An Introduction to Multi-Scale Anisotropic Mesh Adaptation

Adrien Loseille[1]

[1] INRIA - Gamma Project
Rocquencourt, France

with the collaboration of
F. Alauzet, A. Dervieux, C. Dobrzynski, P. Frey, P.L. George,
M. Merhenberger

Journées Estimateurs d'Erreur
Valenciennes, 2008

Course Outline

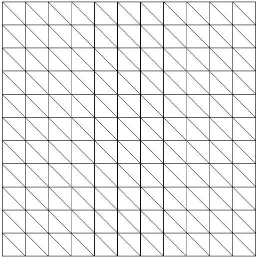
1. Concepts, Motivations and Examples
2. Introduction to Anisotropic Mesh Adaptation
3. Multi-scale mesh adaptation
4. Introduction to Goal Oriented Mesh Adaptation
5. Application dans un contexte industriel

Course Outline

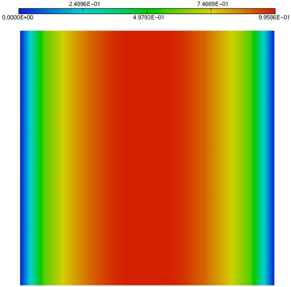
A simple 2D example

We have:

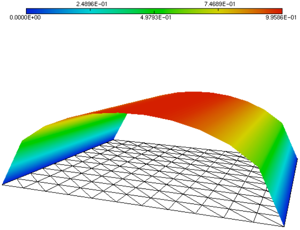
1. a mesh $\mathcal{H}_1$ of $\Omega = [-1,1] \times [-1,1]$
   with $|\mathcal{H}_1| = N = 144$ vertices
2. the function (half-cylinder) $f(x,y) = \sqrt{1 - x^2}$



$\mathcal{H}_1$



Iso-values



$\Pi_h f$ of f on $\mathcal{H}_1$

A simple 2D example

We define the interpolation error by $e = \|f - \Pi_h f\|_{\mathbf{L}^p,\Omega}$

For instance, for $\mathcal{H}_1$:

$\mathbf{L}^1$	$\mathbf{L}^2$	$\mathbf{L}^\infty$
0.029	0.059	0.133

Problematic

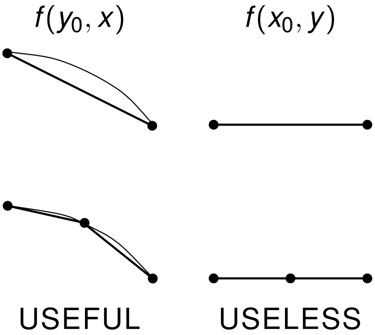
How to reduce the interpolation error with the same number of vertices ?

$$\left\{ \begin{array}{l} \text{Find } \mathcal{H} = \text{Argmin}\|f - \Pi_h f\|_{\mathbf{L}^p} \\ |\mathcal{H}| = N \end{array} \right.$$

A simple 2D example

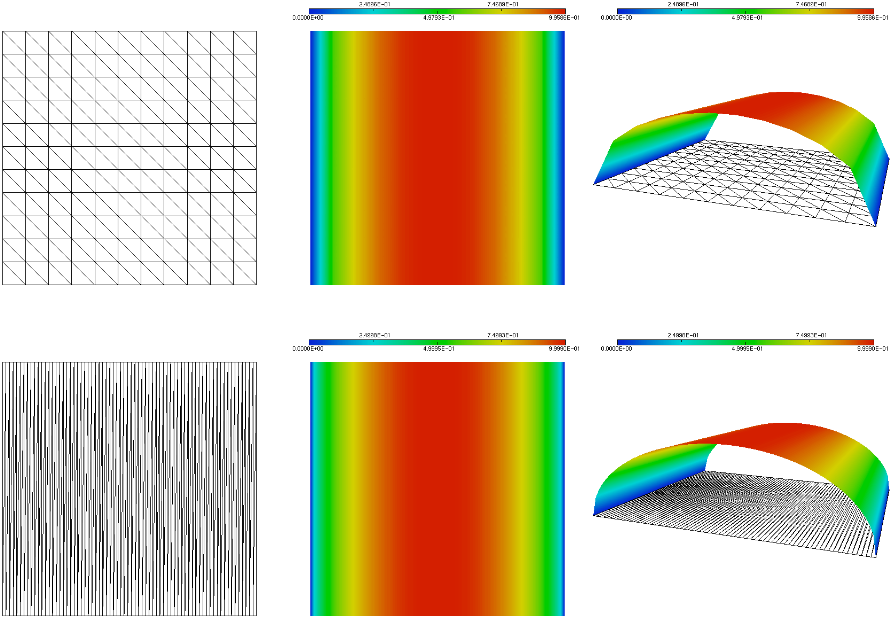
Two local remarks:

1. the largest curvature/variation is in x-direction
2. the function f does not depend on y



$\Longrightarrow$ All vertices on the lines $x = 1$ and $x = -1$

A simple 2D example



A simple 2D example

Interpolation error on both meshes:

L^1	L^2	L^∞	Mesh
0.029	0.059	0.133	$[-1:(2/11):1] \times [-1:(2/11):1]$
0.008	0.005	0.014	$[-1:(2/72):1] \times [-1:2:1]$

$\Longrightarrow$ Manual use of a local information
the anisotropy
of f to improve its representation

How to get an automatic process?

1. using an automatic adaptive mesh generator.
   But how to communicate with it ?
2. measure or quantify this anisotropy to translate it in a comprehensible and usable data for the mesh generator

A simple 2D example

To obtain the second mesh, we want to prescribe at each vertex:

- the largest admissible size in y-direction
- a size in x-direction such that we get the desired number of vertices

$\Longrightarrow$ Replace the Euclidean space $(\mathbb{R}^2, (\ ,\ ))$ by a metric space $(\mathbb{R}^2, (\ ,\ )_{\mathcal{M}})$
$\mathcal{M}$ is a positive definite symmetric matrix

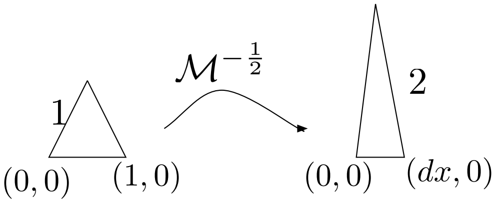
BEFORE	AFTER
$\lVert\overrightarrow{AB}\rVert^2 = {}^t\overrightarrow{AB}\,\overrightarrow{AB}$	$\lVert\overrightarrow{AB}\rVert^2_{\mathcal{M}} = {}^t\overrightarrow{AB}\,\mathcal{M}\,\overrightarrow{AB}$

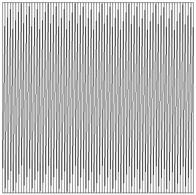
A simple 2D example

To generate $\mathcal{H}_2$, mesh generator works in $([-1,1]\times[-1,1],\mathcal{M})$

$$\mathcal{M} = \begin{pmatrix} \frac{1}{dx^2} & 0 \\ 0 & \frac{1}{2^2} \end{pmatrix}$$

$$\begin{cases} \|{}^t(x,0)\|^2_{\mathcal{M}} = x^2/dx^2 & = \color{red}{1} \text{ if } x = dx \\ \|{}^t(0,y)\|_{\mathcal{M}} = y^2/2^2 & = \color{red}{1} \text{ if } y = 2 \end{cases}$$

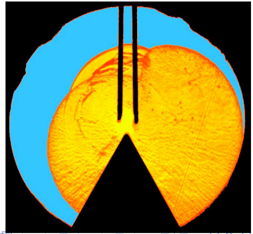




Motivations in CFD

In reality we are confronted with:

- 3D problems with complex geometries
- Unsteady flows



In reality we are confronted with:

- 3D problems
- Complex geometries
- Unsteady flows

$\Longrightarrow$ The problem solution is *a priori* unknown

$\Longrightarrow$ The simulation requires a large number of degrees of freedom

$\Downarrow$

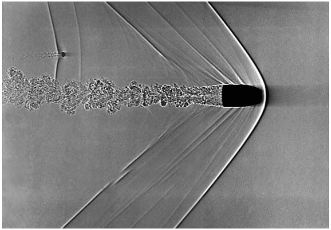
Development of methods in order to reduce the complexity

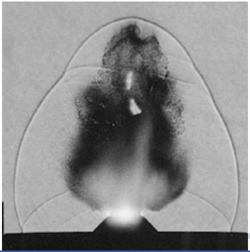
one among them: mesh adaptation

Why Mesh Adaptation in CFD ?

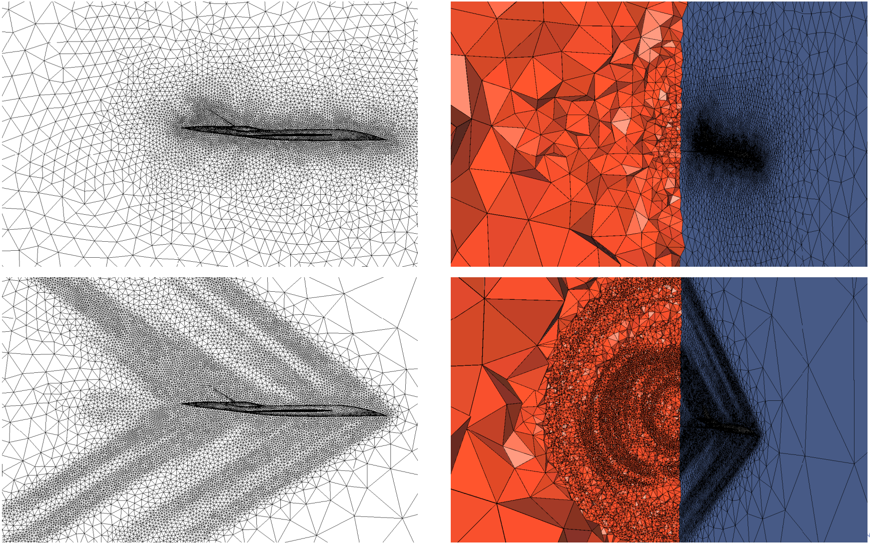
Flow characteristics

- Phenomena are concentrated in small regions of the computational domain
  - ⇝ uniform meshes are not optimal in term of sizes
- Physical phenomena are anisotropic: shock waves, boundary layers, ...
  - ⇝ uniform meshes are not optimal in term of directions
- These regions are moving if the flow is unsteady
  - ⇝ require an uniformly fine mesh in all evolution regions

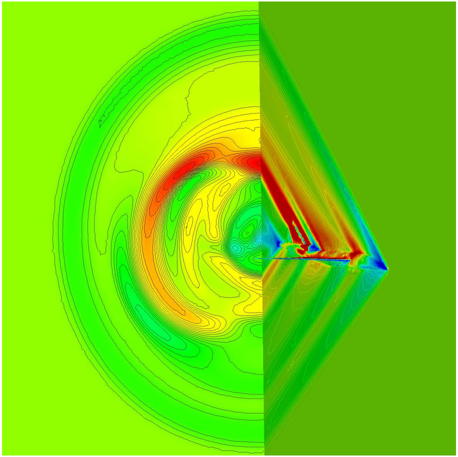


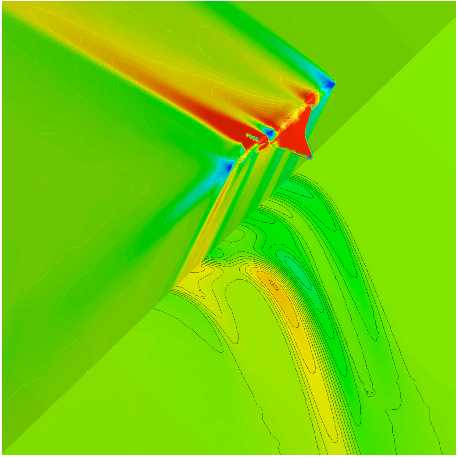


Studying the flow arround a plane to reduce the sonic boom



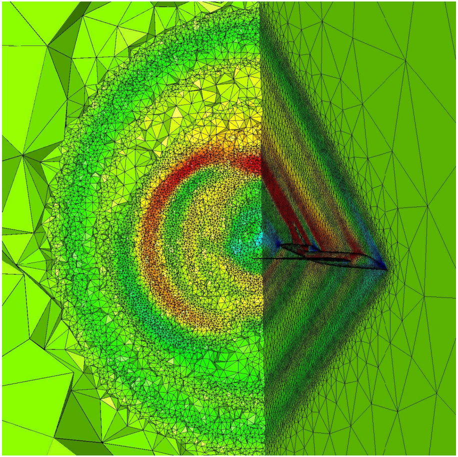
Studying the flow arround a plane to reduce the sonic boom

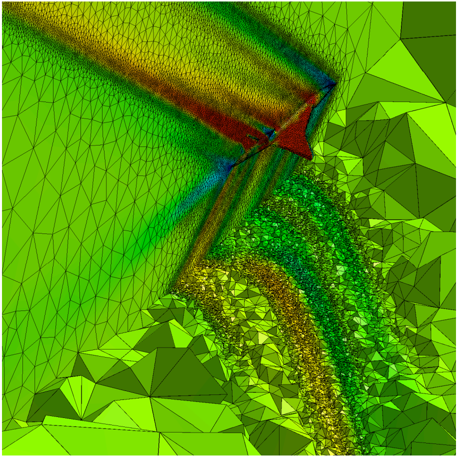




Aeronautic Simulation: SuperSonic Business Jet

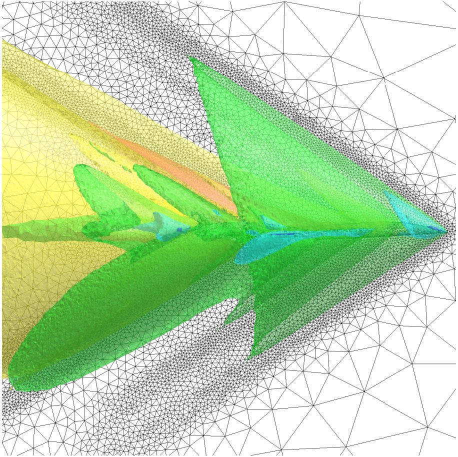
Studying the flow arround a plane to reduce the sonic boom

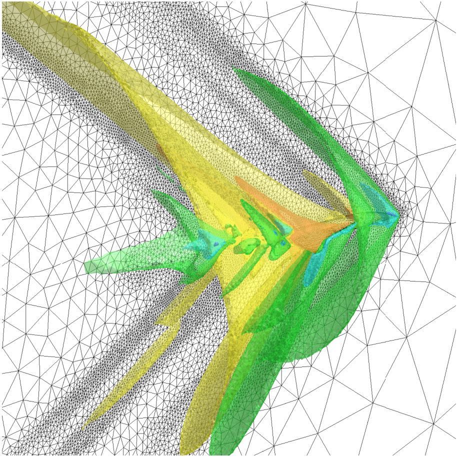




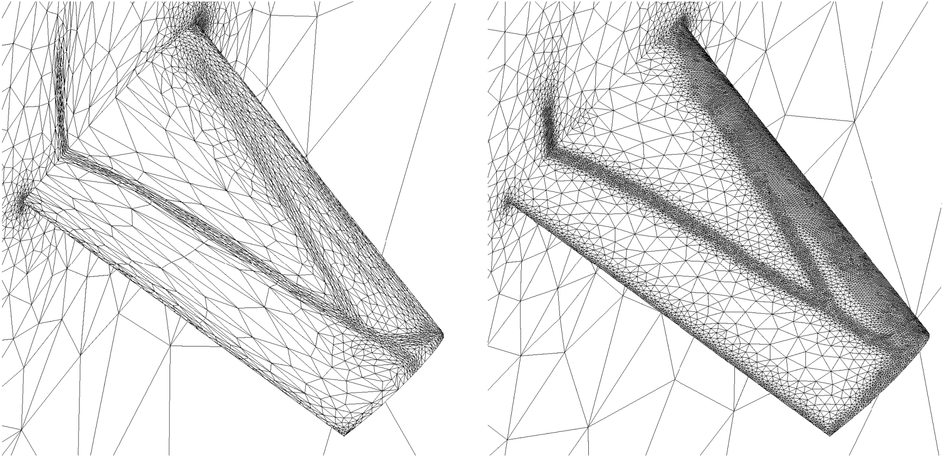
Aeronautic Simulation: SuperSonic Business Jet

Studying the flow arround a plane to reduce the sonic boom

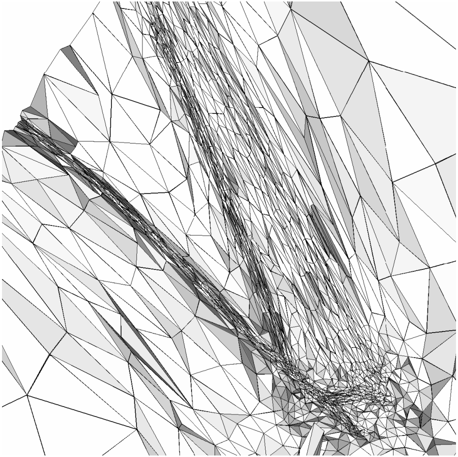


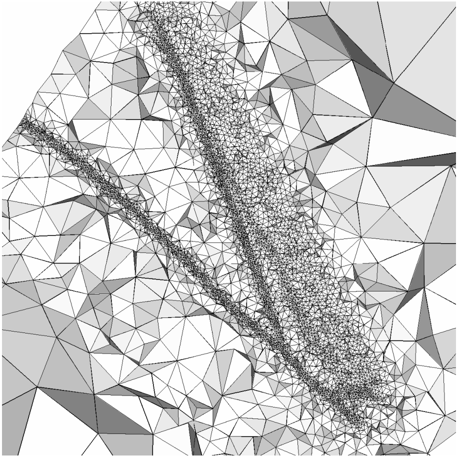


Anisotropic vs. isotropic mesh adaptation

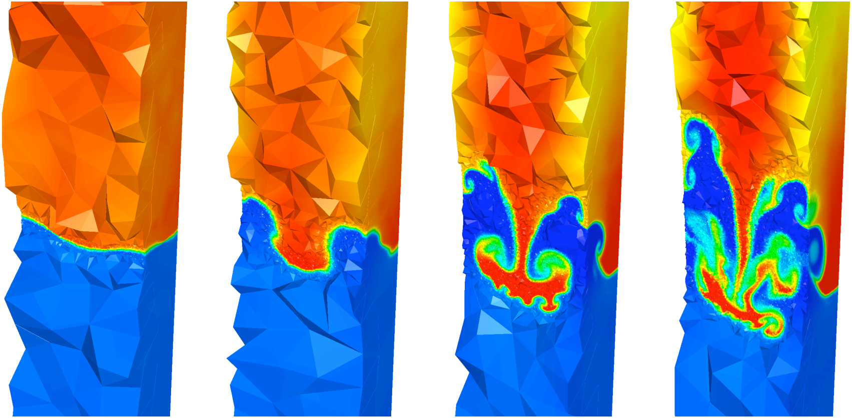


Anisotropic vs. isotropic mesh adaptation



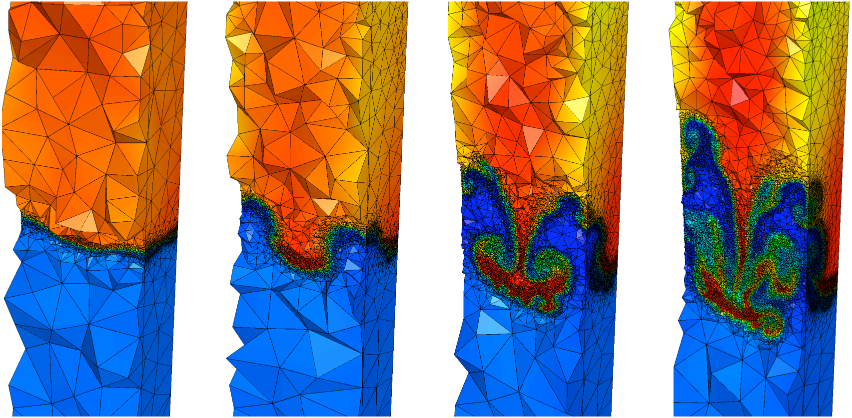


Rayleigh-Taylor Instabilities



$t = 0.2$ $t = 0.8$ $t = 1.4$ $t = 1.8$

Rayleigh-Taylor Instabilities



$t = 0.2$	$t = 0.8$	$t = 1.4$	$t = 1.8$
$nv = 82\,360$	$nv = 185\,905$	$nv = 526\,888$	$nv = 769\,719$
$nt = 477\,324$	$nt = 1\,093\,252$	$nt = 3\,109\,740$	$nt = 4\,536\,589$

A Brief History

1987 J. Peraire et al.

First attempt in 2D : Error measures with directions

They pointed out the directional property of the interpolation error and the idea to generate element with aspect ratio

Local mapping procedure to generate elongated elements

They coupled this with the advancing front technique to generate slightly anisotropic meshes

A Brief History

1987 **J. Peraire et al.**
First attempt in 2D : Error measures with directions

Similar approach have been considered:
R. Löhner 89, **V. Selmin and L. Formaggia** 92

First attempt in 3D. Already !

