A Fixed Mesh Method with Immersed Finite Elements for Solving Interface Inverse Problems

Ruchi Guo and Tao Lin

Department of Mathematics
Virginia Tech

Yanping Lin

Department of Applied Mathematics
The Hong Kong Polytechnic University

Numerical Analysis of
Coupled and Multi-Physics Problems with Dynamic Interfaces
Oaxaca, Mexico

July 29 - August 02, 2018

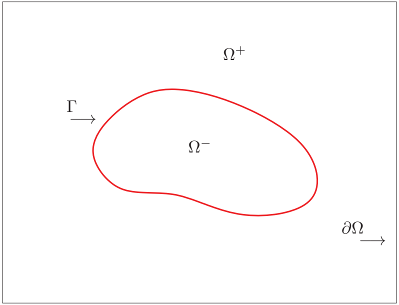
Outline

- Immersed Finite Element (IFE) Methods for Interface Forward Problems
- IFE methods for Interface Inverse Problems
- Numerical Examples
- Conclusions and Future Work

Interface Forward Problems

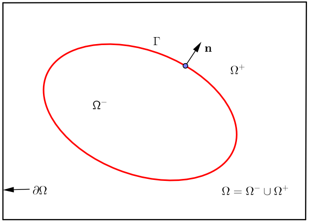
We consider mathematical models of physical phenomenons in a domain that consists of

- multiple materials
- same material of multiple states



This leads to partial differential equations (PDEs) whose coefficients are discontinuous so that the solutions to these PDEs have less desirable global regularity.

Interface Forward Problems



The second order elliptic interface forward problem: find $u(X)$ such that

$$-\nabla \cdot (\beta \nabla u) = f, \quad \text{in} \quad \Omega^- \cup \Omega^+$$
$$u = g, \quad \text{on} \quad \partial\Omega$$

with jump conditions across the interface Γ:

$$[u]_\Gamma = 0, \qquad \left[\beta \frac{\partial u}{\partial n}\right]_\Gamma = 0$$

The coefficient function:

$$\beta(X) = \begin{cases} \beta^+, & \text{in } \Omega^+ \\ \beta^-, & \text{in } \Omega^- \end{cases}$$

The regularity of $u(X)$:

$$u|_{\Omega^s)} \in H^2(\Omega^s), \quad s = \pm$$
$$\text{But} \qquad u \in H^1(\Omega) \backslash H^2(\Omega)$$

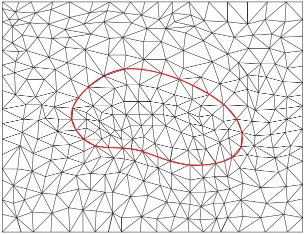
Traditional finite elements:

- The basis functions are independent of the problem.
- The mesh has to be formed according to the problem.

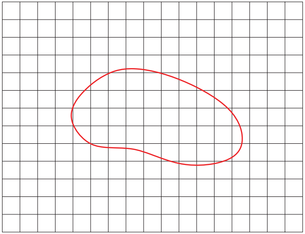
Immersed finite elements (flipping the ideas above):

- The basis functions are constructed according to the interface problem.
- The mesh can be independent of the problem.

Mesh for traditional FE methods

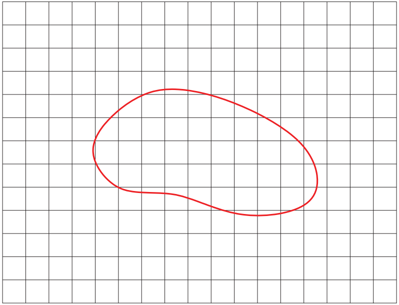


Usable mesh for IFE methods



What is an immersed finite element (IFE) method?

- Of course, it is a finite element (FE) method.
- It can use an interface-independent mesh.
- It uses standard FE functions on all the non-interface elements.
- It uses IFE functions on all interface elements developed according to jump conditions for the interface problem to be solved.
- In a fine mesh, there are far fewer interface elements than non-interface element.



A few advantages of IFE methods:

(Body-Fitting Mesh)

- Mesh: Remeshing.
- Global DOFs: Changing.
- Local Assemble: Difficult.
- Method of Lines: Inapplicable.

(Non-Body-Fitting Mesh)

- Mesh: Fixed.
- Global DOFs: Same.
- Local Assemble: Standard.
- Method of Lines: Applicable.

Finite Element Methods based on Cartesian Meshes

Modifying Formulations for Methods

- penalty finite element method Babuška 1970
- unfitted finite element method Hansbo^2, Olshanskiy, Badia, Massing, ...
- Edge-based correction FE interface (EBC-FEI) method, FE method with flux edge stabilization Guzmán, Sánchez-Uribe, and Sarkis

Modifying Finite Element Functions

- general finite element method *Babuška and Osborn 1983*
- multi-grid finite element method *Cai, Hou and Wu*
- extended finite element method *Moës, Dolbow and Belytschko*
- Modified P_1 finite element methods *Guzmán, Sánchez-Uribe, and Sarkis*

- A 1D Immersed Finite Element (IFE) Method *Zhilin Li 1998*
- IFE and related methods: S. Adjerid, R. Guo, X. He, S. Hou, D.Y. Kwak, Z. Li, T. Lin, Y. Lin, T. Papadopoulo, S.A. Sauter, D. Sheen, S. Vallaghé, R. Warnke, Z. Zhang, and many others.

