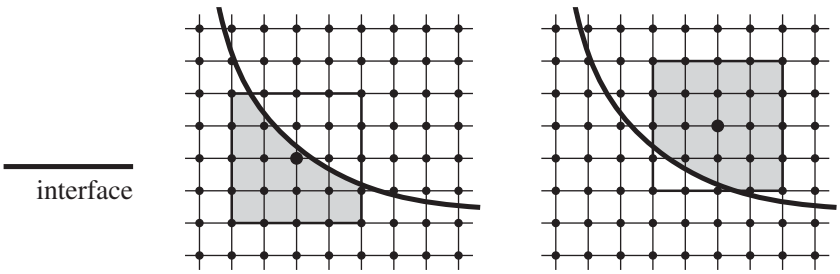

Figure 8. The truncated supports of the weight functions according to the visibility method are shown for two selected nodes.

a standard treatment of a strong discontinuity in the field of meshfree methods [40, 42], and has also been investigated in a mesh-based context by Fries and Belytschko in [43].

The truncated weight function according to the visibility method can be expressed as

$$w_\beta^\star(\mathbf{x}) = \begin{cases} w_\beta(\mathbf{x}) & \text{for } \mathbf{x} \text{ visible from } \mathbf{x}_\beta \\ 0 & \text{instead} \end{cases} \tag{31}$$

where $w_\beta(\mathbf{x})$ is defined in (27). The fact whether $\mathbf{x}$ is visible from $\mathbf{x}_\beta$ can be based on the sign of the level-set function, see Section 2.3. The point $\mathbf{x}$ is visible from $\mathbf{x}_\beta$ if

$$\phi(\mathbf{x}) \cdot \phi(\mathbf{x}_\beta) > 0 \tag{32}$$

As a result of the truncation, the resulting shape functions on each side of the interface are fully decoupled as desired for a strong discontinuity.

3.3. Coupling the PUs

Different sets of shape functions are individually defined over the overlapping subdomains Ω_{FEM} and Ω_{MLS} with respect to the nodal sets I_{FEM} and I_{MLS}, respectively. The subdomains Ω_{FEM} and Ω_{MLS} overlap in the transition area Ω_{trans}. In elements that are in $\Omega_{\text{FEM}} \setminus \Omega_{\text{trans}}$, only finite element shape functions are evaluated and the final shape function $N_i(\mathbf{x})$ used in the approximation follows immediately as

$$N_i(\mathbf{x}) = N_i^{\text{FEM}}(\mathbf{x}) \quad \forall \mathbf{x} \in \Omega_{\text{FEM}} \setminus \Omega_{\text{trans}} \quad \forall i \in I_{\text{FEM}} \tag{33}$$

Analogously, for elements in $\Omega_{\text{MLS}} \setminus \Omega_{\text{trans}}$, only MLS shape functions are present, so that

$$N_i(\mathbf{x}) = N_i^{\text{MLS}}(\mathbf{x}) \quad \forall \mathbf{x} \in \Omega_{\text{MLS}} \setminus \Omega_{\text{trans}} \quad \forall i \in I_{\text{MLS}} \tag{34}$$

In the transition area Ω_{trans}, for each node in $I_{\text{trans}} = I_{\text{FEM}} \cap I_{\text{MLS}}$, a standard finite element and MLS function has been evaluated. In order to obtain one shape function, a coupling is required. A ramp function is used for this purpose which is defined as

$$R(\mathbf{x}) = \sum_{i \in I_{\text{cut}}} N_i^{\text{FEM}}(\mathbf{x}) \tag{35}$$

DOI: 10.1002/fld

It may be seen in Figure 9 that $R(\mathbf{x})=1$ in $\Omega_{\text{MLS}}\setminus\Omega_{\text{trans}}$, and $R(\mathbf{x})=0$ in $\Omega_{\text{FEM}}\setminus\Omega_{\text{trans}}$. It varies continuously between 0 and 1 in Ω_{trans}. Then, the resulting shape function $N_i(\mathbf{x})$ is defined as

$$N_i(\mathbf{x})=N_i^{\text{FEM}}(\mathbf{x})\cdot[1-R(\mathbf{x})]+N_i^{\text{MLS}}(\mathbf{x})\cdot R(\mathbf{x}) \quad \forall\mathbf{x}\in\Omega_{\text{trans}} \;\; \forall i\in I_{\text{trans}} \tag{36}$$

An example of the resulting shape function $N_i(\mathbf{x})$ is depicted in Figure 10.

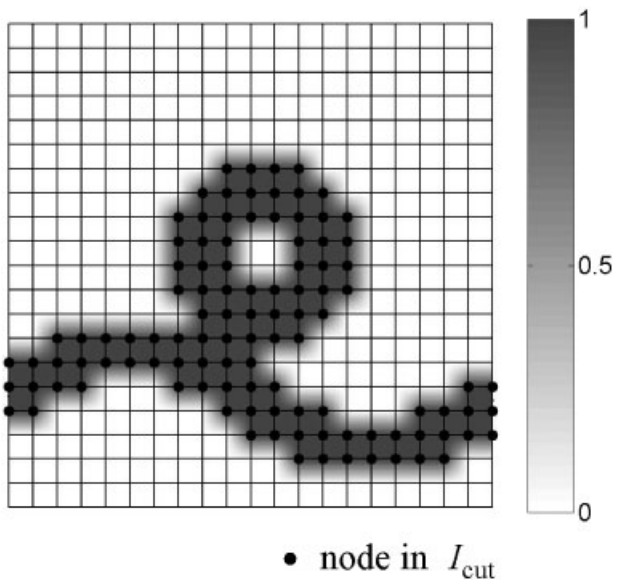

Figure 9. The ramp function $R(\mathbf{x})$ based on I_{cut}.

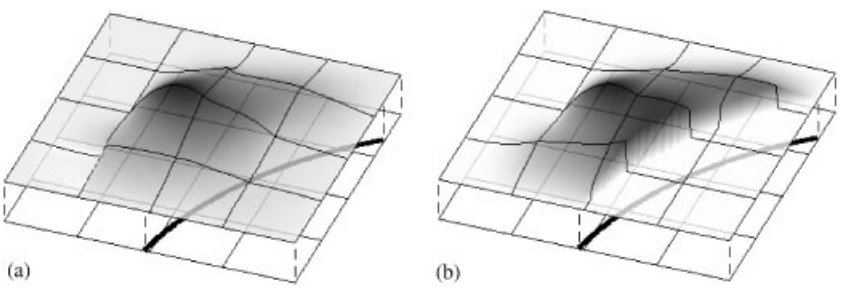

Figure 10. Shape functions of a particular node for the case of a (a) weak discontinuity and (b) strong discontinuity.

DOI: 10.1002/fld

4. DISCRETIZATION

4.1. Governing equations in weak form

In Section 2.1, the strong form of the incompressible Navies–Stokes equations and the definitions of the boundary and initial conditions are given. Following [44–46], the streamline-upwind Petrov–Galerkin (SUPG) and pressure-stabilizing Petrov–Galerkin (PSPG) formulation of the weak form is considered here. The following test and trial spaces, $\mathscr{S}^h$ and $\mathscr{V}^h$, are used for the velocities $\mathbf{u}$ and pressure p

$$\mathscr{S}^h_{\mathbf{u}} = \{\mathbf{u}^h | \mathbf{u}^h \in (\mathscr{H}^{1h})^d, \mathbf{u}^h = \hat{\mathbf{u}}^h \text{ on } \Gamma_{\mathbf{u}}\} \tag{37}$$

$$\mathscr{V}^h_{\mathbf{u}} = \{\mathbf{w}^h | \mathbf{w}^h \in (\mathscr{H}^{1h})^d, \mathbf{w}^h = \mathbf{0} \text{ on } \Gamma_{\mathbf{u}}\} \tag{38}$$

$$\mathscr{S}^h_p = \mathscr{V}^h_p = \{q^h | q^h \in \mathscr{H}^{1h}\} \tag{39}$$

where $\mathscr{H}^{1h} \subseteq \mathscr{H}^1$ is a finite dimensional Hilbert space consisting of the shape functions $\{N_i\}$ defined in the previous section; either abs- and/or sign-enrichment is used for the construction of the velocity and pressure shape functions. The space $\mathscr{H}^1$ is the set of functions which are, together with their first derivatives, square-integrable in Ω. The SUPG/PSPG-stabilized discretized weak form may then be formulated as [44]: Find $\mathbf{u}^h \in \mathscr{S}^h_{\mathbf{u}}$ and $p^h \in \mathscr{S}^h_p$ such that $\forall \mathbf{w}^h \in \mathscr{V}^h_{\mathbf{u}}$ and $\forall q^h \in \mathscr{V}^h_p$,

$$\int_\Omega \mathbf{w}^h \cdot \varrho_i \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h \right) \mathrm{d}\Omega + \int_\Omega \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(\mathbf{u}^h, p^h)\, \mathrm{d}\Omega \tag{40}$$

$$- \int_{\Gamma_{\mathbf{h}}} \mathbf{w}^h \cdot \hat{\mathbf{h}}\, \mathrm{d}\Gamma + \int_\Omega q^h \nabla \cdot \mathbf{u}^h\, \mathrm{d}\Omega + \sum_{j \in \mathscr{Q}} \int_{\Omega^{\mathrm{el}}_j} \tau_j \left(\mathbf{u}^h \cdot \nabla \mathbf{w}^h + \frac{1}{\varrho_i} \nabla q^h \right) \tag{41}$$

$$\cdot \left[\varrho_i \left(\frac{\partial \mathbf{u}^h}{\partial t} + \mathbf{u}^h \cdot \nabla \mathbf{u}^h \right) - \nabla \cdot \boldsymbol{\sigma}\left(\mathbf{u}^h, p^h \right) \right] \mathrm{d}\Omega \tag{42}$$

$$= \int_\Omega \varrho_i \mathbf{w}^h \cdot \mathbf{f}\, \mathrm{d}\Omega + \int_{\Gamma^{\pm}} \gamma \kappa \mathbf{w}^h \cdot \hat{\mathbf{n}}\, \mathrm{d}\Gamma \tag{43}$$

with $i = (1, 2)$, and $\boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ are defined in Equation (3). The stabilization terms are found in Equations (41) and (42) after the summation symbol $\sum_{j \in \mathscr{Q}}$. These terms stabilize oscillations in advection dominated regions and enable equal-order interpolations of the velocities and pressures by circumventing the Babuška–Brezzi condition [47, 48]. The stabilization parameters τ_j are defined as [49]

$$\tau_j = \left[\left(\frac{2}{\Delta t} \right)^2 + \left(\frac{2\,\|\mathbf{u}\|}{h_j} \right)^2 + \left(\frac{4\mu_i}{h_j^2} \right)^2 \right]^{-1/2} \tag{44}$$

where h_j is computed elementwise as [50]

$$h_j = \sqrt{2} \cdot A^{\mathrm{el}}_i / h_{\mathrm{diag}} \tag{45}$$

DOI: 10.1002/fld

with A_j^{el} being the element area and h_{diag} being the larger diagonal distance between the nodes of the quadrilateral element [50].

