An Overview of the GMSH project

Jean-François Remacle[1]

[1] Université catholique de Louvain (UCL), Belgium. http://www.geuz.org/gmsh

December 5, 2011

Generalities

Gmsh is

an OPEN SOURCE finite element grid generator with build-in with pre-post processing capabilities

Outline

- A brief history of the project
- Strategic choices
  - Programming language
  - Building system
  - Licensing
  - Building a community
  - Academic (and financial ?) recognition
- Research topics in GMSH
  - CAD Interface
  - Surface reparametrization
  - Quad-Hex meshing
  - High order – curvilinear meshes

Outline

History

Milestones

- Started in 1996
- First public release in 1998
- Open Source (GPL) in 2003
- Gmsh 2.0 in 2006
- IJNME paper in 2009
- Move form autotools to CMake in 2010
- Gmsh as a library in 2010
  - Python bindings
  - A complete API available
  - Aggregated projects (DGM, Ocean modelling, NL shells....)

Funding

- First funding in 2006 (EDF)
- More recently, IDIHOM, DOMHEX, ONELAB, EASYMESH, GHNOME...

Outline

Strategic choices

Programming Language

- Gmsh is written only in C++
- We try to impose some kind of programming style
  - Files, functions and variables namings
  - Uniform file indenting
  - Avoid exotic C++ constructs (Partial Template Specialization, multiple inheritance...)

Third party softwares

- The building system allows to ON/OFF any 3rd party contribution
- Use standard packaging tools for installation (macports, apt-get...): FLTK, OpenMPI, PETSc, SLEPc, SWIG-PYTHON...
- If possible, re-distribute: METIS, NETGEN, ANN, Chaco, bamg, MathEx...
- Others are more tricky: OCC, TAUCS, TETGEN, Concorde...

Strategic choices

Building system and related tools

- CMake is the building system of Gmsh since 2010
- Management & community tools
  - Automated nightly builds are available, for Linux, Mac OSX and Windows
  - Two mailing lists are maintained
  - A a time line of changes and the bug tracking database is maintained

Documentation

- A basic web site is maintained (`http://www.geuz.org/gmsh`)
- A complete user-documentation is available
- A tutorial introducing all key features and concepts is included in all the versions in the tutorial directory.
- Scientific aspects of algorithms are detailed in journal papers
- A programmer's guide is high on our todo list

Strategic choices

Licensing

- GMSH is licensed under the GPL
  - The GPL is viral (that's Microsoft's interpretation)
  - In reality, the GPL is a very good protection for the (2) copyright's owners
- Double-licensing
  - Allow to include parts of the software in closed source

Third party softwares

- All 3rd party softwares have to be compatible with the GPL if redistributed
  - No problem with LGPL third parties (ANN)
  - Some specific arrangements have been done with G. Karypis (Metis) or with F. Hecht (Bamg)
  - Some 3rd party softwares are NOT compatible: NETGEN, Concorde. A great care has to be taken in order to maintain that compatibility.

Strategic choices

The Community

- There are more than 1,000 daily users of Gmsh, regarding to the mailing list
- Since 2006, more and more industrials are interrested in Gmsh
- Only a few ($\simeq 15$) are commiting in the source
- A lot of users are active on the mailing list

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Strategic choices

Academic Recognition

- Publish or perish
- PETSc users manual (2004): 920 citations (Scholar)
- LAPACK Users' guide (1999): 3278 citations (Scholar)
- The Gmsh paper (`DOI:10.1002/nme.2579`) has been published in dec. 2009 in IJNME
- Its citation count is 281 (Scholar) and 91 (Thompson ISI Web of science)
- It is the paper of 2009 that is the most cited in IJNME: `http://twitter.com/#!/WileyNumEng/status/25871124914`

Financial Recognition

- Is it adequate to talk about money ?
- Double licensing is an opportunity

Outline

Research and developpement in Gmsh

New stuff in Gmsh

- Remeshing surfaces with parametrizations
- High order/curvilinear meshes
- Quad and Hex meshing