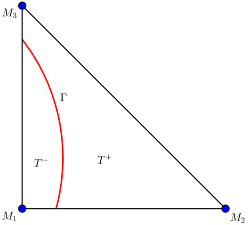
On a non-interface element T: we use standard FEs: (T, Π_T, Σ_T) with

$$\begin{aligned} \Pi_T &= span\{\psi_i, 1 \le i \le d\} \subset \mathcal{P} \\ \Sigma_T &= \{\omega_i, 1 \le i \le d\} \end{aligned}$$

What can we do on an interface element T? We propose:

IFEs: $(T, \tilde{\Pi}_T, \Sigma_T)$

$\tilde{\Pi}_T = span\{\phi_i, 1 \le i \le d\}$



- $\phi_i, 1 \le i \le d$ "satisfy" the interface jump condition
- $\phi_i, 1 \le i \le d$ cannot be polynomials, but they can be piecewise/macro polynomials

$$\phi_i(X) = \begin{cases} \phi_i^-(X) \in \mathcal{P}, & X \in T^- \\ \phi_i^+(X) \in \mathcal{P}, & X \in T^+ \end{cases} \qquad \omega_i(\phi_j) = \delta_{ij}, 1 \le i, j \le d$$

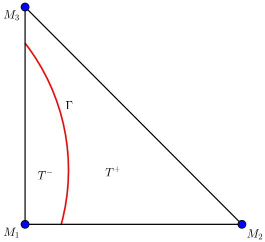
On a non-interface element T: we use standard FE: $(T, \Pi_T, \mathbf{\Sigma_T})$ with

$$\begin{aligned} \Pi_T &= span\{\psi_i, 1 \le i \le d\} \subset \mathcal{P} \\ \Sigma_T &= \{\omega_i, 1 \le i \le d\} \end{aligned}$$

What can we do on an interface element T?

IFE: $(T, \tilde{\Pi}_T, \mathbf{\Sigma_T})$

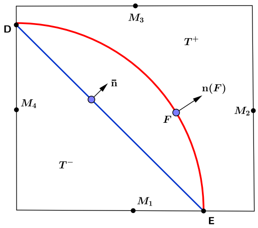
$\tilde{\Pi}_T = span\{\phi_i, 1 \le i \le d\}$



- $\phi_i, 1 \le i \le d$ "satisfy" the interface jump condition
- $\phi_i, 1 \le i \le d$ cannot be polynomials, but they can be piecewise/macro polynomials

$$\phi_i(X) = \begin{cases} \phi_i^-(X) \in \mathcal{P}, & X \in T^- \\ \phi_i^+(X) \in \mathcal{P}, & X \in T^+ \end{cases} \qquad \omega_i(\phi_j) = \delta_{ij}, 1 \le i, j \le d$$

Construction of IFE shape functions



$$\phi(X) = \begin{cases} \phi^-(X) \in \mathcal{P}, & X \in T^- \\ \phi^+(X) \in \mathcal{P}, & X \in T^+ \end{cases}$$

Continuity jump condition: $[u]_\Gamma = 0$

Approximate Continuity jump condition:

$$\phi^-|_L = \phi^+|_L \quad \text{and more if needed}$$

A partition of the index set: $\mathcal{I} = \{1, 2, \cdots, d\} = \mathcal{I}^- \cup \mathcal{I}^+$, with $|\mathcal{I}^-| \leq |\mathcal{I}^+|$

$$\phi(X) = \begin{cases} \phi^-(X) = \phi^+(X) + c_0 L(X) & \text{if } X \in T^- \\ \phi^+(X) = \sum_{j \in \mathcal{I}^-} c_j \psi_j(X) + \sum_{j \in \mathcal{I}^+} v_j \psi_j(X) & \text{if } X \in T^+ \end{cases}$$

$$L(X) = \bar{\mathbf{n}} \cdot (X - D), \qquad \text{we have: } \omega_i(\phi) = v_i, \ i \in \mathcal{I}^+$$

The flux jump condition: $\left[\beta \frac{\partial u}{\partial n}\right]_\Gamma = 0$

Approximate flux jump condition:

$$\beta^- \nabla \phi_T^-(F) \cdot \mathbf{n}(F) = \beta^+ \nabla \phi_T^+(F) \cdot \mathbf{n}(F)$$

Degrees of freedom on $\mathcal{I}^-$: $\omega_i(\phi) = v_i, \ i \in \mathcal{I}^-$

$$\text{Then} \quad (I + k\,\boldsymbol{\delta}\boldsymbol{\gamma}^T)\mathbf{c} = \mathbf{b}, \quad \text{where} \quad \mathbf{c} = (c_i)_{i\in\mathcal{I}^-}$$

$$\text{and} \quad c_0 = k\left(\sum_{i\in\mathcal{I}^-} c_i \nabla\psi_i(F)\cdot\mathbf{n}(F) + \sum_{i\in\mathcal{I}^+} v_i \nabla\psi_i(F)\cdot\mathbf{n}(F)\right)$$