4.2. Time integration

Time integration is performed using the Crank–Nicholson scheme, see e.g. [51],

$$\frac{\partial \mathbf{u}^h}{\partial t} = \frac{\mathbf{u}_{n+1}^h - \mathbf{u}_n^h}{\Delta t}, \quad \mathbf{u}^h = \theta \mathbf{u}_{n+1}^h + (1-\theta)\mathbf{u}_n^h, \quad \theta = \frac{1}{2} \tag{46}$$

The velocities $\mathbf{u}_n^h$ are related to the previous time step. As there is no time derivative in the continuity equation (2), the term $\int_\Omega q^h \nabla \cdot \mathbf{u}^h \, \mathrm{d}\Omega$ becomes directly $\int_\Omega q^h \nabla \cdot \mathbf{u}_{n+1}^h \, \mathrm{d}\Omega$. The pressure is computed fully implicitly [52], i.e. p^h in (40)–(43) is directly replaced by p_{n+1}^h.

In the XFEM, fixed meshes are often used throughout the simulation, so that the mesh velocity is naturally zero. However, the enrichment of the approximation depends upon the interface position which moves in time. That is, the enriched shape functions are time-dependent and a standard time-stepping scheme, e.g. Equation (46), does not take this into account. This issue is worked out in [53]. There, it is shown that space–time finite elements are a natural choice for the XFEM, however, time-stepping methods may also be used as a simplification. In [53], the situation is analyzed for time derivatives of functions that involve *strong* discontinuities. For standard two-fluid flows, it is noted that strong discontinuities are present only in the pressure field and that there is no time derivative of the pressure.

We found that time derivatives of *weakly* discontinuous fields (such as the velocity fields in two-fluid flows) can, in fact, be discretized with very satisfactory results by standard time-stepping schemes. Therefore, the convergence behavior in time was analyzed for $\theta = 1$ (Euler backwards) and $\theta = 0.5$ (Crank–Nicholson) in Equation (46). It was found that the convergence rate was clearly improved for $\theta = 0.5$ compared with first order convergence for $\theta = 1$. That is, although a slight inconsistency in the time-discretization is present when using time stepping in the XFEM context, the convergence in time in incompressible two-fluid problems is still close to optimal for $\theta = 0.5$. Other examples where time-stepping schemes are realized for instationary problems in the frame of the XFEM are given e.g. in [20, 54–57].

4.3. Imposing essential boundary conditions

The construction of a set of shape functions $\{N_i(\mathbf{x})\}$, which is used for the approximations of the velocities and pressure, is discussed in Section 3. Locally, enriched MLS functions are used. These shape functions have a larger support when compared with standard finite element shape functions. Furthermore, MLS functions do in general not have the Kronecker-δ property, i.e.

$$N_i^{\mathrm{MLS}}(\mathbf{x}_j) \neq \delta_{ij} \tag{47}$$

These two aspects—the increased support size and the absence of the Kronecker-δ property—have important consequences for the imposition of essential boundary conditions. These aspects are well-known in the context of meshfree methods. An overview of different techniques to apply boundary conditions in this situation is given in [30, 31].

For the coupled shape functions $N_i(\mathbf{x})$, which are used as test and trial functions, see Section 3, it may be shown that

$$N_i(\mathbf{x}_j) \neq \delta_{ij} \quad \forall i \in I_{\mathrm{MLS}} \;\; \forall j \in I_{\mathrm{cut}} \tag{48}$$

DOI: 10.1002/fld

where I_{MLS} and I_{cut} are defined in Section 3.1. In words, the shape functions of the nodes in I_{MLS} do not have the Kronecker-δ property at the nodes in I_{cut}.

Assume that node k is in I_{cut} and located on the Dirichlet boundary $\Gamma_{\mathbf{u}}$. The velocity component $\hat{u}(\mathbf{x}_k)$ shall be prescribed. Then, the matrix line corresponding to u at node k is replaced by

$$[\ldots, N_{k-1}(\mathbf{x}_k), N_k(\mathbf{x}_k), N_{k+1}(\mathbf{x}_k), \ldots] \tag{49}$$

instead of only

$$[\ldots, 0, 1, 0, \ldots] \tag{50}$$

which would be the case for shape functions having the Kronecker-δ property (e.g. standard finite element shape functions).

Furthermore, it is noted that due to the increased support size, a particular node in I_{MLS} may have a corresponding shape function $N_i(\mathbf{x})$, which is non-zero along the Dirichlet boundary although this node is inside the domain (i.e. not on the Dirichlet boundary itself). This contradicts the requirements for the test space $\mathcal{V}^h_{\mathbf{u}}$ in (38) and must be compensated by an evaluation of the boundary term along the Dirichlet boundary. The boundary terms are present due to the use of the divergence theorem in order to avoid second order terms,

$$\int_\Omega \mathbf{w}^h \nabla \cdot \boldsymbol{\sigma} \, \mathrm{d}\Omega = \int_\Omega \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma} \, \mathrm{d}\Omega - \int_\Gamma \mathbf{w}^h \cdot \boldsymbol{\sigma} \cdot \mathbf{n} \, \mathrm{d}\Gamma \tag{51}$$

The boundary Γ falls into the Dirichlet and Neumann boundary $\Gamma_{\mathbf{u}}$ and $\Gamma_{\mathbf{h}}$, see Section 2.1, therefore,

$$\int_\Gamma \mathbf{w}^h \cdot \boldsymbol{\sigma} \cdot \mathbf{n} \, \mathrm{d}\Gamma = \int_{\Gamma_{\mathbf{u}}} \mathbf{w}^h \cdot \boldsymbol{\sigma} \cdot \mathbf{n} \, \mathrm{d}\Gamma + \int_{\Gamma_{\mathbf{h}}} \mathbf{w}^h \cdot \hat{\mathbf{h}} \, \mathrm{d}\Gamma \tag{52}$$

The first term on the right-hand side vanishes for shape functions that fulfill (38). This, however, is not true for the shape functions defined in Section 3. Therefore, this term needs to be evaluated and influences the system matrix (instead of the Neumann boundary term which influences the right-hand side). It is noted that the integration has only to be realized in the vicinity of an interface cutting the Dirichlet boundary, more precisely in $\Gamma_{\mathbf{u}} \setminus \partial\Omega_{\text{MLS}}$. The additional costs of the evaluation are negligible.

4.4. Integration

In elements not cut by the interface $\Gamma^{\pm}$, the proposed shape functions of the intrinsic XFEM are sufficiently smooth such that standard Gauss integration is suitable. However, in cut elements, the shape functions have jumps or kinks along the interface depending on the enrichment, see Section 3.2.3. This must be considered adequately which is achieved by dividing the elements into integration cells as shown in Figure 11; similar approaches are discussed in [18, 19, 58]. The partitioning depends on the interpolation of the level-set function $\phi(\mathbf{x})$, as discussed in Section 2.3. Here, each cut quadrilateral element is subdivided into two triangulars. In each triangular, linear interpolation functions are employed. Then, the zero-level of the level-set functions, which defines the position of the interface, is piecewise linear. The triangulars are further subdivided into integration cells depending on the interface situation. In each integration cell, standard Gauss points are placed, see Figure 11.

DOI: 10.1002/fld

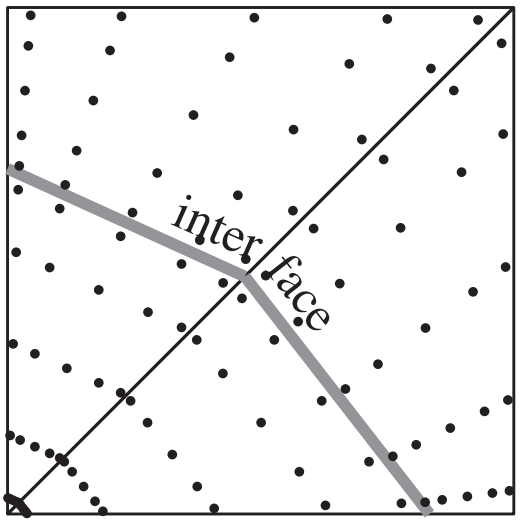

Figure 11. Dividing a cut element into integration domains for standard Gauss integration.

4.5. Moving interfaces in the level-set method

The level-set method is used for the implicit description of the moving interfaces between the two fluids, see Section 2.3. In each time step, the level-set values are advected with the velocity field of the fluid. Therefore, the scalar advection equation

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0 \tag{53}$$

is solved in the domain, where $\mathbf{u}$ are the fluid velocities. In the test cases presented in this work, no inflow and outflow regions are present, that is, the fluids are moving in a closed domain. Then, no Dirichlet boundary conditions are required for the advection equation. The initial condition is

$$\phi(\mathbf{x}, 0) = \hat{\phi}_0(\mathbf{x}) \quad \text{in } \Omega \text{ at } t = 0 \tag{54}$$

The following test and trial spaces are used:

$$\mathscr{S}^h_\phi = \mathscr{V}^h_\phi = \{\phi^h | \phi^h \in \mathscr{H}^{1h}\} \tag{55}$$

and standard bi-linear shape functions are used everywhere in the domain Ω. This is justified due to the fact that the level-set function is smooth everywhere in Ω. The SUPG-stabilized and discretized weak form may be stated as: Find $\phi^h \in \mathscr{S}^h_\phi$ such that $\forall \psi^h \in \mathscr{V}^h_\phi$,

$$\int_\Omega (\psi^h + \tau \mathbf{u}^h \cdot \nabla \psi^h) \cdot \left(\frac{\partial \phi^h}{\partial t} + \mathbf{u}^h \cdot \nabla \phi^h \right) \mathrm{d}\Omega = 0 \tag{56}$$

For the time integration, again the Crank–Nicholson scheme is used, see Section 4.2. A partitioned, *strongly* coupled algorithm is used for the solution of the Navier–Stokes equations (40)–(43) and the advection equation (56).

DOI: 10.1002/fld

An important issue in the level-set method is the reinitialization of the level-set function after regular time-intervals (often in each time step). This is necessary because otherwise the level-set function becomes increasingly steep in local regions leading to unphysical results such as oscillations, e.g. [59]. Furthermore, the distance property of the level-set function is no longer maintained after the transport. Thus, a reinitialization of the level-set function has the aim to maintain the distance property and avoid prohibitively large gradients. Several techniques are given in the literature, see e.g. [59–61].

The reinitialization in this work is realized by taking the transported, discretized values of the level-set function in the new time step, $\phi^h_{n+1}(\mathbf{x}_i)$, $i \in I$, and determining explicitly the intersections of the zero-level with the element edges. Then, for each node in I, the smallest distance to the intersection points is determined, which leads to the reinitialized level-set function $\tilde{\phi}^h_{n+1}(\mathbf{x}_i)$. It is noted that this reinitialization does not change the position of the interface but only values of the level-set function. We find that in the two-dimensional examples considered in this work, this *ad hoc* approach gives satisfactory results in reasonable time (the reinitialization costs less than 1% of the overall computation time).