- **R. Lohner** 90. Anisotropic remeshing coupled with moving bodies!!
- **J. Peraire et al** 92. Results are isotropic.

In 1994, **O. Zienkiewicz and J. Wu.** gave a status.
Even if they had great success with such approach, they said:

"Unfortunately the amount of elongation which can be used in a typical mesh generation by such mapping is small..."

A Brief History

1987 **J. Peraire et al.**
First attempt in 2D : Error measures with directions

1990 **D. Mavriplis.**
Generate stretched elements with a Delaunay approach in 2D

Necessity to obtain very high-aspect-ration triangles in the boundary layer and wake regions.

A Delaunay triangulation is performed in a locally stretched space

A Brief History

1987 **J. Peraire et al.**
First attempt in 2D : Error measures with directions

1990 **D. Mavriplis.**
Generate stretched elements with a Delaunay approach in 2D

1990 **M.G. Vallet and F. Hecht**
Introduce the use of metrics in a Delaunay mesh generator

Generalisation of the previous ideas

Absolute value of the Hessian is a metric

A Delaunay triangulation, edge length computed in the Riemannian metric space

A Brief History

1987 **J. Peraire et al.**
First attempt in 2D : Error measures with directions

1990 **D. Mavriplis.**
Generate stretched elements with a Delaunay approach in 2D

1990 **M.G. Vallet and F. Hecht**
Introduce the use of metrics in a Delaunay mesh generator

1991 **E. D'Azevedo and R.B. Simpson.**
Theoretical analysis of optimal element and metric

Optimality and existence of a metric representing a mesh

A Brief History

1987 **J. Peraire et al.**
First attempt in 2D : Error measures with directions

1990 **D. Mavriplis.**
Generate stretched elements with a Delaunay approach in 2D

1990 **M.G. Vallet and F. Hecht**
Introduce the use of metrics in a Delaunay mesh generator

1991 **E. D'Azevedo and R.B. Simpson.**
Theoretical analysis of optimal element and metric

The fruitful idea of metric was widely exploited for 2D anisotropic mesh adaptation in the 90's and even more today. For instance:

M. Fortin et al. 96, **M. Castro-Diaz et al.** 97, **F. Hecht et al.** 97, **J. Dompierre et al.** 97, **G. Bascaglia and E. Dari** 97, ...

In 2000 first results with "really" 3D anisotropic mesh adaptation for appli.:

A. Tam et al. 00, **C. Pain et al.** 01, **C. Bottasso** 04, **Belhamadia et al.** 04, **X. Li et al.** 05, **C. Gruau and T. Coupez** 05, **P. Frey and F. Alauzet** 05, **C. Dobrzynski et al.** 05, **J. Majewski et al.** 06, **W. Jones et al.** 06.

A Brief History

1987 **J. Peraire et al.**
First attempt in 2D : Error measures with directions

1990 **D. Mavriplis.**
Generate stretched elements with a Delaunay approach in 2D

1990 **M.G. Vallet and F. Hecht**
Introduce the use of metrics in a Delaunay mesh generator

1991 **E. D'Azevedo and R.B. Simpson.**
Theoretical analysis of optimal element and metric

2001 **A. Dervieux. et al.**
Formal calculus on metrics to solve an optimization problem

Outline

Redefining Dot Product

- Euclidean space:

$$\langle \vec{u}\,,\ \vec{v}\rangle = {}^t\vec{u}\ \vec{v} \quad \Longrightarrow \quad \ell(a,b) = \sqrt{{}^t\overrightarrow{ab}\ \overrightarrow{ab}}$$

- Euclidean metric space:
  $\mathcal{M}$: $d \times d$ symmetric definite positive matrix

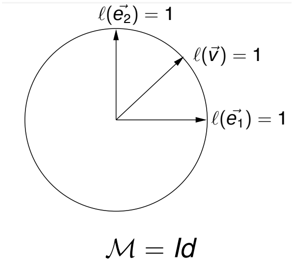
$$\langle \vec{u}\,,\ \vec{v}\rangle_{\mathcal{M}} = {}^t\vec{u}\mathcal{M}\vec{v} \ \Longrightarrow\ \ell_{\mathcal{M}}(a,b) = \sqrt{{}^t\overrightarrow{ab}\ \mathcal{M}\ \overrightarrow{ab}}$$

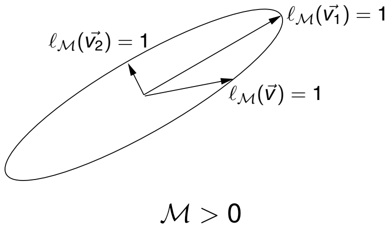
- Riemannian metric space:
  $(x, \mathcal{M}(x))_{x\in\Omega}$

$$\ell_{\mathcal{M}}(a,b) = \int_0^1 \sqrt{{}^t\overrightarrow{ab}\ \mathcal{M}(a + t\overrightarrow{ab})\ \overrightarrow{ab}}\,\mathrm{d}t$$

- Unit ball:

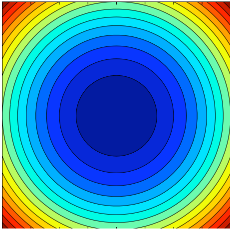
$$\mathcal{E}_{\mathcal{M}(P)} = \left\{ M \,|\, \sqrt{{}^t\overrightarrow{PM}\,\mathcal{M}(P)\,\overrightarrow{PM}} = 1 \right\}$$



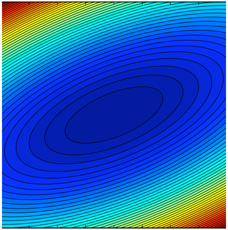


- $\mathcal{M}$ curves space by modifying distance definition

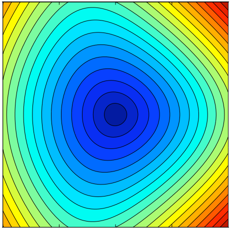
Geometric meaning and examples



l_2

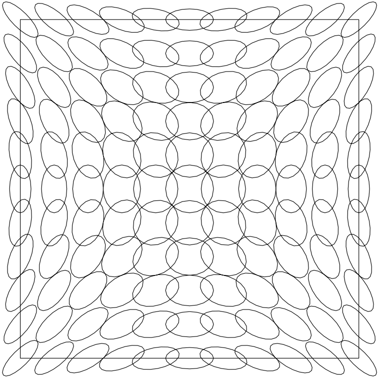


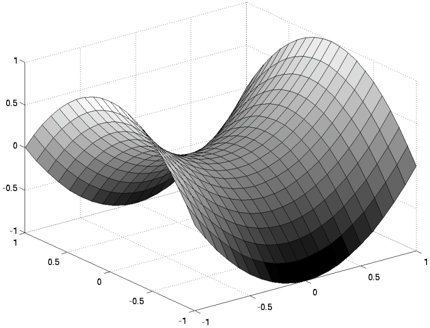
$\mathcal{M}$



$\mathcal{M}(.)$

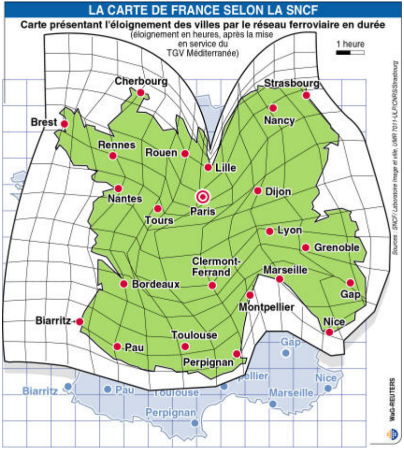
Geometric meaning and examples





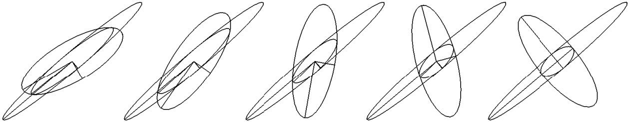
$$\Omega = \{(x, y) \in \mathbb{R}^2 \mid 1 \leq x, y \leq 1\}$$
$$\mathcal{M}(x, y) \Longleftrightarrow z = x^2 - y^2$$

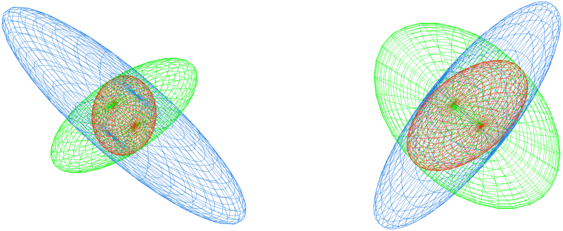
Geometric meaning and examples



Metric intersection: $\mathcal{M}_{1\cap 2} = \mathcal{M}_1 \cap \mathcal{M}_2$

- Strongest constraint in size in each direction is considered
- Resulting metric is given by simultaneous reduction

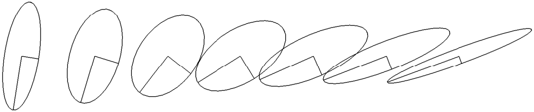


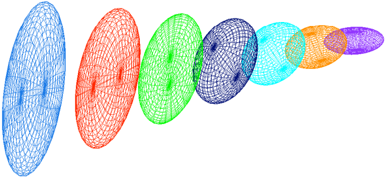


Metric interpolation:

$$\mathbf{x} = \sum_{i=1}^{k} \alpha_i \, \mathbf{x}_i, \text{ avec } \sum_{i=1}^{k} \alpha_i = 1,$$

$$\mathcal{M}(\mathbf{x}) = \exp\left(\sum_{i=1}^{k} \alpha_i \ln(\mathcal{M}(\mathbf{x}_i))\right)$$





Mesh Adaptation Concept

- Main idea:

  Modify discretization of Ω to control solution's accuracy

- **Fundamental concept: Unit mesh**

To adapt $\Longleftrightarrow$ Prescribe h in each direction at each point

$\Updownarrow$ Work in adequate Riemannian/metric space

Prescribe a metric $\mathcal{M}$ at each point

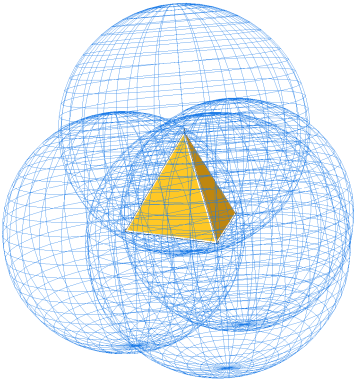
Mesh Adaptation Concept

$\mathcal{H}$ unit mesh $\iff \forall \vec{e},\ \ell_{\mathcal{M}}(\vec{e}) = 1$ and $\forall K,\ |K|_{\mathcal{M}} = \begin{cases} \sqrt{3}/4 & \text{in 2D} \\ \sqrt{2}/12 & \text{in 3D} \end{cases}$

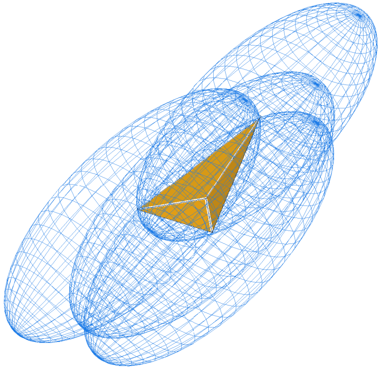
- Regular elements with respect to a metric

elemmetrictiny.mp4

Mesh Adaptation Concept



I_3

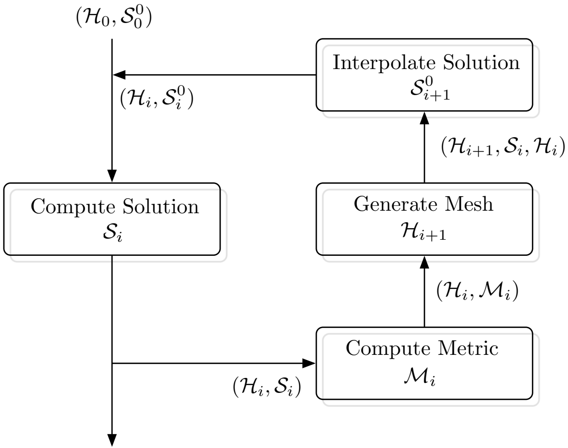


$\mathcal{M}$

Mesh Adaptation Algorithm

Mesh adaptation is a non-linear problem

$\Longrightarrow$ an iterative process is required to converge the couple mesh-solution



Example of Metric Construction

- **Aim:**
  equidistribute the interpolation error on the mesh to control the accuracy of the solution

- **Principle:**
  look at the solution such as a cartesian surface

  This approach is geometric $\Longrightarrow$ it's independent of the problem

Theorem:

Let K be a triangle of the mesh of the studied domain.

Let u be a two times differentiable function.

Then, we have the following bound of the interpolation error in $\mathbf{L}^\infty$ norm:

$$\begin{aligned} \| \, u - \Pi_h u \, \|_{\infty,K} &\leq c_d \max_{x,y,z \in K} \langle \overrightarrow{yz} \, , \, |H_u(x)| \, \overrightarrow{yz} \, \rangle \\ &\leq c_d \max_{x \in K} \max_{e \in E_K} \langle \vec{e} \, , \, |H_u(x)| \, \vec{e} \, \rangle \end{aligned}$$

where $c_d = \dfrac{1}{2}\left(\dfrac{d}{d+1}\right)^2$ and $|H_u| = \mathcal{R} \, |\Lambda| \, \mathcal{R}^{-1}$ with $|\Lambda| = diag(|\lambda_i|)$

Proof:

From a Taylor expension with integral rest of $e = u - \Pi_h u$ at a vertex a of K, for all vertex x inside K we have:

$$\begin{aligned}(u - \Pi_h u)(a) &= (u - \Pi_h u)(x) + \langle \overrightarrow{ax}\,, \nabla(u - \Pi_h u)(x)\rangle \\ &+ \int_0^1 (1-t)\,\langle \overrightarrow{ax}\,, H_u(x + t\overrightarrow{xa})\,\overrightarrow{ax}\rangle\,dt\,.\end{aligned}$$

It exists a real $\lambda \leq 2/3$ such that $\overrightarrow{ax} = \lambda \overrightarrow{aa'}$, thus:

$$\begin{aligned}|e(x)| &= \left|\int_0^1 (1-t)\,\lambda^2\,\langle \overrightarrow{aa'}\,, H_u(a + t\overrightarrow{xa})\,\overrightarrow{aa'}\rangle\,dt\right|, \\ &\leq \frac{4}{9}\max_{y\in aa'} |\langle \overrightarrow{aa'}\,, H_u(y)\,\overrightarrow{aa'}\rangle|\,\left|\int_0^1 (1-t)\,dt\right|.\end{aligned}$$

We finally find:

$$\|u - \Pi_h u\|_{\infty,K} \leq \frac{2}{9}\max_{y\in K} \langle \overrightarrow{aa'}\,, |H_u(y)|\,\overrightarrow{aa'}\rangle\,.$$

Example of Metric Construction

- Let $\overline{\mathcal{M}}(K)$ be an adequate metric tensor. Then, we estimate on each element K the error by:

$$\varepsilon_K = c_d \max_{e \in E_K} \langle \vec{e}\,, \overline{\mathcal{M}}(K)\, \vec{e}\,\rangle\,.$$

- A metric providing a unit length !

If we put: $\mathcal{M}(K) = \dfrac{c_d\, \overline{\mathcal{M}}(K)}{\varepsilon_K} \implies \max_{e \in E_K} \langle \vec{e}\,, \mathcal{M}(K)\, \vec{e}\,\rangle = 1\,.$

- Thus, to equidistribute the interpolation error, a unit mesh $\mathcal{H}$ is generated in the metric $\mathcal{M}$:

$$l_{\mathcal{M}}(\vec{e}_i) = \sqrt{{}^t\vec{e}_i\, \mathcal{M}\, \vec{e}_i} = 1 \quad \text{for each edge} \quad \vec{e}_i \in \mathcal{H}\,.$$

Remark: we bring us to a local analysis on each edge.

