Simulation of soft tissues using innovative non-conforming Finite Element Methods (PhiFEM)

Author: Desmond Roussel NZOYEM

Supervisors: Michel DUPREZ, Stéphane COTIN
Referent teachers: Christophe PRUD'HOMME, Joubine AGHILI

Course: M2 CSMI Project 2020/2021

Institution: University of Strasbourg

Contents

Project description

Project description

Environment and context

What is the environment?

1. **Inria:** where digital health is a main research topic

What is the environment?

1. **Inria:** where digital health is a main research topic
2. **MIMESIS:**
   - real-time simulations for per-operative guidance
   - data-driven simulation dedicated to patient-specific modeling

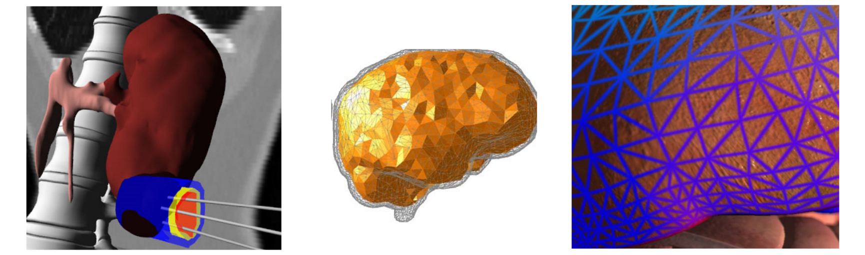


Figure : A few projects at MIMESI.

What are we trying to do?

The goal is to **develop a Finite Element Method (FEM) adapted to computer-assisted surgery**.

What are we trying to do?

The goal is to **develop a Finite Element Method (FEM) adapted to computer-assisted surgery**.

The FEM should be:

- quick;

What are we trying to do?

The goal is to **develop a Finite Element Method (FEM) adapted to computer-assisted surgery**.

The FEM should be:

- quick;
- precise;

What are we trying to do?

The goal is to **develop a Finite Element Method (FEM) adapted to computer-assisted surgery**.

The FEM should be:

- quick;
- precise;
- patient-specific.

Project description

Objectives

What is our goal?

- **Main objective**: develop a ϕ-FEM technique for the dynamics of soft tissues.

What is our goal?

- **Main objective**: develop a ϕ-FEM technique for the dynamics of soft tissues.
- **Intermediate objectives**:
  1. Understand the ϕ-FEM technique in question.

What is our goal?

- **Main objective**: develop a ϕ-FEM technique for the dynamics of soft tissues.
- **Intermediate objectives**:
  1. Understand the ϕ-FEM technique in question.
  2. Reproduce the results from a preliminary study (the Poisson problem).

What is our goal?

- **Main objective**: develop a ϕ-FEM technique for the dynamics of soft tissues.
- **Intermediate objectives**:
  1. Understand the ϕ-FEM technique in question.
  2. Reproduce the results from a preliminary study (the Poisson problem).
  3. Develop a ϕ-FEM technique for the linear elasticity equation.

What is our goal?

- **Main objective**: develop a ϕ-FEM technique for the dynamics of soft tissues.
- **Intermediate objectives**:
  1. Understand the ϕ-FEM technique in question.
  2. Reproduce the results from a preliminary study (the Poisson problem).
  3. Develop a ϕ-FEM technique for the linear elasticity equation.
  4. Use ϕ-FEM on body organ geometries.

Presentation of ϕ-FEM

Presentation of ϕ-FEM

The classic FEM framework

What is FEM?

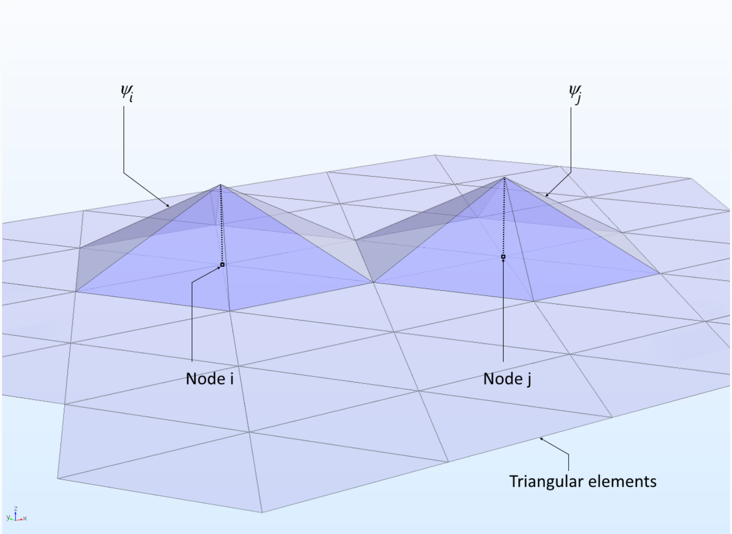


Figure : Nodal functions in the Finite Element Method (FEM) (Cyclopedia, 2017).