References

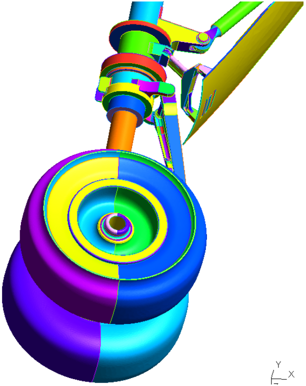
J.-F. Remacle, C. Geuzaine, G. Compère, E. Marchandise
High Quality Surface Remeshing Using Harmonic Maps, IJNME, 83, pp. 403-425, 2010.

E. Marchandise, M.Willemet, G. Bricteux, C. Geuzaine, J-F. Remacle
Quality meshing based on STL triangulations for biomedical simulations, IJNMBE, 83, pp. 876-889, 2010.

E. Marchandise, C. Carton de Wiart, W. Vos, C. Geuzaine, .J-F. Remacle,
High Quality Surface Remeshing Using Harmonic Maps: PARTII: Surfaces with high genus and of large aspect ratio, IJNME, 86, pp 1303-1321, 2011

Reparametrization

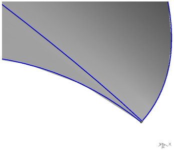
CAD data is not suitable for FE analysis

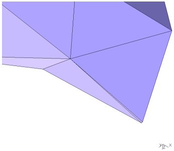


- Geometric models contain a lot of patches
- Example: CAD data issued form CATIA^{TM}: 852 patches
- Are not suitable for FE analysis
  1. small model edges are present in the patches
  2. that lead to triangles of poor quality
- Reparametrize through existing patches could be highly useful
  1. 291 surface patches remaining
  2. suitable CFD mesh build from remeshed surface

Remeshing:

based on cross-patch parametrization
Marcum 1999, Aftomosis 1999





- Geometric models contain a lot of patches
- Example: CAD data issued form CATIA^{TM}: 852 patches
- Are not suitable for FE analysis
    1. small model edges are present in the patches
    2. that lead to triangles of poor quality
- Reparametrize through existing patches could be highly useful
    1. 291 surface patches remaining
    2. suitable CFD mesh build from remeshed surface

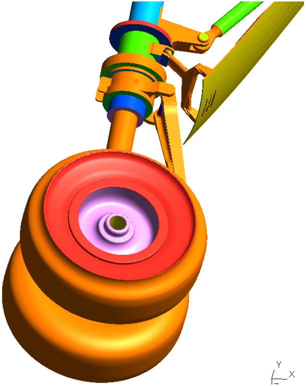
Remeshing:

based on cross-patch parametrization
Marcum 1999, Aftomosis 1999

- Geometric models contain a lot of patches
- Example: CAD data issued form CATIA^{TM}: 852 patches
- Are not suitable for FE analysis
  1. small model edges are present in the patches
  2. that lead to triangles of poor quality
- Reparametrize through existing patches could be highly useful
  1. 291 surface patches remaining
  2. suitable CFD mesh build from remeshed surface

Remeshing:

based on cross-patch parametrization
Marcum 1999, Aftomosis 1999



- Geometric models contain a lot of patches
- Example: CAD data issued form CATIA^{TM}: 852 patches
- Are not suitable for FE analysis
  1. small model edges are present in the patches
  2. that lead to triangles of poor quality
- Reparametrize through existing patches could be highly useful
  1. 291 surface patches remaining
  2. suitable CFD mesh build from remeshed surface

Remeshing:

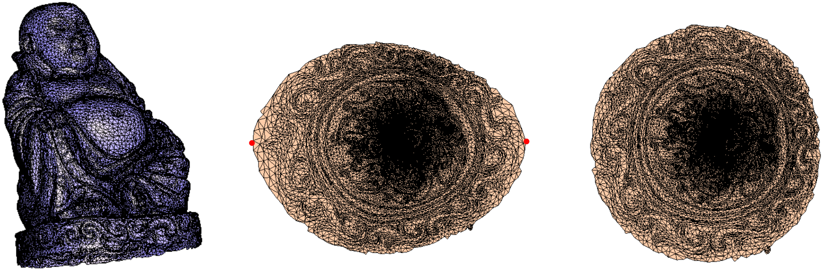
based on cross-patch parametrization
Marcum 1999, Aftomosis 1999

Reparametrization

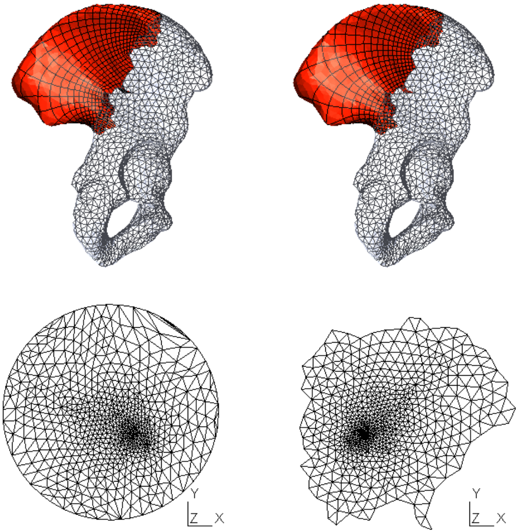
Computing Conformal maps

The eigenvector $\mathbf{u}^*$ (Fiedler vector) associated to the smallest eigenvalue λ, i.e. $L_C\mathbf{u}^* = \lambda\mathbf{u}^*$ is the solution to the constrained minimization problem:

$$\mathbf{u}^* = \underset{\mathbf{u},\mathbf{u}^t\mathbf{e}=0,\mathbf{u}^t\mathbf{u}=1}{\arg\min} \quad \mathbf{u}^t L_C \mathbf{u} \tag{1}$$



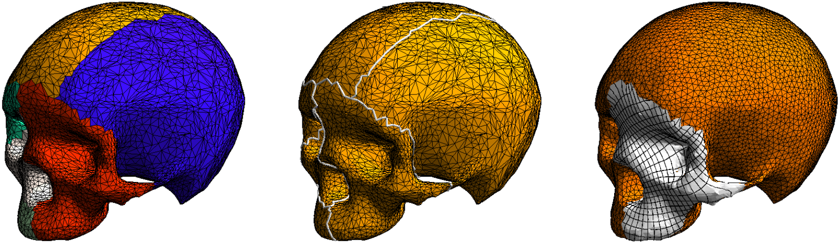
Laplacian and Conformal harmonic maps



Laplacian

Conformal

Automatic remeshing

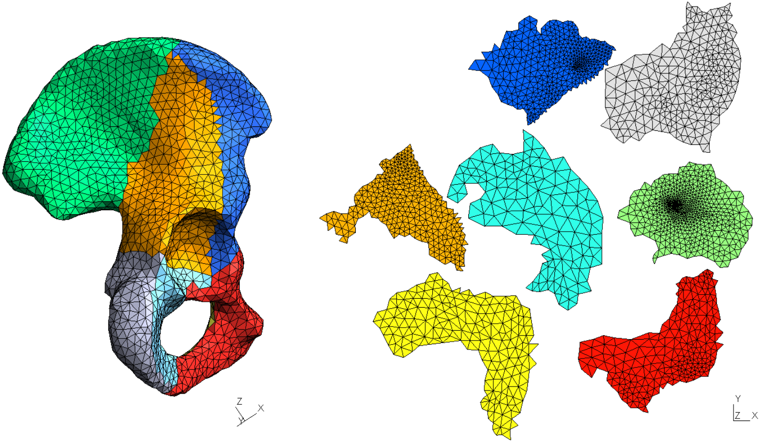


a) b) c)

Remeshing algorithm.