4.6. Surface tension

The surface tension term in Equation (43) involves the curvature κ of the interface $\Gamma^{\pm}$. For interfaces defined by the level-set function $\phi(\mathbf{x}, t)$ with $\|\phi\|=1$, which holds for the signed-distance function used herein, the curvature may be computed as $\kappa=\Delta\phi=\nabla\cdot\nabla\phi$. However, it is shown in Appendix A, that the situation is delicate for reinitialized level-set functions, and that κ computed by the reinitialized level-set function is often not useful. Therefore, it is desirable, to reformulate the surface tension term in Equation (43) by means of the Laplace–Beltrami operator, see e.g. [16, 39],

$$\int_{\Gamma^{\pm}} \gamma\kappa \mathbf{w}^h \cdot \hat{\mathbf{n}}\, \mathrm{d}\Gamma = -\int_{\Gamma^{\pm}} \gamma \underline{\nabla}\,\mathrm{id} : \underline{\nabla}\mathbf{w}^h\, \mathrm{d}\Gamma + \text{boundary terms} \tag{57}$$

where $\underline{\nabla} f = \nabla f - (\nabla f \cdot \hat{\mathbf{n}})\hat{\mathbf{n}}$, and id is an identity mapping on the interface. The important advantage of this formulation is that the curvature κ is not needed. The boundary terms are only present for *open* interfaces, i.e. those which cut the boundary Γ of the domain Ω. For the description of the boundary terms, we reduce ourselves to two-dimensional domains Ω: Then, an open interface $\Gamma^{\pm}$ is assumed to cut the boundary at P_1 and P_2, see Figure 12(a), and the boundary terms are

$$\gamma[\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{P_1}^{P_2} = \gamma[\mathbf{w}^h(P_2)\cdot\hat{\mathbf{t}}(P_2) - \mathbf{w}^h(P_1)\cdot\hat{\mathbf{t}}(P_1)] \tag{58}$$

with $\hat{\mathbf{t}}$ being the tangent vector on $\Gamma^{\pm}$.

Furthermore, as shown in Appendix B, the situation simplifies drastically for polygonal interfaces, which means the interface consists of straight line segments. Assume an *open* polygon with n points $B_1, \ldots, B_n$, with $B_1 = P_1$ and $B_n = P_2$. There are $n-1$ corresponding segments s_i between B_i and B_{i+1}, $i = 1, \ldots, n-1$, with a constant tangent vector $\hat{\mathbf{t}}_i$ in each segment, see Figure 12(b). The surface tension integral over the interface can then be replaced by the following summation:

$$\int_{\Gamma^{\pm}} \gamma\kappa\mathbf{w}^h \cdot \hat{\mathbf{n}}\, \mathrm{d}\Gamma = -\gamma \sum_{i=1}^{n-1} \hat{\mathbf{t}}_i [\mathbf{w}^h]_{B_i}^{B_{i+1}} + \gamma[\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{P_1}^{P_2} \tag{59}$$

$$= \gamma \sum_{i=2}^{n-1} \mathbf{w}^h(B_i)(\hat{\mathbf{t}}_i - \hat{\mathbf{t}}_{i-1}) \tag{60}$$

DOI: 10.1002/fld

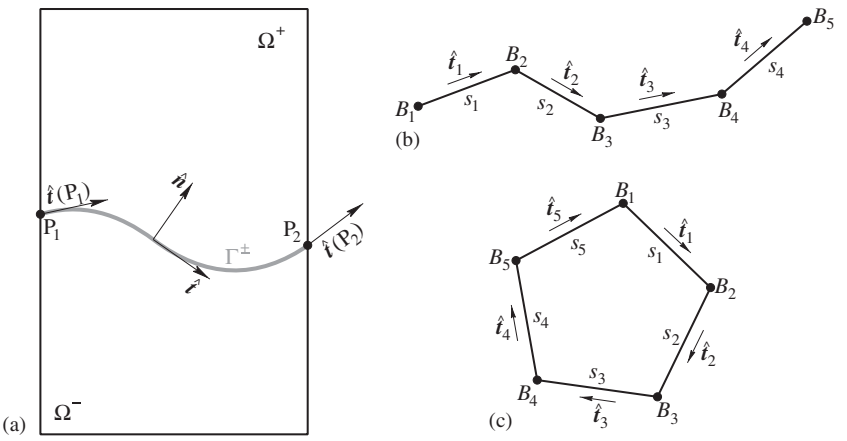

Figure 12. (a) Example of an open interface $\Gamma^{\pm}$ with intersection points P_1 and P_2 with the domain boundary Γ, (b) and (c) show examples of an open and closed polygonal interface, respectively.

see Appendix B for the proof. For a *closed* polygon with n points $B_1, \ldots, B_n$ and n segments $s_1, \ldots, s_n$, where the last segment s_n is between B_n and B_1, see Figure 12(c), the situation is similar

$$\oint_{\Gamma^{\pm}} \gamma\kappa \mathbf{w}^h \cdot \hat{\mathbf{n}}\, \mathrm{d}\Gamma = -\gamma \sum_{i=1}^{n} \hat{\mathbf{t}}_i [\mathbf{w}^h]_{B_i}^{B_{i+1}} \tag{61}$$

$$= \gamma \sum_{i=1}^{n} \mathbf{w}^h(B_i)(\hat{\mathbf{t}}_i - \hat{\mathbf{t}}_{i-1}) \tag{62}$$

where $B_{n+1} = B_1$ and $\hat{\mathbf{t}}_0 = \hat{\mathbf{t}}_n$.

5. NUMERICAL RESULTS

Throughout this paper, we restrict ourselves to two-dimensional domains and bi-linear finite elements. However, a generalization to three dimensions and different elements is possible and from a mathematical point of view is straightforward. Concerning the extension to three dimensions, it is mentioned that from an implementational point of view, the decomposition of the elements into $\mathcal{Q}_{\mathrm{FEM}}$ and $\mathcal{Q}_{\mathrm{MLS}}$ (Section 3.1), the construction of MLS shape functions (Section 3.2), and the coupling of the shape functions (Section 3.3) are easily parallelized. It remains to be shown how the intrinsic XFEM in three dimensions performs in the context of iterative solvers.

The test cases considered here include tank sloshing, a collapsing water column, and bubble flows and various Eötvös numbers. The results of the intrinsic XFEM are compared with those obtained by the classical FEM and the standard XFEM. The approximation of two-fluid flows by means of the classical FEM has been realized successfully in a number of different ways. For example, front tracking methods adjust the underlying mesh by following the moving interface. Front-capturing methods describe the interface implicitly (e.g. with the level-set method) and often

DOI: 10.1002/fld

refine the mesh near the interface. Both approaches can be employed in the classical FEM context and rely, for optimal performance, on suitable manipulations of the mesh.

Some of the classical FEM results discussed in this work are obtained by a front-capturing method on the same *fixed* mesh which is also used for the corresponding XFEM computation. As a consequence, the improvement in the results for the intrinsic XFEM can be directly traced back to the capability to capture the strong and/or weak discontinuities in the velocity and pressure fields appropriately, which is in contrast to the classical FEM. For these classical FEM results, the integration over the cut elements is realized as discussed in Section 4.4 for the intrinsic XFEM. The corresponding material properties are assigned at the integration points. Thereby, no integration error is introduced as would be the case when using standard Gauss integration in the cut elements. The performance of this classical FEM is then directly related to its approximation properties.

5.1. Patch test for two-fluid flows

In order to demonstrate the features of the intrinsic XFEM applied to two-fluid flows we start with considering two situations where the exact solution can be found with the enriched shape functions. The first situation involves a weak discontinuity in the pressure field, the second a strong discontinuity. Only the *stationary* Navier–Stokes equations are considered, and the velocity fields are zero in both cases.

5.1.1. Patch test 1: weak discontinuity in the pressure field. Assume a rectangular domain $\Omega = (0, x^\star) \times (0, y^\star)$, where two different fluids are separated by a horizontal interface $\Gamma^\pm$ at height $h^\star$ which cuts directly through the elements, see Figure 13(a) for a sketch of the situation. Gravitation is acting as a volume force $f_y = -g$. The level-set function is negative in the lower fluid Ω^- and positive in the upper Ω^+. The viscosity of the fluids is identical, $\mu = \mu^- = \mu^+$. The density of the lower fluid is larger than the upper, i.e. $\varrho^- > \varrho^+$. Then, the exact solution is stationary with $\mathbf{u}(\mathbf{x}) = \mathbf{0}$ and the pressure consists of only the hydrostatic pressure, see Figure 13(b),

$$p(\mathbf{x}) = \begin{cases} \varrho^+ \cdot g \cdot (y^\star - y) & \forall \mathbf{x} \in \Omega^+ \\ \varrho^+ \cdot g \cdot (y^\star - h^\star) + \varrho^- \cdot g \cdot (h^\star - y) & \forall \mathbf{x} \in \Omega^- \end{cases} \tag{63}$$

The velocities are prescribed along the whole boundary of Ω, and $p(x, y = y^\star) = 0$ along the upper boundary of the domain.

The solution (63) can be found exactly when using the shape functions of the intrinsic XFEM as proposed in Section 3, e.g. for the mesh shown in Figure 13(a). In contrast, this is not the case for standard FEM computations because standard finite element shape functions are not able to represent the inner-element kink in the exact solution. Anyway, acceptable solutions are obtained for this test case also by standard FEM (e.g. by under-integration in the cut elements). It is mentioned, that, obviously, for this specific test case it would not have been a problem to construct a mesh with element edges aligning with the interface. Then, also the standard FEM would be able to find the exact solution.

5.1.2. Patch test 2: strong discontinuity in the pressure field. A quadratic domain Ω has dimensions $(-x^\star, x^\star) \times (-x^\star, x^\star)$. One fluid Ω^- is inside a circle of radius $r^\star$ around $(0, 0)$, see Figure 14(a). The curvature on the interface has the constant value of $\kappa^\star = 1/r^\star$, and the surface tension coefficient

DOI: 10.1002/fld

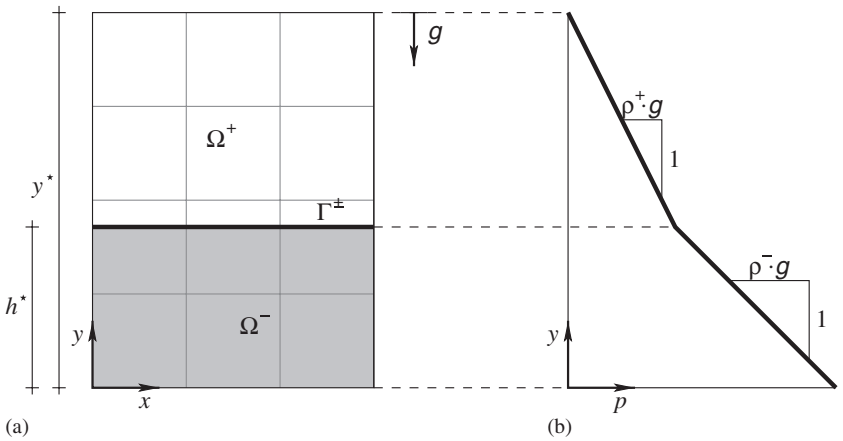

Figure 13. (a) Problem statement and example mesh for the first fluid patch test, (b) exact solution of the pressure.