Presentation of ϕ-FEM

Immersed boundary methods

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;
- moving/evolving boundaries.

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;
- moving/evolving boundaries.

Common examples are:

- **XFEM** (De Cicco and Taheri, 2018)

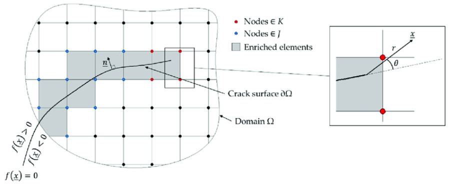


Figure : XFEM

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;
- moving/evolving boundaries.

Common examples are:

- **XFEM** (De Cicco and Taheri, 2018)
- **CutFEM** (Burman and Hansbo, 2010)

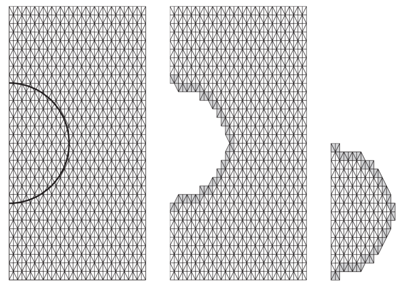


Figure : CutFEM

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;
- moving/evolving boundaries.

Common examples are:

- **XFEM** (De Cicco and Taheri, 2018)
- **CutFEM** (Burman and Hansbo, 2010)
- **SBM** (Main and Scovazzi, 2018)

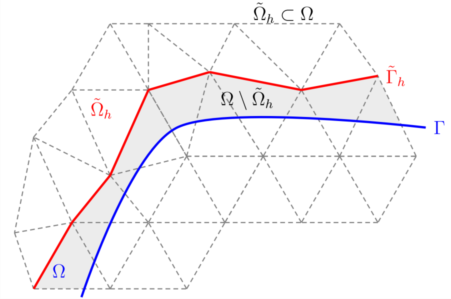


Figure : SBM

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;
- moving/evolving boundaries.

Common examples are:

- **XFEM** (De Cicco and Taheri, 2018)
- **CutFEM** (Burman and Hansbo, 2010)
- **SBM** (Main and Scovazzi, 2018)

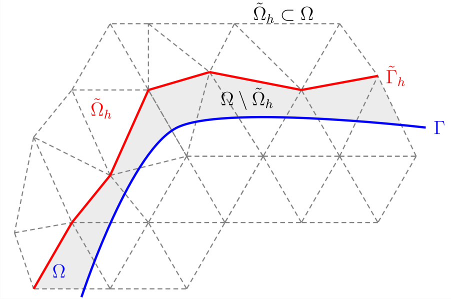


Figure : SBM

Note: These 3 techniques are efficient, but their implementation generates other difficulties, among other things, the quadrature (on the boundary).

What are immersed boundary methods?

In the case of quasi-incompressibility for example, it is necessary to use hexahedral meshes in order to avoid locking phenomena. However, there is no 3D mesh generator capable of meshing any geometry in a hexahedral manner.

One answer is to use **immersed boundary methods**. They can handle:

- complex geometries;
- non-body conforming grids;
- moving/evolving boundaries.

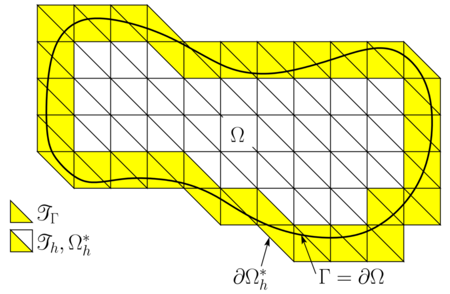


Figure : ϕ-FEM

Common examples are:

- **XFEM** (De Cicco and Taheri, 2018)
- **CutFEM** (Burman and Hansbo, 2010)
- **SBM** (Main and Scovazzi, 2018)

Note: These 3 techniques are efficient, but their implementation generates other difficulties, among other things, the quadrature (on the boundary). This lead to ϕ-FEM (Duprez and Lozinski, 2020).