Theorem. (Guo and Lin, IMA J. Numer. Anal, 2017) IFE function ϕ satisfying the approximate jump conditions is uniquely determined by degrees of freedom $\omega_i(\phi) = v_i, i \in \mathcal{I}$.

$$\text{Let} \quad \mathbf{v}_i = (v_j)_1^d, \quad v_j = \begin{cases} 1, & \text{if } j = i \\ 0, & \text{otherwise} \end{cases} \quad i = 1, 2, \cdots, d \quad \longrightarrow \quad \text{IFE} \quad \phi_i(X; \Gamma)$$

Local IFE space on $T \in \mathcal{T}_h$:

$$S_h(T) = \begin{cases} span\{\psi_i(X), 1 \le i \le d\}, & \text{when } T \text{ is a non-interface element} \\ span\{\phi_i(X;\Gamma), 1 \le i \le d\}, & \text{when } T \text{ is an interface element} \end{cases}$$

Remark: FE shape functions $\psi_i(X)$ are independent of the interface, but FE shape functions $\phi_i(X;\Gamma)$ intrinsically depend on the interface.

Properties of IFE basis functions:

Bounds of IFE basis functions: Then there exits a constant C, independent of interface location, such that

$$|\phi_{i,T}|_{k,\infty,T} \leqslant Ch^{-k}, \quad i \in \mathcal{I}, k = 0, 1, \ \forall T \in \mathcal{T}_h^i.$$

Partition of Unity: On every interface element T, we have

$$\sum_{i\in\mathcal{I}} \phi_{i,T}(X) \equiv 1, \quad \nabla \left(\sum_{i\in\mathcal{I}} \phi_{i,T}(X) \right) = \sum_{i\in\mathcal{I}} \nabla\phi_{i,T}(X) = 0$$

Two Fundamental Identities: For every interface element T, the following identities hold:

$$\sum_{i\in\mathcal{I}} (M_i - X)\phi_{i,T}^-(X) + \sum_{i\in\mathcal{I}^+} (\overline{M}^-(F) - I)^T (M_i - \overline{X}_i)\phi_{i,T}^-(X) = \mathbf{0}, \ \forall X \in T^-$$
$$\sum_{i\in\mathcal{I}} (M_i - X)\phi_{i,T}^+(X) + \sum_{i\in\mathcal{I}^-} (\overline{M}^+(F) - I)^T (M_i - \overline{X}_i)\phi_{i,T}^+(X) = \mathbf{0}, \ \forall X \in T^+$$

Consistence: $\phi_{i,T}(X) \longrightarrow \psi_{i,T}(X)$

Approximation Capability. There exits a constant $C > 0$ independent of the interface location such that on every interface element T the following holds

$$\|I_{h,T}u - u\|_{0,T^-} + h|I_{h,T}u - u|_{1,T^-} \leqslant Ch^2\|u\|_{2,T}$$

Interface forward and inverse problems

The interface forward problem: given $\Gamma, \beta, f, g_D, g_N$, find u such that

$$\begin{aligned} \text{PDE:} \quad & -\nabla \cdot (\beta \nabla u) = f, \quad \text{in } \Omega^- \cup \Omega^+ \\ \text{Boundary Condition:} \quad & u = g_D, \quad \text{on } \partial\Omega_D \subseteq \partial\Omega, \quad \frac{\partial u}{\partial \mathbf{n}} = g_N, \quad \text{on } \partial\Omega_N \subseteq \partial\Omega \\ \text{Jump Condition:} \quad & [u]|_\Gamma = 0, \quad [\beta \nabla u \cdot \mathbf{n}]|_\Gamma = 0 \end{aligned}$$

$$\Gamma, \beta, f, g_D, g_N \quad \longrightarrow \quad u$$

The interface inverse problem: given β, f, g_D, g_N and some data/measurements about u, find the curve Γ

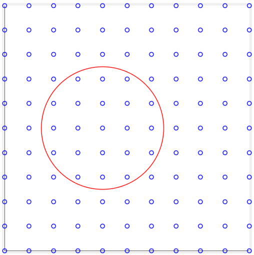
$$\beta, f, g_D, g_N \text{ and some data/measurements about } u \quad \longrightarrow \quad \Gamma$$

A Shape Optimization Method for Interface Inverse Problems

Find Γ^* such that

$$\Gamma^* = argmin\ \mathcal{J}(\Gamma), \quad \mathcal{J}(\Gamma) = \int_{\Omega_0} J(\Gamma, u(\Gamma), X(\Gamma))dX$$

- $u(\Gamma)$ solves the interface forward problem corresponding to the interface Γ
- Integrand $J(\Gamma, u(\Gamma), X(\Gamma))$ is problem dependent according to the available data.
- the spacial variable X depends on the interface Γ because Ω^-, Ω^+ changes w.r.t. Γ



