A Brief Introduction to Unfitted Finite Element Methods

Xinpei Zhang

University of Chinese Academy of Sciences
FEM Reading Group at Oxford

March 6, 2026

Outline

What is Unfitted FEM

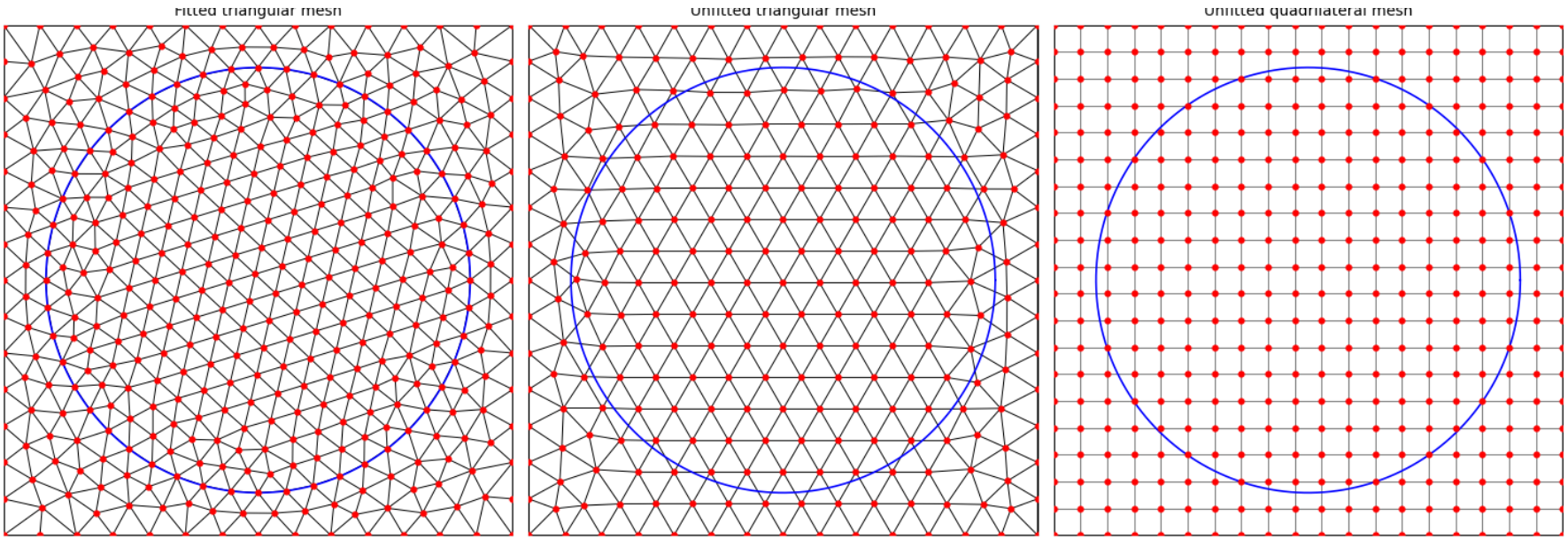


Figure: Comparision of fitted and unfitted meshes.

Definition

Unfitted Finite Element Method (UnfittedFEM) is a finite element approach in which the computational mesh does not conform to the geometry of the domain or interfaces.

What is Unfitted FEM

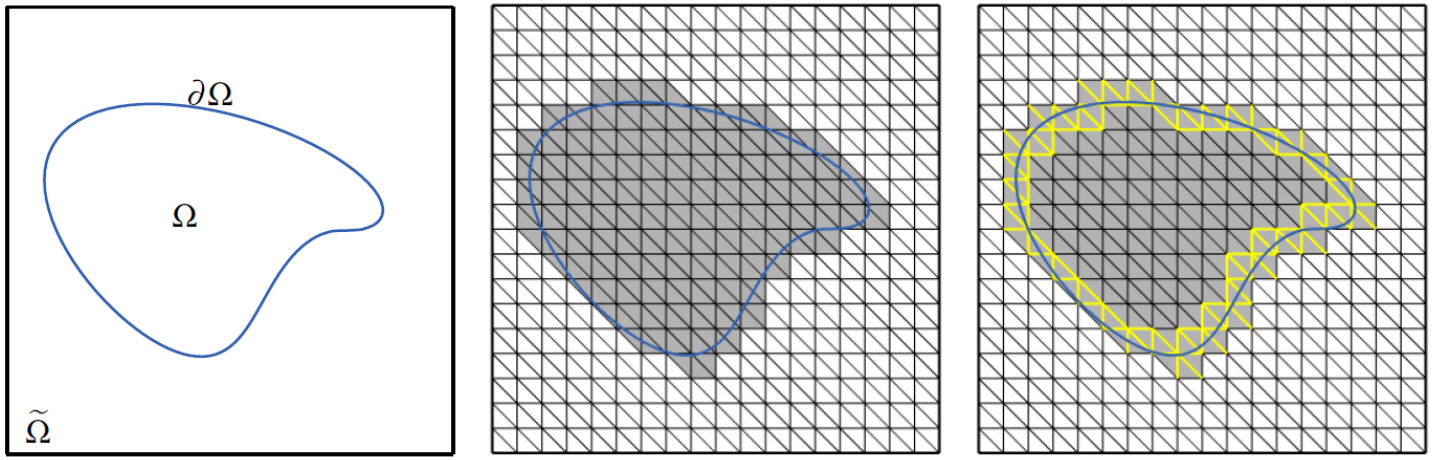


Figure 2.1. (a) The domain $\Omega \subset \mathbb{R}^2$ and the background domain $\widetilde{\Omega}$. (b) The background mesh $\widetilde{\mathcal{T}}_h$, the active mesh $\mathcal{T}_h$, (consisting of the grey triangles), and the active domain Ω_h (the grey region). (c) Edges marked in yellow illustrate the edges in the set $\mathcal{F}_h$, where the stabilization defined in equation (2.14) applies.