The ideas behind ϕ-FEM

The main ideas are (for Dirichlet boundary conditions):

1. Define the domain using a level-set function ϕ.
2. Then make that function carry the solution e.g. $u = \phi w + g$.

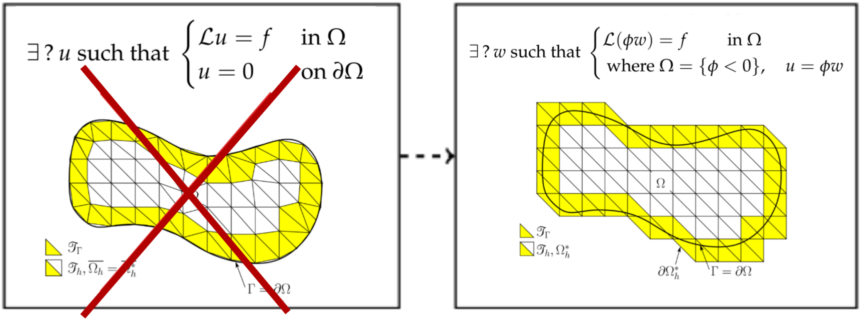


Figure : From Classic FEM (on the left) to ϕ-FEM (on the right)

Results

Results

The Poisson problem

Why the Poisson problem?

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases}$$

The Poisson problem (Poisson's equation) is used in various areas of Mathmetics and Physics (Newtonian gravity, electrostatics, fluid dynamics, etc.).

Figure : Siméon Denis Poisson (Wikipedia, 2020).

Why the Poisson problem?

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases}$$

The Poisson problem (Poisson's equation) is used in various areas of Mathmetics and Physics (Newtonian gravity, electrostatics, fluid dynamics, etc.).

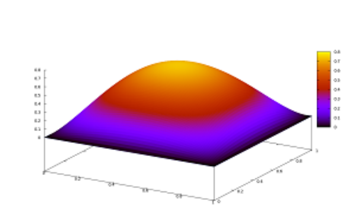


Figure : Displacement of a clampled membrane.

Figure : Siméon Denis Poisson (Wikipedia, 2020).

It can model the displacement u of a clamped membrane to which a constant force f is applied.

Simulating this equation is critical if we are to understand the dynamics of human organs.

Theoretical framework for the Poisson problem

The Poisson problem:

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{1}$$

Theoretical framework for the Poisson problem

The Poisson problem:

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{1}$$

Classic FEM

Find $u_h \in V_h^{(k)}$ (subspace of splines in $H_0^1(\Omega_h)$) such that

$$a_h(u_h, v_h) = l_h(v_h), \;\; \forall\, v_h \in V_h^{(k)},$$

where

$$a_h(u, v) = \int_{\Omega_h} \nabla u \cdot \nabla v\,,$$

$$l_h(v) = \int_{\Omega_h} fv\,.$$

Theoretical framework for the Poisson problem

The Poisson problem:

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{1}$$

Classic FEM

Find $u_h \in V_h^{(k)}$ (subspace of splines in $H_0^1(\Omega_h)$) such that

$$a_h(u_h, v_h) = l_h(v_h), \;\; \forall\, v_h \in V_h^{(k)},$$

where

$$a_h(u, v) = \int_{\Omega_h} \nabla u \cdot \nabla v \,,$$

$$l_h(v) = \int_{\Omega_h} fv \,.$$

ϕ-FEM (Duprez and Lozinski, 2020)

First, write $u_h = \phi_h w_h$. Then find $w_h \in V_h^{(k)}$ such that

$$a_h(w_h, v_h) = l_h(v_h) \text{ for all } v_h \in V_h^{(k)},$$

where

$$a_h(w, v) = \int_{\Omega_h} \nabla(\phi_h w)\cdot\nabla(\phi_h v) - \int_{\partial\Omega_h} \frac{\partial}{\partial n}(\phi_h w)\phi_h v + G_h(w, v) \,,$$

$$l_h(v) = \int_{\Omega_h} f\phi_h v + G_h^{rhs}(v).$$

Theoretical framework for the Poisson problem

The Poisson problem:

$$\begin{cases} -\Delta u = f & \text{in } \Omega \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{1}$$

Classic FEM

Find $u_h \in V_h^{(k)}$ (subspace of splines in $H_0^1(\Omega_h)$) such that

$$a_h(u_h, v_h) = l_h(v_h), \ \ \forall\, v_h \in V_h^{(k)},$$

where

$$a_h(u, v) = \int_{\Omega_h} \nabla u \cdot \nabla v\,,$$

$$l_h(v) = \int_{\Omega_h} fv\,.$$

ϕ-FEM (Duprez and Lozinski, 2020)

First, write $u_h = \phi_h w_h$. Then find $w_h \in V_h^{(k)}$ such that

$$a_h(w_h, v_h) = l_h(v_h) \text{ for all } v_h \in V_h^{(k)},$$

where

$$a_h(w, v) = \int_{\Omega_h} \nabla(\phi_h w)\cdot\nabla(\phi_h v) - \int_{\partial\Omega_h} \frac{\partial}{\partial n}(\phi_h w)\phi_h v + G_h(w, v)\,,$$

$$l_h(v) = \int_{\Omega_h} f\phi_h v + G_h^{rhs}(v).$$

$G_h(w, v)$ and $G_h^{rhs}(w, v)$ are stabilization terms, needed for the well-posedness of the discrete ϕ-FEM system.