- Compute $G = 2, N_B = 0, \eta_{G=0}$
- If needed cut mesh into different partitions of zero genus,
- Remesh the lines at the interfaces between partition
- Compute mapping for every partition and remesh the partition in the parametric space ($\mathbf{u}(\mathbf{x})$ coordinates visible for one partition).

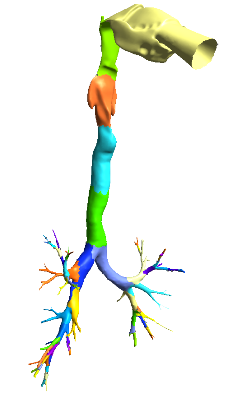
Automatic remeshing



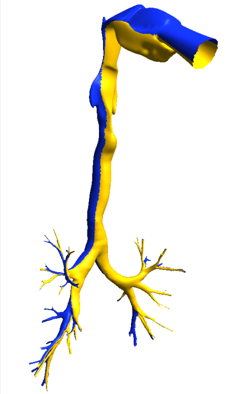
Different mesh partitions of a hemi-pelvis in the parametric space (conformal map)

How to partition the mesh ?

Partitioning a human lung of aspect ratio $\eta = 89$



Multilevel

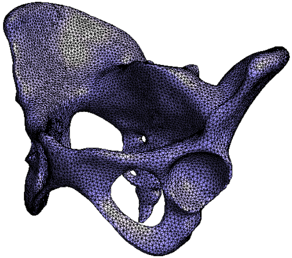


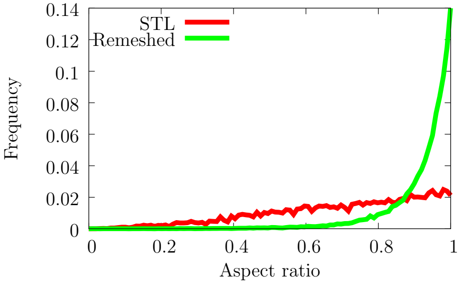
Max-cut mesh partitionner
Based on multiscale Laplace map

E. Marchandise, et al. *High Quality Surface Remeshing Using Harmonic Maps: PARTII: Surfaces with high genus and of large aspect ratio*, IJNME, online Jan 2011

High quality meshing for the Laplacian harmonic map

Plot of the quality histogram with high mean quality $\bar{\kappa} = 0.94$:

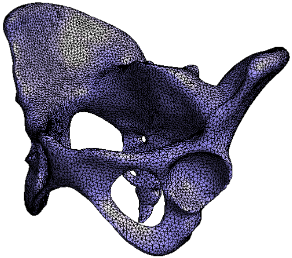


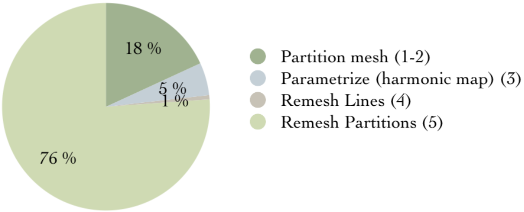


$$\kappa = \alpha \frac{\text{inscribed radius}}{\text{circumscribed radius}} = 4 \frac{\sin \hat{a} \sin \hat{b} \sin \hat{c}}{\sin \hat{a} + \sin \hat{b} + \sin \hat{c}}, \tag{2}$$

High quality meshing for the Laplacian harmonic map

Time spent in the different steps of the automatic remeshing



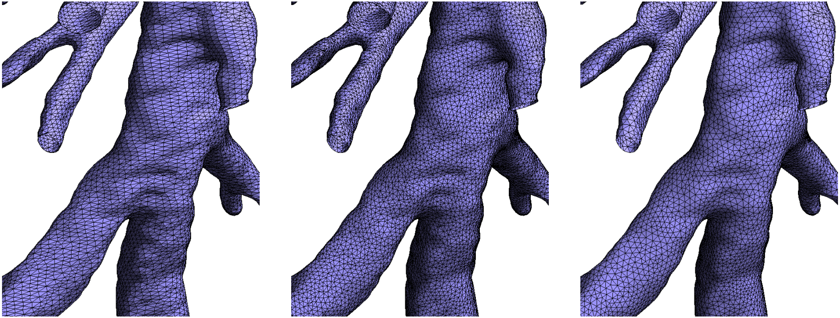


The total time for remeshing the initial mesh of $25k$ triangles is $9s$.