Figure: Fictitious Domain [3]

Why Unfitted FEM?

- **Flexibility**: Easily handle complex geometries, moving boundaries, time dependent domains and multi-physics coupling problems.
- **Efficiency**: Potentially reduce computational cost by using simpler meshes and avoiding remeshing.
- **Parallelization**: Unfitted meshes can be more easily partitioned for parallel computing.

Recent studies(2020) indicate that **38%** of CAE simulation project time is allocated to preprocessing. This consists of CAD model repair, meshing, contact definition, material properties application, boundary condition and load assignment. [9]

Problems with Unfitted FEM

- **Stability**: Small cut elements can lead to ill-conditioned system matrices. [2]
- **Boundary Condition Enforcement**: Imposing Dirichlet conditions is non-trivial.
- **Accuracy**: Accuracy is sacrificed by mesh does not align with the physical boundaries, (weak) boundary condition enforcement methods, integration over cut elements, jump over interfaces, etc.

Goal

Enhancing robustness of the (UnfittedFEM) method without sacrificing accuracy. [2]

Immersed Boundary Method

Let us start with the simplest question:

- No volume, only boundary(curve, like the flexible leaflet of a human heart). No mass. 2D case for simplicity.
- Background mesh is uniform quadrilateral grid.
- Dirichlet boundary condition.
- Viscous incompressible fluid flow, governed by the Navier-Stokes equations. Low Reynolds number.

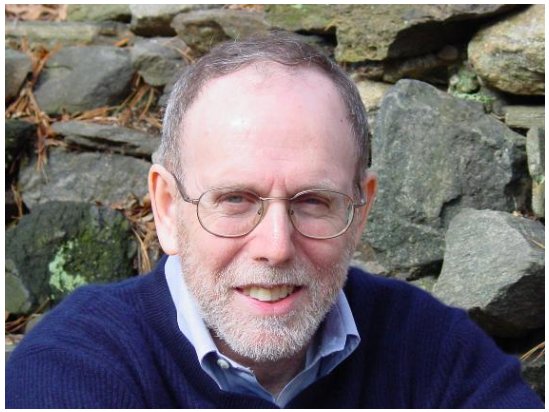


Figure: Charles S. Peskin

Then we got the **Immersed Boundary Method** (Peskin, 1972) [8]!

Euler description and Lagrange description

The nondimensional form of the Navier-Stokes equations is:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \Delta \mathbf{u} + \mathbf{F}, \\ \nabla \cdot \mathbf{u} = 0, \end{cases} \tag{1}$$

Euler description: $\mathbf{u}(\mathbf{x}, t)$, $p(\mathbf{x}, t)$, $\mathbf{F}(\mathbf{x}, t)$, $\mathbf{x} \in \mathbb{R}^2$, where $\mathbf{u}$ is the fluid velocity, p is the pressure, and $\mathbf{F}$ is the force density.
Lagrange description: $\mathbf{X}(\mathbf{q}, t)$, $\mathbf{f}(\mathbf{q}, t)$, $\mathbf{q} \in \mathbb{R}$, where $\mathbf{X}$ is the position of the immersed boundary, and $\mathbf{f}$ is the force density on the immersed boundary.
Lagrange to Euler:

$$\mathbf{F}(\mathbf{x}, t) = \int_{X(s,t) \in B} \mathbf{f}(\mathbf{s}, t)\delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t))d\mathbf{s} \tag{2}$$

where δ is the Dirac delta function, and B is the moving immersed boundary.
Euler to Lagrange:

$$\mathbf{X}_t(\mathbf{s}, t) = \int_R \mathbf{u}(\mathbf{x}, t)\delta(\mathbf{x} - \mathbf{X}(\mathbf{s}, t))d\mathbf{x} \tag{3}$$

where R is the region occupied by the fluid.

Helmholtz-Hodge decomposition and operator P

Helmholtz-Hodge decomposition: Any vector field $\mathbf{F}$ can be decomposed into a divergence-free part and a gradient of a scalar potential:

$$\mathbf{F} = \mathbf{F}_{\text{div-free}} + \nabla\phi, \tag{4}$$

where $\nabla \cdot \mathbf{F}_{\text{div-free}} = 0$.

Projection operator P: The projection operator P projects a vector field onto the space of divergence-free vector fields:

$$P\mathbf{F} = \mathbf{F}_{\text{div-free}}. \tag{5}$$

Proposition

The projection operator P has the following properties:

- *P is linear.*
- *For any scalar function ϕ, $P(\nabla\phi) = 0$.*
- *For any vector $\mathbf{u}$, such that $\nabla \cdot \mathbf{u} = 0$, $P\mathbf{u} = \mathbf{u}$.*

New Equations

Applying the projection operator P to the Navier-Stokes equations, we get:

$$\begin{cases} \mathbf{u}_t = P(-\mathbf{u}\cdot\nabla\mathbf{u} + \nabla^2\mathbf{u} + \mathbf{F}), \\ \mathbf{X}_t = \int_R \mathbf{u}(\mathbf{x},t)\delta(\mathbf{x}-\mathbf{X}(\mathbf{s},t))d\mathbf{x}, \\ \mathbf{F}(\mathbf{x},t) = \int_{X(s,t)\in B} \mathbf{f}(\mathbf{s},t)\delta(\mathbf{x}-\mathbf{X}(\mathbf{s},t))d\mathbf{s} \\ \mathbf{f} = M(\mathbf{X}). \end{cases} \tag{6}$$

where M is a nonlinear operator describing the elastic properties of the boundary. Then discretize the Dirac delta function δ to D_{ij} at the grid points x_{ij}, which satisfies:

- $\lim_{h\to 0}\sum_{i,j} h^2 D_{i,j}(\mathbf{x}_k)\varphi(ih,jh) = \varphi(\mathbf{x}_k)$ for any continuous function φ.
- $D_{i,j}(\mathbf{x}_k)$ is continuous in $\mathbf{x}_k$.
- $\frac{1}{4} = \sum_{\substack{i \text{ odd or even} \\ j \text{ odd or even}}} h^2 D_{i,j}(\mathbf{x}_k)$,

After Discretization

A simple choice of $D_{i,j}$ is:

$$D_{i,j}(\mathbf{x}_k) = D_{ij}(\alpha h, \beta h) = \begin{cases} \frac{1}{16h^2}(2-|\alpha-i|)(2-|\beta-j|), |\alpha-i|<2, |\beta-j|<2 \\ 0, \text{otherwise} \end{cases} \tag{7}$$

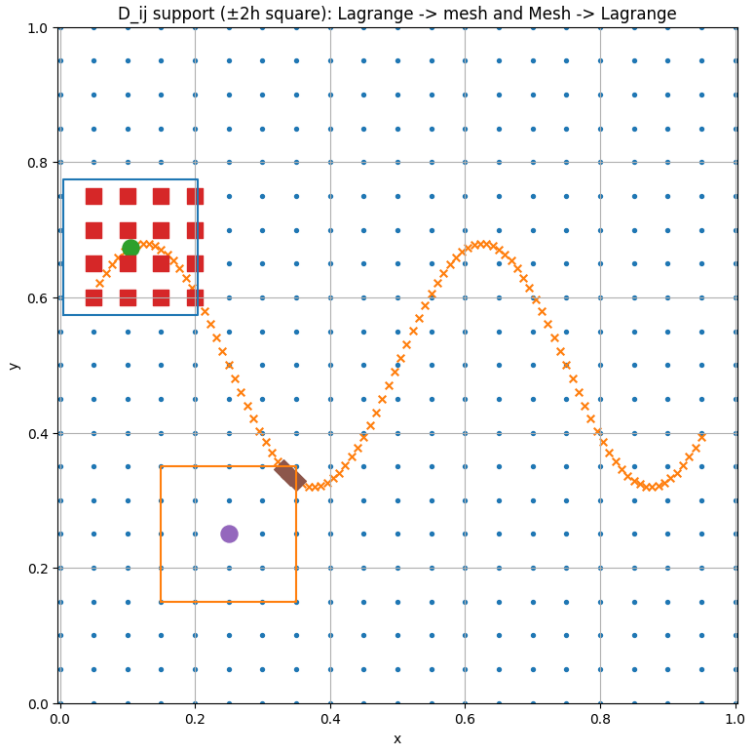
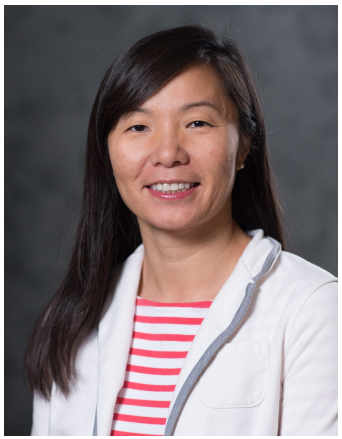


Figure: The discretized delta function $D_{i,j}$.

Problems of Immersed Boundary Method

- Weak assumptions: We want volume and mass to be non-zero.
- Easy mesh: We want to use quasi-uniform mesh, with refinement.
- Low accuracy: Only $\mathcal{O}(h)$ in space (especially for D_{ij}).
- Hard to solve: High Reynolds number, stiffness, small movement of the boundary cause great forces (force is $\mathcal{O}(\frac{1}{h})$), etc.

We need better Dirac function discretization, better time stepping methods, better transform between Euler and Lagrange, etc.



(a) Lucy Zhang

(b) Axel Gerstenberger

(c) Xiaodong Wang

(d) Wing K.Liu

We got the **Immersed Finite Element Method** (Zhang et al., 2004) [10]!

Immersed Finite Element Method

First we need a more complex implementaion of the Dirac delta function - the Reproducing Kernel Particle Method (RKPM).
For every function $u(\mathbf{x})$, it's reproducing function is derived as:

$$u^R(\mathbf{x}) = \int_{-\infty}^{+\infty} u(\mathbf{y})\phi(\mathbf{x}-\mathbf{y}, h)d\mathbf{y} \tag{8}$$

with a peojection operator or a window function ϕ. Specially, up to n-th order polynomial can be reproduced, i.e.

$$\int_{-\infty}^{+\infty} \mathbf{y}^m \phi(\mathbf{x}-\mathbf{y})d\mathbf{y} = \mathbf{x}^m, m = 0, 1, ..., n \tag{9}$$

We introduce a correction function $C(\mathbf{x};\mathbf{y})$ (a polynomial) to ensure the reproducing property:

$$\int_\Omega C(\mathbf{x};\mathbf{y})\frac{1}{r}\phi(\frac{\mathbf{x}-\mathbf{y}}{r})d\mathbf{y} = 1 \tag{10}$$

For not uniform meshes, the discretized delta function is defined as:

$$\phi_i(\mathbf{x}) = C(\mathbf{x};\mathbf{x}-\mathbf{x}_i)\frac{1}{r}W(\frac{\mathbf{x}-\mathbf{x}_i}{r})\Delta\mathbf{x}_i \tag{11}$$