IFE Method for the Interface Inverse Problems

The interface forward problem: find u such that

$$\begin{aligned}
\text{PDE:} \quad & -\nabla\cdot(\beta\nabla u) = f, \quad \text{in } \Omega^- \cup \Omega^+ \\
\text{Boundary Condition:} \quad & u = g_D, \quad \text{on } \partial\Omega_D \subseteq \partial\Omega, \quad \frac{\partial u}{\partial \mathbf{n}} = g_N, \quad \text{on } \partial\Omega_N \subseteq \partial\Omega \\
\text{Jump Condition:} \quad & [u]|_\Gamma = 0, \quad [\beta\nabla u\cdot\mathbf{n}]|_\Gamma = 0 \\
& \Downarrow \\
\text{Operator Format:} \quad & \mathcal{A}(\Gamma)u(X(\Gamma);\Gamma) = F
\end{aligned}$$

Three Discretizations:

1. Discretization of the interface: Choose a parametrization of the interface curve with the control points $\boldsymbol{\alpha} = (\alpha_j)_{j\in\mathcal{D}}$, i.e., $\Gamma \approx \Gamma(\boldsymbol{\alpha})$. In the computations for shape optimization, we deal with a parameterized interface forward problem:

$$\begin{aligned}
\mathcal{A}(\Gamma)u(X(\Gamma);\Gamma) = F \quad & \Rightarrow \quad \mathcal{A}(\Gamma(\boldsymbol{\alpha}))u(X(\Gamma(\boldsymbol{\alpha}));\Gamma(\boldsymbol{\alpha})) = F \\
& \Rightarrow \quad \mathcal{A}(\boldsymbol{\alpha})u(X(\boldsymbol{\alpha});\boldsymbol{\alpha}) = F
\end{aligned}$$

$$u(\Gamma) = u(X(\Gamma);\Gamma) \quad \Rightarrow \quad u(X(\boldsymbol{\alpha});\boldsymbol{\alpha}) = u(\boldsymbol{\alpha})$$

2. IFE Discretization for the parameterized interface forward problem:

IFE solution: $$u_h(X(\boldsymbol{\alpha});\boldsymbol{\alpha}) = \sum_{i=1}^{|\mathring{\mathcal{N}}_h|} u_i(\boldsymbol{\alpha})\phi_i(X(\boldsymbol{\alpha}),\boldsymbol{\alpha}) + \sum_{i=|\mathring{\mathcal{N}}_h|+1}^{|\mathcal{N}_h|} g_D(X_i)\phi_i(X(\boldsymbol{\alpha}),\boldsymbol{\alpha})$$

Then $$\mathcal{A}(\boldsymbol{\alpha})u(X(\boldsymbol{\alpha});\boldsymbol{\alpha}) = F \quad \Rightarrow \quad \mathbf{A}(\boldsymbol{\alpha})\mathbf{u}_h(\boldsymbol{\alpha}) = \mathbf{F}(\boldsymbol{\alpha})$$

where

$$\mathbf{u}_h(\boldsymbol{\alpha}) = \left(u_i(\boldsymbol{\alpha})\right)_{i=1}^{|\mathring{\mathcal{N}}_h|}$$

Optimal IFE discretization (Lin, Lin and Zhang, SIAM J. Numer. Anal, 2015):

$$\|u(\cdot(\boldsymbol{\alpha});\boldsymbol{\alpha}) - u_h(\cdot(\boldsymbol{\alpha});\boldsymbol{\alpha})\|_0 + h\,|u(\cdot(\boldsymbol{\alpha});\boldsymbol{\alpha}) - u_h(\cdot(\boldsymbol{\alpha});\boldsymbol{\alpha})|_1 \le Ch^2$$

Hence, $\mathbf{A}(\boldsymbol{\alpha})\mathbf{u}_h(\boldsymbol{\alpha}) = \mathbf{F}(\boldsymbol{\alpha})$ is an accurate descretization of $\mathcal{A}(\boldsymbol{\alpha})u(X(\boldsymbol{\alpha});\boldsymbol{\alpha}) = F$

3. IFE discretization of the objective functional:

$$\begin{aligned}\text{functional} \quad \mathcal{J}(\Gamma) &= \int_{\Omega_0} J(\Gamma, u(\Gamma), X(\Gamma))dX \\ &\approx \int_{\Omega_0} J(\Gamma, u_h(\Gamma), X(\Gamma))dX \\ &= \int_{\Omega_0} J_h(\boldsymbol{\alpha}, \mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha}))dX = \mathcal{J}_h(\boldsymbol{\alpha}) \quad \text{function}\end{aligned}$$

For many interface inverse problems, this IFE discretization of the objective functional is also optimal:

$$|\mathcal{J}(\Gamma) - \mathcal{J}_h(\boldsymbol{\alpha})| \leq Ch^2$$

A Critical Feature: The discretizations for the interface forward problem and the objective functional are optimal regardless of the location of the parameterized interface $\Gamma(\boldsymbol{\alpha})$ in a fixed mesh.