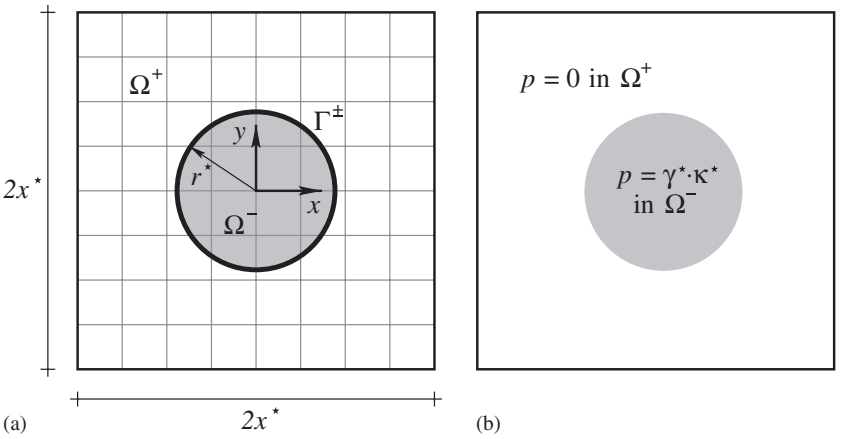

Figure 14. (a) Problem statement and example mesh for the second fluid patch test, (b) exact solution of the pressure.

is $\gamma^\star$. Density and viscosity of the two fluids are chosen identical, and no volume forces are present, $\mathbf{f}=\mathbf{0}$. The exact solution of the velocities is $\mathbf{u}(\mathbf{x})=\mathbf{0}$ and the pressure is

$$p(\mathbf{x}) = \begin{cases} 0 & \forall \mathbf{x} \in \Omega^+ \\ \gamma^\star \cdot \kappa^\star & \forall \mathbf{x} \in \Omega^- \end{cases} \tag{64}$$

see Figure 14(b). The boundary conditions are prescribed as for the first patch test. Again, the intrinsic XFEM is able to represent the exact solution (64) exactly without any element edges aligning with the circular interface. In contrast, classical FEM computations for this type of problem

DOI: 10.1002/fld

are known to achieve poor accuracy due to the jump in the pressure field, see e.g. [16]. It is noted that only in this test case, the exact level-set representation of the interface,

$$\phi(\mathbf{x}) = \|\mathbf{x}\| - r^\star \tag{65}$$

has been used instead of its approximation $\phi^h(\mathbf{x})$.

5.2. Sloshing tank

A two-fluid flow in a sloshing tank is considered next. The setup of the test case follows the description in [22]. The domain Ω is a container with height 1.5 m and width 1.0 m. The two fluids in Ω^- and Ω^+ are separated by a sinusoidal interface

$$\Gamma^\pm = \{(x, y) : y = 1.01 + 0.1 \cdot \sin((x - 0.5) \cdot \pi), 0 \leqslant x \leqslant 1\} \tag{66}$$

see Figure 15(a). The density of the fluids in Ω^- and Ω^+ is $\varrho^- = 1000\,\mathrm{kg/m^3}$ and $\varrho^+ = 1\,\mathrm{kg/m^3}$, respectively, and the viscosity is $\mu^- = 1\,\mathrm{kg/s/m}$ and $\mu^+ = 0.01\,\mathrm{kg/s/m}$. No surface tension is considered here. A volume force (gravitation) of $f_y = -g = -1.0\,\mathrm{m/s^2}$ is considered. Slip-conditions are assumed along the walls of the tank, and $p = 0\,\mathrm{N/m^2}$ is set along the upper boundary. The situation is observed for $t = (0, 20\,\mathrm{s})$.

For the intrinsic XFEM, the domain is discretized by 20×30, 40×60, and 80×120 elements. The velocity and pressure approximations are abs-enriched according to Section 3.2.3. The results are compared with those obtained by the standard FEM, where also a 160×240 element mesh is used. The time step in the Crank–Nicholson scheme is chosen as $\Delta t = 0.05\,\mathrm{s}$.

Firstly, we are interested in the area conservation of the two fluids. The domain Ω^- has the area $A^- = 1.0 \times 1.01 = 1.01\,\mathrm{m^2}$. As both fluids are immiscible and incompressible, this area should be conserved. However, it is well-known that the level-set method, in fact, does not conserve the area exactly [14]. As well as the position of the interface is only an approximation of the real position,

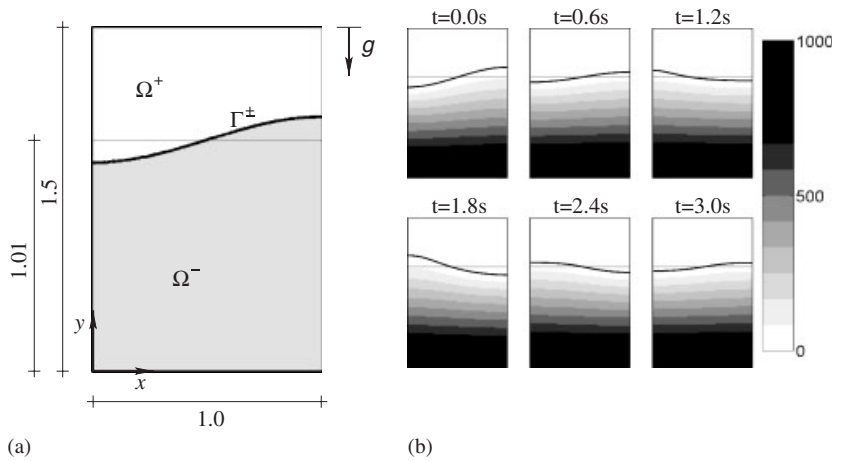

Figure 15. (a) Problem statement for the sloshing tank test case, (b) position of the interface and pressure field at different times.

DOI: 10.1002/fld

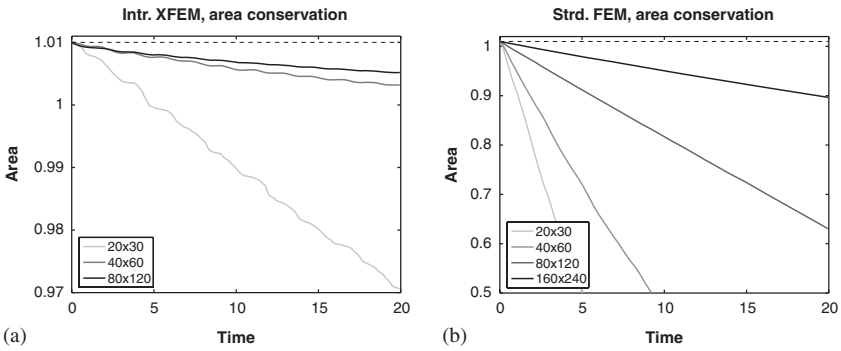

Figure 16. (a) and (b) show the change in the area A^- after $t=20\,\mathrm{s}$ for the intrinsic XFEM and classical FEM, respectively. Note the different scaling of the y-axis.

this also holds for the area A^-. Figure 16(a) and (b) show the development of the area of domain Ω^- over time for the intrinsic XFEM and classical FEM (with interface-capturing and a fixed mesh). For both methods, the area conservation clearly improves with mesh refinement. However, a dramatic difference can be observed between the performance of the intrinsic XFEM and the standard FEM: The intrinsic XFEM maintains about 97% of the area A^- (after 20 s) already on the rough 20×30 mesh, whereas the classical FEM only maintains about 90% of the area even on the fine 160×240 mesh. Clearly, the performance of the classical FEM would lead to results of similar quality than those obtained by the intrinsic XFEM once the mesh is manipulated during the computation, e.g. with a front-tracking scheme [3]. The aim here is to show that for *fixed* meshed, enriched meshes improve the accuracy significantly.

The position where the interface $\Gamma^{\pm}$ meets the left boundary is called height $h^{\star}$. This height is depicted over time in Figure 17 for some of the meshes used. It can be seen that due to the viscous damping the amplitude of the height decreases. The frequency of the oscillation is 0.279 Hz. A clear convergence to the solution given in [22] can be observed for the intrinsic XFEM. For the classical FEM, only the solution on the (fixed) 160×240 mesh is shown, the lack of area conservation overshadows the whole solution. It is mentioned again that an interface tracking algorithm would also be well suited for this test case as there are no topological changes in the interface position throughout the simulation. Then, the results would be of a similar quality than those obtained by the intrinsic XFEM on a fixed mesh.

5.3. Collapsing water column

This test case considers a collapsing water column in a domain $\Omega=0.584\,\mathrm{m}\times 0.45\,\mathrm{m}$, as described e.g. in [22, 62, 63]. Experimental data are found in [64]. In the initial situation, the fluid in Ω^- is placed on the left-hand side as a water column with dimensions $a\times b=0.146\,\mathrm{m}\times 0.292\,\mathrm{m}$, see Figure 18(a) for a sketch of the situation. The rest of the domain belongs to $\Omega^+=\Omega\setminus\Omega^-$. The fluid properties are $\rho^-=1000\,\mathrm{kg/m^3}$, $\rho^+=1\,\mathrm{kg/m^3}$, and $\mu^-=10^{-3}\,\mathrm{kg/s/m}$, $\mu^+=10^{-5}\,\mathrm{kg/s/m}$. No surface tension is considered, and a volume force (gravitation) is chosen as $f_y=-g=-9.81\,\mathrm{m/s^2}$. Slip-conditions are assumed along the boundaries of the domain, and $p=0\,\mathrm{N/m^2}$ is set at the upper boundary. The situation is observed for $t=(0, 0.3\,\mathrm{s})$ with $\Delta t=3\times 10^{-4}$. See Figure 19 for

DOI: 10.1002/fld

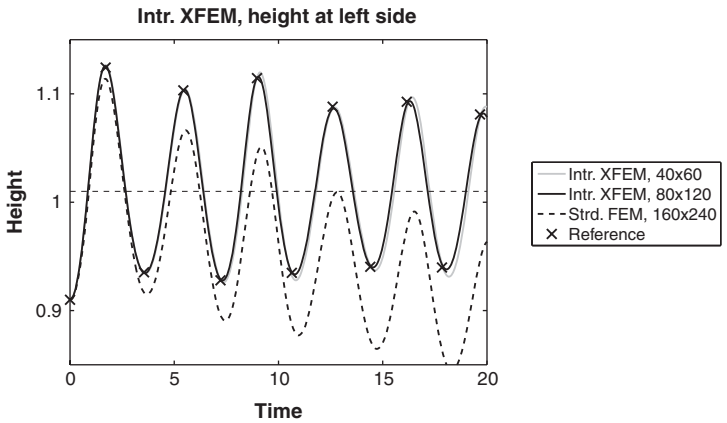

Figure 17. Interface position $h^\star$ at the left wall of the tank for different meshes.