RKPM

An example of the RKPM is cubic B-spline:

$$W(q) = \begin{cases} \frac{2}{3} - q^2 + \frac{1}{2}q^3, & 0 \le q < 1 \\ \frac{1}{6}(2-q)^3, & 1 \le q < 2 \\ 0, & q \ge 2 \end{cases} \tag{12}$$

And for d-dimensional case(with normalization):

$$\phi_j(X) = \frac{\phi_h^{(d)}(x_j - X)}{\sum_{k \in S(X)} \phi_h^{(d)}(x_k - X)\,\Delta V_k}. \tag{13}$$

where $\phi_h^{(d)}(x - X) = \frac{1}{h^d} \prod_{m=1}^{d} W\left(\frac{|x_m - X_m|}{h}\right)$., and $S(X)$ is the set of all nodes within the support of $\phi_h^{(d)}$ centered at X, ΔV_k is the volume associated with node k.

By RKPM we can achieve $\mathcal{O}(h^2)$ or $\mathcal{O}(h^3)$ even higher accuracy in space, and quasi-uniform meshes and refinement are allowed now.

Curve to Solid

We improve the original immersed boundary method, make the boundary have volume and mass, into a solid.
Still, fluid use Euler description and unfitted mesh, but the solid use Lagrange description and fitted mesh.
Still, force and velocity are transferred by the discretized delta function.
We skip some details of the implementation.
Questions remain:

- Accuracy?
- Stability?
- Consistency of mass and volume? Jump over interfaces?

Immersed Interface Method

(a) LeVeque Randall J

(b) Zhilin Li

Based on differential format, we get an idea to deal with the jump over interfaces by modifying the finite difference stencils near the interface, and get the **Immersed Interface Method** (LeVeque and Randall J, 1994) [5].

Immersed Interface Method

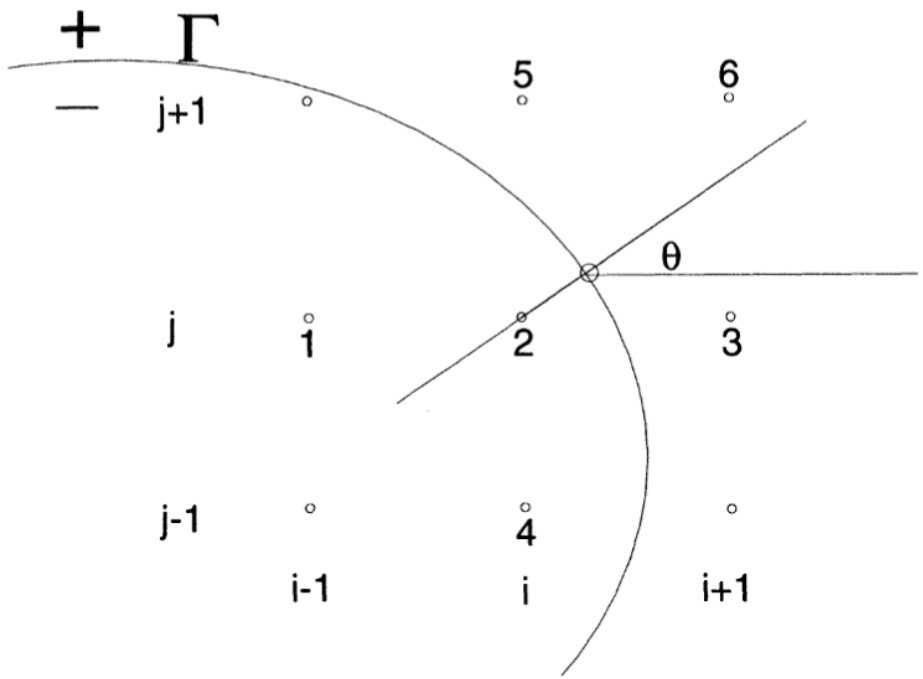


FIG. 2. *The geometry at an irregular grid point* (i, j). *The coefficients* γ_1 *through* γ_6 *will be determined for the stencil points labelled* 1–6. *The circled point on* Γ *is the point* (x_i^*, y_j^*).

Figure: Six points finite difference stencil and projection of the interface onto the grid.

Based on the five-point difference scheme, the IIM projects boundaries onto grid points via the direction of the normal vector and incorporates jump-related calculations within the difference scheme for correction. Due to the introduction of new unknowns, they extended the difference scheme to a six-point configuration.

eXtended Finite Element Method (XFEM)

To deal with small crack problems, eXtended Finite Element Method (XFEM) was proposed by Belytschko and Black in 1999, and improved by (Nicolas MOËS et al, 1999) [6].

Figure: Ted Belytschko

Figure: Nicolas MOËS

eXtended Finite Element Method (XFEM)

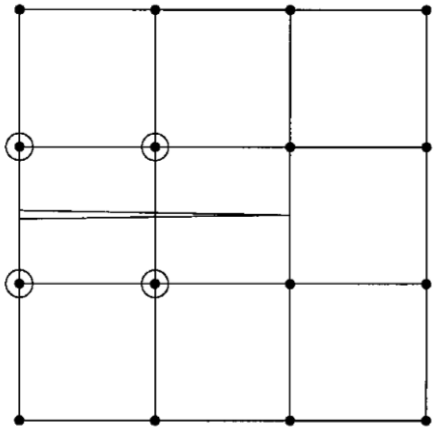


Figure 4. Crack not aligned with a mesh, the circled nodes are enriched with the discontinuous function