Shape Optimization Problem: Find Γ^* such that

$$\Gamma^* = argmin\ \mathcal{J}(\Gamma), \quad \mathcal{J}(\Gamma) = \int_{\Omega_0} J(\Gamma, u(\Gamma), X(\Gamma))dX$$

subject to $\quad \mathcal{A}(\Gamma)u(X(\Gamma);\Gamma) = F$

$\Downarrow$ on a fixed mesh

Constrained Optimization Problem: Find $\boldsymbol{\alpha}^*$ such that

$$\boldsymbol{\alpha}^* = argmin\ \mathcal{J}_h(\boldsymbol{\alpha}), \quad \mathcal{J}_h(\boldsymbol{\alpha}) = \int_{\Omega_0} J_h(\boldsymbol{\alpha}, \mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha}))dX$$

subject to $\quad \mathbf{A}(X(\boldsymbol{\alpha}),\boldsymbol{\alpha})\mathbf{u}_h(\boldsymbol{\alpha}) - \mathbf{F}(X(\boldsymbol{\alpha}),\boldsymbol{\alpha}) = 0$

Gradient for the constrained optimization problem:

Material derivative formula for the objective function:

$$
\begin{aligned}
D_{\alpha_j}\mathcal{J}_h(\boldsymbol{\alpha}) &= \int_{\Omega_0} D_{\alpha_j} J_h dX + \int_{\Omega_0} J_h \,\mathrm{div}\left(\mathbf{V}^j\right) dX \\
&= \frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h + \int_{\Omega_0} \frac{\partial J_h}{\partial \alpha_j} dX + \int_{\Omega_0} \left(\frac{\partial J_h}{\partial X}\right)^T \mathbf{V}^j dX + \int_{\Omega_0} J_h \,\mathrm{div}\left(\mathbf{V}^j\right) dX
\end{aligned}
$$

Haslinger and Mäkinen(2003) (Introduction to Shape Optimization) and others

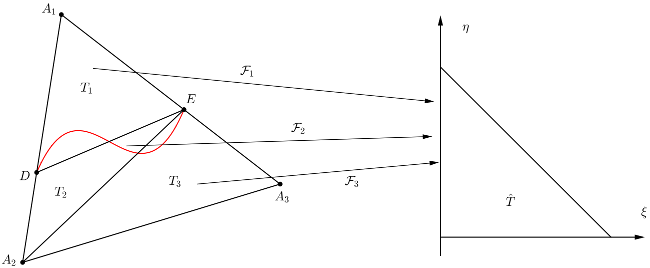
Three important ingredients:

- The Velocity Fields: $\mathbf{V}^j$
- The Shape derivative of the IFE functions
- $\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h$.

1. Velocity Fields

$$\mathbf{V}^j(X) = \frac{\partial X(\boldsymbol{\alpha})}{\partial \alpha_j} = \begin{cases} \mathbf{V}^j_T(X) = \mathbf{0}, & \text{on non-interface elements} \\ \mathbf{V}^j_T(X) = (D_{\alpha_j}\mathbf{J}_i)\mathbf{J}_i^{-1}(X - A_i), & \text{on interface elements} \end{cases}$$

$\mathbf{J}_k$ is the Jacobian of the affine mapping $\mathcal{F}_k, k = 1, 2, 3$



Haslinger and Mäkinen(2003) (Introduction to Shape Optimization) and others

- $\mathbf{V}^j(X)$ vanishes on all non-interface elements $\rightarrow$ efficiency
- $\mathbf{V}^j(X)$ has explicit formula in terms of the design variables $\boldsymbol{\alpha}$ $\rightarrow$ accuracy
- $\mathbf{V}^j(X)$ is an H^1 function on the whole domain Ω

2. Shape Derivatives of IFE Shape Functions

On every non-interface elements, the IFE functions are standard FE functions; hence, they are independent of the interface such that their derivatives w.r.t. to α_j is zero

On each interface elements, IFE functions intrinsically depends on $\boldsymbol{\alpha}$:

$$\phi(X) = \begin{cases} \phi^-(X) = \phi^+(X) + c_0(\boldsymbol{\alpha})L(X, \boldsymbol{\alpha}) & \text{if } X \in T^- \\ \phi^+(X) = \sum_{i \in \mathcal{I}^-} c_i(\boldsymbol{\alpha})\psi_i(X) + \sum_{i \in \mathcal{I}^+} v_i \psi_j(X) & \text{if } X \in T^+ \end{cases}$$

we have

$$\frac{\partial \phi}{\partial \alpha_j} = \begin{cases} \frac{\partial \phi^-(X)}{\partial \alpha_j} = \frac{\partial \phi^+(X)}{\partial \alpha_j} + \frac{\partial c_0}{\partial \alpha_i} L + c_0 \frac{\partial L}{\partial \alpha_i} & \text{if } X \in T^- \\ \frac{\partial \phi^+(X)}{\partial \alpha_j} = \sum_{i \in \mathcal{I}^-} \frac{\partial c_i}{\partial \alpha_j} \psi_i(X) & \text{if } X \in T^+ \end{cases}$$

We have formulas of c_i, c_0 and L in terms of interface-mesh points D and E which allow us to calculate their shape derivatives straightforwardly such as

$$\frac{\partial c_i}{\partial \alpha_j} = \frac{\partial c_i}{\partial D} \frac{\partial D}{\partial \alpha_j} + \frac{\partial c_i}{\partial E} \frac{\partial E}{\partial \alpha_j}$$

and $\frac{\partial D}{\partial \alpha_j}$, $\frac{\partial E}{\partial \alpha_j}$ can be computed elementarily.

