

# A Volumetric Extrapolation Method for Weak Imposition of Interface Conditions on Level Sets

Dmitri Kuzmin, Jan-Phillip Bäcker

TU Dortmund University, Institute of Applied Mathematics

WCCM-APCOM 2022

### Sharp interface formulation

- XFEM, unfitted Nitsche FEM, CutFEM
- small cut cells may cause ill conditioning
- severe time step restrictions or instability

### Surrogate interface formulation

- shifted boundary method
- extrapolation of boundary conditions to mesh
- complex closest-point projection algorithms

### Diffuse interface formulation

- immersed boundary, phase field methods
- construction of approximate delta functions

Interface  $\Gamma(t) = \{\mathbf{x} \in \Omega : \phi(\mathbf{x}, t) = 0\}$  determined by  $\phi(\mathbf{x}, t)$  such that

- $\phi > 0$  in  $\Omega_+(t)$ ,  $\Omega_+(t)$  remains fully embedded in  $\Omega$
- $\phi < 0$  in  $\overline{\Omega} \setminus \overline{\Omega_+(t)}$ ,
- $\mathbf{n}_\pm = \mp \frac{\nabla \phi}{|\nabla \phi|}$  extended unit outward normal

Interface  $\Gamma(t) = \{\mathbf{x} \in \Omega : \phi(\mathbf{x}, t) = 0\}$  determined by  $\phi(\mathbf{x}, t)$  such that

- $\phi > 0$  in  $\Omega_+(t)$ ,  $\Omega_+(t)$  remains fully embedded in  $\Omega$
- $\phi < 0$  in  $\overline{\Omega} \setminus \overline{\Omega_+(t)}$ ,
- $\mathbf{n}_\pm = \mp \frac{\nabla\phi}{|\nabla\phi|}$  extended unit outward normal

### Linear transport equation

$$\frac{\partial\phi}{\partial t} + \mathbf{v} \cdot \nabla\phi = 0 \quad \text{in } \Omega$$

### Conservation laws in evolving domains

$$\begin{aligned}\frac{\partial u}{\partial t} + \nabla \cdot [\mathbf{f}(u) - \kappa \nabla u] &= 0 && \text{in } \Omega_+(t), \\ [\mathbf{f}(u) - \kappa \nabla u] \cdot \mathbf{n}_+ &= g_\Gamma(u, u_\Gamma) && \text{on } \Gamma(t), \\ u &= u_0 && \text{in } \Omega_+(0)\end{aligned}$$

$$u(\mathbf{x}, t) \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^d, d \in \{2, 3\}, t \geq 0,$$

### Conservation laws in evolving domains

$$\begin{aligned}\frac{\partial u}{\partial t} + \nabla \cdot [\mathbf{f}(u) - \kappa \nabla u] &= 0 && \text{in } \Omega_+(t), \\ [\mathbf{f}(u) - \kappa \nabla u] \cdot \mathbf{n}_+ &= g_\Gamma(u, u_\Gamma) && \text{on } \Gamma(t), \\ u &= u_0 && \text{in } \Omega_+(0)\end{aligned}$$

$u(\mathbf{x}, t) \in \mathbb{R}$ ,  $\mathbf{x} \in \mathbb{R}^d$ ,  $d \in \{2, 3\}$ ,  $t \geq 0$ ,

$$g_\Gamma(u, u_\Gamma)(\mathbf{x}) = f_{\mathbf{n}}(u(\mathbf{x}), u_\Gamma(\mathbf{x})) - \kappa \lim_{\varepsilon \rightarrow 0} \frac{u_\Gamma(\mathbf{x}) - u(\mathbf{x} - \varepsilon \mathbf{n}_+(\mathbf{x}))}{\varepsilon}$$

### Dirichlet-type condition

$$s_D(w, u) = \int_{\Omega \setminus \Omega_+(t)} \gamma_\Omega w (u - u_-) dx$$

$\Rightarrow$  weak imposition of  $u = u_-$  in  $\Omega \setminus \overline{\Omega_+(t)}$