Contribution of Mesh Adaptation

It is well known that mesh adaptation

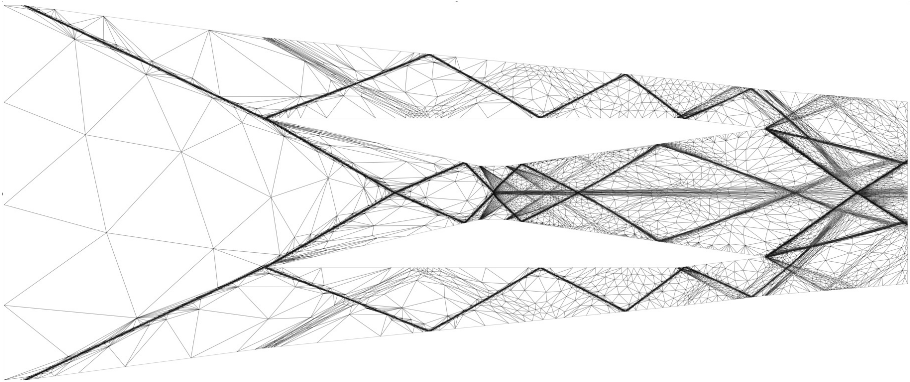
- Improve the ratio between solution accuracy and the inverse of the number of dof
  - gain in CPU, memory and storage
  - visualization facilitated
- Error estimate detects physical phenomena and captures their behavior
- Mesh adaptation is automatic

Outline

Is there anisotropy ?

A supersonic flow inside a Scramjet reactor.

Anisotropic result in 1997:

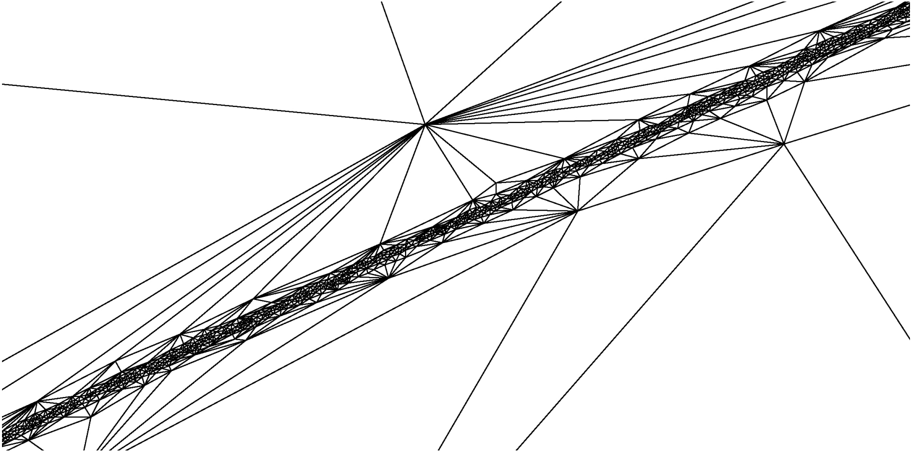


$\approx 90,000$ vertices

Is there anisotropy ?

A supersonic flow inside a Scramjet reactor.

Anisotropic result in 1997: a zoom



Is there anisotropy ?

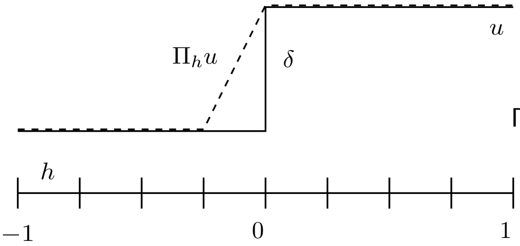
A supersonic flow inside a Scramjet reactor.

Due to

- Flow solvers oscillations in shocks if not TVD
- The use of L^∞ norm in the error estimate

A very simple example in 1D

Is there anisotropy ?



$$\Pi_h u(x) = \begin{cases} 0 & \text{if} \quad x \in [-1, 0] \\ \frac{\delta}{h} x & \text{if} \quad x \in [0, h] \\ \delta & \text{if} \quad x \in [h, 1]\,. \end{cases}$$

$$\int_0^h \left| \delta(1 - \frac{x}{h}) \right|^p \mathrm{d}x = \frac{h\delta^p}{p+1} \quad \Longrightarrow \quad \| u - \Pi_h u \|_{[0,h],\mathbf{L}^p} = \delta \left(\frac{h}{p+1} \right)^{\frac{1}{p}}$$

$$\Longrightarrow \quad \text{Convergence in } \mathcal{O}(\frac{1}{p})$$

$$\| u - \Pi_h u \|_{[-1,1],\mathbf{L}^\infty} = \delta \quad \Longrightarrow \quad \text{No convergence} \mathcal{O}(1)$$

Is there anisotropy ?

A supersonic flow inside a Scramjet reactor.

Due to

- Flow solvers oscillations in shocks if not TVD
- The use of L^∞ norm in the error estimate

$\Longrightarrow$ saved by h_{min}

The non convergence for shocks leads us to work on

1. Continuous mesh model
2. Interpolation error control in $\mathbf{L}^p$ norm

Continuous Element Model

Definition

Let $(\mathbb{R}^n, \mathcal{M})$ an euclidean metric space, the unit ball $\mathcal{B}_{\mathcal{M}}$ is called continuous unit element, in short continuous element. We will denote it by $\mathcal{M}$ when there will be no confusion.

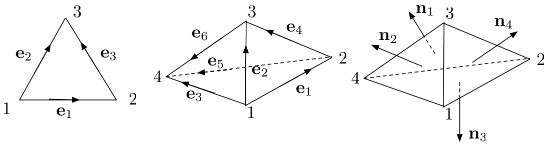
This definition provides sizes and orientation for the continuous element.

Definition

A discrete element K is unit for an euclidean metric space $(\mathbb{R}^n, \mathcal{M})$ if: $\quad \forall i = 1, \dots, \dfrac{n(n+1)}{2}, \quad \ell_{\mathcal{M}}(\vec{e}_i) = 1\,.$

The volume is given by: $|K|_{\mathcal{M}} = \dfrac{\sqrt{n+1}}{2^{n/2}\, n!}\,.$

Continuous Element Model



- if $|K|$ is the Euclidean volume of K:

$$|K|\sqrt{\det(\mathcal{M})} = \frac{\sqrt{2}}{12},$$

- if $(\mathbf{e})_{i=1...6}$ is the edges list of K and H a symmetric matrix:

$$\sum_{i=1}^{6} {}^t\mathbf{e}_i H \mathbf{e}_i = 2\,\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}}),$$

- if $\{\mathbf{n}_i\}_{i=1...4}$ are outward normals of faces of K:

$$\sum_{i=1}^{4} {}^t\mathbf{n}_i H \mathbf{n}_i = \frac{\mathrm{trace}(\mathcal{M}^{\frac{1}{2}} H \mathcal{M}^{\frac{1}{2}})}{\det(\mathcal{M})}.$$

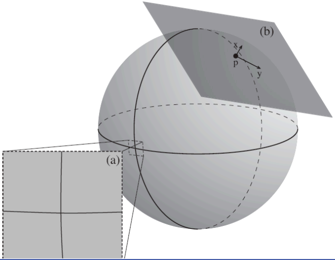
Continuous Mesh Model

Definition

A continuous mesh $\mathcal{M}(.)$ defined on Ω is the riemannian metric space $(\mathbf{x}, \mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$.

Analogy with the differential geometry:

DG: a riemannian metric space $(\mathbf{x}, \mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is defined by "patching together" its tangent spaces $T_{\mathbf{x}}\Omega$ (at each point $\mathbf{x}$) supplied with a metric $\mathcal{M}(\mathbf{x})$ which are euclidean metric spaces.



Continuous Mesh Model

Definition

A continuous mesh $\mathcal{M}(.)$ defined on Ω is the riemannian metric space $(\mathbf{x}, \mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$.

Analogy with the differential geometry:

DG: a riemannian metric space $(\mathbf{x}, \mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ is defined by "patching together" its tangent spaces $T_{\mathbf{x}}\Omega$ (at each point $\mathbf{x}$) supplied with a metric $\mathcal{M}(\mathbf{x})$ which are euclidean metric spaces.

CM: Locally, continuous elements are modeled by euclidean metric spaces (the unit ball). The continuous mesh is the gathering of all the continuous elements.

Continuous Mesh Model

Proposition

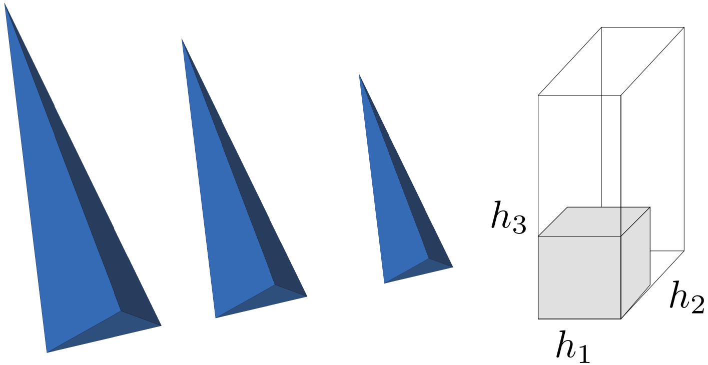
A continuous mesh $\mathcal{M}(.)$ is a function composed of the following local parameters:

- A density: $d = \left(\prod_{k=1}^{n} h_k\right)^{-1} = \left(\prod_{k=1}^{n} \lambda_k\right)^{\frac{1}{2}}$,
- n anisotropic quotients: $r_i = h_i \left(\prod_{k=1}^{n} h_k\right)^{-\frac{1}{n}}$.

Locally, the metric $\mathcal{M}(\mathbf{a})$ is reformulated as:

$$\mathcal{M}(\mathbf{a}) = d^{\frac{2}{n}}(\mathbf{a})\,\mathcal{R}(\mathbf{a}) \begin{pmatrix} r_1^{-2}(\mathbf{a}) & & \\ & \ddots & \\ & & r_n^{-2}(\mathbf{a}) \end{pmatrix} {}^t\mathcal{R}(\mathbf{a}).$$

Continuous Mesh Model



Continuous Mesh Model

Proposition

A continuous mesh $\mathcal{M}(.)$ is a function composed of the following local parameters:

- A density: $d = \left(\prod_{k=1}^{n} h_k\right)^{-1} = \left(\prod_{k=1}^{n} \lambda_k\right)^{\frac{1}{2}}$,
- n anisotropic quotients: $r_i = h_i \left(\prod_{k=1}^{n} h_k\right)^{-\frac{1}{n}}$.

The complexity $\mathcal{C}$ of the continuous mesh is defined by:

$$\mathcal{C}(\mathcal{M}) = \int_{\Omega} d(\mathbf{x})\, d\mathbf{x} = \int_{\Omega} \sqrt{\det(\mathcal{M}(\mathbf{x}))}\, d\mathbf{x}.$$

Continuous Mesh Model

Embedded continuous mesh:

Mother continuous mesh is defined by:

$$\mathcal{M}_1 = \frac{\mathcal{M}}{\int_\Omega \sqrt{\det(\mathcal{M}(\mathbf{x}))}\,\mathrm{d}\mathbf{x}}$$

A family of continuous mesh indexed by the complexity N, $(\mathcal{M}_N)_{N=1\ldots\infty}$, that behaves similarly is deduced:

$$\mathcal{M}_N = N\,\mathcal{M}_1 = \frac{N}{\int_\Omega \sqrt{\det(\mathcal{M}(\mathbf{x}))}\,\mathrm{d}\mathbf{x}}\,\mathcal{M}$$

$\Longrightarrow$ asymptotic analysis could be performed

$\Longrightarrow$ convergence study for continuous meshes

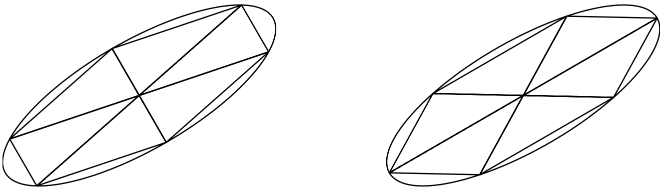
Unit Mesh

Weaker definition

A simplicial mesh $\mathcal{H}$ is unit for a continuous mesh $\mathcal{M}(.)$ if all its discrete elements K are quasi-unit.

Is there unicity ?

Obviously no.



Unit Mesh

Weaker definition

A simplicial mesh $\mathcal{H}$ is unit for a continuous mesh $\mathcal{M}(.)$ if all its discrete elements K are quasi-unit.

Two unit meshes of complexity N are equivalent if:

$$\mathcal{H}_1 \mathbf{R}_{\mathcal{M}} \mathcal{H}_2 \Longleftrightarrow \left\{ \begin{array}{l} \mathcal{H}_1 \text{ is unit for } \mathcal{M}_N \\ \mathcal{H}_2 \text{ is unit for } \mathcal{M}_N, \end{array} \right.$$

They represent different discrete projections of the continuous mesh $\mathcal{M}(.)$. For instance, different mesh generators.

Minimizing the Interpolation Error in $\mathbf{L}^p$-norm

An ill-posed problem

Find $\mathcal{H}_{opt}$ having N vertices such that

$$\mathcal{H}_{opt}(u) = \mathrm{Arg}\min_{\mathcal{H}} \|u - \Pi_h u\|_{\mathcal{H},\mathbf{L}^p(\Omega)}$$

$$\Downarrow$$

A well-posed problem

Find $\mathcal{M}_{opt}$ of complexity N such that

$$\mathcal{E}_{\mathcal{M}_{opt}}(u) = \min_{\mathcal{M}} \|u - \pi_{\mathcal{M}} u\|_{\mathcal{M},\mathbf{L}^p(\Omega)}$$

Given a continuous mesh $(\mathbf{x}, \mathcal{M}(\mathbf{x}))_{\mathbf{x}\in\Omega}$ and a function u on Ω

Goal: Propose consistent definitions of

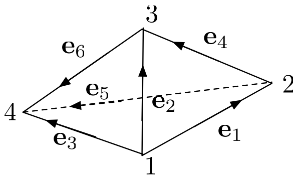
a continuous interpolation error $\|u - \pi_{\mathcal{M}} u\|_{\mathbf{L}^1(\Omega)}$

Roadmap:

1. Classical interpolation error estimates in $\mathbf{L}^p$ norms
2. Local study : u is quadratic and $\mathcal{M}(\mathbf{x}) = \mathcal{M}_0$
3. Asymptotic analysis: u and $\mathcal{M}$ are varying smoothly in Ω

Continuous interpolation

1. Classical interpolation error estimates in $\mathbf{L}^p$ norms:



- u is quadratic: $u(\mathbf{x}) = \frac{1}{2}\, {}^t\mathbf{x} H \mathbf{x}$
- K is a tetrahedron

- We define

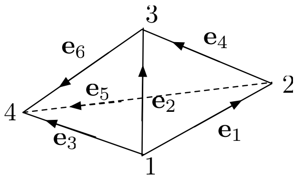
$$l_{ij} = {}^t e_i\, H\, e_j \;\text{ and }\; l_i = {}^t e_i\, H\, e_i$$

- Estimate in $\mathbf{L}^1$

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{V_K}{40} \left| -3(l_1 + l_2 + l_3) + 2(l_{12} + l_{13} + l_{23}) \right|$$

Continuous interpolation

2. Local study : u is quadratic and $\mathcal{M}(\mathbf{x}) = \mathcal{M}_0$:



- u is quadratic: $u(\mathbf{x}) = \frac{1}{2}\, {}^t\mathbf{x} H \mathbf{x}$
- K is a tetrahedron

- Estimate in $\mathbf{L}^1$

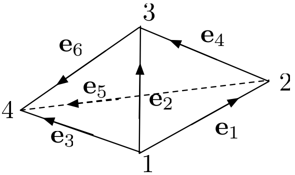
$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{V_K}{40} \left| -3(l_1 + l_2 + l_3) + 2(l_{12} + l_{13} + l_{23}) \right|$$

Main idea

$\mathcal{M}$ is a metric space, what happens when K is unit in $\mathcal{M}$?

$${}^t e_i \, \mathcal{M} \, e_i = 1$$

2. Local study : u is quadratic and $\mathcal{M}(\mathbf{x}) = \mathcal{M}_0$:



- u is quadratic: $u(\mathbf{x}) = \frac{1}{2}\, {}^t\mathbf{x} H \mathbf{x}$
- K is a tetrahedron

- Estimate in $\mathbf{L}^1$

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{V_K}{40} \left| -3(l_1 + l_2 + l_3) + 2(l_{12} + l_{13} + l_{23}) \right|$$

For all K unit for $\mathcal{M}$ we have

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{\sqrt{2}}{240} \underbrace{\det(\mathcal{M}^{-\frac{1}{2}})}_{mapping} \underbrace{\operatorname{trace}(\mathcal{M}^{-\frac{1}{2}}\, H\, \mathcal{M}^{-\frac{1}{2}})}_{Anisotropic\ part}$$

Continuous interpolation

For all K unit for $\mathcal{M}$ we have

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{\sqrt{2}}{240} \underbrace{\det(\mathcal{M}^{-\frac{1}{2}})}_{mapping} \underbrace{\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}})}_{Anisotropic\ part}$$

- Error decomposition

$$\frac{\|u - \Pi_h u\|_{\mathbf{L}^1(K)}}{|K|} = \frac{\sqrt{2}}{240} \left(d^{-\frac{2}{3}} \underbrace{\sum_{i=1}^{3} r_i^2 \, {}^t\vec{v}_i \, H \, \vec{v}_i}_{local,\ independent\ of\ |K|} \right)$$

- $d \to \infty$ when $h_i \to 0$: control the level of accuracy
- $\sum_{i=1}^{3} r_i^2 \, {}^t\vec{v}_i \, H \, \vec{v}_i$ measures orientation deviance between H and $\mathcal{M}$

Continuous interpolation

For all K unit for $\mathcal{M}$ we have

$$\|u - \Pi_h u\|_{\mathbf{L}^1(K)} = \frac{\sqrt{2}}{240} \underbrace{\det(\mathcal{M}^{-\frac{1}{2}})}_{mapping} \underbrace{\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}} H \mathcal{M}^{-\frac{1}{2}})}_{Anisotropic\ part}$$

Existence of a continuous interpolate $\pi_{\mathcal{M}}$?

- Continuous counterpart to K is the unit ball $\mathcal{B}_{\mathcal{M}}$ of $\mathcal{M}$
- Continuous counterpart of Π_h is the linear Clément interpolation $\pi_{\mathcal{M}}$

Continuous interpolation

if $u = \frac{1}{2}\,{}^t\mathbf{x}\,H\,\mathbf{x} + {}^t\mathbf{x}\,G + b$, then $\pi_{\mathcal{M}} u = {}^t\mathbf{x}\,A + c$:

$$\begin{aligned} \int_{\mathcal{B}_{\mathcal{M}}} (f(\mathbf{x}) - \pi_{\mathcal{M}}(\mathbf{x}))\,\mathrm{d}\mathbf{x} &= 0, \\ \int_{\mathcal{B}_{\mathcal{M}}} (f(\mathbf{x}) - \pi_{\mathcal{M}}(\mathbf{x}))\,x_i\,\mathrm{d}\mathbf{x} &= 0, \end{aligned} \quad \Longrightarrow \quad \begin{aligned} &\frac{2}{15}\,\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}}\,H\,\mathcal{M}^{-\frac{1}{2}}) + \frac{4}{3}\,(b-c) = 0, \\ &\frac{4}{15}\,\mathcal{M}^{-\frac{1}{2}}\,(G-A) = 0. \end{aligned}$$

$$\begin{aligned} (f(x) - \pi_{\mathcal{M}}(x)) &= \frac{1}{10}\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}}\,H\,\mathcal{M}^{-\frac{1}{2}}) + \frac{1}{2}\,{}^t\mathbf{x}\,H\,\mathbf{x} \\ &= \frac{1}{10}\mathrm{trace}(\mathcal{M}^{-\frac{1}{2}}\,H\,\mathcal{M}^{-\frac{1}{2}}) + O(\|\mathbf{x}\|_2^2). \end{aligned}$$

Discrete-Continuous equivalence

There exist a constant C that only depends on the dimension:

$$(u - \pi_{\mathcal{M}} u)(\mathbf{a}) = C\,\frac{\|u - \Pi_h u\|_{\mathbf{L}^1(K)}}{|K|} \text{ for all } K \text{ unit in } \mathcal{M}.$$

If u and $\mathcal{M}$ are analytically given, interpolation error can be computed without having a discrete mesh.