$$G_h(w, v) := \sigma h \sum_{E\in\mathcal{F}_h^\Gamma} \int_E \left[\frac{\partial}{\partial n}(\phi_h w)\right]\left[\frac{\partial}{\partial n}(\phi_h v)\right] + \sigma h^2 \sum_{T\in\mathcal{T}_h^\Gamma} \int_T \Delta(\phi_h w)\Delta(\phi_h v)\,,$$

$$G_h^{rhs}(v) := -\sigma h^2 \sum_{T\in\mathcal{T}_h^\Gamma} \int_T f\Delta(\phi_h v)\,.$$

Numerical solution for the Poisson problem

Classic FEM

$$\begin{cases} \Omega = \left\{(x,y) \in \mathbb{R}^2 : \left(x - \frac{1}{2}\right)^2 + \left(y - \frac{1}{2}\right)^2 < \frac{1}{8}\right\} \\ u(x,y) = -\left(\frac{1}{8} - \left(x - \frac{1}{2}\right)^2 - \left(y - \frac{1}{2}\right)^2\right) \exp(x) \sin(2\pi y) \\ f(x,y) = -\frac{\partial^2 u}{\partial x^2}(x,y) - \frac{\partial^2 u}{\partial y^2}(x,y) \end{cases}$$

ϕ-FEM

$$\begin{cases} \mathcal{O} = [0,1] \times [0,1] \\ \phi(x,y) = -\frac{1}{8} + \left(x - \frac{1}{2}\right)^2 + \left(y - \frac{1}{2}\right)^2 \\ u(x,y) = \phi(x,y) \times \exp(x) \times \sin(2\pi y) \\ f(x,y) = -\frac{\partial^2 u}{\partial x^2}(x,y) - \frac{\partial^2 u}{\partial y^2}(x,y) \\ \sigma = 20 \end{cases}$$

Numerical solution for the Poisson problem

Classic FEM

$$\begin{cases} \Omega = \left\{(x,y) \in \mathbb{R}^2 : \left(x-\frac{1}{2}\right)^2 + \left(y-\frac{1}{2}\right)^2 < \frac{1}{8}\right\} \\ u(x,y) = -\left(\frac{1}{8} - \left(x-\frac{1}{2}\right)^2 - \left(y-\frac{1}{2}\right)^2\right)\exp(x)\sin(2\pi y) \\ f(x,y) = -\frac{\partial^2 u}{\partial x^2}(x,y) - \frac{\partial^2 u}{\partial y^2}(x,y) \end{cases}$$

ϕ-FEM

$$\begin{cases} \mathcal{O} = [0,1] \times [0,1] \\ \phi(x,y) = -\frac{1}{8} + \left(x-\frac{1}{2}\right)^2 + \left(y-\frac{1}{2}\right)^2 \\ u(x,y) = \phi(x,y) \times \exp(x) \times \sin(2\pi y) \\ f(x,y) = -\frac{\partial^2 u}{\partial x^2}(x,y) - \frac{\partial^2 u}{\partial y^2}(x,y) \\ \sigma = 20 \end{cases}$$

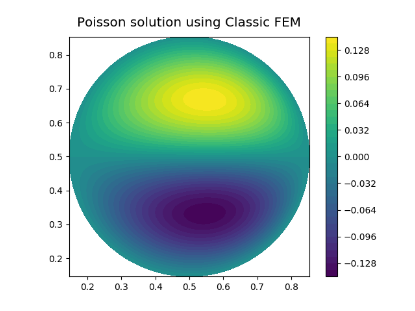


Figure : Classic FEM (39642 cells)

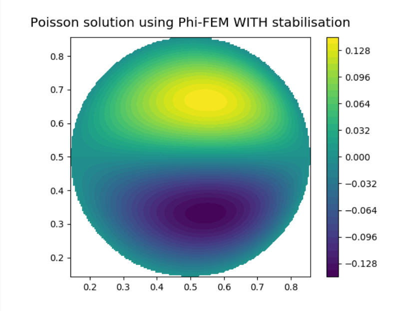


Figure : ϕ-FEM (39936 cells)

Convergence study for the Poisson Problem

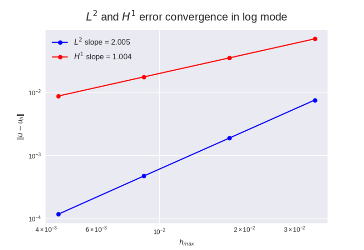


Figure : Classic FEM

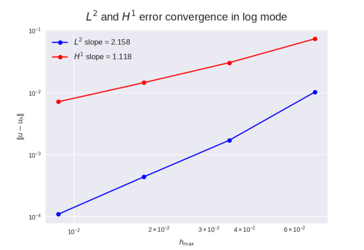


Figure : ϕ-FEM

Problem	Technique	L^2 slope	H^1 slope
Poisson	Classic FEM	2.005	1.004
	ϕ-FEM	2.158	1.118

Table : Convergence rates.