### Dirichlet-type condition

$$s_D(w, u) = \int_{\Omega \setminus \Omega_+(t)} \gamma_\Omega w (u - u_-) dx$$

$\Rightarrow$  weak imposition of  $u = u_-$  in  $\Omega \setminus \overline{\Omega_+(t)}$

### Neumann-type condition

$$s_N(w, u) = \int_{\Omega \setminus \Omega_+(t)} \gamma_\Omega \nabla w \cdot (\nabla u - \mathbf{g}_-) dx$$

$\Rightarrow$  harmonic extension into  $\Omega \setminus \overline{\Omega_+(t)}$



Weak form

$$\int_{\Omega_+(t)} w \frac{\partial u}{\partial t} d\mathbf{x} - \int_{\Omega_+(t)} \nabla w \cdot [\mathbf{f}(u_\Gamma) - \kappa \nabla u] d\mathbf{x} - s(w, u) \\ + \int_{\Gamma(t)} w [\mathbf{f}(u_\Gamma) \cdot \mathbf{n}_+ - \kappa \partial_n u] ds = 0$$

## Fictitious domain formulation

---

Weak form

$$\int_{\Omega_+(t)} w \frac{\partial u}{\partial t} d\mathbf{x} - \int_{\Omega_+(t)} \nabla w \cdot [\mathbf{f}(u_\Gamma) - \kappa \nabla u] d\mathbf{x} - s(w, u) \\ + \int_{\Gamma(t)} w [\mathbf{f}(u_\Gamma) \cdot \mathbf{n}_+ - \kappa \partial_n u] ds = 0$$

## Fictitious domain formulation

$$\frac{d}{dt} \int_{\Omega} H(\phi) w u d\mathbf{x} - \int_{\Omega} H(\phi) \nabla w \cdot [\mathbf{f}(u_\Gamma) - \kappa \nabla u] d\mathbf{x} + s(w, u) \\ + \int_{\Gamma(t)} w [\mathbf{f}(u_\Gamma) \cdot \mathbf{n}_+ - v_n u_\Gamma - \kappa \partial_n u] ds = 0, \quad \forall w \in V(\Omega)$$

$$u_\Omega = H(\phi)u + (1 - H(\phi))u_-$$

### Dirichlet-type ghost penalties

$$s_D(w, u) = \int_{\Omega} \gamma_{\Omega} w (u - u_{\Omega}) d\mathbf{x}$$

$$u_\Omega = H(\phi)u + (1 - H(\phi))u_-$$

### Dirichlet-type ghost penalties

$$s_D(w, u) = \int_{\Omega} \gamma_{\Omega} w (u - u_{\Omega}) dx$$

$$\mathbf{g}_{\Omega} = H(\phi)\nabla u + (1 - H(\phi))\mathbf{g}_-$$

### Neumann-type ghost penalties

$$s_N(w, u) = \int_{\Omega} \gamma_{\Omega} \nabla w \cdot (\nabla u - \mathbf{g}_{\Omega}) dx$$

### Dirichlet penalty form

$$\begin{aligned} \frac{d}{dt} \int_{\Omega} H_{\varepsilon}(\phi) w u d\mathbf{x} - \int_{\Omega} H_{\varepsilon}(\phi) \nabla w \cdot [\mathbf{f}(u) - \kappa \nabla u] d\mathbf{x} + \int_{\Omega} \gamma_{\Omega} w (u - u_{\Omega}) d\mathbf{x} \\ + \int_{\Omega} w G(\phi, u, u_{\Gamma}) \delta_{\varepsilon} |\nabla \phi| d\mathbf{x} = 0, \quad \forall w \in V(\Omega) \end{aligned}$$