- Complete abstraction of the DISCRETE mesh
- Well-posed model for error of the form (Exact for quadratic function):

$$\int_{\Omega} |u - \pi_{\mathcal{M}} u|^p$$

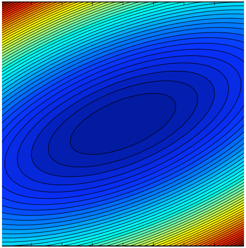
Consequences

- Calculus of variations
- GLOBAL continuous optimisation
- ..., general mathematical tools on Riemann spaces.

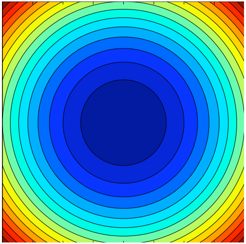
- Local optimization

$$\sum_{i=1}^{3} r_i^2 \, {}^t\vec{v}_i \, H \, \vec{v}_i \text{ is minimal if } \begin{cases} \mathcal{M} \text{ is aligned with } H \\ r_i^2 = \dfrac{\det(H)^{\frac{1}{3}}}{\lambda_i} \end{cases}$$

- Geometric meaning of optimal orientation and anisotropic ratios



$e_{I_3}(\mathbf{a})$



$e_{\mathcal{M}}(\mathbf{a})$

Global Optimality

Local error becomes:

$$e_{\mathcal{M}}(\mathbf{a}) = d^{-\frac{2}{n}} \, |\det(H(\mathbf{a}))|^{\frac{1}{n}}$$

Global calculus of variation stage

Minimize the local continuous error model in $\mathbf{L}^p$-norm:

$$\text{find } \mathcal{M} \text{ such that } \mathcal{E}(\mathcal{M}) = \min_{\mathcal{M}} \int_{\Omega} \left(|e_{\mathcal{M}}(\mathbf{x})|^p \, \mathrm{d}\mathbf{x} \right)^{\frac{1}{p}}$$

$$\text{under the constraint } \mathcal{C}(\mathcal{M}) = N.$$

Resolution in 1D:

We have $\mathcal{M}(a) = \lambda(a) = h^{-2}(a) = d^2(a)$, thus:

$$e_{\mathcal{M}}(a) = c\, h^2(a)|u''(a)| = c\, d^{-2}(a)|u''(a)|$$

We solve the problem for the density:

$$\min_{\mathcal{M}} \mathcal{E}(\mathcal{M})^p = \min_{d} \int_\Omega c\, d^{-2p}(x)\, |u''(x)|^p \, \mathrm{d}x$$

under the constraint

$$\mathcal{C}(d) = \int_\Omega d(x)\, \mathrm{d}x = N$$

Resolution in 1D:

From the Euler-Lagrange optimality condition we obtain:

$$d(a) = cte\, |u''(a)|^{\frac{p}{2p+1}}$$

The constant is evaluated thanks to the constraint:

$$\begin{aligned} d_{opt}(a) &= \frac{N}{\displaystyle\int_\Omega |u''(x)|^{\frac{p}{2p+1}}\, \mathrm{d}x} |u''(a)|^{\frac{p}{2p+1}} \\ h_{opt}(a) &= \frac{\displaystyle\int_\Omega |u''(x)|^{\frac{p}{2p+1}}\, \mathrm{d}x}{N} |u''(a)|^{-\frac{p}{2p+1}} \end{aligned}$$

We deduce:

$$\mathcal{E}(\mathcal{M}_{opt}) = \frac{\left(\displaystyle\int_\Omega |u''(x)|^{\frac{p}{2p+1}}\, \mathrm{d}x\right)^{\frac{2p+1}{p}}}{N^2}$$

Global Optimality

Resolution in 1D:

$$\mathbf{L}^1 \ : \ h_{opt}(a) = \frac{\displaystyle\int_\Omega |u''(x)|^{\frac{1}{3}}\,\mathrm{d}x}{N}\,|u''(a)|^{-\frac{1}{3}}$$

$$\mathbf{L}^2 \ : \ h_{opt}(a) = \frac{\displaystyle\int_\Omega |u''(x)|^{\frac{2}{5}}\,\mathrm{d}x}{N}\,|u''(a)|^{-\frac{2}{5}}$$

$$\mathbf{L}^\infty \ : \ h_{opt}(a) = \frac{\displaystyle\int_\Omega |u''(x)|^{\frac{1}{2}}\,\mathrm{d}x}{N}\,|u''(a)|^{-\frac{1}{2}}$$

Notice that for $\mathbf{L}^\infty$ we have:

$$e^\infty_{opt}(a) = \frac{\left(\displaystyle\int_\Omega |u''(x)|^{\frac{1}{2}}\,\mathrm{d}x\right)^2}{N^2} \quad \text{and} \quad \mathcal{E}^\infty(\mathcal{M}_{opt}) = \frac{\left(\displaystyle\int_\Omega |u''(x)|^{\frac{1}{2}}\,\mathrm{d}x\right)^2}{N^2}$$

$\Longrightarrow$ Equi-distribution principle

Global Optimality

Optimal metric

$$\mathcal{M}_{\mathbf{L}^p} = \underset{❶}{D_{\mathbf{L}^p}} \quad \underset{❷}{(\det|H_u|)^{\frac{-1}{2p+3}}} \quad \underset{❸}{\mathcal{R}_u^{-1}} \quad \underset{❹}{|\Lambda|} \quad \mathcal{R}_u$$

1. Global normalisation: to reach the targeted # of points N

$$D_{\mathbf{L}^p} = N^{\frac{2}{3}} \left(\int_\Omega (\det|H_u|)^{\frac{p}{2p+3}} \right)^{-\frac{2}{3}} \quad \text{and} \quad D_{\mathbf{L}^\infty} = N^{\frac{2}{3}} \left(\int_\Omega (\det|H_u|)^{\frac{1}{2}} \right)^{-\frac{2}{3}}$$

2. Local normalisation: sensitivity to small solution variations, depends on $\mathbf{L}^p$ norm
3. Optimal directions equal to Hessian eigenvectors
4. Diagonal matrix of absolute values of Hessian eigenvalues

Global Optimality

Theorem

- The optimal metric $\mathcal{M}_{\mathbf{L}^p}$ is unique
- Optimal interpolation error in $\mathbf{L}^p$-norm reads

$$\mathcal{E}(\mathcal{M}_{\mathbf{L}^p}) = nN^{-\frac{2}{n}} \left(\int_\Omega (\det |H_u|)^{\frac{p}{2p+n}} \right)^{\frac{2p+n}{pn}}$$

- Second-order of convergence for the interpolation error is asymptotically predicted, indeed: $\mathcal{E}(\mathcal{M}_{\mathbf{L}^p}) \leq \dfrac{Cst}{N^{2/n}}$

We may expect

- Global second-order asymptotic mesh convergence for the mesh adaptation process

Contribution to High-Order Method

- High-order shock capturing methods are used to model accurately physical phenomena
- They generally correspond to second-order scheme

On industrial problem global mesh convergence is rarely second-order !!!

$\Longrightarrow$ Second-order convergence is observed for smooth flows

$\Longrightarrow$ Only asymptotic second-order convergence is reached for flows with steep gradients (Navier-Stokes)

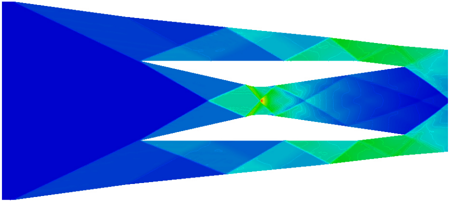
$\Longrightarrow$ For inviscid flow containing singularities second-order is not reached (Euler)

Illustration on a simple example

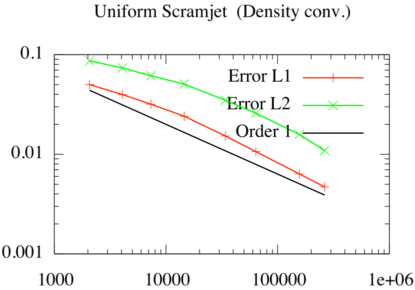
Contribution to High-Order Method

- Mach 3 flow in a scramjet

  Global mesh convergence on a sequence of uniform mesh



Density iso-values ≈ 150M vertices



Density convergence

⟹ Order one is only reached with a second-order scheme

Contribution to High-Order Method

From the continuous metric theory, we may expect

- Global second-order asymptotic mesh convergence arises for numerical solutions with singularities

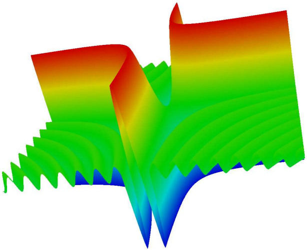
1D example: a step

- $\mathbf{L}^p$ interpolation error corresponds to a $\frac{1}{p}$ order of convergence
- $\Longrightarrow$ $\mathbf{L}^p$ continuous metric imposes $2p$ points in the singularities.

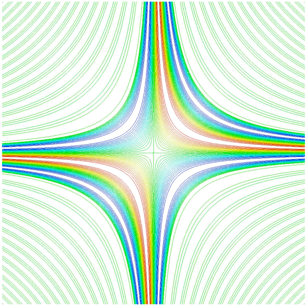
Remark: this explain why h_{min} was required for the previous theory.

2D Analytical Examples

$$\forall (x,y) \in [0;1]^2 \quad f_1(x,y) = \begin{cases} 0.1\sin(50xy) & \text{if } xy \le \pi/50 \\ \sin(50xy) & \text{if } xy \le 2\pi/50 \\ 0.1\sin(50xy) & \text{elsewhere} \end{cases}$$

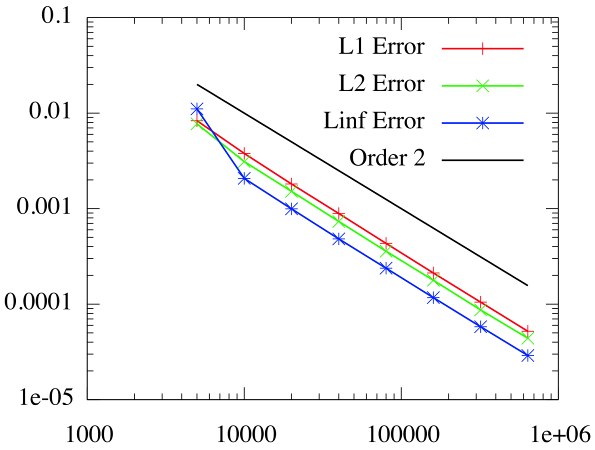


f_1



f_1 isovalues

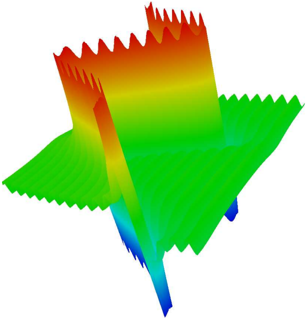
2D Analytical Examples



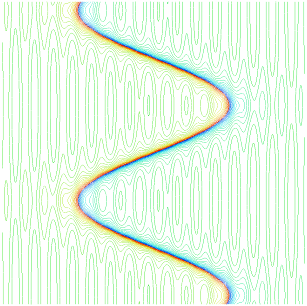
f_1 mesh convergence order

2D Analytical Examples

$$\forall (x,y) \in [0;1]^2 \quad f_2(x,y) = 0.1\sin(50x) + atan\left(\frac{0.1}{\sin(5y)-2x}\right)$$

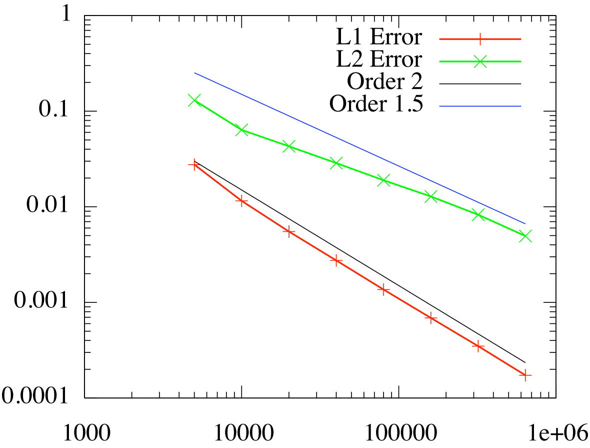


f_2



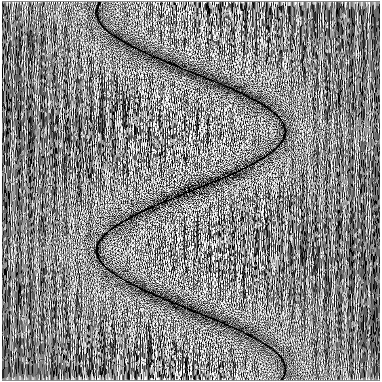
f_2 isovalues

2D Analytical Examples

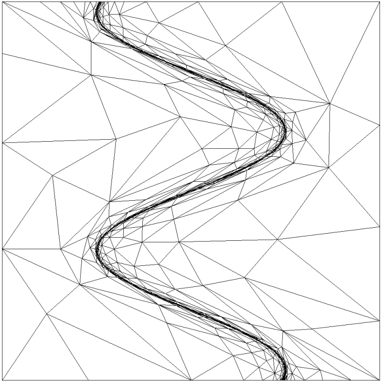


f_2 mesh convergence order

2D Analytical Examples



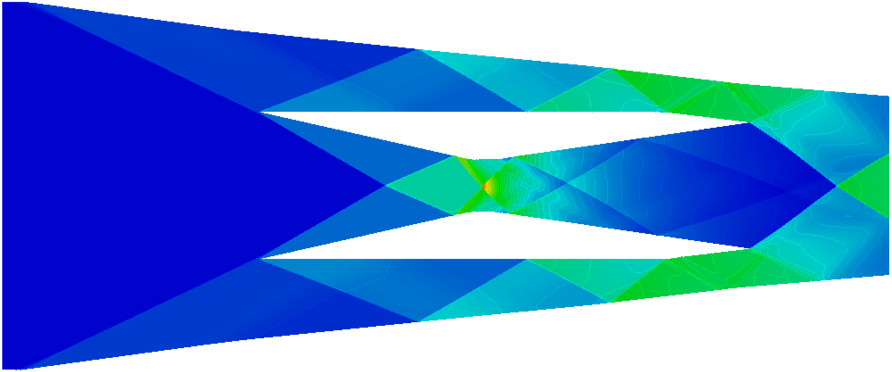
L^1



L^∞

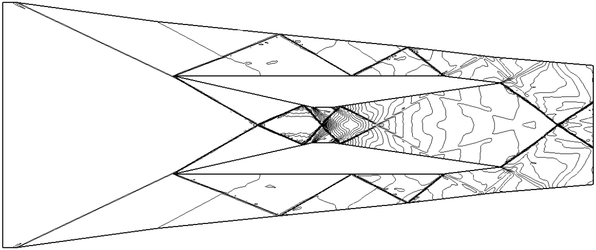
Scramjet Internal Flow

- Mach 3 flow in scramjet: density iso-values

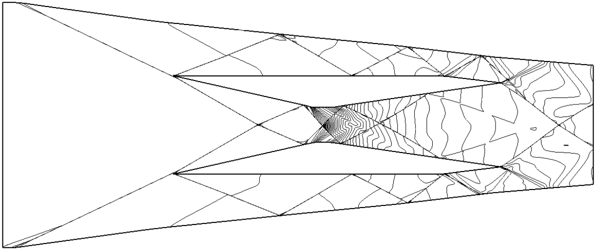


Scramjet Internal Flow

- Density iso-lines:

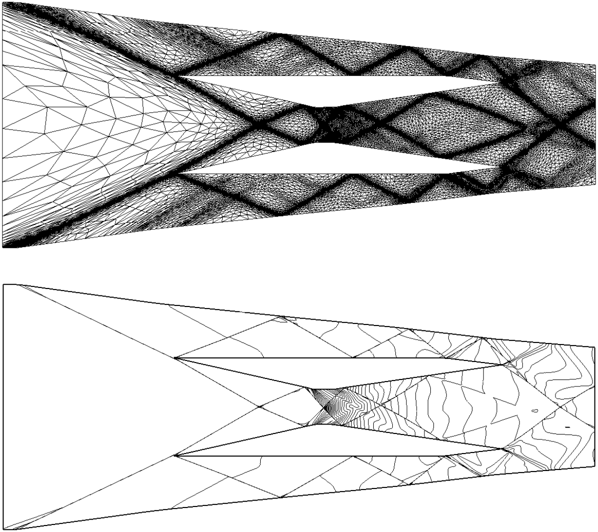


uniform mesh
156, 414 vertices



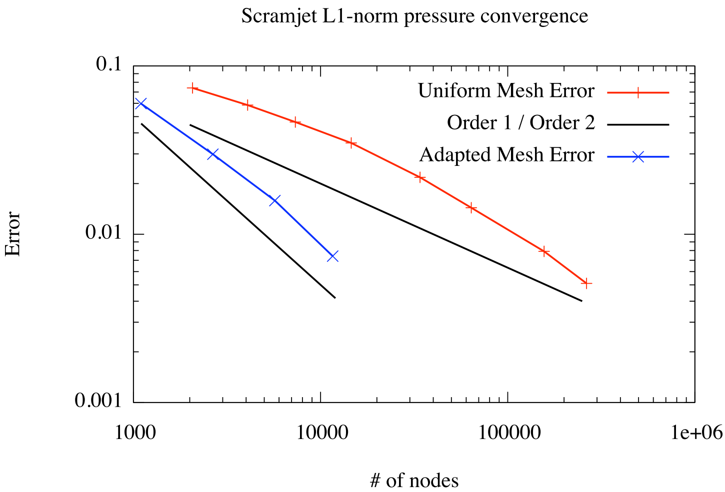
adapted mesh
with $\mathbf{L}^1$ norm
22, 566 vertices

- Adapted mesh with $\mathbf{L}^1$ norm $22,566$ vertices:



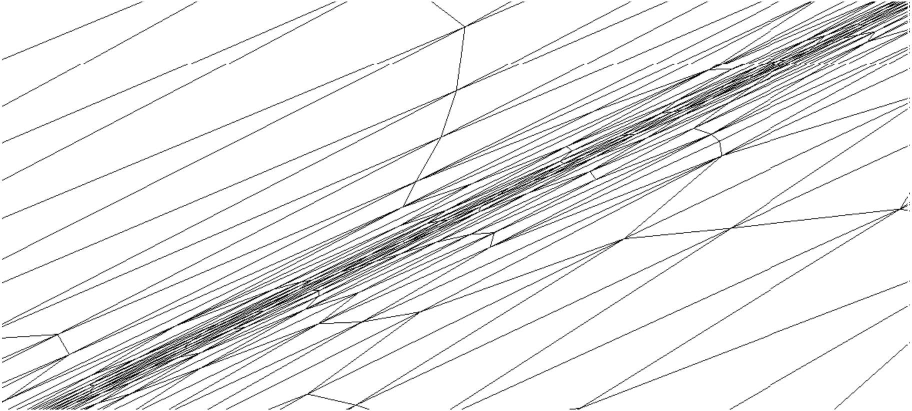
Scramjet Internal Flow

- Mach 3 flow in scramjet: density convergence for $\mathbf{L}^1$ norm:



Scramjet Internal Flow

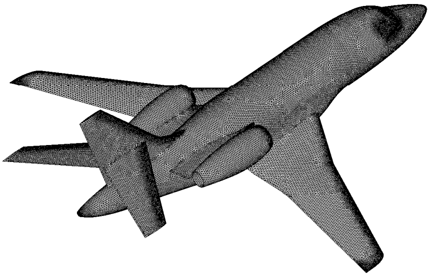
- Is there anisotropy ?