Convergence study for the Poisson Problem

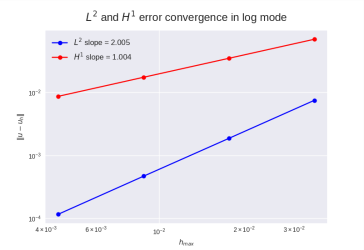


Figure : Classic FEM

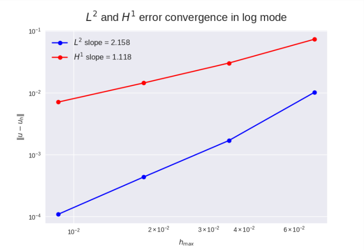


Figure : ϕ-FEM

Problem	Technique	L^2 slope	H^1 slope
Poisson	Classic FEM	2.005	1.004
	ϕ-FEM	2.158	1.118

Table : Convergence rates.

Note: The convergence rates are clearly better with ϕ-FEM (Duprez, 2020).

Results

The elasticity equation

Why use the equations of linear elasticity?

$$\begin{cases} \nabla \cdot \sigma(u) + f = 0 & \text{in } \Omega \\ \sigma(u) = \lambda(\nabla \cdot u)\mathcal{I} + \mu(\nabla u + \nabla u^T) & \text{in } \Omega \\ u = g & \text{on } \partial\Omega \end{cases}$$

Analysis of structures is one of the major activities of modern engineering, which likely makes the PDE modeling the deformation of elastic bodies the most popular PDE in the world (FEniCS, 2020).

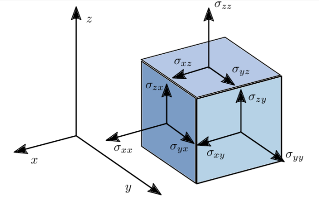


Figure : The nine quantities in the stress tensor $\sigma(\cdot)$ on a representative volume (Semanticscholar, 2020).

Why use the equations of linear elasticity?

$$\begin{cases} \nabla \cdot \sigma(u) + f = 0 & \text{in } \Omega \\ \sigma(u) = \lambda(\nabla \cdot u)\mathcal{I} + \mu(\nabla u + \nabla u^T) & \text{in } \Omega \\ u = g & \text{on } \partial\Omega \end{cases}$$

Analysis of structures is one of the major activities of modern engineering, which likely makes the PDE modeling the deformation of elastic bodies the most popular PDE in the world (FEniCS, 2020).

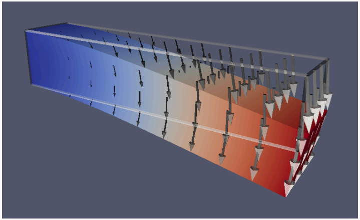


Figure : Gravity-induced deflection in a clamped beam (FEniCS, 2020).

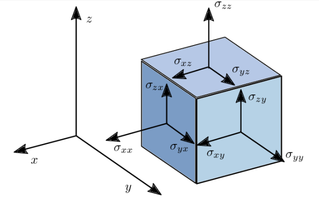


Figure : The nine quantities in the stress tensor $\sigma(\cdot)$ on a representative volume (Semanticscholar, 2020).

It takes into account variables not considered in the Poisson problem; namely the direction of the displacement vector u. Also, if we added the Neumann condition $\sigma(u) \cdot n = g_N$, it would represent a force applied on the boundary.

Theoretical framework for the elasticity equation

The elasticity equation:

$$\begin{cases} \nabla \cdot \sigma(u) + f = 0 & \text{in } \Omega \\ \sigma(u) = \lambda(\nabla \cdot u)\mathcal{I} + \mu(\nabla u + \nabla u^T) & \text{in } \Omega \\ u = g & \text{on } \partial\Omega \end{cases} \tag{2}$$

Theoretical framework for the elasticity equation

The elasticity equation:

$$\begin{cases} \nabla \cdot \sigma(u) + f = 0 & \text{in } \Omega \\ \sigma(u) = \lambda(\nabla \cdot u)\mathcal{I} + \mu(\nabla u + \nabla u^T) & \text{in } \Omega \\ u = g & \text{on } \partial\Omega \end{cases} \tag{2}$$