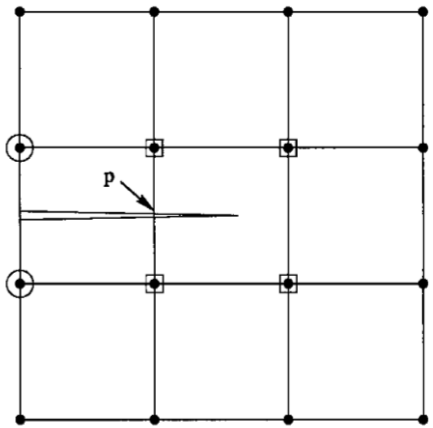


Figure 5. Crack not aligned with a mesh, the circled nodes are enriched with the discontinuous function and the squared nodes with the tip enrichment functions. Enrichment with only the discontinuous function shortens the crack to point p

$$\mathbf{u}^h = \underbrace{\sum_{i\in I} \mathbf{u}_i \phi_i}_{\text{Continuous}} + \underbrace{\sum_{j\in J} \mathbf{b}_j \phi_j H(\mathbf{x})}_{\text{Strong discontinuity}} + \underbrace{\sum_{k\in K_1} \phi_k \left(\sum_{l=1}^{4} \mathbf{c}_k^{l1} F_l^1(\mathbf{x}) \right)}_{\text{Crack tip 1}} + \underbrace{\sum_{k\in K_2} \phi_k \left(\sum_{l=1}^{4} \mathbf{c}_k^{l2} F_l^2(\mathbf{x}) \right)}_{\text{Crack tip 2}} \tag{14}$$

The approximation takes the form above, where I is the set of all nodes, J is the set of nodes whose support is cut by the crack, K_1 and K_2 are the sets of nodes whose support contains crack tip 1 and crack tip 2, respectively. $H(\mathbf{x})$ is the Heaviside function, and $F_l^1(\mathbf{x})$, $F_l^2(\mathbf{x})$ are the crack tip enrichment functions.

eXtended Finite Element Method (XFEM)

Let $\mathbf{x} \in \Omega$, $\quad \mathbf{x}^*$ = closest point to $\mathbf{x}$ on the crack C,

and $\mathbf{e}_n$ be the unit normal at $\mathbf{x}^*$.

Discontinuous jump (Haar) function

$$H(\mathbf{x}) = \begin{cases} +1, & (\mathbf{x} - \mathbf{x}^*) \cdot \mathbf{e}_n > 0, \\ -1, & (\mathbf{x} - \mathbf{x}^*) \cdot \mathbf{e}_n < 0. \end{cases}$$

Near-tip enrichment functions at crack tip 1 (local polar coordinates (r_1, θ_1))

$$F_l^1(\mathbf{x}) = \left\{ \sqrt{r_1} \sin\left(\frac{\theta_1}{2}\right),\ \sqrt{r_1} \cos\left(\frac{\theta_1}{2}\right),\ \sqrt{r_1} \sin\left(\frac{\theta_1}{2}\right) \sin\theta_1,\ \sqrt{r_1} \cos\left(\frac{\theta_1}{2}\right) \sin\theta_1 \right\}.$$

Near-tip enrichment functions at crack tip 2 (local polar coordinates (r_2, θ_2))

$$F_l^2(\mathbf{x}) = \left\{ \sqrt{r_2} \sin\left(\frac{\theta_2}{2}\right),\ \sqrt{r_2} \cos\left(\frac{\theta_2}{2}\right),\ \sqrt{r_2} \sin\left(\frac{\theta_2}{2}\right) \sin\theta_2,\ \sqrt{r_2} \cos\left(\frac{\theta_2}{2}\right) \sin\theta_2 \right\}.$$

Example on XFEM

Domain:

$$\Omega = (0,1)^3$$

Interface (sphere):

$$\Gamma: \ (x-0.5)^2 + (y-0.5)^2 + (z-0.5)^2 = 0.35^2$$

Subdomains:

$$\Omega^- = \text{inside sphere}, \qquad \Omega^+ = \Omega \setminus \overline{\Omega^-}$$

PDE:

$$-\nabla \cdot (a(x)\nabla u) = f \quad \text{in } \Omega^- \cup \Omega^+$$

$$a(x) = \begin{cases} 0.1, & x \in \Omega^- \\ 2, & x \in \Omega^+ \end{cases}$$

Exact solution:

$$u = \begin{cases} \cos(xyz), & x \in \Omega^- \\ \sin(x+y+z), & x \in \Omega^+ \end{cases}$$

Jump conditions on Γ:

$$[u] = u_1 - u_2, \qquad [a\nabla u] \cdot n = 0.1\nabla u_1 \cdot n - 2\nabla u_2 \cdot n$$

Parallel Hierarchical Grid

PHG (Parallel Hierarchical Grid) is a parallel program development platform specifically designed for three-dimensional adaptive finite element analysis, currently under development by the State Key Laboratory of Scientific and Engineering Computing(Institute of Computational Mathematics and Scientific/Engeering Computing, CAS). Its core lies in a distributed hierarchical grid structure. Presently, PHG handles three-dimensional tetrahedral coordinated grids as its primary mesh objects. Developed in C language, PHG implements parallelism through MPI message-passing communication.