3. The Discretized Adjoint Method for $\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h$:

From $\mathbf{A}(\boldsymbol{\alpha})\mathbf{u}_h(\boldsymbol{\alpha}) = \mathbf{F}(\boldsymbol{\alpha})$

$$D_{\alpha_j}\mathbf{A}\ \mathbf{u}_h + \mathbf{A}\ D_{\alpha_j}\mathbf{u}_h = D_{\alpha_j}\mathbf{F}$$

$$\Rightarrow \quad D_{\alpha_j}\mathbf{u}_h = \mathbf{A}^{-1}(D_{\alpha_j}\mathbf{F} - D_{\alpha_j}\mathbf{A}\ \mathbf{u}_h), \quad j = 1, 2, \cdots, |\boldsymbol{\alpha}|$$

$$\Rightarrow \quad \frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h = \left(\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h}\right)^T \mathbf{A}^{-1}(D_{\alpha_j}\mathbf{F} - D_{\alpha_j}\mathbf{A}\ \mathbf{u}_h)$$

- Compute $\mathbf{Y}^T = \left(\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h}\right)^T \mathbf{A}^{-1}$ by solving for $\mathbf{Y}$ from $\mathbf{A}^T\mathbf{Y} = \frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h}$ so that

$$\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h = \mathbf{Y}^T(D_{\alpha_j}\mathbf{F} - D_{\alpha_j}\mathbf{A}\ \mathbf{u}_h), \quad j = 1, 2, \cdots, |\boldsymbol{\alpha}|$$

- $D_{\alpha_j}\mathbf{F}$ and $D_{\alpha_j}\mathbf{A}$ are computed efficiently by the standard local assembling procedure only over interface elements
- $D_{\alpha_j}\mathbf{F}$ and $D_{\alpha_j}\mathbf{A}$ can be computed accurately by formulas based on shape derivatives of the IFE basis functions and the velocity fields

In the IFE system $\mathbf{A}(\boldsymbol{\alpha})\mathbf{u}_h(\boldsymbol{\alpha}) = \mathbf{F}(\boldsymbol{\alpha})$, the matrix $\mathbf{A}(\boldsymbol{\alpha})$ is assembled from local matrices such as

$$\mathbf{K}_T = \begin{cases} \left(\int_T \beta \nabla \psi_{p,T} \cdot \nabla \psi_{q,T} dX\right)_{p,q \in \mathcal{I}}, & T \text{ is non-interface element} \\ \left(\int_T \beta \nabla \phi_{p,T} \cdot \nabla \phi_{q,T} dX\right)_{p,q \in \mathcal{I}}, & T \text{ is interface element} \end{cases}$$

Then, $D_{\alpha_j}\mathbf{A}$ is assembled from corresponding local matrices such as

$$D_{\alpha_j}\mathbf{K}_T = \mathbf{0}, \quad T \text{ is non-interface element}$$

but when T is an interface element:

$$\begin{aligned} D_{\alpha_j}\mathbf{K}_T &= \left(\int_T \beta \nabla \frac{\partial \phi_{p,T}}{\partial \alpha_j} \cdot \nabla \phi_{q,T} dX\right)_{p,q \in \mathcal{I}} + \left(\int_T \beta \nabla \frac{\partial \phi_{p,T}}{\partial \alpha_j} \cdot \nabla \phi_{q,T} dX\right)^T_{p,q \in \mathcal{I}} \\ &+ \left(\sum_{i=1}^{3} \int_{T_i} \beta \nabla \phi_{p,T} \cdot \nabla \phi_{q,T} \, dX \; tr\left((D_{\alpha_j}\mathbf{J}_i)\mathbf{J}_i^{-1}\right)\right)_{p,q \in \mathcal{I}} \end{aligned}$$

Haslinger and Mäkinen(2003) (Introduction to Shape Optimization) and others

Recall Material derivative formula for the objective function:

$$D_{\alpha_j}\mathcal{J}_h(\boldsymbol{\alpha}) = \frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h + \int_{\Omega_0} \frac{\partial J_h}{\partial \alpha_j} dX + \int_{\Omega_0} \left(\frac{\partial J_h}{\partial X}\right)^T \mathbf{V}^j dX + \int_{\Omega_0} J_h \,\mathrm{div}\left(\mathbf{V}^j\right) dX$$

- $\int_{\Omega_0} \frac{\partial J_h}{\partial \alpha_j} dX$, $\int_{\Omega_0} \left(\frac{\partial J_h}{\partial X}\right)^T \mathbf{V}^j dX$ and $\int_{\Omega_0} J_h \,\mathrm{div}\left(\mathbf{V}^j\right) dX$ can be calculated by explicit formulas efficiently because shape derivatives of the IFE functions and velocity fields vanish on all non-interface elements

- $\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h$ is computed by the standard discretized adjoint method together with the shape derivatives of IFE basis functions.