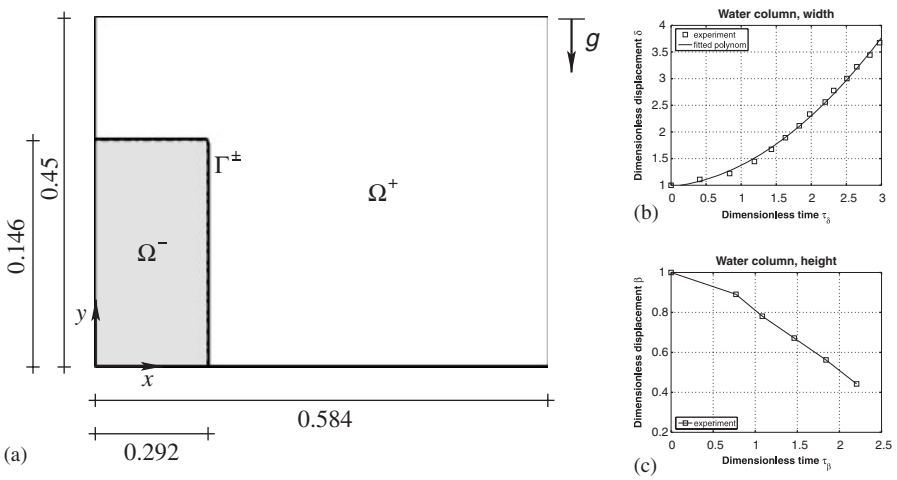

Figure 18. (a) Problem statement for the collapsing water column test case, (b) and (c) show experimental data by Martin and Moyce [64].

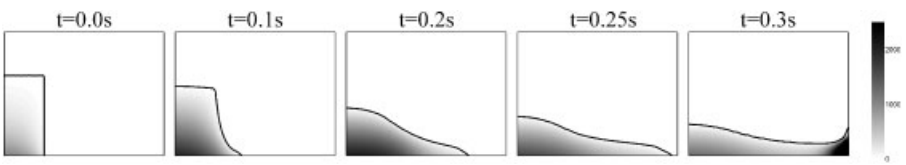

Figure 19. Interface position and pressure field at some selected points in time for the collapsing water column test case.

DOI: 10.1002/fld

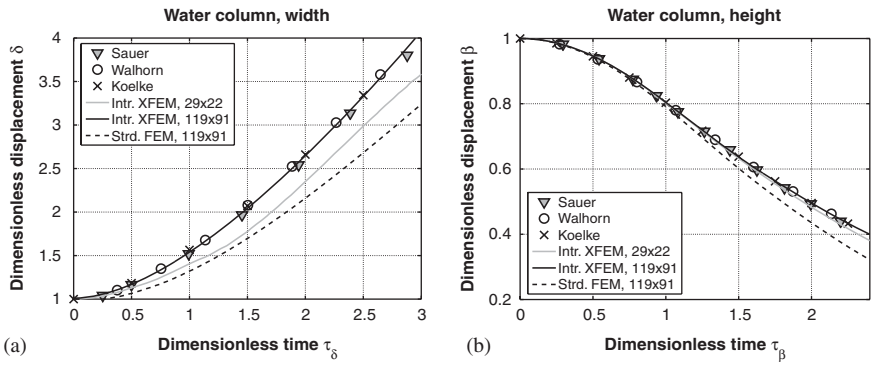

Figure 20. Dimensionless displacements over time for the collapsing water column test case.

interface positions and pressure fields at some selected points in time. The meshes used have 29×22 and 119×91 elements.

In Figure 20, we use the dimensionless times

$$\tau_\delta = t\sqrt{2g/a}, \quad \tau_\beta = t\sqrt{2g/b} \tag{67}$$

and displacements

$$\delta = x^\star/a, \quad \beta = y^\star/b \tag{68}$$

for displaying the results of the collapsing water column test case. The variables $x^\star(t)$ and $y^\star(t)$ are the intersection points of the interface $\Gamma^\pm$ with the bottom and left wall of the domain, respectively. Clearly, $x^\star(t=0)=a$ and $y^\star(t=0)=b$.

The solution of the intrinsic XFEM is compared with reference solutions given in [22, 62, 63], see Figure 20. In [22], the standard XFEM is used with space–time finite elements. Reference [62] uses an implicit description by the volume-of-fluid approach, see e.g. [4]. A front-tracking method is employed in [63]. An excellent agreement of the results obtained by the different approaches is found. Again, only the classical FEM on a fixed mesh gives unsatisfactory results.

5.4. Rising bubble

Bubble flows at different Eötvös numbers are considered next, experimental data are found e.g. in [65]. The external forces, densities, and viscosities are kept constant for all bubble flows. Only the diameter d of the circular bubble is changed; the bubble area is denoted by Ω^-. The domain Ω depends on the bubble diameter and is given as $\Omega=2d\times4d$, see Figure 21(a). The fluid properties are $\rho^-=1\,\mathrm{kg/m^3}$, $\rho^+=1000\,\mathrm{kg/m^3}$, and $\mu^-=10^{-3.5}\,\mathrm{kg/s/m}$, $\mu^+=10^{-1.5}\,\mathrm{kg/s/m}$. The surface tension coefficient is $\gamma=10^{-3}\,\mathrm{kg/s^2}$, and gravitational forces are $f_y=-g=-0.01\,\mathrm{m/s^2}$. The Morton number is

$$Mo=\frac{g\cdot(\mu^+)^4\cdot(\rho^+-\rho^-)}{(\rho^+)^2\cdot\gamma^3}=0.01 \tag{69}$$

DOI: 10.1002/fld

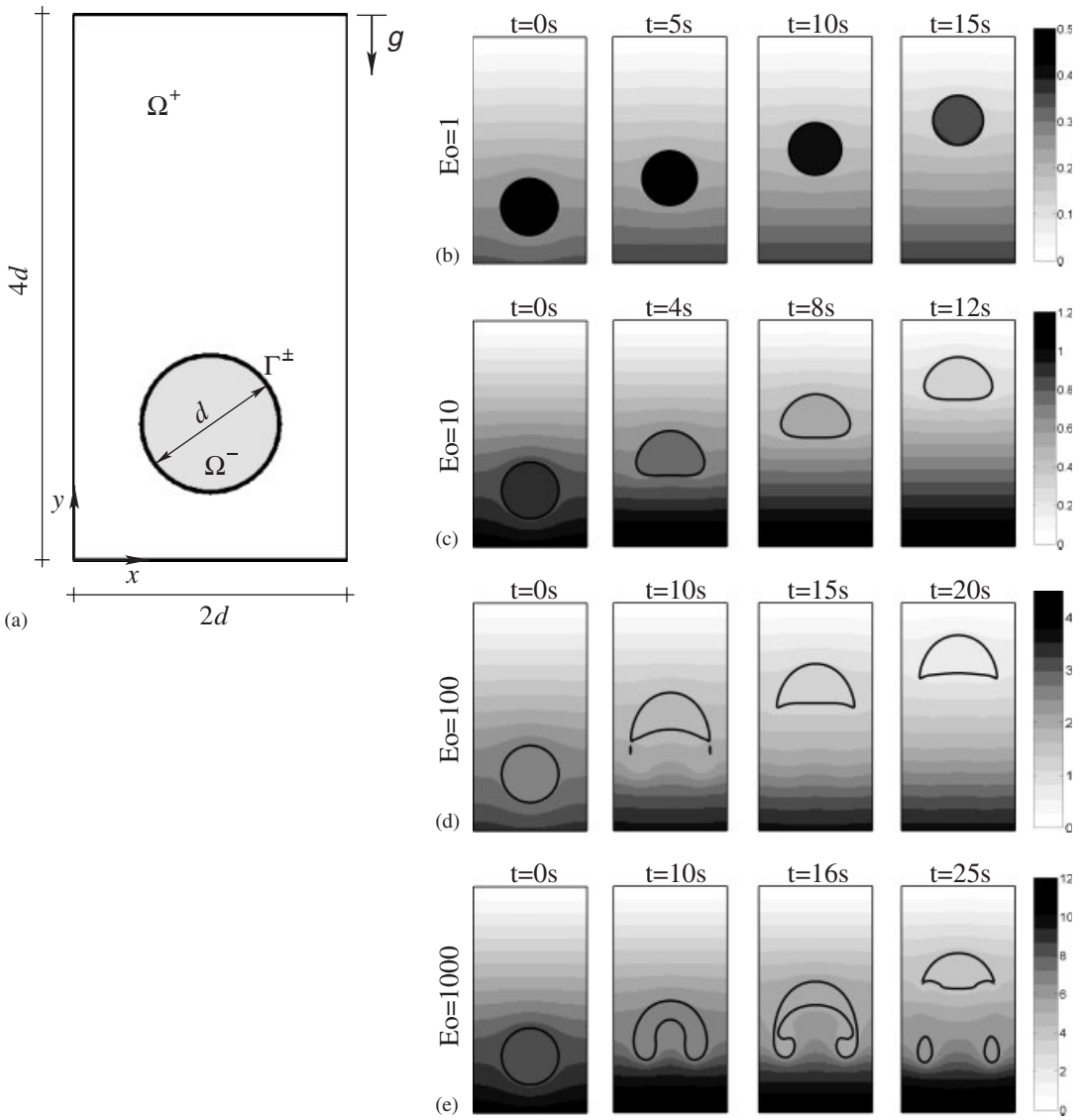

Figure 21. (a) Problem statement for the rising bubble test case, (b)–(e) show results at different Eötvös numbers, Eo = {1, 10, 100, 1000}, respectively.

for all examples considered here. Four different diameters are chosen as $d = \{0.01, 10^{-1.5}, 0.1, 10^{-0.5}\,\mathrm{m}\}$, and with

$$Eo = \frac{g \cdot (\rho^+ - \rho^-) \cdot d^2}{\gamma} \tag{70}$$

DOI: 10.1002/fld

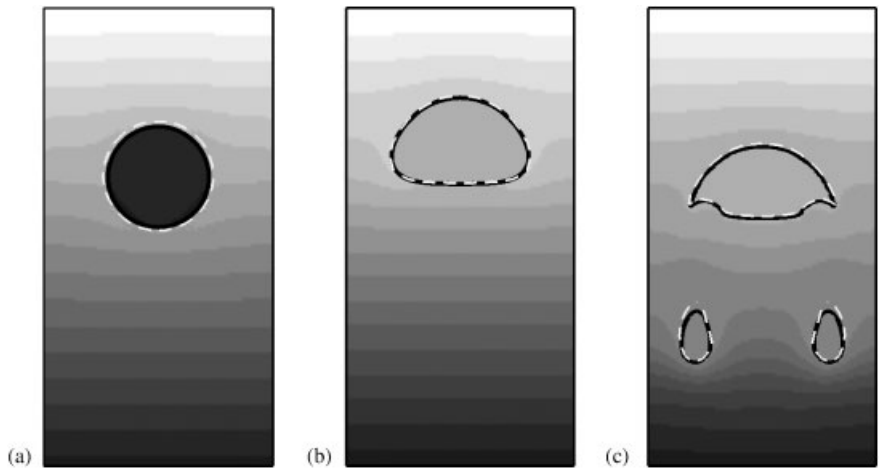

Figure 22. Comparison of the interface positions obtained by the intrinsic XFEM and standard XFEM [22] (dashed white lines) for different Eötvös numbers: (a) Eo = 1; (b) Eo = 10; and (c) Eo = 1000.

four Eötvös numbers $Eo = \{1, 10, 100, 1000\}$ result. Slip-conditions are assumed along the boundaries of the domain, and $p = 0\,\mathrm{N/m^2}$ is set at the upper boundary. The situation is observed in different time frames $t_{\mathrm{end}} = \{15\,\mathrm{s}, 12.5\,\mathrm{s}, 20\,\mathrm{s}, 25\,\mathrm{s}\}$, depending on the diameter of the bubble, or, in other words, on the Eötvös number. For each case, 1000 time steps are used. The spatial resolution is 80×160 elements. Figure 21 shows results of the bubble flows at selected points in time. For low Eötvös numbers, surface tension effects dominate the problem and stabilize the bubble. In contrast, for high Eötvös numbers, the surface tension is negligible and the form of the bubble is characterized by the external forces, densities and viscosities.