### Dirichlet penalty form

$$\begin{aligned} \frac{d}{dt} \int_{\Omega} H_{\varepsilon}(\phi) w u d\mathbf{x} - \int_{\Omega} H_{\varepsilon}(\phi) \nabla w \cdot [\mathbf{f}(u) - \kappa \nabla u] d\mathbf{x} + \int_{\Omega} \gamma_{\Omega} w (u - u_{\Omega}) d\mathbf{x} \\ + \int_{\Omega} w G(\phi, u, u_{\Gamma}) \delta_{\varepsilon} |\nabla \phi| d\mathbf{x} = 0, \quad \forall w \in V(\Omega) \end{aligned}$$

### Gaussian regularization

$$H_{\varepsilon}(\phi) = \frac{1}{2} \left( 1 + \operatorname{erf} \left( \frac{\pi \phi}{3\varepsilon} \right) \right), \quad \delta_{\varepsilon} = H'_{\varepsilon}(\phi) = \frac{1}{\varepsilon} \sqrt{\frac{\pi}{9}} \exp \left( \frac{-\pi^2 \phi^2}{9\varepsilon^2} \right)$$

## Extrapolation using level sets

---

To be defined: continuous extensions  $U, \partial_n U, V$  for calculating

$$G(\phi_h, u_h, u_\Gamma) = F - \kappa \partial_n U - VU, \quad \mathbf{v}_h = -V \mathbf{q}_h, \quad u_{\Omega, h}(u_h, U)$$

## Extrapolation using level sets

---

To be defined: continuous extensions  $U, \partial_n U, V$  for calculating

$$G(\phi_h, u_h, u_\Gamma) = F - \kappa \partial_n U - VU, \quad \mathbf{v}_h = -V \mathbf{q}_h, \quad u_{\Omega, h}(u_h, U)$$

Main steps:

- 1 closest-point search
- 2 gradient reconstruction
- 3 extrapolation



## Extrapolation using level sets

---

To be defined: continuous extensions  $U, \partial_n U, V$  for calculating

$$G(\phi_h, u_h, u_\Gamma) = F - \kappa \partial_n U - VU, \quad \mathbf{v}_h = -V \mathbf{q}_h, \quad u_{\Omega, h}(u_h, U)$$

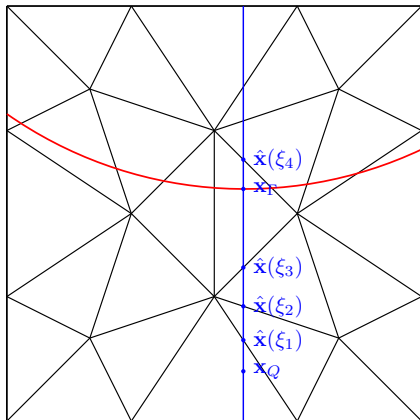
Main steps:

- 1 closest-point search
- 2 gradient reconstruction
- 3 extrapolation

Requirements: simplicity, efficiency, accuracy

Interface pointer

$$\mathbf{n}_Q := -\mathbf{q}_h(\mathbf{x}_Q) \approx \mathbf{n}_+(\mathbf{x}_Q)$$

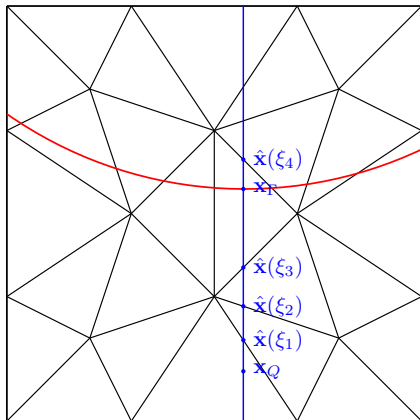


Interface pointer

$$\mathbf{n}_Q := -\mathbf{q}_h(\mathbf{x}_Q) \approx \mathbf{n}_+(\mathbf{x}_Q)$$

Exact distance function  $\phi$

$$\mathbf{x}_\Gamma := \mathbf{x}_Q + \phi(\mathbf{x}_Q)\mathbf{n}_Q$$



Interface pointer

$$\mathbf{n}_Q := -\mathbf{q}_h(\mathbf{x}_Q) \approx \mathbf{n}_+(\mathbf{x}_Q)$$