CPU time $< 100s$ for mesh of $1.e^6$ elements

High quality meshing

Comparisons with other techniques for surface remeshing:



a) b) c)

Figure: Remeshing of human lungs: a) part of the initial STL triangulation, b) remeshed geometry with Mimics (after 2 steps) and c) remeshed lung based on the Harmonic mapping remeshing procedure.

High quality meshing

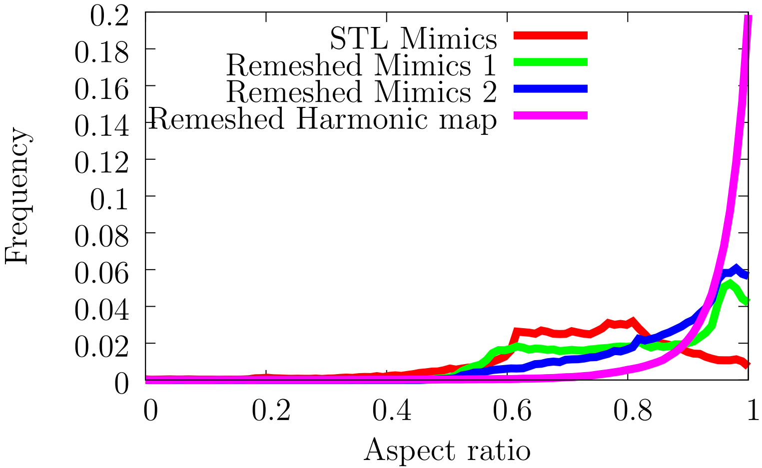
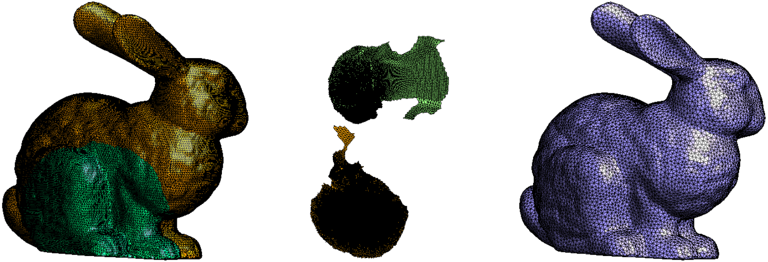


Figure: Remeshing of human lungs with the presented algorithm as compared with a commercial package such as Mimics.

Efficient surface remeshing

Remeshing of the bunny mesh model of $70k$ triangles



Remeshing	Number of partitions	Partition time (s)	Parametrization time (s)	Total remeshing time (s)
LSCM Levy	23	30	95	–
Eck	88	-	-	33.5
ABF++ Zayer	2	-	13	-
LinABF Zayer	2	-	2	-
Present work				
* laplacian part.	2	16.7	1.4	25
* metis part.	10	7	1.4	14

Research and developpement in Gmsh

New stuff in Gmsh

- Remeshing surfaces with parametrizations
- High order/curvilinear meshes
- Quad and Hex meshing

References

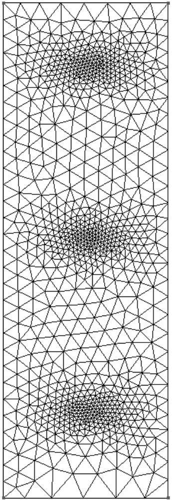
J.-F. Remacle et. al.
Blossom-Quad: a non-uniform quadrilateral mesh generator using a minimum cost perfect matching algorithm , IJNME, In press, 2011.