Core	CPU Time	Run Time
1	276	275
2	314	156
4	295	75
8	337	43
18	472	27
36	703	21
72 (2 machines)	1117	22

Table: Performance comparison with different numbers of cores

Control Boundaries Using Functions?(Fictitious Domain Methods)

We know how to control boundaries with unfitted meshes now. But can we control boundaries with functions?
It means, embedding complex regions into a larger, simpler region, construct a mesh on the larger region, and then recover the original problem through constraints.
Absolutely!
I'll introduce:

- Lagrange multiplier method
- Nitsche's method
- Ghost penalty method

Lagrange Multiplier Method

(Babuška 1973) [1] introduced the Lagrange multiplier method to finite element methods in a very early stage. For a elliptic problem with Dirichlet boundary:

$$\begin{cases} -\Delta u + u = f, & \text{in } \Omega, \\ u = g, & \text{on } \partial\Omega, \end{cases} \tag{15}$$

The classical technique is to minimize the quadratic functional:

$$J(v) = \frac{1}{2}\int_\Omega \left[|\nabla v|^2 + v^2\right] - \int_\Omega fv dx \tag{16}$$

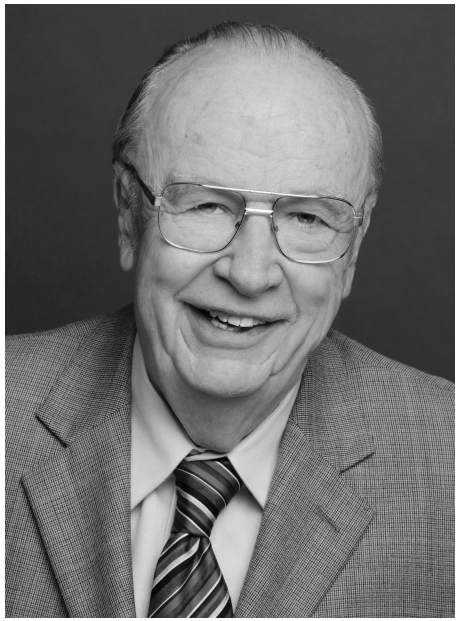


Figure: Ivo Babuška

Then we introduce a Lagrange multiplier λ:

$$L(v, \lambda) = J(v) + \oint_{\partial\Omega} \lambda(v - g) ds \tag{17}$$

$\lambda \in H^{-1/2}(\partial\Omega)$, means $\frac{\partial u}{\partial n}$ in physics.

Lagrange Multiplier Method

Introduce bilinear forms:

$$B(u, \lambda; v, \mu) = \int_{\Omega} (\nabla u \cdot \nabla v + uv) dx - \int_{\partial\Omega} (\lambda v + \mu u) ds \tag{18}$$

and two functionals:

$$F(u, \lambda) = \int_{\Omega} f u dx, \quad G(u, \lambda) = -\oint_{\partial\Omega} \lambda g ds$$

At stationary point (u_0, λ_0), we have for all (v, μ):

$$B(u_0, \lambda_0; v, \mu) = F(v, \mu) + G(v, \mu) \tag{19}$$

Lagrange Multiplier Method

Theorem (Existence and Uniqueness)

Bilinear form B in space $H = H^1(\Omega) \times H^{-1/2}(\partial\Omega)$ satisfies the inf-sup condition and boundedness, then the solution (u_0, λ_0) is unique.

Theorem (Equivalence)

Let $f \in L_2(\Omega)$, $g \in H^{\frac{1}{2}}(\partial\Omega)$ and let $(u_0, \lambda_0) \in H$ be such that (19) holds for all $(v, \mu) \in H$. Further let $w \in H^1(\Omega)$ be the weak solution of the Dirichlet problem. Then $u_0 = w$ and $\lambda_0 = \partial w/\partial n$.

Lagrange Multiplier Method

Theorem (Discrete inf–sup condition)

Let

$$M_h = S^{(1)}_{h_1,k_1}(\Omega) \times S^{(2)}_{h_2,k_2}(\partial\Omega)$$

be the finite element space introduced above, where $S^{(1)}_{h_1,k_1}(\Omega)$ *is a* (t_1, k_1)*–regular system and* $S^{(2)}_{h_2,k_2}(\partial\Omega)$ *is a strongly* (t_2, k_2)*–regular system, with* $k_1 \geq 1$ *and* $k_2 \geq \frac{1}{2}$.

Assume further that the mesh parameters satisfy $h_1 \geq Kh_2$, *for some constant* $K > 0$ *sufficiently large and independent of* h_1, h_2.

Then there exists a constant $\beta > 0$, *independent of* h_1 *and* h_2, *such that for every* $(u_h, \lambda_h) \in M_h$, $\sup_{(v_h,\mu_h)\in M_h} \frac{B\big((u_h,\lambda_h),(v_h,\mu_h)\big)}{\|(v_h,\mu_h)\|_{H^1(\Omega)\times H^{-1/2}(\partial\Omega)}} \geq \beta \|(u_h, \lambda_h)\|_{H^1(\Omega)\times H^{-1/2}(\partial\Omega)}$.

Equivalently, the discrete Babuška–Brezzi (inf–sup) condition holds on M_h, *and the constant* β *is independent of the mesh size.*

Lagrange Multiplier Method

Theorem (Optimal convergence in H^1 and L^2 norms)

Under the assumptions of former Theorems, let (u, λ) denote the exact solution of the saddle–point formulation, and let $(u_h, \lambda_h) \in M_h$ be the corresponding finite element solution. Assume

$$f \in H^k(\Omega), \quad k \geq 0, \qquad g \in H^l(\partial\Omega), \quad l \geq \tfrac{1}{2}.$$

Then there exists a constant $C > 0$, independent of h_1, h_2, such that
H^1–estimate:

$$\|u - u_h\|_{H^1(\Omega)} + \|\lambda - \lambda_h\|_{H^{-1/2}(\partial\Omega)} \leq C\Big(h_1^{\mu_1} \|f\|_{H^k(\Omega)} + h_2^{\mu_2} \|g\|_{H^l(\partial\Omega)} \Big),$$

where $\mu_1 = \min\{k+1, t_1, k_1+1\}, \qquad \mu_2 = \min\{l+\tfrac{1}{2}, t_2, k_2+\tfrac{1}{2}\}.$
L^2–estimate:

$$\|u - u_h\|_{L^2(\Omega)} \leq C\Big(h_1^{\mu_1+1} \|f\|_{H^k(\Omega)} + h_2^{\mu_2+1} \|g\|_{H^l(\partial\Omega)} \Big).$$

In particular, the method converges with optimal order in both $H^1(\Omega)$ and $L^2(\Omega)$.