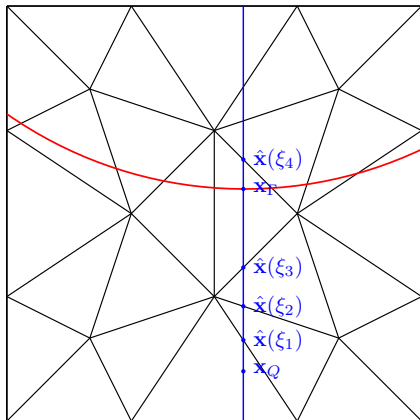
Exact distance function  $\phi$

$$\mathbf{x}_\Gamma := \mathbf{x}_Q + \phi(\mathbf{x}_Q)\mathbf{n}_Q$$

Numerical approximation  $\phi_h$

$$\hat{\mathbf{x}}(\xi) = \mathbf{x}_Q + \xi \text{sign}(\phi_h(\mathbf{x}_Q))\mathbf{n}_Q, \quad \xi \in \mathbb{R}$$

$$\phi_h(\mathbf{x}_\Gamma) = 0 \text{ at } \mathbf{x}_\Gamma = \hat{\mathbf{x}}(\xi_\Gamma)$$



Interface pointer

$$\mathbf{n}_Q := -\mathbf{q}_h(\mathbf{x}_Q) \approx \mathbf{n}_+(\mathbf{x}_Q)$$

Exact distance function  $\phi$

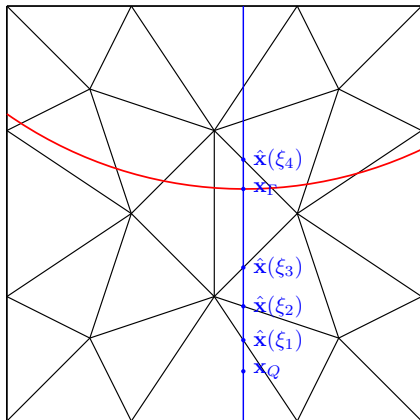
$$\mathbf{x}_\Gamma := \mathbf{x}_Q + \phi(\mathbf{x}_Q)\mathbf{n}_Q$$

Numerical approximation  $\phi_h$

$$\hat{\mathbf{x}}(\xi) = \mathbf{x}_Q + \xi \text{sign}(\phi_h(\mathbf{x}_Q))\mathbf{n}_Q, \quad \xi \in \mathbb{R}$$

$$\phi_h(\mathbf{x}_\Gamma) = 0 \text{ at } \mathbf{x}_\Gamma = \hat{\mathbf{x}}(\xi_\Gamma)$$