J.-F. Remacle. et al.
A frontal Delaunay quad mesh generator using the L^{∞} norm, IJNME, Accepted, 2011.

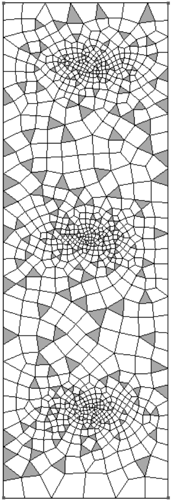
What we aim at

Indirect approach: a non-optimal matching algorithm

$h(x,y) = 0.1 + 0.08\sin(3x)\cos(6y)$, 836 quads and 240 triangles



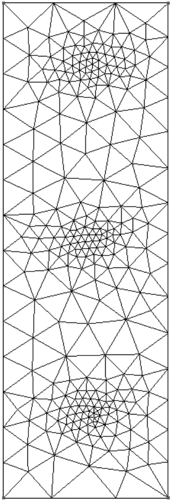
$\tau = 0.888825$



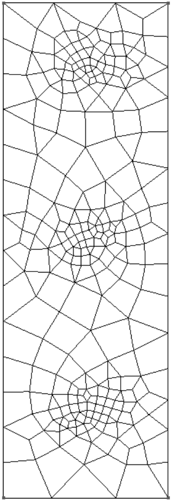
$\tau = 0.855715$

Indirect approach: a full-quad approach

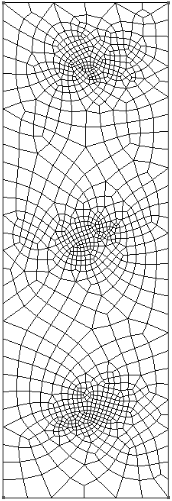
$h(x,y) = 0.1 + 0.08\sin(3x)\cos(6y)$, only quads



$\tau = 0.854955$



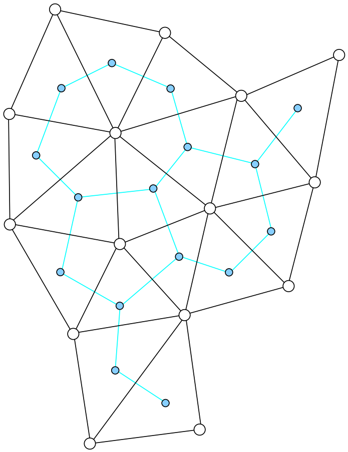
$\tau = 0.838941$



$\tau = 0.790898$

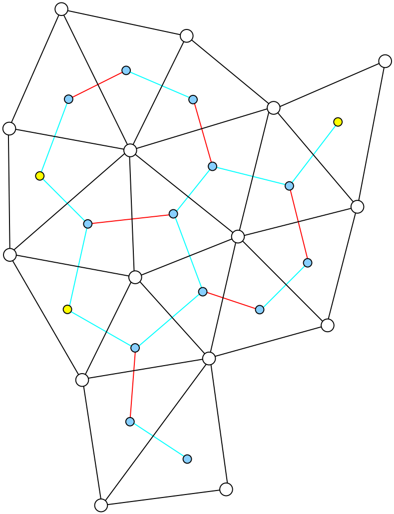
Indirect approach: the Blossom-Quad approach

We build $G(V, E, c)$ an undirected weighted graph. Here, V is the set of n_V vertices, E is the set of n_E undirected edges and $c(E) = \sum c(e_{ij})$ is an edge-based cost function, i.e., the sum of all weights associated to every edge $e_{ij} \in E$ of the graph.

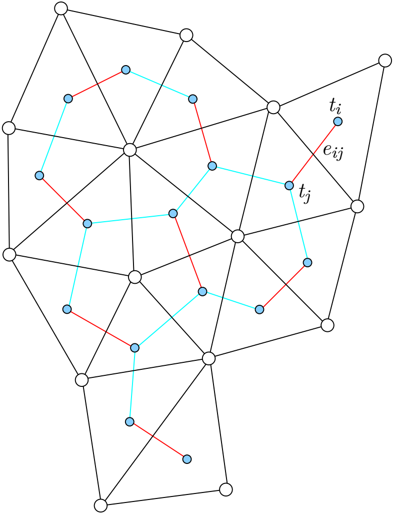


Indirect approach: the Blossom-Quad approach

A *matching* is a subset $E' \subseteq E$ such that each node of V has at most one incident edge in E'.