Classic FEM

First $V = \left[H_0^1(\Omega)\right]^d$, and $\gamma G = g$. Then find $u \in G + V$ such that

$$a(u, v) = l(v), \quad \forall v \in V\,,$$

where a and l are defined as

$$a(u, v) = \int_\Omega \sigma(u) : \varepsilon(v)\,,$$

and

$$l(v) = \int_\Omega f \cdot v\,.$$

Theoretical framework for the elasticity equation

The elasticity equation:

$$\begin{cases} \nabla \cdot \sigma(u) + f = 0 & \text{in } \Omega \\ \sigma(u) = \lambda(\nabla \cdot u)\mathcal{I} + \mu(\nabla u + \nabla u^T) & \text{in } \Omega \\ u = g & \text{on } \partial\Omega \end{cases} \tag{2}$$

Classic FEM

First $V = \left[H_0^1(\Omega)\right]^d$, and $\gamma G = g$. Then find $u \in G + V$ such that

$$a(u, v) = l(v), \quad \forall v \in V\,,$$

where a and l are defined as

$$a(u, v) = \int_\Omega \sigma(u) : \varepsilon(v)\,,$$

and

$$l(v) = \int_\Omega f \cdot v\,.$$

ϕ-FEM (Duprez and Nzoyem, 2020)

First, $u_h = \phi w + g$. Then find $w \in V$ such that:

$$a(w, v) = l(v), \quad \forall v \in V\,,$$

where a and l are defined as

$$a(w, v) = \int_\Omega \sigma(\phi w) : \varepsilon(\phi v) - \int_{\partial\Omega} (\sigma(\phi w) \cdot n) \cdot (\phi v) + G(w, v)\,,$$

and

$$l(v) = \int_\Omega f \cdot (\phi v) + \int_{\partial\Omega} (\sigma(g) \cdot n) \cdot (\phi v) - \int_\Omega \sigma(g) : \varepsilon(\phi v) + G_{rhs}(v)\,.$$

Theoretical framework for the elasticity equation

The elasticity equation:

$$\begin{cases} \nabla\cdot\sigma(u)+f=0 & \text{in } \Omega \\ \sigma(u)=\lambda(\nabla\cdot u)\mathcal{I}+\mu(\nabla u+\nabla u^T) & \text{in } \Omega \\ u=g & \text{on } \partial\Omega \end{cases} \tag{2}$$

Classic FEM

First $V=\left[H_0^1(\Omega)\right]^d$, and $\gamma G=g$. Then find $u\in G+V$ such that

$$a(u,v)=l(v),\quad \forall v\in V\,,$$

where a and l are defined as

$$a(u,v)=\int_\Omega \sigma(u):\varepsilon(v)\,,$$

and

$$l(v)=\int_\Omega f\cdot v\,.$$

ϕ-FEM (Duprez and Nzoyem, 2020)

First, $u_h=\phi w+g$. Then find $w\in V$ such that:

$$a(w,v)=l(v),\quad \forall v\in V\,,$$

where a and l are defined as

$$a(w,v)=\int_\Omega \sigma(\phi w):\varepsilon(\phi v)-\int_{\partial\Omega}(\sigma(\phi w)\cdot n)\cdot(\phi v)+G(w,v)\,,$$

and

$$l(v)=\int_\Omega f\cdot(\phi v)+\int_{\partial\Omega}(\sigma(g)\cdot n)\cdot(\phi v)-\int_\Omega \sigma(g):\varepsilon(\phi v)+G_{rhs}(v)\,.$$

$G(w,v)$ and $G^{rhs}(w,v)$ are stabilization terms, needed for the well-posedness of the discrete ϕ-FEM system.

$$G(w,v)=\sigma_{pen}h^2\sum_{E\in\mathcal{T}_h^\Gamma}\int_E(\nabla\cdot\sigma(\phi w))\cdot(\nabla\cdot\sigma(\phi v))+\sigma_{pen}h\sum_{F\in\mathcal{F}_h^\Gamma}\int_F[\sigma(\phi w)\cdot n]\cdot[\sigma(\phi v)\cdot n]$$

$$G_{rhs}(v)=-\sigma_{pen}h^2\sum_{E\in\mathcal{T}_h^\Gamma}\int_E(f+\nabla\cdot\sigma(g))\cdot(\nabla\cdot\sigma(\phi v))-\sigma_{pen}h\sum_{F\in\mathcal{F}_h^\Gamma}\int_F[\sigma(g)\cdot n]\cdot[\sigma(\phi v)\cdot n]$$

Numerical solution for the elasticity equation

Classic FEM

$$\begin{cases} \Omega = \left\{(x,y) \in \mathbb{R}^2 : \left(x-\frac{1}{2}\right)^2 + \left(y-\frac{1}{2}\right)^2 < \frac{1}{8}\right\} \\ u(x,y) = \begin{pmatrix} 2x + \sin(x)\exp(y) \\ \frac{x}{2} + \cos(x) - 1 \end{pmatrix} \\ f(x,y) = -\nabla \cdot \sigma(u(x,y)) \\ g(x,y) = u(x,y) \end{cases}$$