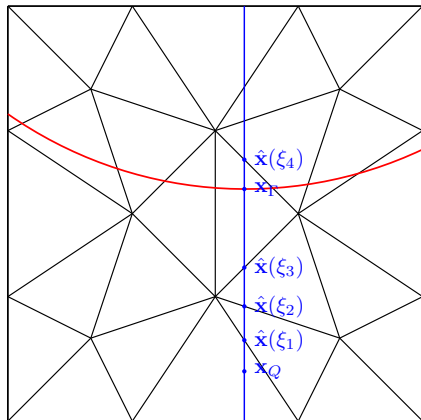
$\Rightarrow$  simple line search



## Closest point search

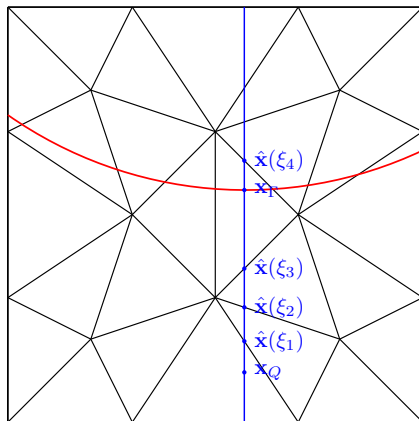
### Search algorithm

- Set  $\xi_0 = 0$



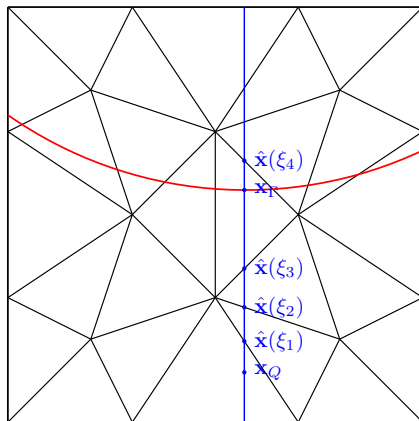
### Search algorithm

- Set  $\xi_0 = 0$
- For  $i > 1$ : Find next intersection  $\hat{x}(\xi_i)$ ,  $\xi_i > \xi_{i-1}$  of  $\hat{x}(\xi)$  with boundary of a mesh cell boundary



### Search algorithm

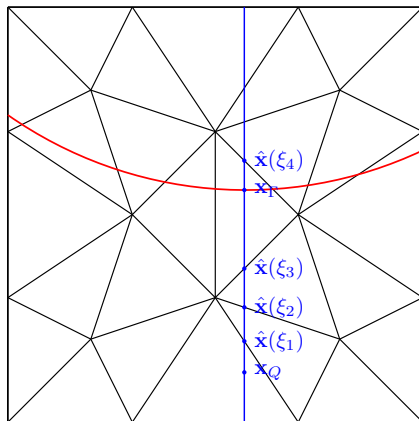
- Set  $\xi_0 = 0$
- For  $i > 1$ : Find next intersection  $\hat{\mathbf{x}}(\xi_i)$ ,  $\xi_i > \xi_{i-1}$  of  $\hat{\mathbf{x}}(\xi)$  with boundary of a mesh cell boundary
- If  $\phi(\hat{\mathbf{x}}(\xi_i))\phi(\hat{\mathbf{x}}(\xi_{i-1})) < 0$  for  $i = m$  exit loop





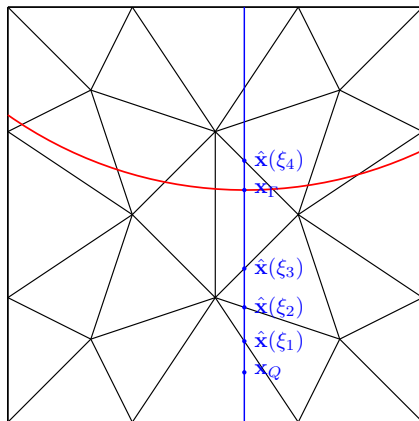
### Search algorithm

- Set  $\xi_0 = 0$
- For  $i > 1$ : Find next intersection  $\hat{\mathbf{x}}(\xi_i)$ ,  $\xi_i > \xi_{i-1}$  of  $\hat{\mathbf{x}}(\xi)$  with boundary of a mesh cell boundary
- If  $\phi(\hat{\mathbf{x}}(\xi_i))\phi(\hat{\mathbf{x}}(\xi_{i-1})) < 0$  for  $i = m$  exit loop
- Solve linear/quadratic equation to find root  $\xi_\Gamma \in [\xi_{m-1}, \xi_m]$  of  $\phi(\hat{\mathbf{x}}(\xi))$