A matching is perfect if each node of V has exactly one incident edge in E'. A perfect matching is optimum if $c(E')$ is minimum among all possible perfect matchings.

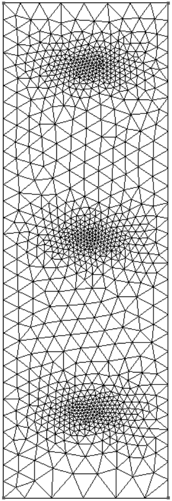


Indirect approach: the Blossom-Quad approach

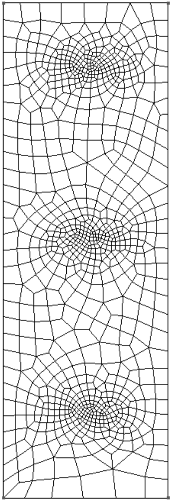
- In 1965, Edmonds invented the *Blossom algorithm* that solves the problem of optimum perfect matching in polynomial time. A straightforward implementation of Edmonds's algorithm requires $\mathcal{O}(n_V^2 n_E)$ operations.
- Since then, the worst-case complexity of the Blossom algorithm has been steadily improving. Both Lawler and Gabow achieved a running time of $\mathcal{O}(n_V^3)$. Galil, Micali and Gabow improved it to $\mathcal{O}(n_V n_E \log(n_V))$. The current best known result in terms of n_V and n_E is $\mathcal{O}(n_V(n_E + \log n_V))$.
- There is also a long history of computer implementations of the Blossom algorithm, starting with the Blossom I code of Edmonds, Johnson and Lockhart. In this paper, our implementation makes use of the Blossom IV code of Cook and Rohe[1] that has been considered for several years as the fastest available implementation of the Blossom algorithm. Gmsh redistributes this piece of code.

[1] `http://www2.isye.gatech.edu/?wcook/blossom4/`.

Indirect approach: Blossom-Quad
$h(x,y) = 0.1 + 0.08\sin(3x)\cos(6y)$, only quads



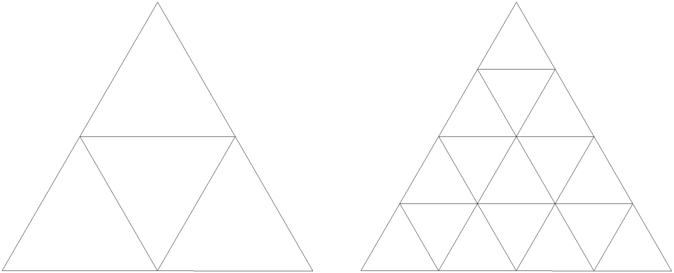
$\tau = 0.888825$



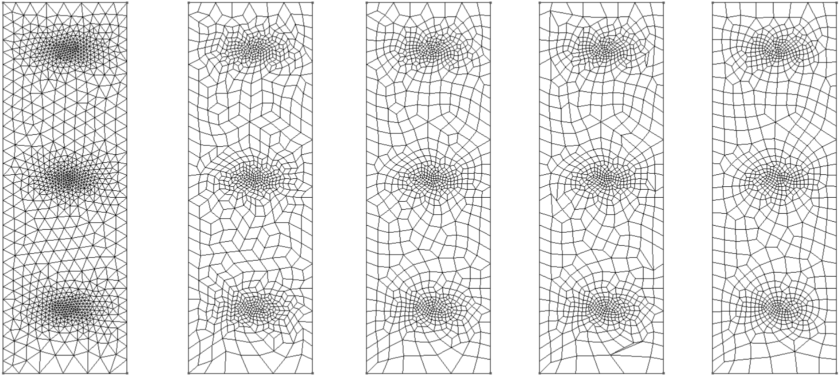
$\tau = 0.839313$

Existence of perfect matchings

- There is no guarantee that even one single perfect matching exists in a given graph.
- Consider the meshes below. It is obvious that no perfect matching exists for the coarsest one.
- The following result, known as Tutte's theorem, proves that none of the two meshes contains a perfect matching.