In Figure 22, the results of the intrinsic XFEM are compared with those obtained by a standard XFEM procedure with space–time finite elements [22]. It can be seen that the form of the bubble coincides in both methods and also agrees well with experimental results for the bubble form as given e.g. in [65]. There is a small difference in the interface positions as a result of the limited mesh resolution and the different time integration schemes used.

6. CONCLUSION

The intrinsic XFEM is applied for the simulation of two-fluid flows. The approximation space used in the intrinsic XFEM is able to represent inner-element jumps and kinks exactly. This approximation space is built by standard finite element shape functions in the majority of the domain and special enriched moving least-squares functions in the vicinity of the interface. Inbetween the FE and MLS regions, a coupling of the two classes of shape functions is realized. Most importantly, the resulting approximation in the intrinsic XFEM has the same number of unknowns as a classical FE approximation. The computational work for the evaluation of the MLS shape functions is increased, however, these functions are only needed locally near the interface.

DOI: 10.1002/fld

The interfaces in the proposed method are described implicitly by the level-set method. This method provides a natural way to consider the movement of the interfaces by a transport equation. The enrichment of the shape functions is directly based on the level-set function. Time integration is realized by time-stepping schemes with satisfactory accuracy although they do not fully consider the time-dependency of the shape functions. Therefore, space–time finite elements in the context of the intrinsic XFEM are one possible extension of the proposed method. Special issues in the XFEM in general are the treatment of essential boundary conditions and the integration of the weak form. Surface tension effects are considered by means of the Laplace–Beltrami operator and a simplification which is valid for polygonal interfaces; this can be also applied analogously in classical finite element simulations.

The numerical results illustrate the capability of the intrinsic XFEM to represent jumps and kinks exactly. Standard test cases are considered such as the simulation of tank-sloshing, a collapsing water column and bubble flows at different Eötvös numbers. Excellent results are obtained which are in very good agreement with results obtained by the standard XFEM and finite element results in the context of interface tracking methods. We believe that the intrinsic XFEM is a good solution for the simulation of two-fluid flow problems on fixed meshes.

APPENDIX A

We show that the explicit determination of the curvature κ by means of the level-set function, $\kappa = \Delta\phi(\mathbf{x})$, performs poorly for reinitialized level-set functions. For simplicity, a stationary level-set function

$$\phi(\mathbf{x}) = \|\mathbf{x}\| - r, \quad r \in \mathbb{R} \tag{A1}$$

is considered (in two dimensions), where $\|\cdot\|$ is the Euclidean norm. The exact level-set function (A1) defines a circular interface with radius r around $\mathbf{x} = \mathbf{0}$. The curvature $\kappa(\mathbf{x})$ follows as

$$\kappa(\mathbf{x}) = \Delta\phi(\mathbf{x}) = 1/\|\mathbf{x}\| \tag{A2}$$

The procedure for determining the reinitialized level-set function $\tilde{\phi}$ is as follows: In a first step, discrete points $\tilde{\mathbf{x}}_j$, $j = 1, \ldots, \tilde{n}$, are determined along the interface, clearly, $\phi(\tilde{\mathbf{x}}_j) = 0$. For example, $\tilde{\mathbf{x}}_j$ are the intersection points of the interface with element edges.

For each point $\mathbf{x}$ in the domain, the shortest distance to all points $\tilde{\mathbf{x}}_j$, $j = 1, \ldots, \tilde{n}$, is determined. Then, the reinitialized level-set function is defined as

$$\tilde{\phi}(\mathbf{x}, \tilde{\mathbf{x}}_j) = \pm \min \|\mathbf{x} - \tilde{\mathbf{x}}_j\| \quad \forall j = 1, \ldots, \tilde{n} \tag{A3}$$

where the sign depends on which side of the polygon through $\tilde{\mathbf{x}}_j$ the point $\mathbf{x}$ lies. For the definition of the curvature, an alternative definition of $\tilde{\phi}(\mathbf{x}, \tilde{\mathbf{x}}_j)$ is useful. Consider a Voronoi diagram depending on the points $\tilde{\mathbf{x}}_j$. The Voronoi-cell of node $\tilde{\mathbf{x}}_k$ is denoted by V_k and for all $\mathbf{x} \in V_k$, $\tilde{\mathbf{x}}_k$ is the nearest node. Consequently,

$$\tilde{\phi}(\mathbf{x}, \tilde{\mathbf{x}}_j) = \pm \|\mathbf{x} - \tilde{\mathbf{x}}_k\| \quad \forall \mathbf{x} \in V_k, \quad k = 1, \ldots, \tilde{n} \tag{A4}$$

The curvature of the reinitialized level-set function is

$$\tilde{\kappa}(\mathbf{x}, \tilde{\mathbf{x}}_j) = \pm \frac{1}{\|\mathbf{x} - \tilde{\mathbf{x}}_k\|} \quad \forall \mathbf{x} \in V_k, \quad k = 1, \ldots, \tilde{n} \tag{A5}$$

DOI: 10.1002/fld

Figure A1(a) and (b) show the level-set function ϕ and curvature κ, respectively. Depending on the nodes $\tilde{\mathbf{x}}_j,\ j=1,\ldots,\tilde{n}$, Figure A1(c) and (d) show the resulting reinitialized level-set function $\tilde{\phi}(\mathbf{x})$ and curvature $\tilde{\kappa}(\mathbf{x})$. For an increasing number of nodes $\tilde{\mathbf{x}}_j$ on the circular interface, ϕ and $\tilde{\phi}$ look more and more identical. However, this is not true for the curvature. The following differences

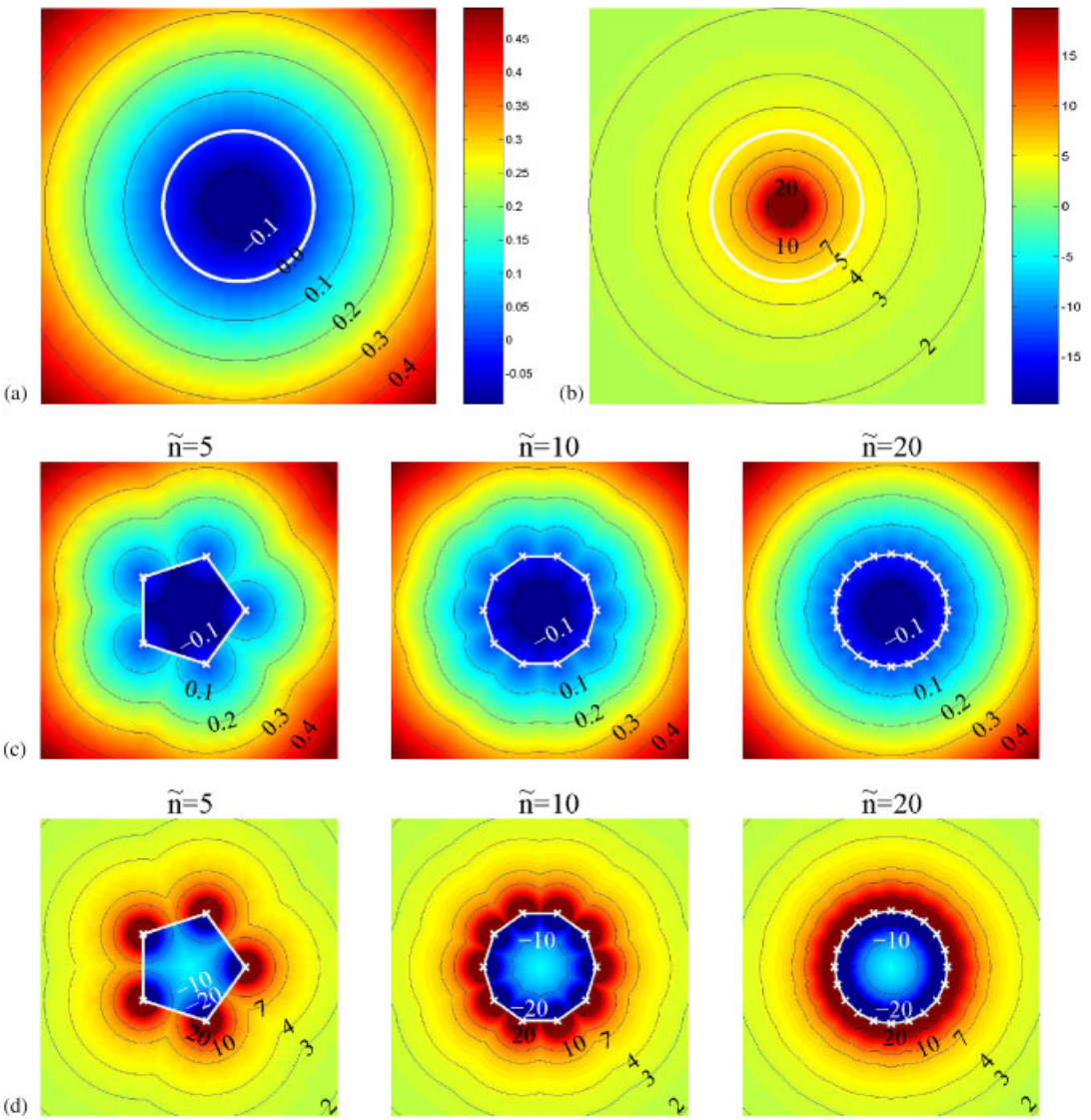

Figure A1. (a) and (b) show the level-set function $\phi(\mathbf{x})$ and curvature $\kappa(\mathbf{x})$. (c) and (d) show the reinitialized level-set function $\tilde{\phi}(\mathbf{x})$ and curvature of reinitialized level-set function $\tilde{\kappa}(\mathbf{x})$ for different nodes $\mathbf{x}_j,\ j=1,\ldots,\tilde{n}$.

DOI: 10.1002/fld

in the curvatures κ and $\tilde{\kappa}$ are found:

- The curvature $\tilde{\kappa}$ is singular at each node $\tilde{\mathbf{x}}_j$, $j=1,\ldots,\tilde{n}$, whereas κ is smooth on the interface.
- The curvature $\tilde{\kappa}$ inside the circle of radius r has the opposite sign as the curvature κ ($\tilde{\phi}$ is concave with respect to $\mathbf{0}$, whereas ϕ is convex). Therefore, $\tilde{\kappa}$ is discontinuous across the interface, whereas κ has the same sign on both sides.
- Away from the interface, the curvature $\tilde{\kappa}$ has kinks along the edges of the Voronoi diagram, i.e. it is C^0-continuous there. In contrast, κ is C^∞-continuous except at $\mathbf{x}=\mathbf{0}$.