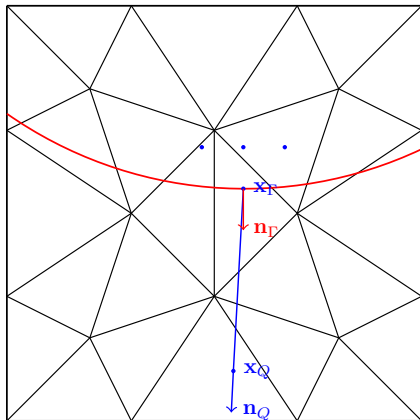
### Search algorithm

- Set  $\xi_0 = 0$
- For  $i > 1$ : Find next intersection  $\hat{\mathbf{x}}(\xi_i)$ ,  $\xi_i > \xi_{i-1}$  of  $\hat{\mathbf{x}}(\xi)$  with boundary of a mesh cell boundary
- If  $\phi(\hat{\mathbf{x}}(\xi_i))\phi(\hat{\mathbf{x}}(\xi_{i-1})) < 0$  for  $i = m$  exit loop
- Solve linear/quadratic equation to find root  $\xi_\Gamma \in [\xi_{m-1}, \xi_m]$  of  $\phi(\hat{\mathbf{x}}(\xi))$
- Set  $\mathbf{x}_\Gamma = \hat{\mathbf{x}}(\xi_\Gamma)$



Unit outward normal vector

$$\mathbf{n}_\Gamma = -\frac{\mathbf{q}_h(\mathbf{x}_\Gamma)}{|\mathbf{q}_h(\mathbf{x}_\Gamma)|} \approx \mathbf{n}_+(\mathbf{x}_\Gamma)$$



## Gradient reconstruction

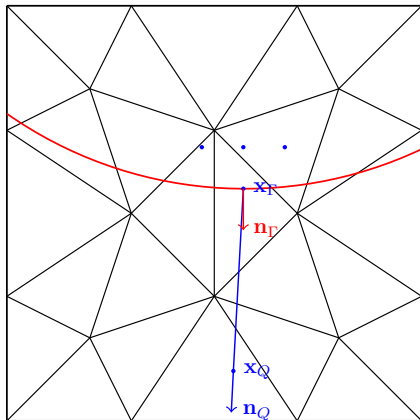
Unit outward normal vector

$$\mathbf{n}_\Gamma = -\frac{\mathbf{q}_h(\mathbf{x}_\Gamma)}{|\mathbf{q}_h(\mathbf{x}_\Gamma)|} \approx \mathbf{n}_+(\mathbf{x}_\Gamma)$$

Three point interpolation stencil:

$$\mathcal{S}_{2D}(\mathbf{x}_\Gamma) = \{\mathbf{x}_P - 0.5\varepsilon\boldsymbol{\tau}_\Gamma, \mathbf{x}_P, \mathbf{x}_P + 0.5\varepsilon\boldsymbol{\tau}_\Gamma\}$$

$$\mathbf{x}_P = \mathbf{x}_\Gamma - \varepsilon\mathbf{n}_\Gamma, \quad \boldsymbol{\tau}_\Gamma \perp \mathbf{n}_\Gamma$$



## Gradient reconstruction

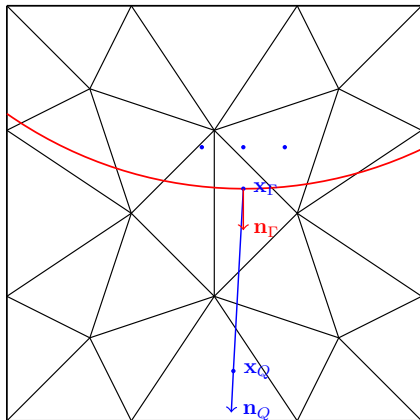
Unit outward normal vector

$$\mathbf{n}_\Gamma = -\frac{\mathbf{q}_h(\mathbf{x}_\Gamma)}{|\mathbf{q}_h(\mathbf{x}_\Gamma)|} \approx \mathbf{n}_+(\mathbf{x}_\Gamma)$$

Three point interpolation stencil:

$$\mathcal{S}_{2D}(\mathbf{x}_\Gamma) = \{\mathbf{x}_P - 0.5\varepsilon\boldsymbol{\tau}_\Gamma, \mathbf{x}_P, \mathbf{x}_P + 0.5\varepsilon\boldsymbol{\tau}_\Gamma\}$$

$$\mathbf{x}_P = \mathbf{x}_\Gamma - \varepsilon\mathbf{n}_\Gamma, \quad \boldsymbol{\tau}_\Gamma \perp \mathbf{n}_\Gamma$$