Nitsche's Method

In 1972, J. Nitsche proposed a method to weakly enforce Dirichlet boundary conditions without introducing additional unknowns, which is now known as Nitsche's method [7].
In 2002, Mr and Mrs Hansbo proposed a Nitsche's method for unfitted finite element methods [4].
For a stationary heat conduction problem with Dirichlet boundary and jump over interface:

$$\begin{cases} -\nabla \cdot (\alpha \nabla u) = f, & \text{in } \Omega = \Omega_1 \cup \Omega_2, \\ u = g, & \text{on } \partial\Omega, \\ [u] = 0, & \text{on } \Gamma, \\ [\alpha \nabla u \cdot n] = 0, & \text{on } \Gamma, \end{cases} \tag{20}$$

Where $[u] = u_1 - u_2$ denotes the jump of u across the interface Γ,

Figure: Anita Hansbo

Figure: Peter Hansbo

Nitsche's Method

Assume $f \in L^2(\Omega)$ and $g \in H^{1/2}(\Gamma)$. The origin variational formulation is to find $u \in H_0^1(\Omega)$ such that for all $v \in H_0^1(\Omega)$:

$$(\alpha \nabla u, \nabla v)_\Omega = (f, v)_\Omega + (g, v)_\Gamma \tag{21}$$

The method is defined by the variational problem of finding $U = (U_1, U_2)$ in $V^h = V_1^h \times V_2^h$, where $V_i^h = \left\{\phi_i \in H^1(\Omega_i) : \phi_i|_{K_i} \text{ is linear, } \phi_i|_{\partial\Omega} = 0\right\}$ such that

$$a_h(U, \phi) = L(\phi), \quad \forall \phi \in V^h, \tag{22}$$

where

$$a_h(U, \phi) = \underbrace{(\alpha_i \nabla U_i, \nabla \phi_i)_\Omega}_{\text{Bulk diffusion term}} - \underbrace{([U], \{\alpha \nabla_{\boldsymbol{n}} \phi\})_\Gamma}_{\text{Consistency term}} - \underbrace{(\{\alpha \nabla_{\boldsymbol{n}} U\}, [\phi])_\Gamma}_{\text{Symmetry term}} + \underbrace{(\lambda [U], [\phi])_\Gamma}_{\text{Penalty term}}. \tag{23}$$

with λ sufficiently large, and

$$L(\phi) := (f, \phi)_\Omega + (\kappa_2 g, \phi_1)_\Gamma + (\kappa_1 g, \phi_2)_\Gamma. \tag{24}$$

Where $\kappa_i|_K = \frac{|K_i|}{|K|} = \frac{meas(K_i)}{meas(K)}$, and $\{\phi\} := (\kappa_1 \phi_1 + \kappa_2 \phi_2)$ is a convex combination of $\phi = \phi_1 + \phi_2$ along Γ.

Nitsche's Method

Along with their assumptions to ensure a fine mesh:

- triangulation is non-degenerate.
- Γ intersects each element boundary ∂K exactly twice, and each (open) edge at most once.
- Within each cut element K, the exact interface Γ_K can be represented as a normal perturbation of the straight-line discrete interface $\Gamma_{K,h}$.

Then a_h is coercive and consistent, then the stability and uniqueness of the solution is guaranteed.
It satisfies:

$$\|u - U\|_E \leq Ch\|u\|_{2,\Omega_1\cup\Omega_2}, \qquad \|u - U\|_{0,\Omega} \leq Ch^2\|u\|_{2,\Omega_1\cup\Omega_2},$$

i.e., first-order convergence in the energy norm and second-order convergence in the L^2-norm.
For any continuous linear functional $J(\cdot)$ on $L^2(\Omega)$, the error satisfies the residual-based bound

$$|J(u - U)| \leq C\,E(U)\,\|J\|_{0,\Omega},$$

where $E(U)$ is a computable residual-type error estimator.

Ghost Penalty

Now we have the strong and weak methods to control boundaries with functions.
Sometimes the intersect between cells and physical domain is very small. "If the cut results in elements with very small intersections with the physical domain, the system matrix may be very ill-conditioned." [2]
To solve this problem, Erik Burman proposed the Ghost Penalty method in 2010 [2].

Figure: Erik Burman

Standard Possion problem with Dirichlet boundary condition:

$$\begin{cases} -\Delta u = f, & \text{in } \Omega, \\ u = g, & \text{on } \partial\Omega, \end{cases} \tag{25}$$

Ghost Penalty

Assume $V_h^k := \left\{ v_h \in H^1(\Omega_{\mathcal{T}}) : v_h|_T \in P_k(T) \right\}$. Consider the following non-symmetric fictitious domain method inspired by Nitsche's method: find $u_h \in V_h^k$ such that for all $v_h \in V_h^k$:

$$a_h(u, v) = \int_\Omega f v_h dx$$

where

$$a_h(u, v) = (\nabla u_h, \nabla v_h)_\Omega - \langle \nabla u_h \cdot n, v_h \rangle_{\partial\Omega} + \langle u_h, \nabla v_h \cdot n \rangle_{\partial\Omega} + \frac{\gamma}{h} \langle u_h, v_h \rangle_{\partial\Omega}, \quad \gamma > 0. \tag{26}$$

He defines the ghost penalty as

$$s_h(u_h, v_h) := \sum_{l=1}^{N_{\mathcal{P}}} s_l(u_h, v_h), \text{ where } s_l(u_h, v_h) := \int_{\mathcal{P}_l} h_{\mathcal{P}_l}^{-2} (u_h - \pi_l u_h) v_h \, \mathrm{d}x. \tag{27}$$

where $N_{\mathcal{P}}$ is the number of patches, π_l is the L^2-projection onto $P_k(\mathcal{P}_l)$, and $\mathcal{P}_l$ is the patch of elements around the cut elements.