These aspects have important consequences for the computation of the curvature on discrete meshes. Assume that

$$\phi^h(\mathbf{x}) = \sum_{i\in I} N_i^{\mathrm{FEM}}(\mathbf{x})\phi_i \tag{A6}$$

with $\phi_i=\phi(\mathbf{x}_i)$ from Equation (A1) is the discrete level-set function. Analogously, $\tilde{\phi}^h(\mathbf{x},\tilde{\mathbf{x}}_j)$ is defined as the discrete reinitialized level-set function with $\tilde{\mathbf{x}}_j$ being the intersection points of the interface with element edges. The curvature κ^h of $\phi^h(\mathbf{x})$ is determined by simple finite differences

$$\kappa^h = \frac{\partial^2\phi^h}{\partial x^2} + \frac{\partial^2\phi^h}{\partial x^2} \tag{A7}$$

$$= 1/h^2\cdot[\phi^h(x_{i-1},y_i) - 2\cdot\phi^h(x_i,y_i) + \phi^h(x_{i+1},y_i)] \tag{A8}$$

$$+1/h^2\cdot[\phi^h(x_i,y_{i-1}) - 2\cdot\phi^h(x_i,y_i) + \phi^h(x_i,y_{i+1})] \tag{A9}$$

and $\tilde{\kappa}^h$ is determined in the same way depending on $\tilde{\phi}^h(\mathbf{x},\tilde{\mathbf{x}}_j)$. Results for κ^h and $\tilde{\kappa}^h$ are shown in Figure A2(a) and (b) on a regular 55×55 mesh. It may be seen that $\tilde{\kappa}^h$ is dramatically different from κ^h and dominated by severe oscillations. Other computations of $\tilde{\kappa}^h$, e.g. by means of finite elements, lead to similar results. Therefore, in this work we avoid the explicit computation of the curvature by means of the level-set function and proceed as in Appendix B. It is mentioned that the explicit computation of the curvature has been realized by a special smoothing procedure in [66].

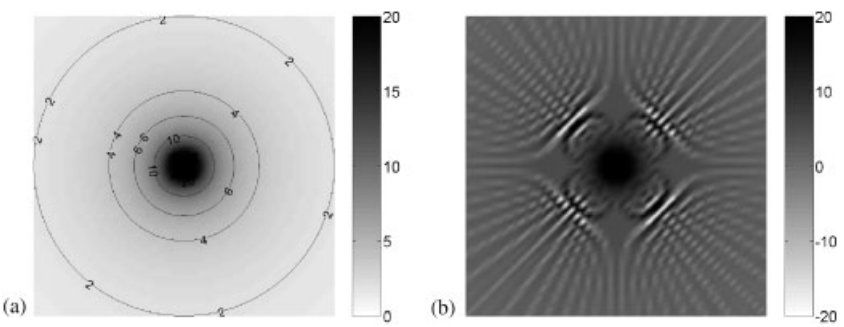

Figure A2. Curvature obtained by finite differences on a regular 55×55 mesh. (a) and (b) show $\kappa^h(\mathbf{x})$ and $\tilde{\kappa}^h(\mathbf{x})$, respectively, computed by means of $\phi^h(\mathbf{x})$ and $\tilde{\phi}^h(\mathbf{x})$.

DOI: 10.1002/fld

APPENDIX B

Surface tension is modeled with the interface integral

$$\int_{\Gamma^{\pm}} \gamma \kappa \mathbf{w}^h \cdot \hat{\mathbf{n}} \, \mathrm{d}\Gamma \tag{B1}$$

see Equation (43). In this work, the interface consists in an open or closed *polygon* as shown in Figure 12(b) and (c). It is now shown how the surface tension integral can be evaluated without an explicit computation of the curvature $\kappa(\mathbf{x})$. For explicit computations of κ, the interested reader is referred to [67, 68] and references given therein.

Assume three points B_1, B_2, B_3 which define an open polygon $\Gamma^{\pm}$ consisting of two straight lines $s_1 = \overline{B_1 B_2}$ and $s_2 = \overline{B_2 B_3}$ as e.g. shown in Figure B1. Each of the two lines has a constant normal vector $\hat{\mathbf{n}}_i$ and tangent vector $\hat{\mathbf{t}}_i$, $i = 1, 2$. The angle between the two lines is denoted by α. The curvature κ is zero along the straight lines, however, at the edge, κ has a 'peak' similar to a Dirac-δ function. The contribution of κ in the integral (B1) is defined in the framework of distributions [69]. As a consequence, the integral is neither zero nor infinite.

In order to evaluate the integral (B1) for this particular choice of $\Gamma^{\pm}$, the kink between the two lines is smoothed in the following way. A parameter ε is introduced, which defines the two points

$$C_1 = B_2 - \varepsilon \cdot \hat{\mathbf{t}}_1 \quad \text{and} \quad C_2 = B_2 + \varepsilon \cdot \hat{\mathbf{t}}_2 \tag{B2}$$

The segment of the circle Γ_c starts at C_1 and ends at C_2 and its radius is given as

$$r = \tan \frac{\alpha}{2} \cdot \varepsilon \tag{B3}$$

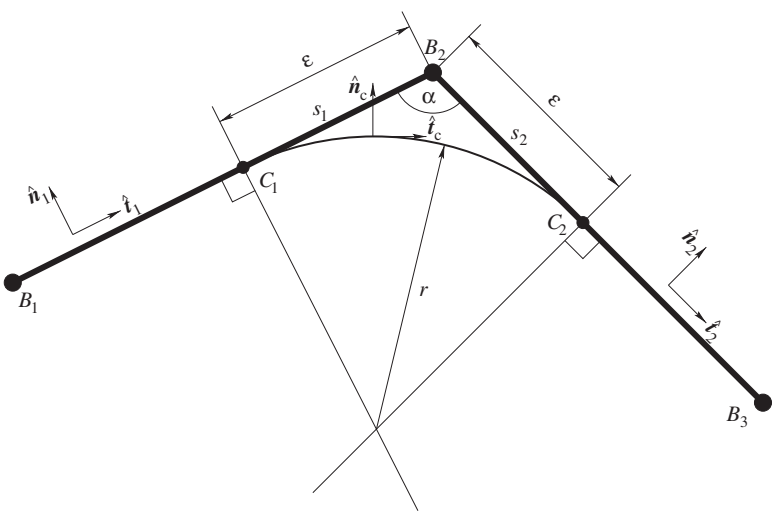

Figure B1. Sketch of the situation used for the proof.

DOI: 10.1002/fld

The normal and tangent vector on the segment of the circle is called $\hat{\mathbf{n}}_c$ and $\hat{\mathbf{t}}_c$. The curvature of the circle is constant $\kappa_c = \pm 1/r$, where the sign is positive for counter-clockwise integrations, and negative for clockwise integrations (as in Figure B1). It is noted that $\hat{\mathbf{t}}_c(C_1) = \hat{\mathbf{t}}_1$ and $\hat{\mathbf{t}}_c(C_2) = \hat{\mathbf{t}}_2$. From differential geometry [70], we have

$$\int_{\Gamma_c} \kappa_c \cdot \hat{\mathbf{n}}_c \, \mathrm{d}\Gamma = \hat{\mathbf{t}}_c(C_2) - \hat{\mathbf{t}}_c(C_1) \tag{B4}$$

$$= \hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1 \tag{B5}$$

It is interesting that although the segment of the circle Γ_c and the curvature κ_c depend on ε, the integral is, in fact, independent on ε. Let us now evaluate (B1) along the smoothed interface $\Gamma_c^{\pm}$ between B_1 and B_3. Firstly,

$$\int_{\Gamma_c^{\pm}} \gamma \kappa \mathbf{w}^h \cdot \hat{\mathbf{n}} \, \mathrm{d}\Gamma = \int_{\Gamma_c} \gamma \, \kappa_c \mathbf{w}^h \cdot \hat{\mathbf{n}}_c \, \mathrm{d}\Gamma \tag{B6}$$

because the linear segments $\overline{B_1 C_1}$ and $\overline{C_2 B_3}$ have zero-curvature and, therefore, no influence in the integral. Furthermore, with (B6) and $\varepsilon \longrightarrow 0$, it follows that

$$\int_{\Gamma_c} \gamma \kappa_c \mathbf{w}^h \cdot \hat{\mathbf{n}}_c \, \mathrm{d}\Gamma \overset{\varepsilon \to 0}{\longrightarrow} \gamma \mathbf{w}^h(B_2) \cdot (\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1) = \int_{\Gamma^{\pm}} \gamma \kappa \mathbf{w}^h \cdot \hat{\mathbf{n}} \, \mathrm{d}\Gamma \tag{B7}$$

It is noted that the same result can be obtained by evaluating the right-hand side of (57) for this situation, i.e.

$$-\int_{\Gamma^{\pm}} \gamma \underline{\nabla} \mathrm{id} : \underline{\nabla} \mathbf{w}^h \, \mathrm{d}\Gamma + \gamma [\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{B_1}^{B_3} \tag{B8}$$

$$= -\int_{s_1} \gamma \underline{\nabla} \mathrm{id} : \underline{\nabla} \mathbf{w}^h \, \mathrm{d}\Gamma - \int_{s_2} \gamma \underline{\nabla} \mathrm{id} : \underline{\nabla} \mathbf{w}^h \, \mathrm{d}\Gamma + \gamma [\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{B_1}^{B_3} \tag{B9}$$

Therefore, the special situation of *one* straight line s between P_1 and P_2 is considered. In this case, $\kappa = 0$, and from (57) follows immediately

$$\int_s \gamma \underline{\nabla} \mathrm{id} : \underline{\nabla} \mathbf{w}^h \, \mathrm{d}\Gamma = \gamma [\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{P_1}^{P_2} \tag{B10}$$

Consequently, for Equation (B9) follows:

$$-\gamma [\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{B_1}^{B_2} - \gamma [\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{B_2}^{B_3} + \gamma [\mathbf{w}^h \cdot \hat{\mathbf{t}}]_{B_1}^{B_3} \tag{B11}$$

$$= -\gamma [\mathbf{w}^h(B_2) \cdot \hat{\mathbf{t}}_1 - \mathbf{w}^h(B_1) \cdot \hat{\mathbf{t}}_1] - \gamma [\mathbf{w}^h(B_3) \cdot \hat{\mathbf{t}}_2 - \mathbf{w}^h(B_2) \cdot \hat{\mathbf{t}}_2] \tag{B12}$$

$$+ \gamma [\mathbf{w}^h(B_3) \cdot \hat{\mathbf{t}}_2 - \mathbf{w}^h(B_1) \cdot \hat{\mathbf{t}}_1] \tag{B13}$$

$$= \gamma \mathbf{w}^h(B_2) \cdot (\hat{\mathbf{t}}_2 - \hat{\mathbf{t}}_1) \tag{B14}$$

This is the same result than obtained in Equation (B7). The situation is generalized straightforward for polygonal interfaces with more than two segments. The resulting expressions are given in Section 4.6.

DOI: 10.1002/fld

ACKNOWLEDGEMENTS

The support of the German Research Foundation (DFG) in the frame of the Emmy-Noether-research group 'Numerical methods for discontinuities in continuum mechanics' is gratefully acknowledged.