## Gradient reconstruction

Unit outward normal vector

$$\mathbf{n}_\Gamma = -\frac{\mathbf{q}_h(\mathbf{x}_\Gamma)}{|\mathbf{q}_h(\mathbf{x}_\Gamma)|} \approx \mathbf{n}_+(\mathbf{x}_\Gamma)$$

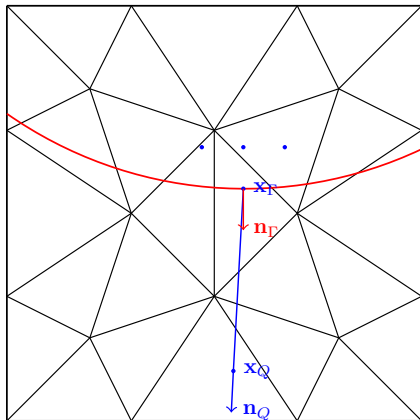
Three point interpolation stencil:

$$\mathcal{S}_{2D}(\mathbf{x}_\Gamma) = \{\mathbf{x}_P - 0.5\varepsilon\boldsymbol{\tau}_\Gamma, \mathbf{x}_P, \mathbf{x}_P + 0.5\varepsilon\boldsymbol{\tau}_\Gamma\}$$

$$\mathbf{x}_P = \mathbf{x}_\Gamma - \varepsilon\mathbf{n}_\Gamma, \quad \boldsymbol{\tau}_\Gamma \perp \mathbf{n}_\Gamma$$

Approximate normal derivative

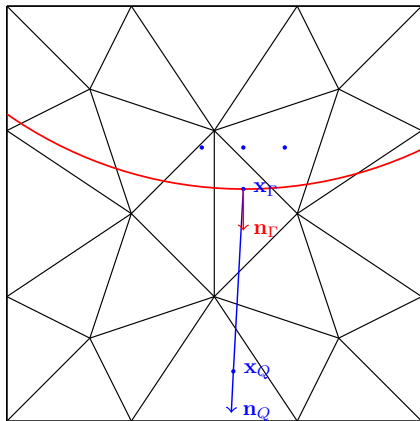
$$\partial_n U(\mathbf{x}_\Gamma) = \frac{u_\Gamma(\mathbf{x}_\Gamma) - u_h(\mathbf{x}_\Gamma - \varepsilon\mathbf{n}_\Gamma)}{\varepsilon}$$



For  $\mathbb{P}_1$  or  $\mathbb{Q}_1$  elements:

Normal derivatives: constant extrapolation

$$\partial_n U(\mathbf{x}_Q) = \partial_n U(\mathbf{x}_\Gamma)$$



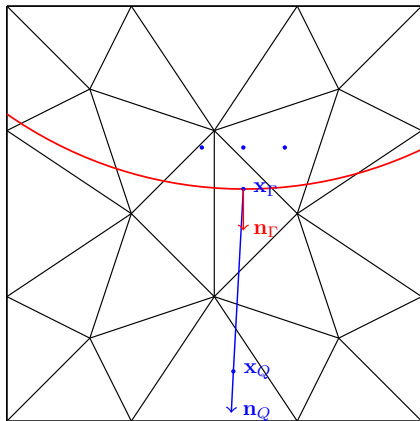
For  $\mathbb{P}_1$  or  $\mathbb{Q}_1$  elements:

Normal derivatives: constant extrapolation

$$\partial_n U(\mathbf{x}_Q) = \partial_n U(\mathbf{x}_\Gamma)$$

Solution values: linear extrapolation

$$U(\mathbf{x}_Q) = u_\Gamma(\mathbf{x}_\Gamma) + \nabla U(\mathbf{x}_\Gamma) \cdot (\mathbf{x}_Q - \mathbf{x}_\Gamma)$$