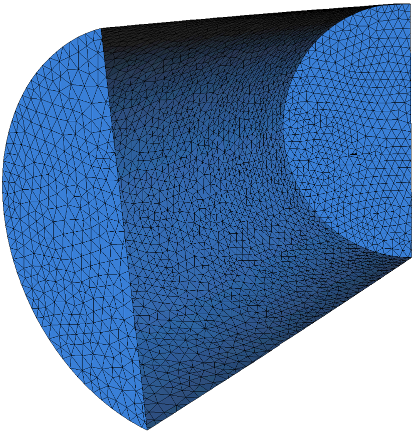


Transonic Falcon

Falcon at transonic speed cruise. Inviscid computation.

$\approx$ 12 millions tetrahedra.

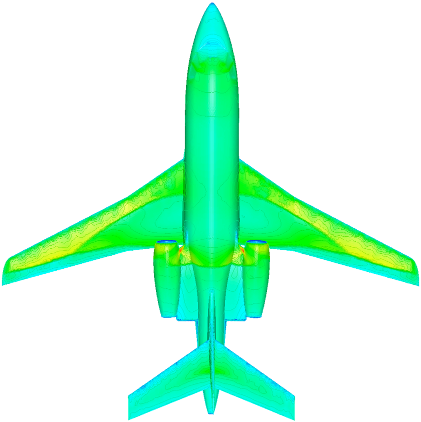


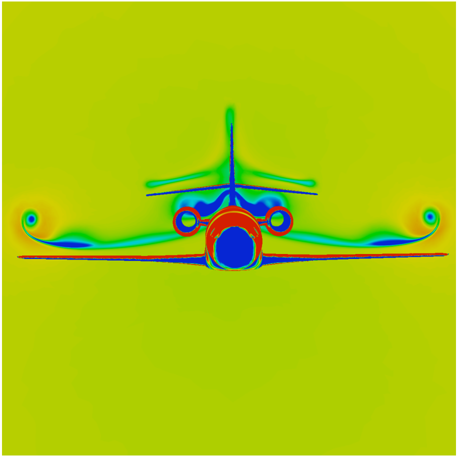


Transonic Falcon

Falcon at transonic speed cruise. Inviscid computation.

$\approx$ 12 millions tetrahedra.

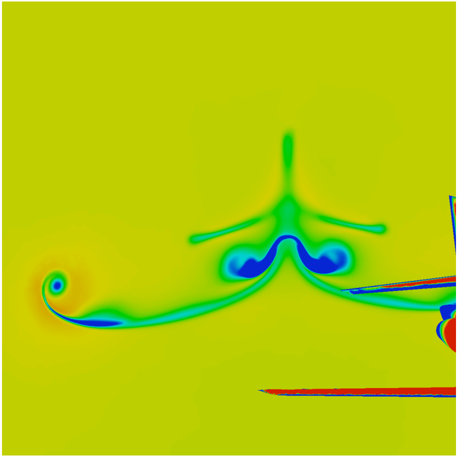


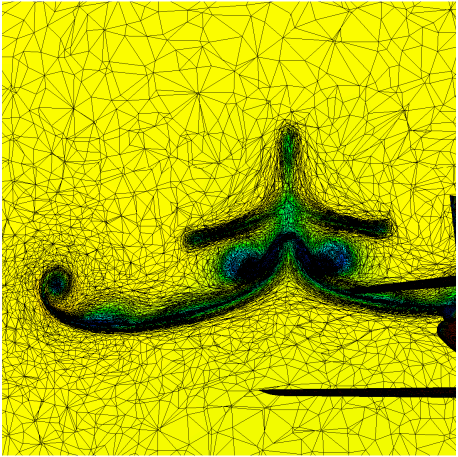


Transonic Falcon

Falcon at transonic speed cruise. Inviscid computation.

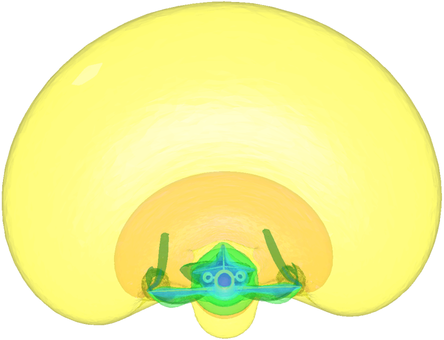
$\approx$ 12 millions tetrahedra.

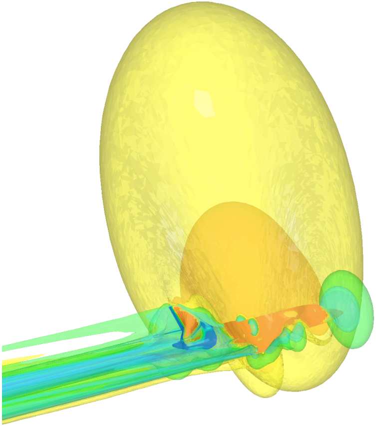




Falcon at transonic speed cruise. Inviscid computation.

$\approx$ 12 millions tetrahedra.



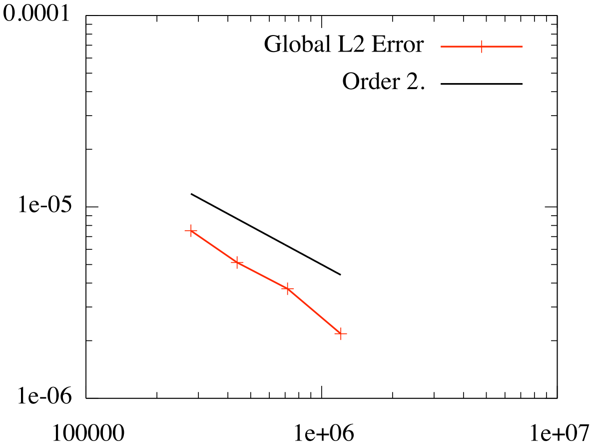


Transonic Falcon

Falcon at transonic speed cruise. Inviscid computation.

$\approx$ 12 millions tetrahedra.

Subsonic aircraft Mach number convergence for $\mathbf{L}^2$ norm:



Outline

Motivation for Goal Oriented Mesh Adaptation

Real-life problems

u is solution of a PDE and only its numerical approximate u_h is given on the current mesh.

- Hessian-based criteria easily produce optimal specification for mesh density and local stretching. But they rely on local interpolation error and link with PDE's is difficult.

  Castro-Diaz *et al.* IJNMF 1997, Habashi *et al.* IJNMF 2000, Huang JCP 2005, Frey-Alauzet CMAME 2005, Courty *et al.* ANM 2006

- PDE-based estimates are closer to PDE goals but produce stretching criteria less easy to exploit.

  Venditti-Darmofal JCP 2002, Formagggia *et al.* ANM 2004

Motivation for Goal Oriented Mesh Adaptation

We evaluate the performance of a solution W by means of a functional $j(W)$.
For instance: the drag, the lift, the temperature at the exit, ...

Linearizing the functional leads to:

$$j(W) \approx j(W_h) + (g, W - W_h) + R$$

We aim at controlling the approximation error on this functional

$$j(W) - j(W_h) = (g, W - W_h)$$

Motivation for Goal Oriented Mesh Adaptation

We aim at controlling the approximation error on this functional

$$j(W) - j(W_h) = (g, W - W_h)$$

Practically,

Let us consider the Euler equation : $div(F(W)) = 0$

We want to control the drag: $j(W) = \int_\gamma (pn_x)^2$ under the constraint $div(F(W)) = 0$.

By linearization we obtain: $g = \dfrac{\partial j(W)}{\partial W} = 2 \displaystyle\int_\gamma \dfrac{\partial p}{\partial W} n_x$

Be-careful: $g \notin L^2(\Omega) \implies$ a regularization is required

Goal Oriented MA for Euler Equations

The goal is then to control the approximation error of the solution weighted by the cost function:

$$\underbrace{(g, W - W_h)}_{\textit{Approximation error}} = \underbrace{(g, W - \Pi_h W)}_{\textit{Interpolation error}} + \underbrace{(g, \Pi_h W - W_h)}_{\textit{Implicit error}}$$

- Interpolation error: independent of the problem
  $\Longrightarrow$ previous analysis
- Implicit error: dependent of the problem
  $\Longrightarrow$ is a fct of the discrete adjoint and interpolation error

A Weighted $\mathbf{L}^1$ Minimization Problem

$$\underbrace{(g, W - W_h)}_{\textit{Approximation error}} = \underbrace{(g, W - \Pi_h W)}_{\textit{Interpolation error}} + \underbrace{(g, \Pi_h W - W_h)}_{\textit{Implicit error}}$$

We have seen: $\mathcal{M}_{opt}^{\mathbf{L}^p}(\mathbf{x}) = \mathcal{M}_{opt}^{\mathbf{L}^p}(|H(u(\mathbf{x}))|)$

We consider the following discrete optimization problem:

$$\min_{\mathcal{H}} \mathcal{E}(\mathcal{H}) = \int_{\Omega_h} |g(\mathbf{x})| \, |u(\mathbf{x}) - \Pi_h(\mathbf{x})| \, \mathrm{d}\mathbf{x}$$

In the continuous framework we get:

$$\min_{\mathcal{M}} \mathcal{E}(\mathcal{M}) = \int_{\Omega} |g(\mathbf{x})| \, |u(\mathbf{x}) - \pi_{\mathcal{M}}(\mathbf{x})| \, \mathrm{d}\mathbf{x}$$

The resolution leads to:

$$\mathcal{M}_{opt}(\mathbf{x}) = \mathcal{M}_{opt}^{\mathbf{L}^1}(|g(\mathbf{x})| \, |H(u(\mathbf{x}))|)$$

Goal Oriented MA for Euler Equations

Euler equations could be written

$$\text{div}\mathcal{F}(W) = 0$$

Variational statement assuming $W \in V = [\mathbf{H}^1(\Omega)]^5$

$$\int_\Omega \phi\, \nabla.\mathcal{F}(W)\, \text{d}\Omega - \int_\Gamma \phi\, \bar{\mathcal{F}}(W).n\, \text{d}\Gamma = 0 \qquad \forall \phi \in V$$

Mixed-Element-Volume discretization for $W_h \in V_h$
where $V_h = \{\phi_h \in V \,|\, \phi_h \text{ linear by element}\}$:

$$\int_\Omega \phi_h\, \nabla.\mathcal{F}_h(W_h)\, \text{d}\Omega - \int_\Gamma \phi_h\, \bar{\mathcal{F}}_h(W_h).n\, \text{d}\Gamma + \mathcal{D}_h(W_h, \phi_h) = 0 \quad \forall \phi_h \in V_h$$

Goal Oriented MA for Euler Equations

$$\underbrace{(g, W - W_h)}_{\textit{Approximation error}} = \underbrace{(g, W - \Pi_h W)}_{\textit{Interpolation error}} + \underbrace{(g, \Pi_h W - W_h)}_{\textit{Implicit error}}$$

Discrete adjoint P_h solution of the problem:

$(P_h, \mathcal{A}_h(W_h)\,\phi) = (g, \phi)$ where $\mathcal{A}_h$ is the discrete Jacobian

We formally deduce using the VF and the MEV discretization:

$$\begin{aligned}
(g, \Pi_h W - W_h) &= (P_h, \mathcal{A}_h(W_h)(\Pi_h W - W_h)) \\
&\approx \int_\Omega P_h \nabla.(\mathcal{F}(W) - \Pi_h \mathcal{F}(W))\,\mathrm{d}\Omega + \mathrm{BT} \\
&= -\int_\Omega \nabla.P_h\,(\mathcal{F}(W) - \Pi_h \mathcal{F}(W))\,\mathrm{d}\Omega + \mathrm{BT}
\end{aligned}$$

Goal Oriented MA for Euler Equations

$$\underbrace{(g, W - W_h)}_{\text{Approximation error}} = \underbrace{(g, W - \Pi_h W)}_{\text{Interpolation error}} + \underbrace{(g, \Pi_h W - W_h)}_{\text{Implicit error}}$$

We deduce

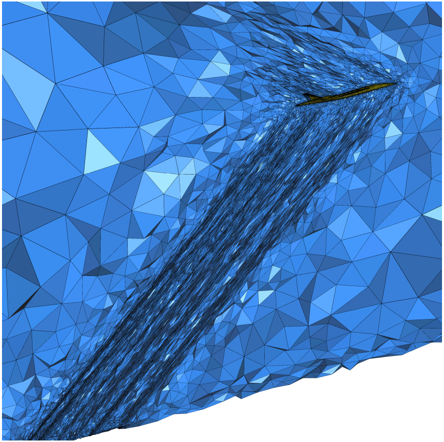
$$(|g|, |W - W_h|) \leq (|g|, |W - \Pi_h W|) + \int_\Omega |\nabla . P_h| \, |\mathcal{F}(W) - \Pi_h \mathcal{F}(W)| \, d\Omega + \mathrm{BT}$$

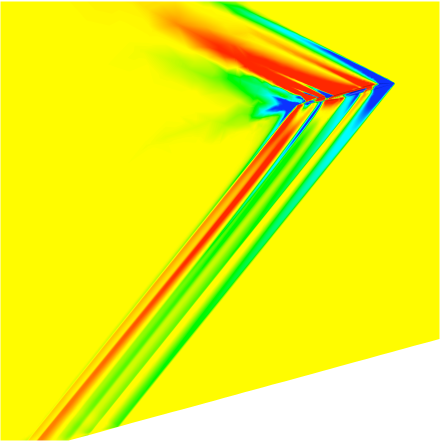
The continuous framework leads to

$$\mathcal{M}_{opt}^{go} = \mathcal{M}_{opt}^{\mathbf{L}^1}\left(\sum_{i=1}^{5} |g(W_i)| \, |H(W_i)| + \sum_{i=1}^{5} (\sum_{j=1}^{3} |\nabla_{x_j} P_h(W_i)| \, |H(\mathcal{F}_{x_j}(W_i))|)\right)$$

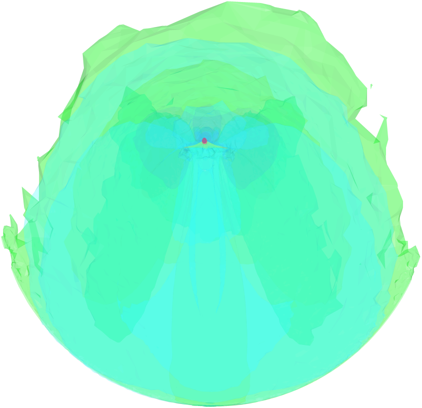
Result on a supersonic aircraft: Un-symmetric effects