Ghost Penalty

Lemma (Extended coercivity)

There exists $\alpha_s > 0$ such that, for all $v_h \in V_h^k$,

$$\alpha_s \|v_h\|^2_{1,h,\Omega_T} \leq \|v_h\|^2_{1,h,\Omega} + s_h(v_h, v_h).$$

Lemma (Condition number bound)

Let $G = A + S$ denote the system matrix associated with the bilinear form

$$A_h(u_h, v_h) = a_h(u_h, v_h) + s_h(u_h, v_h). \tag{28}$$

Then the condition number of G satisfies the upper bound $\kappa(G) \leq C_A h_{\min}^{-2}$, where the constant

$$C_A = M_A C_I^2 C_P^2 \alpha_A^{-1} \left(\frac{\mu_{\max}}{\mu_{\min}} \right)$$

is independent of the boundary/mesh intersection $\partial\Omega \cap \mathcal{T}_h$.

Ghost Penalty

Lemma (Optimal convergence)

Assume that problem admits a solution $u \in H^{k+1}(\Omega)$ *and that* u_h *is the solution of* (28). *Then there holds*

$$\|u - u_h\|_{1,h,\Omega} \leq Ch^k |u|_{H^{k+1}(\Omega)}.$$

Remark

If we use Nitsche's method + Ghost Penalty, or Lagrange multiplier method + Ghost Penalty, we can achieve optimal convergence and good condition number even for very small cut elements.
Then we get the **CutFEM**, published by Burman et al in 2015, and improved further in recent years [3].

Other UnfittedFEM Methods

- **Finite Cell Method**: Embedded the physical domain into a larger simple domain and extend the PDE with a small stiffness in the fictitious region, solving everything on a background mesh.
- φ-**FEM**: Use a level-set function φ to implicitly describe the boundary and construct modified finite element spaces that encode boundary conditions without explicit cut-cell integration.
- **Shifted Boundary Method**: Replace boundary integrals on the true geometry by integrals on a nearby mesh-aligned surrogate boundary and correct the error using Taylor expansions.
- **TraceFEM**: Define finite element functions on a bulk mesh and restrict (trace) them onto an embedded surface where the PDE is posed.
- Zhiming Chen and Yong Liu introduced a method that on arbitrary smooth complex domains, an automatic elimination of small-cell construction shapes enables the formation of regular meshes, achieving arbitrary high-order consistent finite element methods without ghost penalty. This approach yields stable and optimal hp error estimates. [11]

Conclusion

- Unfitted finite element methods are powerful tools for solving PDEs on complex domains without requiring body-fitted meshes.
- One way to control boundaries is use Dirac delta functions to transfer forces and velocities between fluid and solid.
- Another way is to project boundaries onto grid points, or use enrichment functions to capture discontinuities.
- Third way is to embed complex regions into larger, simpler regions and recover the original problem and stabilize small cut elements through constraints, such as Lagrange multipliers, Nitsche's method, and Ghost penalty method.

References I

[1] Ivo Babuška, *The finite element method with lagrangian multipliers*, Numerische Mathematik **20** (1973), no. 3, 179–192.

[2] Erik Burman, *Ghost penalty*, Comptes Rendus Mathematique **348** (2010), no. 21-22, 1217–1220.

[3] Erik Burman, Peter Hansbo, Mats G Larson, and Sara Zahedi, *Cut finite element methods*, Acta Numerica **34** (2025), 1–121.

[4] Anita Hansbo and Peter Hansbo, *An unfitted finite element method, based on nitsche' s method, for elliptic interface problems*, Computer methods in applied mechanics and engineering **191** (2002), no. 47-48, 5537–5552.

[5] Randall J LeVeque and Zhilin Li, *The immersed interface method for elliptic equations with discontinuous coefficients and singular sources*, SIAM Journal on Numerical Analysis **31** (1994), no. 4, 1019–1044.

[6] Nicolas Moës, John Dolbow, and Ted Belytschko, *A finite element method for crack growth without remeshing*, International journal for numerical methods in engineering **46** (1999), no. 1, 131–150.

References II

[7] J Nitsche, *On dirichlet problems using subspaces with nearly zero boundary conditions*, The mathematical foundations of the finite element method with applications to partial differential equations, Elsevier, 1972, pp. 603–627.

[8] Charles S Peskin, *Flow patterns around heart valves: a numerical method*, Journal of computational physics **10** (1972), no. 2, 252–271.

[9] G. Roth, *C_jun_20_americas_250*, Presentation Recording, NAFEMS Americas, June 2020, NAFEMS Resource Center. Accessed 2026-02-24.

[10] Lucy Zhang, Axel Gerstenberger, Xiaodong Wang, and Wing Kam Liu, *Immersed finite element method*, Computer Methods in Applied Mechanics and Engineering **193** (2004), no. 21-22, 2051–2067.

[11] 陈志明 and 刘勇, 任意区域的网格自动生成和任意高阶有限元方法, 中国科学: 数学 **54** (2024), no. 03, 337–354.