Moving embedded boundary

$$\Gamma(t) = \{(x, y) \in \mathbb{R}^2 : (x - 0.5)^2 + (y - 0.5)^2 = (0.25 + 0.15t)^2\} = \partial\Omega_+(t)$$

Parabolic test

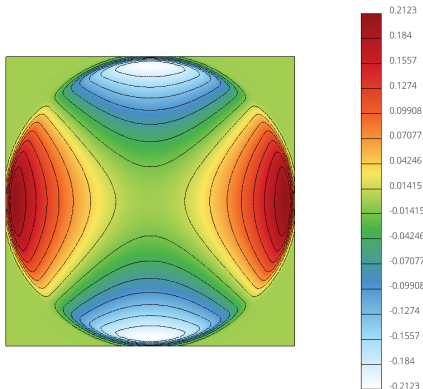
$$\begin{aligned} \frac{\partial u}{\partial t} - \Delta u &= 0 && \text{in } \Omega_+(t), \\ u &= u_\Gamma && \text{on } \Gamma(t), \\ u &= u_0 && \text{in } \Omega_+(0) \end{aligned}$$

$$u_\Gamma(x, y, t) = u_0(x, y) = u_{ex}(x, y, t) = (x - 0.5)^2 - (y - 0.5)^2$$

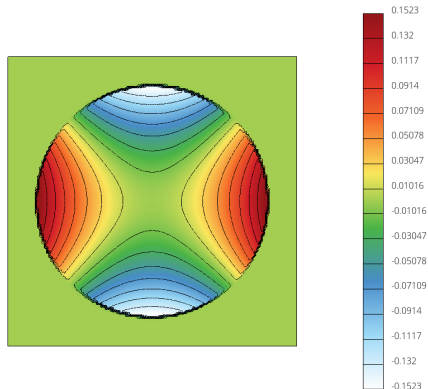
- Dirichlet ghost penalties with  $\gamma_{\Omega,D} = h^{-1}$
- Crank-Nicolson scheme

$h^{-1}$	$(\Delta t)^{-1}$	full DGP	EOC	damped DGP	EOC
16	5	6.30e-03		6.30e-03	
32	10	1.16e-03	2.44	1.16e-03	2.44
64	20	2.86e-04	2.02	2.86e-04	2.02
128	40	6.95e-05	2.04	6.95e-05	2.04
256	80	1.67e-05	2.06	1.67e-05	2.06
512	160	4.24e-06	1.98	4.24e-06	1.98
1024	320	1.10e-06	1.95	1.10e-06	1.95

(a)  $u_h$ , damped DGP



(b)  $H_\epsilon(\phi)u_h$ , damped DGP



### Summary




- approximation of surface integrals by volume integrals
- fast closest-point search algorithm
- new way to define and calculate compact-support extensions
- narrow-band integration of terms involving extended fluxes, ghost penalties and velocity fields

## Summary

- approximation of surface integrals by volume integrals
- fast closest-point search algorithm
- new way to define and calculate compact-support extensions
- narrow-band integration of terms involving extended fluxes, ghost penalties and velocity fields

## Outlook

- theoretical studies of proposed approach
- new approximate delta functions with compact support
- application to interface problems and PDE systems

-  D. Kuzmin and J.-P. Bäcker, An unfitted finite element method using level set functions for extrapolation into deformable diffuse interfaces. *J. Comput. Phys.* **461** (2022) 111218.
-  S. Zahedi and A.K. Tornberg, Delta function approximations in level set methods by distance function extension. *J. Comput. Phys.* **229** (2010) 2199–2219.
-  K. Li, N. M. Atallah, A. Main, and G. Scovazzi, The Shifted Interface Method: A flexible approach to embedded interface computations. *Int. J. Numer. Meth. Engrg.* **121** (2020) 492–518.

**Thank you for your attention!**

jan-philip.baecker@tu-dortmund.de