ϕ-FEM

$$\begin{cases} \mathcal{O} = [0,1] \times [0,1] \\ \phi(x,y) = -\frac{1}{8} + \left(x-\frac{1}{2}\right)^2 + \left(y-\frac{1}{2}\right)^2 \\ u(x,y) = \begin{pmatrix} 2x + \sin(x)\exp(y) \\ \frac{x}{2} + \cos(x) - 1 \end{pmatrix} \\ f(x,y) = -\nabla \cdot \sigma(u(x,y)) \\ g(x,y) = u(x,y) + \phi(x,y) \begin{pmatrix} \sin(x) \\ \exp(y) \end{pmatrix} \\ \sigma_{pen} = 20 \end{cases}$$

Numerical solution for the elasticity equation

Classic FEM

$$
\begin{cases}
\Omega = \left\{(x,y) \in \mathbb{R}^2 : (x-\frac{1}{2})^2 + (y-\frac{1}{2})^2 < \frac{1}{8}\right\} \\
u(x,y) = \begin{pmatrix} 2x + \sin(x)\exp(y) \\ \frac{x}{2} + \cos(x) - 1 \end{pmatrix} \\
f(x,y) = -\nabla \cdot \sigma(u(x,y)) \\
g(x,y) = u(x,y)
\end{cases}
$$

ϕ-FEM

$$
\begin{cases}
\mathcal{O} = [0,1] \times [0,1] \\
\phi(x,y) = -\frac{1}{8} + (x-\frac{1}{2})^2 + (y-\frac{1}{2})^2 \\
u(x,y) = \begin{pmatrix} 2x + \sin(x)\exp(y) \\ \frac{x}{2} + \cos(x) - 1 \end{pmatrix} \\
f(x,y) = -\nabla \cdot \sigma(u(x,y)) \\
g(x,y) = u(x,y) + \phi(x,y) \begin{pmatrix} \sin(x) \\ \exp(y) \end{pmatrix} \\
\sigma_{pen} = 20
\end{cases}
$$

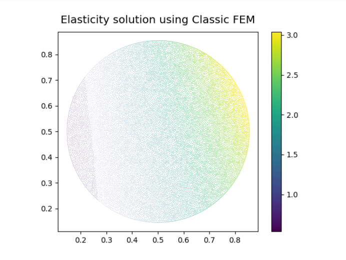


Figure : Classic FEM

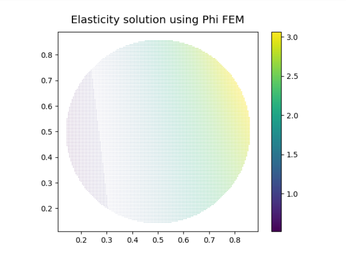
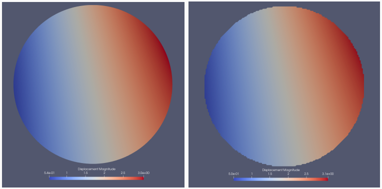


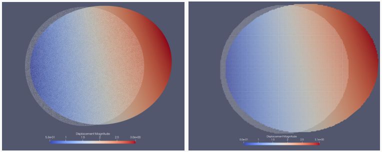
Figure : ϕ-FEM

Numerical solution for the elasticity equation (cont.)



(a) Classic FEM **(b)** ϕ-FEM

Figure : Solution (deformation) magnitude in Paraview.



(a) Classic FEM **(b)** ϕ-FEM

Figure : Solution warped by vector in Paraview.

Convergence study for the elasticity equation

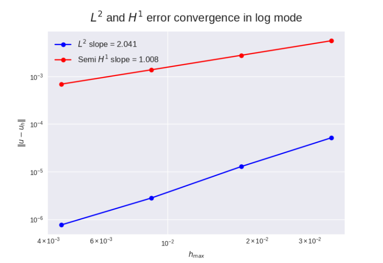


Figure : Classic FEM

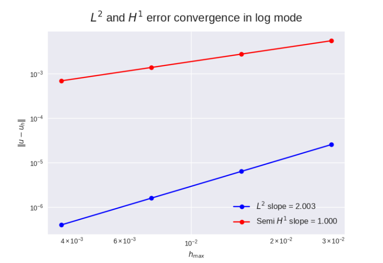


Figure : ϕ-FEM

Problem	Technique	L^2 slope	H^1 slope
Elasticity	Classic FEM	2.041	1.008
	ϕ-FEM	2.003	1.000

Table : Convergence rates.