- For typical interface inverse problems, formulas for

$$\frac{\partial J_h}{\partial \alpha_j}, \quad \frac{\partial J_h}{\partial X}, \quad \frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h}$$

  can be easily derived

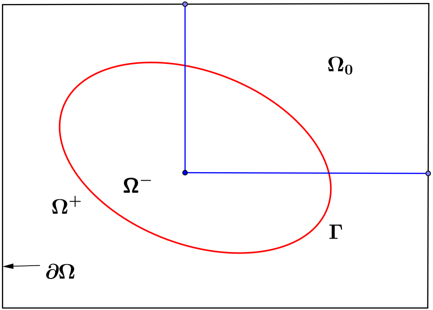
Algorithm Summary

Algorithm The IFE Shape Optimization Algorithm

1: Generate a fixed mesh and choose an initial interface curve discretized by $\boldsymbol{\alpha}$.
2: Loop until convergence.
3: Prepare data for the interface-mesh intersection points:
  a: use $\boldsymbol{\alpha}$ to generate the parametric curve $\Gamma(\boldsymbol{\alpha})$;
  b: find the interface-mesh intersection points, interface edges and interface elements.
4: Prepare matrices and vectors for the IFE systems, and compute the cost function:
  a: assemble matrices and vectors $\mathbf{A}, \mathbf{F}$ for the IFE systems;
  b: compute the PPIFE solutions $\mathbf{u}$ and compute the objective function $\mathcal{J}_h(\boldsymbol{\alpha})$.
5: Compute the shape sensitivities or the gradient:
  a: prepare the velocity fields $\mathbf{V}^j$, $j \in \mathcal{D}$, and shape derivatives of IFE shape functions;
  b: compute $D_{\alpha_j}\mathcal{J}_h(\boldsymbol{\alpha}) = \frac{\partial \mathcal{J}_h}{\partial \mathbf{u}_h} \cdot D_{\alpha_j}\mathbf{u}_h + \int_{\Omega_0} \frac{\partial J_h}{\partial \alpha_j} dX + \int_{\Omega_0} \left(\frac{\partial J_h}{\partial X}\right)^T \mathbf{V}^j dX + \int_{\Omega_0} J_h \operatorname{div}\left(\mathbf{V}^j\right) dX$
  c: form the gradient of the objective function: $\nabla_{\boldsymbol{\alpha}}\mathcal{J}_h(\boldsymbol{\alpha})$
6: Update $\boldsymbol{\alpha}$ by a chosen gradient-based optimization algorithm.
7: End loop

$$\boldsymbol{\alpha}_0 \xrightarrow{J_h(\boldsymbol{\alpha}_0),\ \nabla_{\boldsymbol{\alpha}}\mathcal{J}_h(\boldsymbol{\alpha}_0)} \boldsymbol{\alpha}_1 \xrightarrow{J_h(\boldsymbol{\alpha}_1),\ \nabla_{\boldsymbol{\alpha}}\mathcal{J}_h(\boldsymbol{\alpha}_1)} \boldsymbol{\alpha}_2 \longrightarrow \cdots$$

Example 1: Heat distribution design



Give: Given all data functions in the interface forward problem and $\bar{u}(X)$ on $\Omega_0 \subseteq \Omega$
Find: Γ such that the solution $u(X)$ to the interface forward problem is close to $\bar{u}(X)$

The objective functional is given by

$$\mathcal{J}(\Gamma) = \int_{\Omega_0} (u(X(\Gamma);\Gamma) - \bar{u}(X))^2 dX$$

$$\boldsymbol{\alpha}^* = argmin\ \mathcal{J}_h(\boldsymbol{\alpha}), \quad \mathcal{J}_h(\boldsymbol{\alpha}) = \int_{\Omega_0} J_h(\boldsymbol{\alpha}, \mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha}))dX,$$
$$\text{subject to} \quad \mathbf{A}(X(\boldsymbol{\alpha}), \boldsymbol{\alpha})\mathbf{u}_h(\alpha) - \mathbf{F}(X(\boldsymbol{\alpha}), \boldsymbol{\alpha}) = \mathbf{0},$$

where,

$$J_h(\boldsymbol{\alpha}, \mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha})) = \left(\tilde{J}_h(\boldsymbol{\alpha}, \mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha}))\right)^2,$$
$$\tilde{J}_h(\boldsymbol{\alpha}, \mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha})) = \sum_{i=1}^{|\mathring{\mathcal{N}}_h|} u_i(\boldsymbol{\alpha})\phi_i(X(\boldsymbol{\alpha}), \boldsymbol{\alpha}) + \sum_{i=|\mathring{\mathcal{N}}_h|+1}^{|\mathcal{N}_h|} g_D(X_i)\phi_i(X(\boldsymbol{\alpha}), \boldsymbol{\alpha}) - \bar{u}.$$