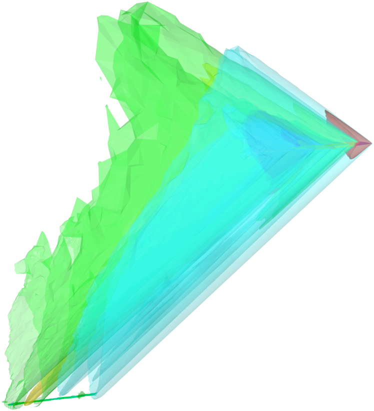
Goal Oriented MA for Euler Equations



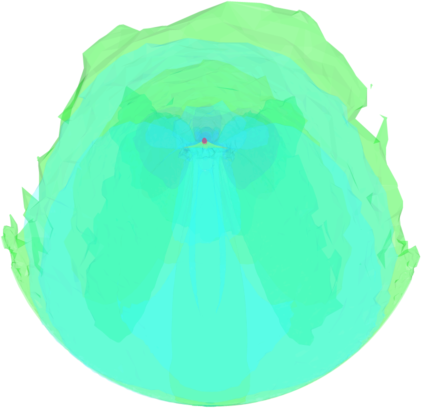


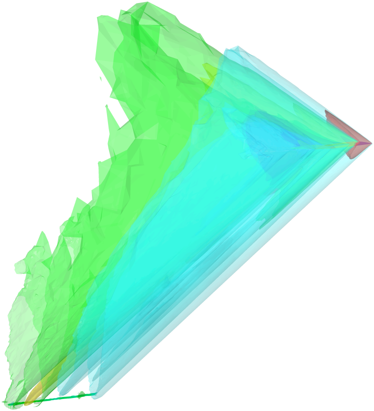
Goal Oriented MA for Euler Equations



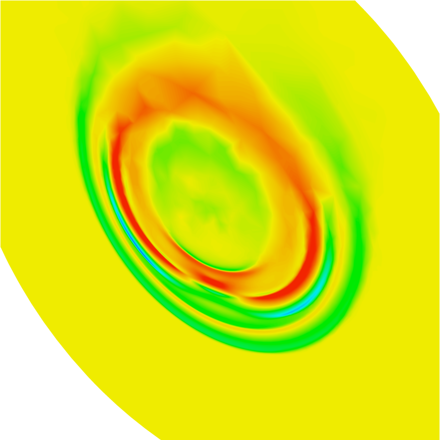


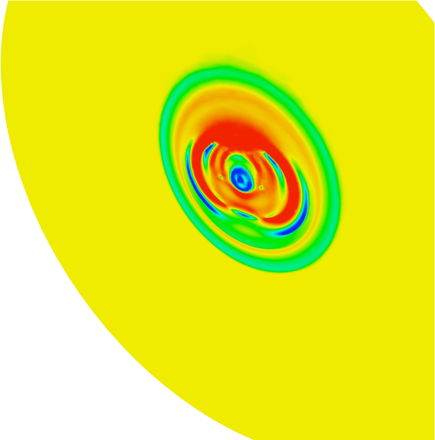
Goal Oriented MA for Euler Equations



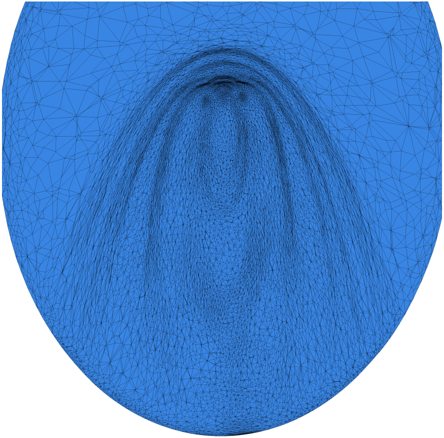


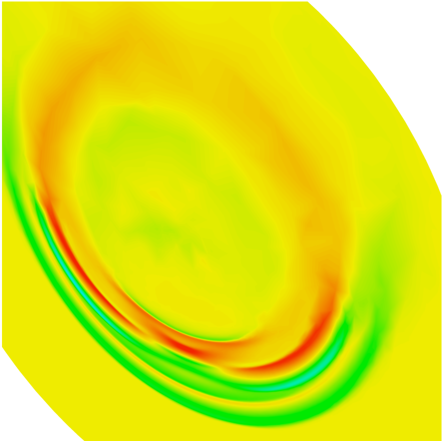
Goal Oriented MA for Euler Equations





Goal Oriented MA for Euler Equations





Outline

Un Contexte Industriel

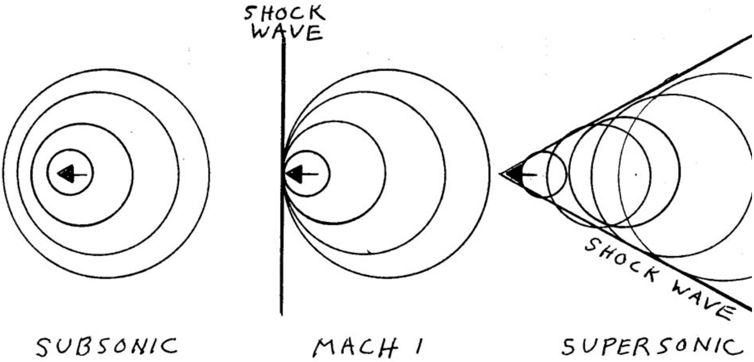
Aujourd'hui,
il y a un gros intérêt à la réalisation d'un avion supersonique

$\Longrightarrow$ un jet d'affaire supersonique : SSBJ

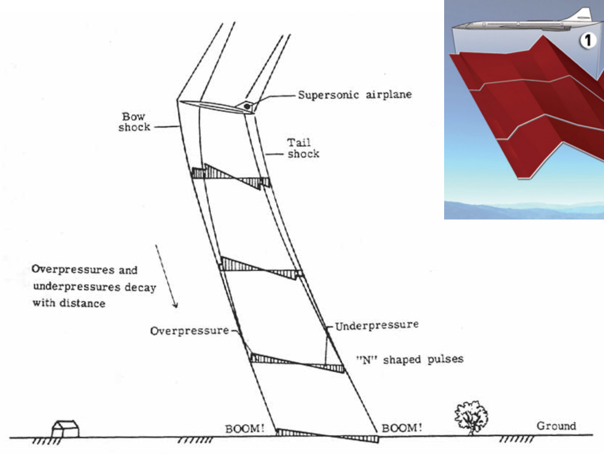
En effet, pour un vol New York $\rightarrow$ Los Angeles (3930 km):

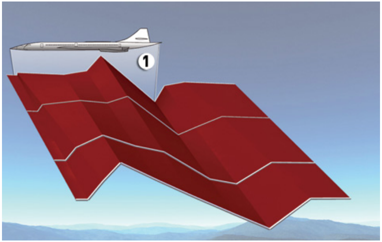
Mach $= 0.85$	≈ 900 km.h^{-1}	$\Longrightarrow$	5 h
Mach $= 0.98$	≈ 1035 km.h^{-1}	$\Longrightarrow$	4 h 20 min
Mach $= 1.8$	≈ 1900 km.h^{-1}	$\Longrightarrow$	2 h 15 min

Le phénomène physique :

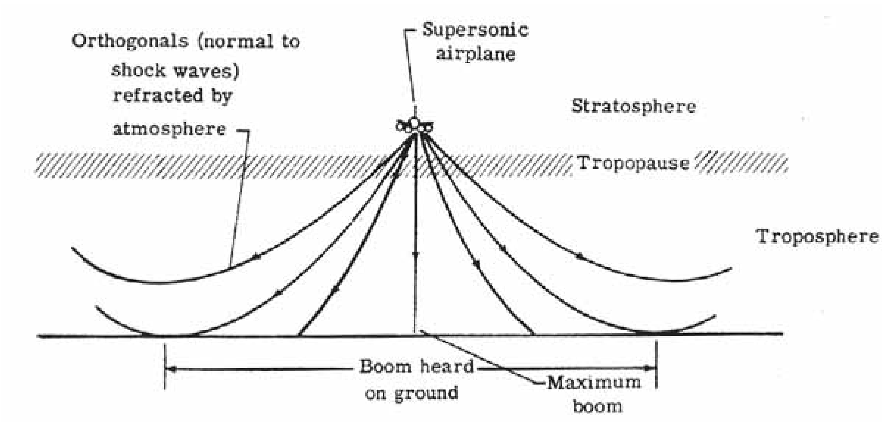


Le phénomène physique :





Le phénomène physique :



Le Bang Sonique

Exemples :

Le Bang Sonique

Impact environnemental :

- Gêne sur les populations : nuisances sonores, contrariété, effraye, ...
- Bruit et vibrations des structures
- Au pire, fenêtres cassées ou dommages des structures

Réglementation actuelle :

- Pas de vol supersonique au-dessus des terres
- Pas de bang sonique mesurable au-dessus des terres

Il faut REDUIRE le bang sonique

Objectif pour un faible bang sonique :

- 15 Pa et 65 dBA

Des essais en vol déjà réalisés par la NASA:

NASA Dryden Flight Research Center Photo Collection
http://www.dfrc.nasa.gov/Gallery/Photo/index.html
NASA Photo: EC03–0210–1 Date: August 2, 2003 Photo By: Carla Thomas

Northrop–Grumman Corporation's modified U.S. Navy F–5E Shaped Sonic Boom Demonstration (SSBD) aircraft.

Profil d'avion modifié

NASA Dryden Flight Research Center Photo Collection
http://www.dfrc.nasa.gov/Gallery/Photo/index.html
NASA Photo: ED06–0184–13 Date: September 27, 2006 Photo By: Carla Thomas

NASA F–15B #836 in flight with Quiet Spike attached.

Utilisation d'une perche

Etat des Lieux

Des projets commerciaux sont très avancés :

- Premier vol en 2011
- Commercialisation en 2013

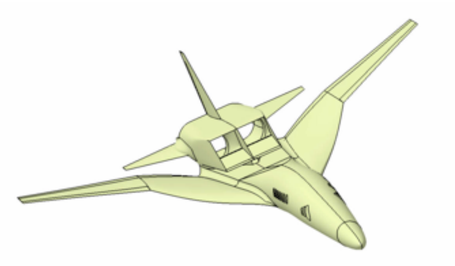
Aerion Corp.

Lockheed Martin

Et en Europe : projet Européen HISAC (HIgh Speed AirCraft)

- Coordonné par Dassault Aviation
- $\approx$ 40 partenaires

Profil faible traînée



Profil faible bang

- Simuler précisément le bang sonique
- Certifier les calculs de signatures au sol

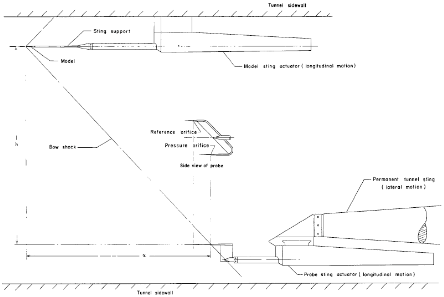
$\Longrightarrow$ Calcul très précis loin de l'avion (plusieurs kilomètres)

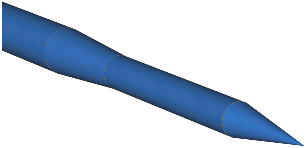
Nécessite les techniques d'adaptation de maillage

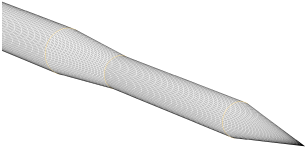
$\Downarrow$

- Analyser des formes d'avions non-conventionnelles
- Optimiser les profils sélectionnés pour minimiser le bang sonique

Nasa spike: Experimental study, NASA Langley, 1968

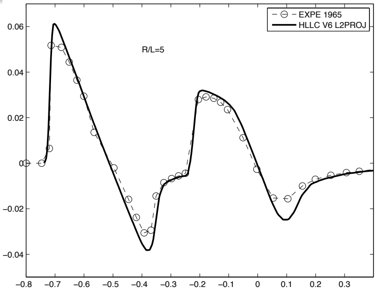


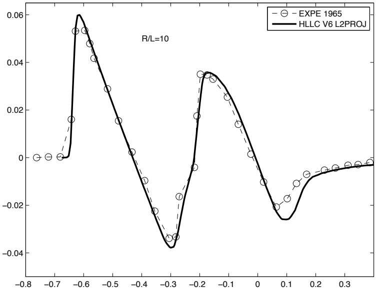


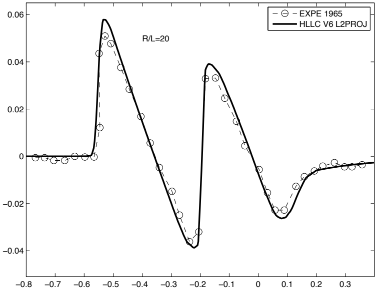


Nasa spike:

- 10M points
- Projection $\mathbf{L}^2$
- Mach 1.41

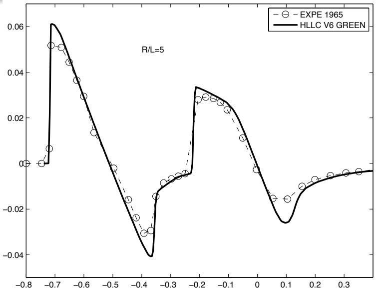


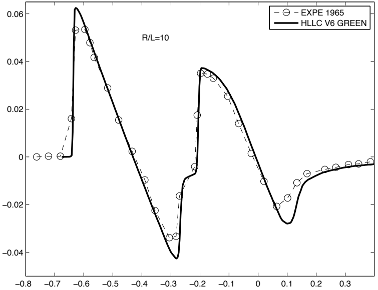


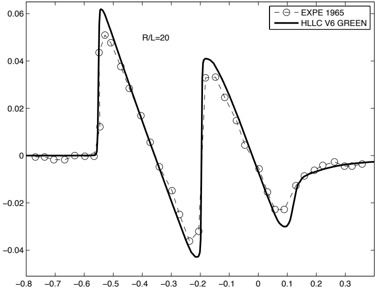


Nasa spike:

- 10M points
- Green
- Mach 1.41



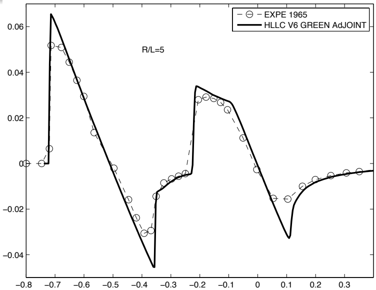


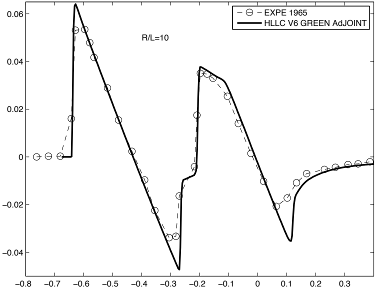


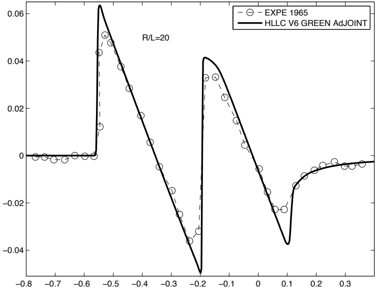
Validations

Nasa spike:

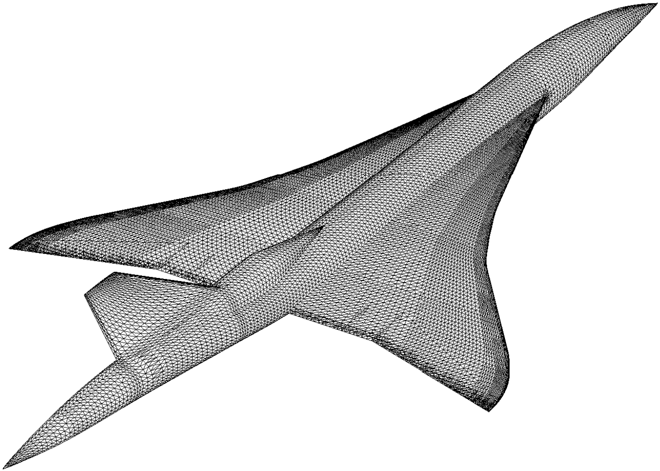
- 10M points
- Green et Adjoint
- Mach 1.41



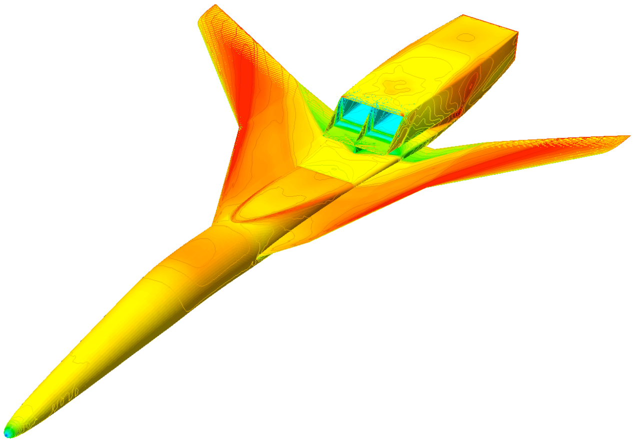




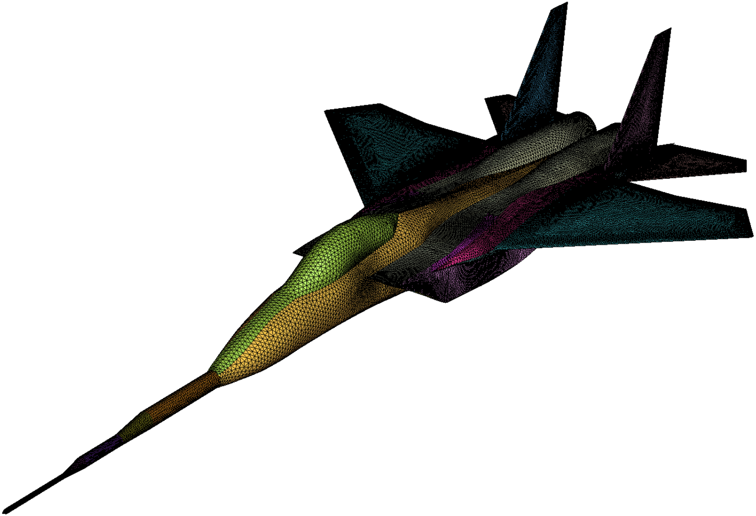
Avions Etudiés



Avions Etudiés



Avions Etudiés



Supersonic Business Jet

Supersonic business jet simulation

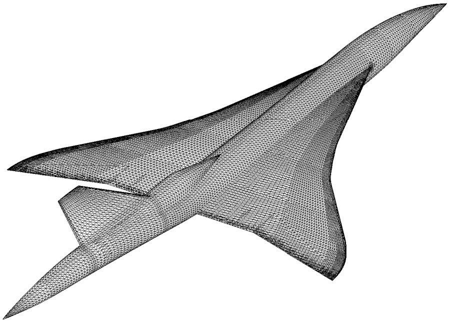
Objective: modelling the sonic boom

Accurate pressure extraction at a distance R from the SSBJ

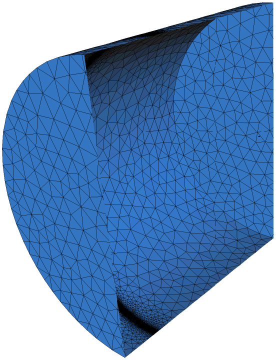
- 1.6 Mach
- an angle of attack of 3 degrees
- an altitude of $45,000$ feet
- aircraft length $L = 36$ meters

Simulation carried out in serial on a MacPro

- 2.66 GHz Intel Xeon processor
- 4 GB of memory
- approximately 48 hours of CPU for the whole process (22 millions of tetrahedra)



Aircraft geometry



Computational domain

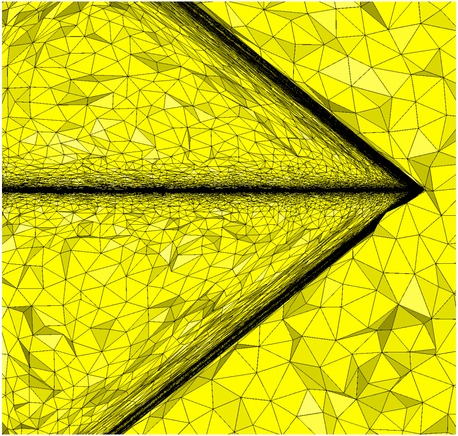
Aircraft size = 36m, mesh size from 1mm to 30cm

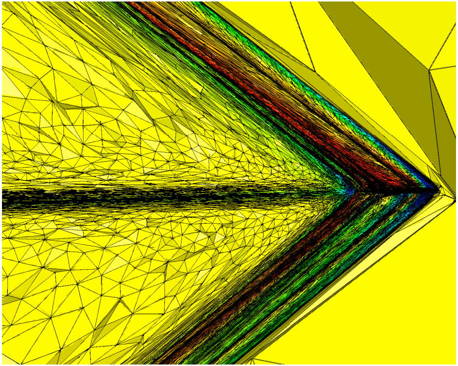
Domain size (meters):

$x : [-225,\ 2025] \quad y : [-1200,\ 1200] \quad z : [-1200,\ 1200]$

Supersonic Business Jet

- Volume adapted mesh with $\mathbf{L}^2$ norm on the Mach Number
  - $\approx$ 3.8 millions vertices
  - $\approx$ 22.3 millions tetrahedra