Convergence study for the elasticity equation

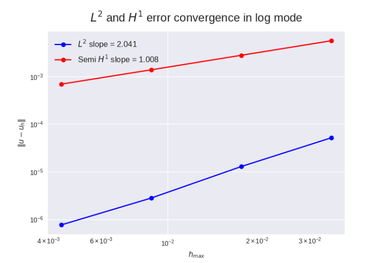


Figure : Classic FEM

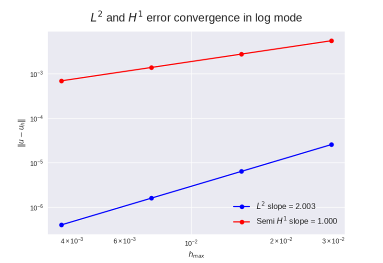


Figure : ϕ-FEM

Problem	Technique	L^2 slope	H^1 slope
Elasticity	Classic FEM	2.041	1.008
	ϕ-FEM	2.003	1.000

Table : Convergence rates.

Note: The convergence rates are almost identical in both cases (Duprez, 2020).

Project summary

Project summary

Work done

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours
- ✓ **Implementing the elasticity equation**: January 19, 2021 : 30 hours

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours
- ✓ **Implementing the elasticity equation**: January 19, 2021 : 30 hours
- × **Running Simulations on organ geometries**: January 19, 2021 : 00 hours

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours
- ✓ **Implementing the elasticity equation**: January 19, 2021 : 30 hours
- × **Running Simulations on organ geometries**: January 19, 2021 : 00 hours

Possible explorations:

- Test the technique on complex geometries.

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours
- ✓ **Implementing the elasticity equation**: January 19, 2021 : 30 hours
- × **Running Simulations on organ geometries**: January 19, 2021 : 00 hours

Possible explorations:

- Test the technique on complex geometries.
- Benchmark the technique for speed efficiency.

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours
- ✓ **Implementing the elasticity equation**: January 19, 2021 : 30 hours
- × **Running Simulations on organ geometries**: January 19, 2021 : 00 hours

Possible explorations:

- Test the technique on complex geometries.
- Benchmark the technique for speed efficiency.
- Deploy the numerical implementation into the SOFA software.

What has been done?

Based on the objectives, the deadlines, and the time estimates we set early in the project:

- ✓ **Understanding ϕ-FEM** : November 3rd, 2020 : 10 hours
- ✓ **Implementing the Poisson problem**: November 10, 2020 : 50 hours
- ✓ **Implementing the elasticity equation**: January 19, 2021 : 30 hours
- × **Running Simulations on organ geometries**: January 19, 2021 : 00 hours

Possible explorations:

- Test the technique on complex geometries.
- Benchmark the technique for speed efficiency.
- Deploy the numerical implementation into the SOFA software.
- Implement the Neumann case (Duprez et al., 2020).

Project summary

Delivered documents

What did I deliver?

As promised:

1. A typewritten report

What did I deliver?

As promised:

1. A typewritten report
2. A Python code base

All are available on this GitHub repository: `https://github.com/master-csmi/2020-m2-mimesis`

References

Burman, Erik and Peter Hansbo (2010). “Fictitious domain finite element methods using cut elements: I. A stabilized Lagrange multiplier method”. In: *Computer Methods in Applied Mechanics and Engineering* 199.41-44, pp. 2680–2686.

Cyclopedia (Nov. 2017). “The Finite Element Method (FEM)”. In: *Cyclopedia*. URL: `https://www.comsol.fr/multiphysics/finite-element-method`.

De Cicco, Davide and Farid Taheri (2018). “Delamination buckling and crack propagation simulations in fiber-metal laminates using xFEM and cohesive elements”. In: *Applied Sciences* 8.12, p. 2440.

Duprez, M. (2020). “Personal notes from M. Duprez on January 27 and 28, 2020”. In.

Duprez, M. and D. R. Nzoyem (2020). “Simulation of soft tissues using innovative non-conforming Finite Elements Methods”. In: pp. 13–15. URL: `https://github.com/master-csmi/2020-m2-mimesis/blob/main/docs/pdfs/report-v2.pdf`.

Duprez, M. et al. (Mar. 2020). “phi-FEM, a finite element method on domains defined by level-sets: the Neumann boundary case”. In: *arXiv*. URL: `https://arxiv.org/abs/2003.11733`.

Duprez, Michel and Alexei Lozinski (2020). “ϕ-FEM: a finite element method on domains defined by level-sets”. In: *SIAM Journal on Numerical Analysis* 58.2, pp. 1008–1028.

FEniCS (2020). “The equations of linear elasticity”. In: URL: `https://fenicsproject.org/pub/tutorial/html/._ftut1008.html`.

Main, Alex and Guglielmo Scovazzi (2018). “The shifted boundary method for embedded domain computations. Part I: Poisson and Stokes problems”. In: *Journal of Computational Physics* 372, pp. 972–995.

Thank you for your kind attention ☺ !
Questions ?