Then, computing $\frac{\partial J_h}{\partial X}$ is straightforward:

$$\frac{\partial J_h}{\partial X} = 2\tilde{J}_h(\mathbf{u}_h(\boldsymbol{\alpha}), X(\boldsymbol{\alpha}), \boldsymbol{\alpha}) \left(\sum_{i=1}^{|\mathring{\mathcal{N}}_h|} u_i(\boldsymbol{\alpha})\nabla\phi_i(X(\boldsymbol{\alpha}), \boldsymbol{\alpha}) + \sum_{i=|\mathring{\mathcal{N}}_h|+1}^{|\mathcal{N}_h|} g_D^1(X_i)\nabla\phi_i(X(\boldsymbol{\alpha}), \boldsymbol{\alpha}) - \nabla\bar{u} \right)$$

Formulas for

$$\frac{\partial \mathcal{J}_h}{\partial \mathbf{u}}, \quad \frac{\partial J_h}{\partial \alpha_j}, \ j = 1, 2, \cdots, |\boldsymbol{\alpha}|$$

can also be easily derived.

(click to play a movie for $\Omega_0 = \Omega$)

(click to play a movie for $\Omega_0 \neq \Omega$)

Some Examples for Interface Inverse Problems

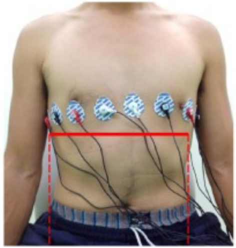
Example 2: Electrical Impedance Tomography

Governing BVP: interface problem for $-\nabla \cdot (\beta \nabla u) = f$

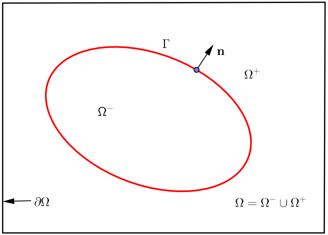
Given both:

$$\text{Dirichlet data:} \quad u|_{\partial\Omega} = g_D \quad \text{on } \partial\Omega,$$
$$\text{Neumann data:} \quad \frac{\partial u}{\partial \mathbf{n}}|_{\partial\Omega_N} = g_N \quad \text{on } \partial\Omega.$$

Find: Γ such that the solution $u(X)$ to the interface forward problem is close to g_D and g_N on $\partial\Omega$



(by Huang, Hung, Wang, Lin)



The objective functional (Kohn-Vogelius type functional) is given by

$$\mathcal{J}(\Gamma) = \int_{\Omega} (u_N(X(\Gamma);\Gamma) - u_D(X(\Gamma);\Gamma))^2 dX$$

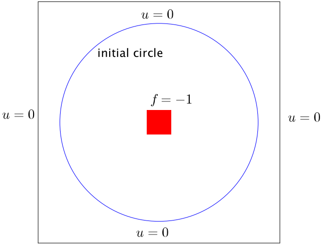
$$\begin{cases} -\nabla \cdot (\beta \nabla u_D) = f \\ [u_D]|_{\Gamma} = 0 \\ [\beta \nabla u \cdot \mathbf{n}]|_{\Gamma} = 0 \\ u_D = g_D \quad \text{on } \partial\Omega \end{cases} \qquad \begin{cases} -\nabla \cdot (\beta \nabla u_N) = f \\ [u_N]|_{\Gamma} = 0 \\ [\beta \nabla u_N \cdot \mathbf{n}]|_{\Gamma} = 0 \\ \frac{\partial u_N}{\partial \mathbf{n}} = g_N \quad \text{on } \partial\Omega \end{cases}$$

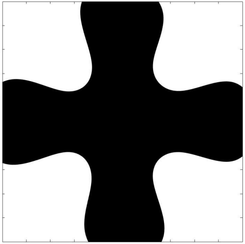
(click to play a movie for the D-N problem)

Example 3: Heat Dissipation Optimization

Find: Γ to minimize the heat dissipation functional:

$$\mathcal{J}(\Gamma, u(\Gamma)) = \int_{\Omega} \beta |\nabla u(X(\Gamma); \Gamma)|^2 dX, \qquad |\Omega^-| < 0.5|\Omega|$$





Conclusion Remarks

- Interface inverse problems are treated as shape optimization problems in which the constraint is the related interface forward problem
- Parametric curve is used to discretize the interface.
- An IFE method is applied to discretize the related interface forward problem and objective functional such that the disretizations to both the constraint and objective functional in the shape optimization problem are optimal regardless the location of the parameterized interface in a fixed mesh.
- With the IFE discretization, the shape optimization is reduced to a constrained optimization problem. The gradient of the objective function in this constrained optimization problem is derived. Both the objective function and its gradient can be implemented accurately and efficiently in the IFE framework.
- IFE methods allow us to solve interface inverse problems with a fixed mesh not only efficiently but also accurately.
- IFE methods have be extended to the interface inverse problems related with interface forward problems of linear elasticity; extensions to interface problems related with interface forward problems associated with other types PDEs are under consideration.

Thank Your