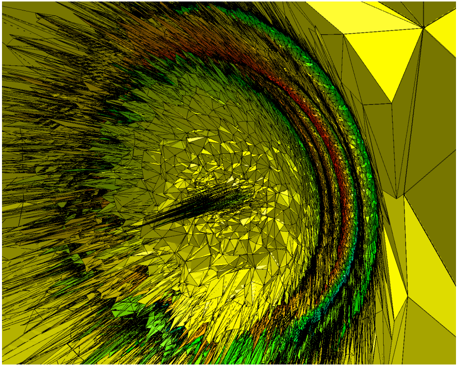




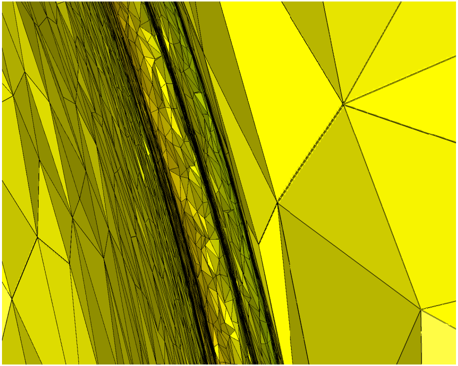
Mesh refinements propagate 2km

Supersonic Business Jet

- Volume adapted mesh with $\mathbf{L}^2$ norm on the Mach Number
  - ≈ 3.8 millions vertices
  - ≈ 22.3 millions tetrahedra



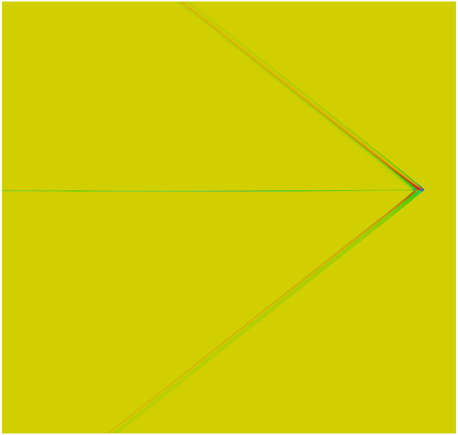
Mesh behind the aircraft

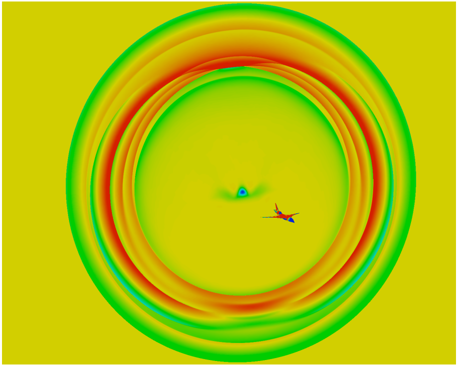


After 2km of propagation
mesh size is ≈ one meter

Supersonic Business Jet

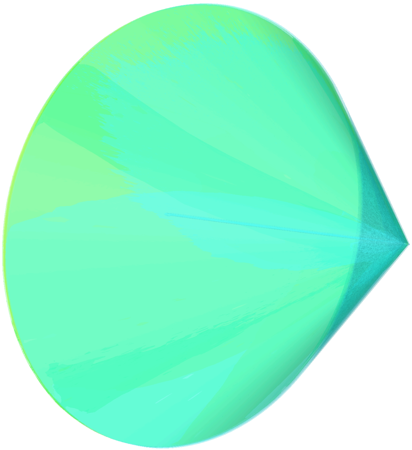
- Mach Number iso-values
  - Solution accurately propagated in the whole domain
  - All shocks are accurately captured

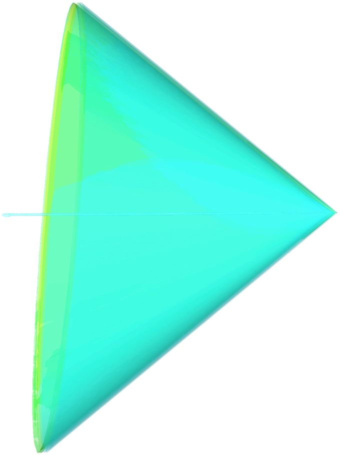




Supersonic Business Jet

- Mach Number iso-surfaces
  - Mach cone clearly appears
  - Solution accurately propagated in the whole domain





Conclusion: Covered topics in this course

- Goals from physical motivations:
  - complex geometries,
  - complex anisotropic phenomena
- Gaols from numerical motivations
  - validate the whole adaptive process,
  - achieve convergence for flows with singularities

$\Longrightarrow$ Mesh adaptation to reduce complexity

Standard framework with the notion of continuous mesh and unit meshes

Automaticity with the notion of continuous mesh and unit meshes

Mesh adaptation is equivalent to mesh optimisation
Error certification is another (complementary) job

Conclusion: Uncovered topics in this course

- Un-steady problems
- High order mesh adaptation
- Mesh adaptation for level-set
- Mesh generation/adaptation techniques
- Interpolation for metric fields, conservative, etc.

f is a smooth function, in the vicinity of $\mathbf{x}_0$:

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)\,(\mathbf{x}-\mathbf{x}_0) + \frac{1}{2}\,{}^t(\mathbf{x}-\mathbf{x}_0)\,H(\mathbf{x}_0)\,(\mathbf{x}-\mathbf{x}_0) + O(\|\mathbf{x}-\mathbf{x}_0\|_2^2)$$

$\mathbf{x}$ belongs to the iso-line $f(\mathbf{x}_0)$ if $f(\mathbf{x}) = f(\mathbf{x}_0)$:

$$\nabla f(\mathbf{x}_0)(\mathbf{x}-\mathbf{x}_0) + \frac{1}{2}\,{}^t(\mathbf{x}-\mathbf{x}_0)\,H(\mathbf{x}_0)\,(\mathbf{x}-\mathbf{x}_0) = 0.$$

Mesh Adaptation for Level Set Application

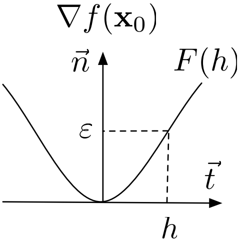
$$\nabla f(\mathbf{x}_0)(\mathbf{x}-\mathbf{x}_0) + \frac{1}{2}{}^t(\mathbf{x}-\mathbf{x}_0)\,H(\mathbf{x}_0)\,(\mathbf{x}-\mathbf{x}_0) = 0.$$

In $2D$, in the local Frenet-frame $(\vec{\mathbf{n}}, \vec{\mathbf{t}})$:

$$\varepsilon(h) = \varepsilon(0) + \varepsilon'(0)\,h + \frac{1}{2}\varepsilon''(0)\,h^2$$

with

$$\frac{\partial F}{\partial h} + \varepsilon'(h)\frac{\partial F}{\partial \varepsilon} = 0$$



$$\varepsilon(h) = -\frac{1}{2}h^2\,\frac{{}^t\vec{t}\,H(\mathbf{x}_0)\,\vec{t}}{\|\nabla f(\mathbf{x}_0)\|_2}$$

Mesh Adaptation for Level Set Application

- along $\vec{n}$:

$$\text{distance} \approx \varepsilon$$

- along $\vec{t}$:

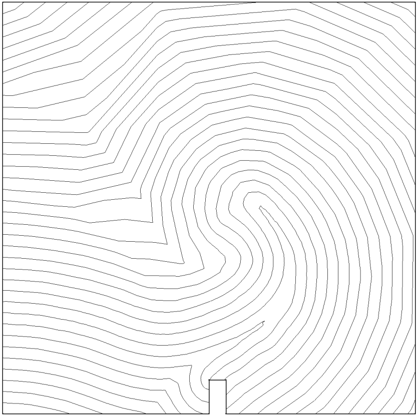
$$\text{distance} \approx \frac{1}{2} \frac{\left| {}^t\vec{t}\, H(\mathbf{x}_0)\, \vec{t} \right|}{\varepsilon \, \|\nabla f(\mathbf{x}_0)\|_2}$$

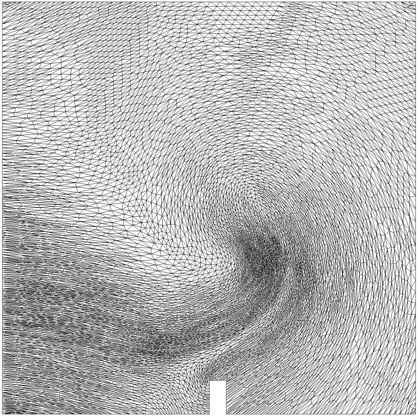
$$\mathcal{M}^{iso}(\varepsilon) = (\vec{n}, \vec{t}) \begin{pmatrix} \frac{1}{\varepsilon^2} & \\ & \frac{1}{2} \frac{\left| {}^t\vec{t}\, H(\mathbf{x}_0)\, \vec{t} \right|}{\varepsilon \, \|\nabla f(\mathbf{x}_0)\|_2} \end{pmatrix} {}^t(\vec{n}, \vec{t})$$

In the case of distance-function for level-set: $\|\nabla f(\mathbf{x}_0)\|_2 = 1$.

Mesh Adaptation for Level Set Application

Practical interpretation: the mesh is following the solution iso-lines





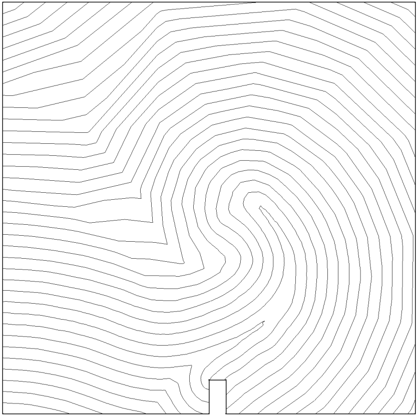
Mesh Adaptation for Level Set Application

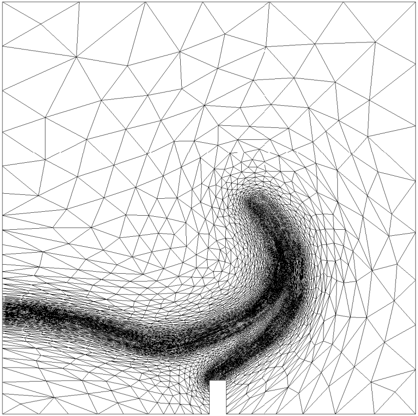
Metric dedicated to the interface:

$$\mathcal{M}_{ls}(x) = \begin{cases} D(\varepsilon)\, R \begin{pmatrix} h_{ls}^{-2} & 0 \\ 0 & \gamma_{\vec{t}} \end{pmatrix} R^t & \text{if } |\Phi(x)| \leq \delta_{ls} \\ D(\varepsilon)\, R \begin{pmatrix} h_{\vec{n}}^{-2} & 0 \\ 0 & h_{\vec{t}}^{-2} \end{pmatrix} R^t & \text{if } |\Phi(x)| > \delta_{ls} \end{cases}$$

- Adapt in a layer with a size $2\,\delta_{ls}$
- Prescribe h_{ls} in the normal direction
- Control the variation in the tangential direction via ε

Mesh Adaptation for Level Set Application



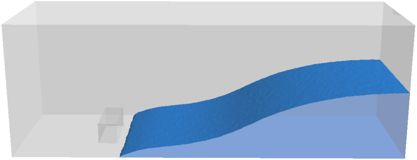


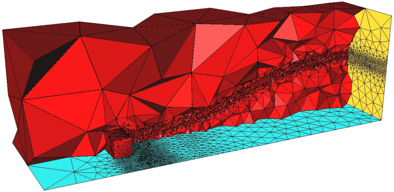
3D Water column falling on a obstacle

Bi-fluids simulation, done by Damien Gueugan, Lemma, Nice.
Solveur: Ananas, Olivier Allain, Lemma, Nice.

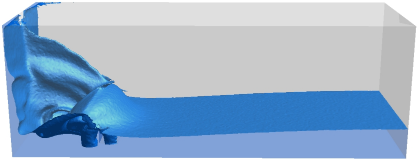
3D Water column falling on a obstacle

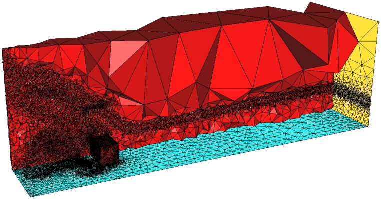
0.4 sec :





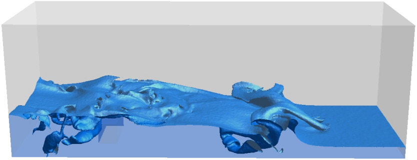
1.2 sec :

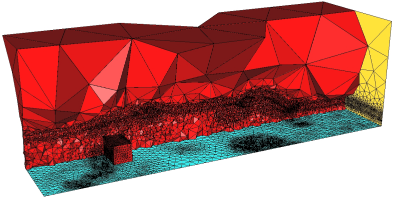




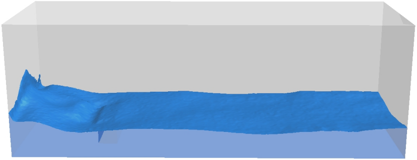
3D Water column falling on a obstacle

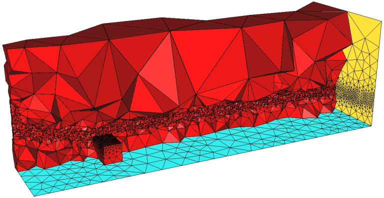
2.4 sec :



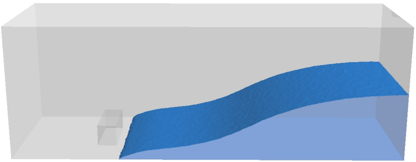


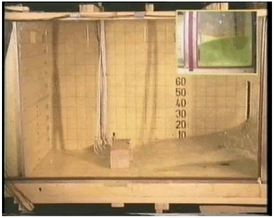
5.6 sec :

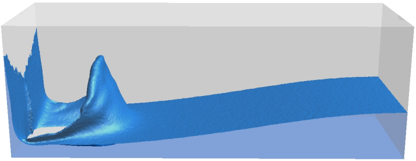


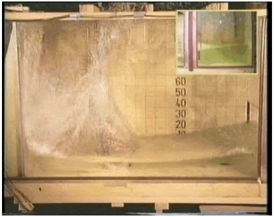


Comparison with experience

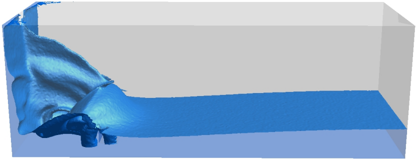


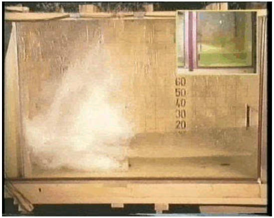


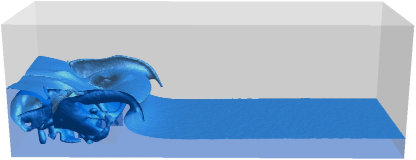


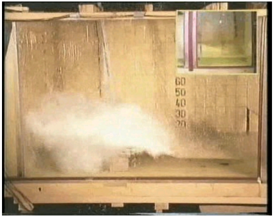


Comparison with experience

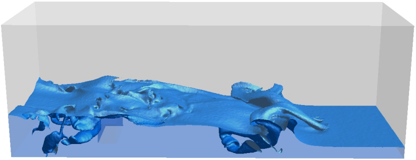


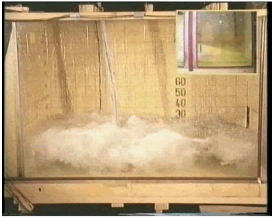


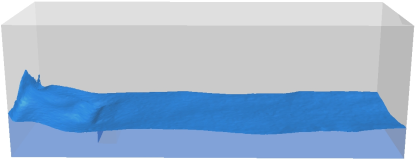


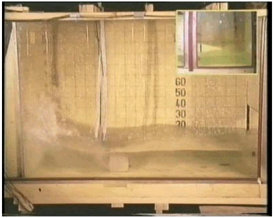


Comparison with experience

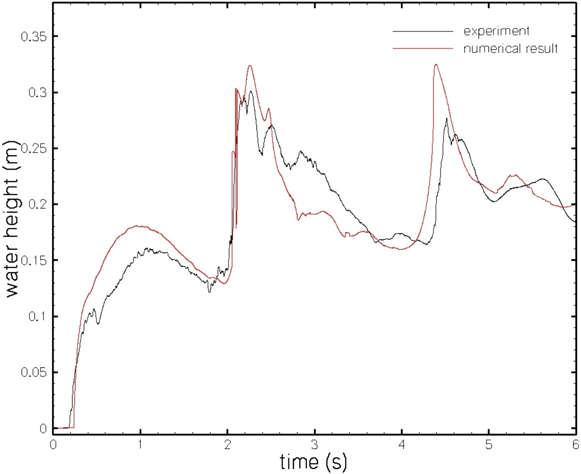






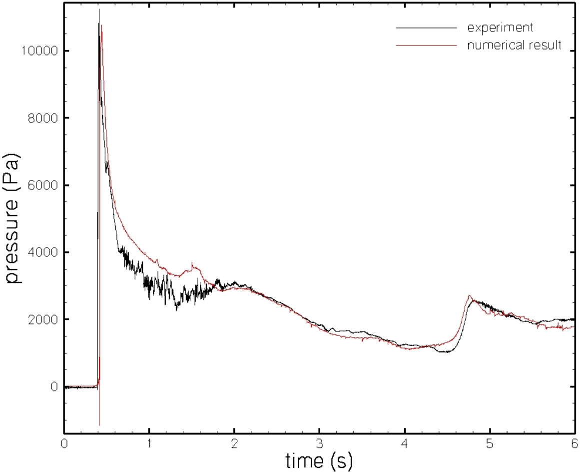


Experimental sensors:



Water high

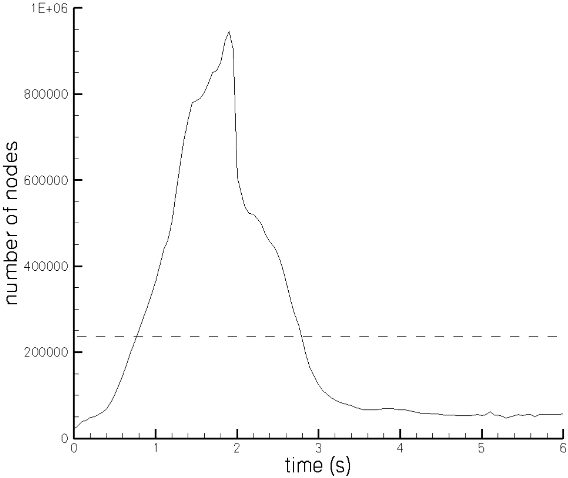
Experimental sensors:



Pressure

Problem size evolution

Evolution of the number of dof:



Outline

Why interpolation error is a reliable error estimate ?