REFERENCES

1. Güler I, Behr M, Tezduyar T. Parallel finite element computation of free-surface flows. *Computational Mechanics* 1999; **23**:117–123.
2. Tezduyar T, Behr M, Liou J. A new strategy for finite element computations involving moving boundaries and interfaces—the deforming-spatial-domain/space-time procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders. *Computer Methods in Applied Mechanics and Engineering* 1992; **94**:353–371.
3. Huerta A, Liu W. Viscous flow with large free surface motion. *Computer Methods in Applied Mechanics and Engineering* 1988; **69**:277–324.
4. Hirt C, Nichols B. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics* 1981; **39**:201–225.
5. Sussman M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics* 1994; **114**:146–159.
6. Sussmann M, Fatemi E, Smereka P, Osher S. An improved level set method for incompressible two-phase flows. *Computers and Fluids* 1998; **27**:663–680.
7. Belytschko T, Liu W, Moran B. *Nonlinear Finite Elements for Continua and Structures*. Wiley: Chichester, 2000.
8. Zienkiewicz O, Taylor R. *The Finite Element Method*, vols. 1–3. Butterworth-Heinemann: Oxford, 2000.
9. Babuška I, Melenk J. The partition of unity method. *International Journal for Numerical Methods in Engineering* 1997; **40**:727–758.
10. Strang G, Fix G. *An Analysis of the Finite Element Method*. Prentice-Hall: Englewood Cliffs, NJ, 1973.
11. Cruchaga M, Celentano D, Breitkopf P, Villon P, Rassineux A. A front remeshing technique for a Lagrangian description of moving interfaces in two-fluid flows. *International Journal for Numerical Methods in Engineering* 2006; **66**:2035–2063.
12. Raad P, Chen S, Johnson D. The introduction of micro cells to treat pressure in free surface fluid flow problems. *Journal of Fluids Engineering* 1995; **117**:683–690.
13. Sussmann M, Fatemi E. An efficient interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on Scientific Computing* 1999; **20**:1165–1191.
14. Osher S, Fedkiw R. *Level Set Methods and Dynamic Implicit Surfaces*. Springer: Berlin, 2003.
15. Gerbeau J, le Bris C, Bercovier M. Spurious velocities in the steady flow of an incompressible fluid subjected to external forces. *International Journal for Numerical Methods in Fluids* 1997; **25**:679–695.
16. Ganesan S, Matthies G, Tobiska L. On spurious velocities in incompressible flow problems with interfaces. *Computer Methods in Applied Mechanics and Engineering* 2006; DOI: 10.1016/j.cma.2006.08.018.
17. Pelletier D, Fortin A, Camarero R. Are FEM solutions of incompressible flows really incompressible? (Or how simple flows can cause headaches!). *International Journal for Numerical Methods in Fluids* 1989; **9**:99–112.
18. Belytschko T, Moës N, Usui S, Parimi C. Arbitrary discontinuities in finite elements. *International Journal for Numerical Methods in Engineering* 2001; **50**:993–1013.
19. Moës N, Dolbow J, Belytschko T. A finite element method for crack growth without remeshing. *International Journal for Numerical Methods in Engineering* 1999; **46**:131–150.
20. Chessa J, Belytschko T. The extended finite element method for two-phase fluids. *ASME Journal of Applied Mechanics* 2003; **70**:10–17.
21. Chessa J. The extended finite element method for two phase and free surface flow. *Dissertation*, Northwestern University, 2003.
22. Kölke A. Modellierung und Diskretisierung bewegter Diskontinuitäten in randgekoppelten Mehrfeldaufgaben. *Dissertation*, Technische Universität Braunschweig, 2005.
23. Fries T, Belytschko T. The intrinsic XFEM: a method for arbitrary discontinuities without additional unknowns. *International Journal for Numerical Methods in Engineering* 2006; **68**:1358–1385.
24. Strouboulis T, Babuška I, Copps K. The design and analysis of the generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2000; **181**:43–69.

25. Strouboulis T, Copps K, Babuška I. The generalized finite element method. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**:4081–4193.
26. Babuška I, Banerjee U, Osborn J. Survey of meshless and generalized finite element methods: a unified approach. *Technical Report 02-40*, TICAM, The University of Texas at Austin, 2002.
27. Hansbo A, Hansbo P. A finite element method for the simulation of strong and weak discontinuities in solid mechanics. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:3523–3540.
28. Areias P, Belytschko T. Letter to the editor. *Computer Methods in Applied Mechanics and Engineering* 2004; **195**:1275–1276.
29. Rabczuk T, Belytschko T, Fernández-Méndez S, Huerta A. Meshfree methods. In *Encyclopedia of Computational Mechanics*, vol. 1, Stein E, De Burst R, Hughes TJR (eds). Wiley: Chichester, 2004.
30. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P. Meshless methods: an overview and recent developments. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:3–47.
31. Fries T, Matthies H. Classification and overview of meshfree methods. *Informatikbericht-Nr. 2003-03*, Technical University Braunschweig, Brunswick, 2003. Available from: http://opus.tu-bs.de/opus/volltexte/2003/418/.
32. Fries T. A stabilized and coupled meshfree/meshbased method for fluid–structure-interaction problems. *Dissertation*, Technical University Braunschweig, Brunswick, Germany, 2005.
33. Hou S, Liu X. A numerical method for solving variable coefficient elliptic equation with interfaces. *Journal of Computational Physics* 2005; **202**:411–445.
34. Belytschko T, Gu L, Lu YY. Fracture and crack growth by element free Galerkin methods. *Modelling and Simulation in Material Science and Engineering* 1994; **2**:519–534.
35. Lancaster P, Salkauskas K. Surfaces generated by moving least squares methods. *Mathematics of Computation* 1981; **37**:141–158.
36. Fries T, Belytschko T. Convergence and stabilization of stress-point integration in mesh-free and particle methods. *International Journal for Numerical Methods in Engineering* 2007; DOI: 10.1002/nme.2198.
37. Chessa J, Wang H, Belytschko T. On the construction of blending elements for local partition of unity enriched finite elements. *International Journal for Numerical Methods in Engineering* 2003; **57**:1015–1038.
38. Fries T. A corrected XFEM approximation without problems in blending elements. *International Journal for Numerical Methods in Engineering* 2007; DOI: 10.1002/nme.2259.
39. Hysing S. A new implicit surface tension implementation for interfacial flows. *International Journal for Numerical Methods in Fluids* 2005; **51**:659–672.
40. Belytschko T, Lu Y, Gu L. Element-free Galerkin methods. *International Journal for Numerical Methods in Engineering* 1994; **37**:229–256.
41. Huerta A, Fernández-Méndez S. Enrichment and coupling of the finite element and meshless methods. *International Journal for Numerical Methods in Engineering* 2000; **48**:1615–1636.
42. Organ D, Fleming M, Terry T, Belytschko T. Continuous meshless approximations for nonconvex bodies by diffraction and transparency. *Computational Mechanics* 1996; **18**:225–235.
43. Fries T, Belytschko T. New shape functions for arbitrary discontinuities without additional unknowns. In *Meshfree Methods for Partial Differential Equations*, Griebel M, Schweitzer M (eds). Lectures Notes in Computational Science and Engineering. Springer: Berlin, submitted.
44. Tezduyar T. Stabilized finite element formulations for incompressible flow computations. In *Advances in Applied Mechanics*, vol. 28, Hutchinson J, Wu T (eds). Academic Press: New York, NY, 1992.
45. Tezduyar T, Mittal S, Ray S, Shih R. Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity–pressure elements. *Computer Methods in Applied Mechanics and Engineering* 1992; **95**:221–242.
46. Brooks A, Hughes T. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1982; **32**:199–259.
47. Babuška I. Error-bounds for finite element method. *Numerische Mathematik* 1971; **16**:322–333.
48. Brezzi F. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. *RAIRO Analyse Numérique* 1974; **R-2**:129–151.
49. Shakib F, Hughes T, Johan Z. A new finite element formulation for computational fluid dynamics: X. The compressible Euler and Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering* 1991; **89**:141–219.
50. Mittal S. On the performance of high aspect ratio elements for incompressible flows. *Computer Methods in Applied Mechanics and Engineering* 2000; **188**:269–287.

51. Donea J, Huerta A. *Finite Element Methods for Flow Problems*. Wiley: Chichester, 2003.
52. Löhner R. *Applied CFD Techniques*. Wiley: Chichester, 2001.
53. Chessa J, Belytschko T. Arbitrary discontinuities in space-time finite elements by level-sets and X-FEM. *International Journal for Numerical Methods in Engineering* 2004; **61**:2595–2614.
54. Belytschko T, Chean H, Xu J, Zi G. Dynamic crack propagation based on loss of hyperbolicity and a new discontinuous enrichment. *International Journal for Numerical Methods in Engineering* 2003; **58**:1873–1905.
55. Chessa J, Smolinski P, Belytschko T. The extended finite element method (XFEM) for solidification problems. *International Journal for Numerical Methods in Engineering* 2002; **53**:1959–1977.
56. Dolbow J, Merle R. Solving thermal and phase change problems with the extended finite element method. *Computational Mechanics* 2002; **28**:339–350.
57. Ji H, Chopp D, Dolbow J. A hybrid extended finite element/level set method for modeling phase transformations. *International Journal for Numerical Methods in Engineering* 2002; **54**:1209–1233.
58. Sukumar N, Chopp D, Moës N, Belytschko T. Modeling holes and inclusions by level sets in the extended finite-element method. *Computer Methods in Applied Mechanics and Engineering* 2001; **190**:6183–6200.
59. Zhao H, Chan T, Merriman B, Osher S. A variational level set approach to multiphase motion. *Journal of Computational Physics* 1996; **127**:179–195.
60. Peng D, Merriman B, Osher S, Zhao H, Kang M. A PDE-based fast local level set method. *Journal of Computational Physics* 1999; **155**:410–438.
61. Sussmann M, Smereka P, Osher S. A level set approach for computing solutions to incompressible two-phase flows. *Journal of Computational Physics* 1994; **114**:146–159.
62. Sauer J. Instationär kavitierende Strömungen—ein neues Modell, basierend auf front-capturing (Vof) und Blasendynamik. *Dissertation*, Universität Karlsruhe, 2000.
63. Walhorn E. Ein simultanes Berechnungsverfahren für Fluid-Struktur-Wechselwirkungen mit finiten Raum-Zeit-Elementen. *Dissertation*, Technische Universität Braunschweig, 2002.
64. Martin J, Moyce W. An experimental study of the collapse of liquid columns on a rigid horizontal plane. *Philosophical Transactions of the Royal Society of London* 1952; **244**:312–324.
65. Clift R, Grace J, Weber M. *Bubbles*, *Drops and Particles*. Academic Press: New York, NY, 1978.
66. Smolianski A. Numerical modelling of two-fluid interfacial flows. *Dissertation*, University of Jyväskylä, 2001.
67. Brickell F, Hsiung C. The total absolute curvature of closed curves in Riemannian manifolds. *Journal of Differential Geometry* 1974; **9**:177–193.
68. Sullivan J. Curvatures of smooth and discrete surfaces. In *Discrete Differential Geometry*, *Oberwolfach Seminars*, vol. 38, Al Bobenko P, Schröder P, Sullivan JM, Ziegler GM (eds). Birkhäuser: Basel, 2008.
69. Lax P. *Functional Analysis*. Wiley: Chichester, 2002.
70. do Carmo M. *Differential Geometry of Curves and Surfaces*. Prentice-Hall: Englewood Cliffs, NJ, 1976.

DOI: 10.1002/fld