- We have the following bound of the interpolation error:

$$\| u - \Pi_h u \|_{\infty,K} \leq c_d \max_{x \in K} \max_{e \in E_K} \langle \vec{e}, |H_u(x)| \, \vec{e} \rangle$$

- Let $\overline{\mathcal{M}}(K)$ be an adequate metric tensor.
  Then, we estimate on each element K the error by:

$$\varepsilon_K = c_d \max_{e \in E_K} \langle \vec{e}, \overline{\mathcal{M}}(K) \, \vec{e} \rangle \, .$$

- A metric providing a unit length !

If we put: $\mathcal{M}(K) = \dfrac{c_d \, \overline{\mathcal{M}}(K)}{\varepsilon_K} \quad \Longrightarrow \quad \max_{e \in E_K} \langle \vec{e}, \mathcal{M}(K) \, \vec{e} \rangle = 1 \, .$

Why interpolation error is a reliable error estimate ?

- u solution of the problem is unknown
- u_h the numerical solution linear by element

Principle: in our approach the theoretical estimate is used indirectly *i.e.*, the estimate is not apply to u_h

Idea: reconstruct a solution $u^* \in \mathcal{C}^1$ from u_h which is higher order and consider:

$$\|u - u_h\| \approx \|u^* - u_h\| \,.$$

We apply our results to u^*

$$\|u^* - \Pi_h u^*\|_{\infty,K} \leq c \max_{x \in K} \max_{e \in E_K} \langle \vec{e}\,,\, |H_{u^*}(x)|\, \vec{e}\,\rangle\,,$$

How to reconstruct u^* such that $\Pi_h u^* = u_h$?

Hessian computation by a double $\mathbb{L}^2$ projection

- **Notations:**

  $\mathcal{H}$ a mesh of the domain $\Omega \subset \mathbb{R}^d$

  $V_h = \{v \in H^1_0(\Omega) \mid \forall K \in \mathcal{H}\,,\; v_{|K} \in P^0\}$ the approximation space

  $\varphi_i \in V_h$ a basis function

  S_i the stencil of φ_i, *i.e.*, $S_i = supp\,\varphi_i$

- P^0 **Clément interpolation operator:**

  Idea: $v \in L^2(\Omega)$ is projected in P^0 function space independently on each $S_i \subset \mathcal{H}$

  We define $\Pi_0 v \in L^2(\Omega)$ by

$$\forall S_i \subset \mathcal{T}_h\,, \qquad \begin{cases} (\Pi_0 v)_{|S_i} \in P^0\,, \\ \displaystyle\int_{S_i} (\Pi_0 v - v)p = 0\,, \quad \forall p \in P^0\,, \end{cases}$$

  and the Clément interpolation operator $\Pi_c \,:\, L^2 \rightarrow V_h$ by

$$\Pi_c\, v = \sum_{j=1}^{n} \Pi_0 v(P_j)\,\varphi_j\,.$$

Hessian computation by a double $\mathbb{L}^2$ projection

- **Remark:** this operator is also called local $\mathbb{L}^2$ projection operator
- **Gradient reconstruction operator:**

$$\nabla_R\, u_h = \Pi_c(\nabla u_h) := \left[\Pi_c(\frac{\partial u_h}{\partial x}),\, \Pi_c(\frac{\partial u_h}{\partial y}),\, \Pi_c(\frac{\partial u_h}{\partial z})\right] .$$

For each S_i, we have by definition, for each $p \in P^0$:

$$\begin{aligned}\int_{S_i} (\Pi_0(\nabla u_h) - \nabla u_h)\, p = 0 \quad &\Longleftrightarrow \quad \int_{S_i} \Pi_0(\nabla u_h) = \int_{S_i} \nabla u_h\,, \\ &\Longleftrightarrow \quad |S_i|\, (\Pi_0(\nabla u_h))_{|S_i} = \sum_{K \in S_i} \int_K \nabla u_h\,, \\ &\Longleftrightarrow \quad (\Pi_0 \nabla u_h))_{|S_i} = \frac{\sum_{K \in S_i} |K| \nabla(u_{h|K})}{|S_i|}\,.\end{aligned}$$

Hessian computation by a double $\mathbb{L}^2$ projection

- **Remark:** this operator is also called local $\mathbb{L}^2$ projection operator
- **Gradient reconstruction operator:**

  For each vertex P_i

$$\nabla_R\, u_h(P_i) = \Pi_c(\nabla u_h)(P_i) = \Pi_0(\nabla u_h)(P_i)\,.$$

We deduce

$$\nabla_R\, u_h(P_i) = \frac{\sum_{K \in S_i} |K| \nabla(u_{h|K})}{|S_i|}\,.$$

Remark: This reconstruction is nothing else than the average of the neighboring triangles' gradients weighted by the areas.

Hessian computation by a double $\mathbb{L}^2$ projection

- **Hessian reconstruction operator:**

$$H_R(u_h) = \nabla_R((\nabla_R u_h)^t) = {}^t\begin{pmatrix} \nabla_R(\Pi_c(\frac{\partial u_h}{\partial x})) \\ \nabla_R(\Pi_c(\frac{\partial u_h}{\partial y})) \\ \nabla_R(\Pi_c(\frac{\partial u_h}{\partial z})) \end{pmatrix},$$

with for each component x_j, $1 \leq j \leq d$:

$$\nabla_R(\Pi_c(\frac{\partial u_h}{\partial x_j}))(P_i) = \frac{\sum_{K\in S_i} |K| \nabla((\Pi_c(\frac{\partial u_h}{\partial x}))_{|K})}{|S_i|}.$$

Hessian computation by a least square method

Idea: Find the best function u^* that fits u_h at the least square sense

- We consider the ball of P: $\mathcal{B}(P)$
- For each $P_i \in \mathcal{B}(P)$, a Taylor expansion provides:

$$\frac{1}{2}\langle \overrightarrow{PP_i},\, H_{u^*}(P)\,\overrightarrow{PP_i}\rangle = u^*(Pi) - u^*(P) - \overrightarrow{PP_i}\,.\,\nabla u^*(P)\,,$$

$$\text{or}\quad \frac{1}{2}(a\,x_i^2 + 2\,bx_iy_i + c\,y_i^2) = u_i^* - u_p^* - (\alpha x_i + \beta y_i)\,,$$

- We get an over-determined linear system of the form $AX = B$.
- Solve it by an approximation at the least square sense:

$$\text{Find } X \in \mathbb{R}^3 \text{ such that } \|AX - B\|^2 = \inf_{Y\in\mathbb{R}^3} \|AY - B\|^2\,.$$

which is equivalent to

$$\text{Find } X \in \mathbb{R}^3 \text{ such that } A^T AX = A^T B\,.$$

Practical metric construction

- We have the following bound of the interpolation error:

$$\| u - \Pi_h u \|_{\infty,K} \leq c_d \max_{x \in K} \max_{e \in E_K} \langle \vec{e}, |H_u(x)| \, \vec{e} \rangle$$

- Let $\overline{\mathcal{M}}(K)$ be an adequate metric tensor.
  Then, we estimate on each element K the error by:

$$\varepsilon_K = c_d \max_{e \in E_K} \langle \vec{e}, \overline{\mathcal{M}}(K) \, \vec{e} \rangle \,.$$

- A metric providing a unit length !

If we put: $\mathcal{M}(K) = \dfrac{c_d \, \overline{\mathcal{M}}(K)}{\varepsilon_K} \implies \max_{e \in E_K} \langle \vec{e}, \mathcal{M}(K) \, \vec{e} \rangle = 1 \,.$

Practical metric construction

- Variables for which we will equidistribute the interpolation error are choosen.
- The desired error threshold ε is set.
- Minimal h_{min} and maximal h_{max} sizes authorized for mesh generation are fixed.
- $\mathcal{M}$ is (discretely) constructed:

$$\mathcal{M} = \mathcal{R}\,\tilde{\Lambda}\,\mathcal{R}^{-1} \quad \text{with} \quad \tilde{\Lambda} = \begin{pmatrix} \tilde{\lambda}_1 & 0 \\ 0 & \tilde{\lambda}_2 \end{pmatrix} \quad \text{and}$$

$$\tilde{\lambda}_i = \min\left(\max\left(\frac{c_d\,|\lambda_i|}{\varepsilon}, \frac{1}{h_{max}^2}\right), \frac{1}{h_{min}^2}\right).$$

Practical metric construction

$$\mathcal{M} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$$

- Eigenvalues computation:

  They are given by $\det(\mathcal{M} - \lambda \mathrm{Id})$

  We set $\Delta = (a - c)^2 + 4\,b^2$, we have

$$\lambda_1 = \frac{a + c + \sqrt{\Delta}}{2} \quad \text{and} \quad \lambda_2 = \frac{a + c - \sqrt{\Delta}}{2}$$

$$\mathcal{M} = \begin{pmatrix} a & b \\ b & c \end{pmatrix}$$

- Eigenvectors computation:

$\vec{v_1} \in \mathrm{Ker}(\mathcal{M} - \lambda_1 \,\mathrm{Id}) \Longrightarrow$ solve the system: $\mathcal{M}\,\vec{v_1} = \lambda_1\,\vec{v_1}$

Let be $\vec{v_1} = {}^t(x_1, y_1)$, we find $\vec{v_1} = \begin{pmatrix} 1 \\ \frac{\lambda_1 - a}{b} \end{pmatrix}$.

Remark: $\vec{v_1} \in (\mathrm{Im}(\mathcal{M} - \lambda_1 \,\mathrm{Id}))^{\perp}$

Choose $\vec{w_1} \in \mathrm{Im}(\mathcal{M} - \lambda_1 \,\mathrm{Id})$ define by:

$$\vec{w_1} = (\mathcal{M} - \lambda_1 \,\mathrm{Id}) \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} a - \lambda_1 \\ b \end{pmatrix}$$

Thus, we get $\vec{v_1} = \begin{pmatrix} -b \\ a - \lambda_1 \end{pmatrix}$.

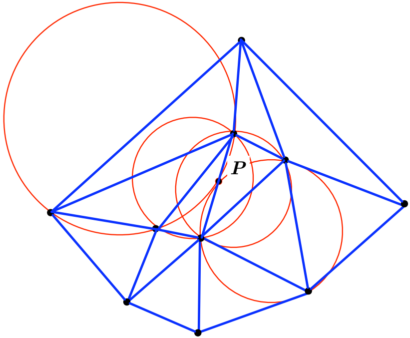
Delaunay Triangulation

- $\mathcal{T}$ is a Delaunay mesh iff

$$\forall (K, K') \in \mathcal{T} \text{ such that } K = adj(K') \text{ we have } \mathring{\mathcal{B}}(K) \cap \mathcal{V}(K') = \emptyset$$

- The Delaunay measure is defined as:

$$\alpha(K, P) = \frac{d(P, O_K)}{r(K)} \quad \Longrightarrow \quad K \in \mathcal{C}_P \text{ iff } \alpha(K, P) \leq 1$$



Incremental method [Hermeline 1980, Bowyer-Watson 1981]:

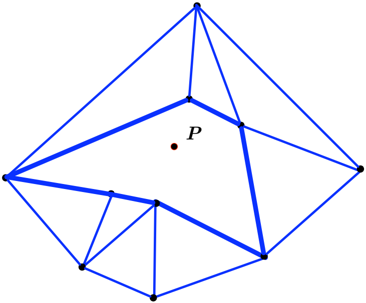
(Delaunay) Triangulation $\mathcal{T}_n$

- $\mathcal{T}$ is a Delaunay mesh iff

$$\forall (K, K') \in \mathcal{T} \text{ such that } K = adj(K') \text{ we have } \mathring{\mathcal{B}}(K) \cap \mathcal{V}(K') = \emptyset$$

- The Delaunay measure is defined as:

$$\alpha(K, P) = \frac{d(P, O_K)}{r(K)} \implies K \in \mathcal{C}_P \text{ iff } \alpha(K, P) \leq 1$$



Remove the cavity $\mathcal{C}_n$ of P

$$\mathcal{T}_n - \mathcal{C}_n$$

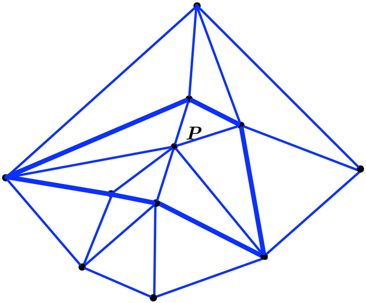
Delaunay Triangulation

- $\mathcal{T}$ is a Delaunay mesh iff

$$\forall (K, K') \in \mathcal{T} \text{ such that } K = adj(K') \text{ we have } \mathring{\mathcal{B}}(K) \cap \mathcal{V}(K') = \emptyset$$

- The Delaunay measure is defined as:

$$\alpha(K, P) = \frac{d(P, O_K)}{r(K)} \quad \Longrightarrow \quad K \in \mathcal{C}_P \text{ iff } \alpha(K, P) \leq 1$$



Star connection of the cavity

$$\mathcal{T}_{n+1} = \mathcal{T}_n - \mathcal{C}_n \cup \mathcal{B}_{n+1}$$

To find O center of the circumscribed ball of tetra K, solve:

$$d(P_i, O) = d(P_j, 0)\,, \quad \forall P_i \neq P_j \in K\,,$$

it yields the linear system:

$$\begin{pmatrix} x_{12} & y_{12} & z_{12} \\ x_{13} & y_{13} & z_{13} \\ x_{14} & y_{14} & z_{14} \end{pmatrix} \cdot \begin{pmatrix} O_x \\ O_y \\ O_z \end{pmatrix} = \begin{pmatrix} d_{12} \\ d_{13} \\ d_{14} \end{pmatrix}\,,$$

with $d_{1i} = (x_i^2 + y_i^2 + z_i^2) - (x_1^2 + y_1^2 + z_1^2)$ and $x_{1i} = 2\,(x_i - x_1)$.

The solution of the system $A\,.\,O = d$ is then obtained using Cramer's formula, the in-radius is simply $r(K) = d(P_1, O)$.

Delaunay Kernel: Anisotropic Case

- The Delaunay kernel remains valid
- Anisotropic Delaunay measure:

$$\alpha(K, P)_{\mathcal{M}} = \frac{\ell_{\mathcal{M}}(P, O_K)}{r_{\mathcal{M}}(K)}$$

- Given a metric tensor $\mathcal{M}(x)$, the previous system becomes:

$$\ell_{\mathcal{M}}(P_i, O) = \ell_{\mathcal{M}}(P_j, O)\,, \quad \forall P_i \neq P_j \in K\,.$$

Then, once solved we have $r_{\mathcal{M}}(K) = \ell_{\mathcal{M}}(P_1, O)$

But which $\mathcal{M}$?

$$\ell_{\mathcal{M}(P_i)}(P_i, O) = \ell_{\mathcal{M}(P_j)}(P_j, O)\,, \;\; \forall P_i \neq P_j \in K$$

$\Longrightarrow$ non linear system

- There are several ways of solving this equations, *i.e.*, to define the Delaunay measure:
  1. $\alpha(K,P)_{\mathcal{M}(P)} \leq 1$,
  2. $\alpha(K,P)_{\mathcal{M}(P)} + \sum_{i=1}^{4} \alpha(K,P)_{\mathcal{M}(P_i)} \leq 5$,
  3. $\alpha(K,P)_{\mathcal{M}(P)} + \sum_{i=1}^{4} \omega_i \alpha(K,P)_{\mathcal{M}(P_i)} \leq 1 + \sum_{i=1}^{4} \omega_i$.

- Star connection of the cavity + connected:

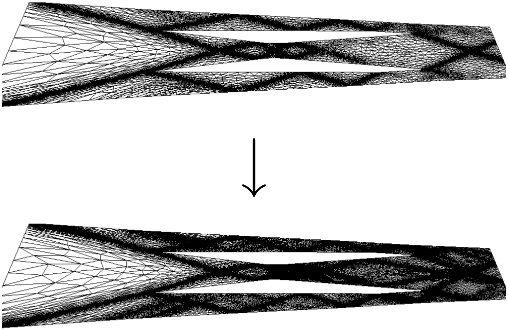
  $\Longrightarrow$ necessary cavity correction phase

Adapted Mesh Generation

- Complete a unit mesh in the metric
- Global or local remeshing
- Anisotropic Delaunay kernel based vertex insertion
- All local mesh modification operations (swap, deletion, ...)
- Three steps algorithm:
  1. mesh analysis
  2. mesh adaptation
  3. mesh optimization

Solution Interpolation

- Previous solution field is transfered on the new mesh



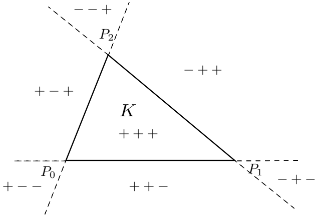
- Two steps algorithm:
  1. Vertices localization in the background mesh
  2. Solution interpolation

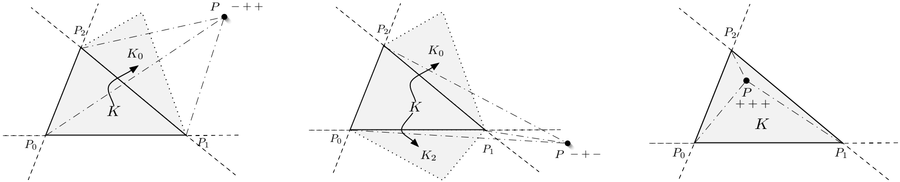
Solution Interpolation

- Some notations:
  - P vertex of the new mesh
  - $K = [P_1, P_2, P_3]$ triangle of the background mesh
  - A_K the area of K
  - K_i the virtual triangle where P replace the vertex P_i
  - A_i the area of K_i
- the three barycentric coordinates of P w.r.t K are $\beta_i = \dfrac{A_i}{A_K}$

Localization:

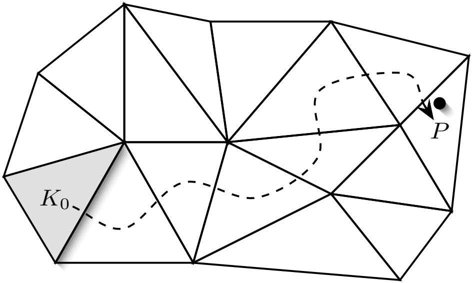
- mesh is oriented $\Longrightarrow$ signed area or volume (positive)
- barycentric coordinates of a vertex for an element:





Localization:

- mesh is oriented $\Longrightarrow$ signed area or volume (positive)
- algorithm based on barycentric using mesh topology to move inside the mesh
  This algorithm has a complexity of $N_{ver} * n_{move}$



- Reduce n_{move} with a scheme using the topology of both meshes

Solution interpolation:

Assuming we have a piecewise linear representation of the solution

- P^1 projection operator is defined by

$$u(P) = \sum_{i=1}^{4} \beta_i(P)\, u(P_i)\,, \quad \text{with} \;\; \beta_i(P) = \frac{A_i}{A_K}\,.$$