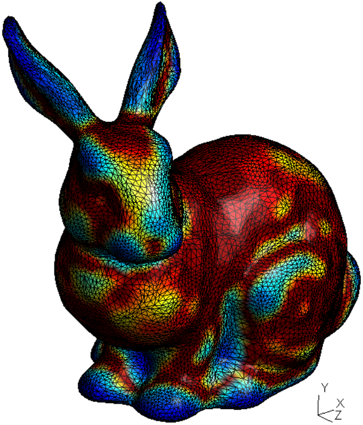
The Blossom-Quad algorithm



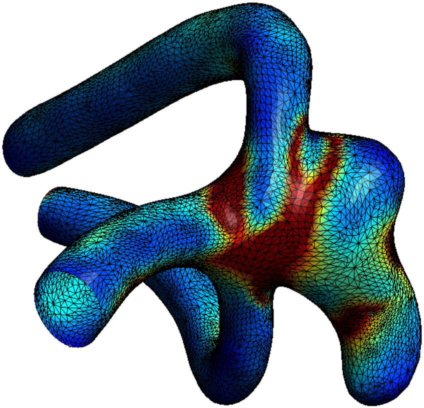
Initial triangulation | Raw Blossom application | Vertex smoothing | Topological optimization | Final mesh

Quad mesh generation applied to STL models

$h(\vec{x}) = \frac{2\pi R(\vec{x})}{N_p}, \quad \text{with } R(\vec{x}) = \frac{1}{\kappa(\vec{x})}, N_P = 50$

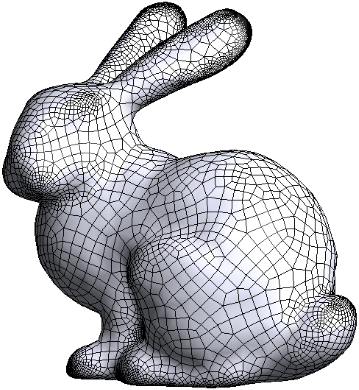


Stanford Bunny

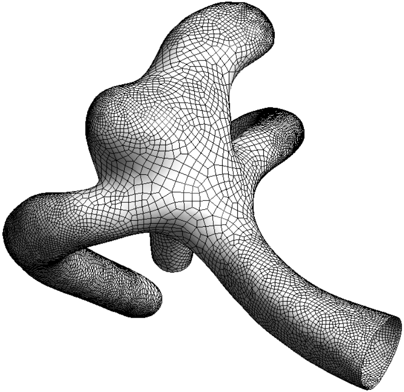


Aneurysm

Quad mesh generation applied to STL models



$\tau = 0.842549$

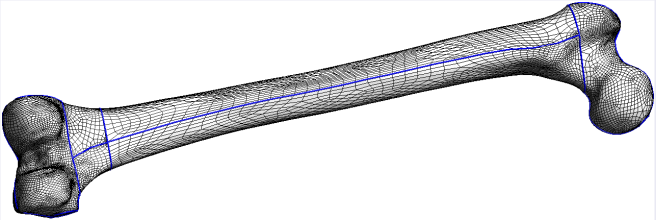


$\tau = 0.856247$

- The overall remeshing procedure for both STL examples takes only $20s$ ($5s$ for the Blossom-Quad).
- The quad-dominant meshing algorithm of [Levy et al.] takes $271s$ for the remeshing of the Stanford bunny.

Anisotropic quad mesh generation applied to a parametric model

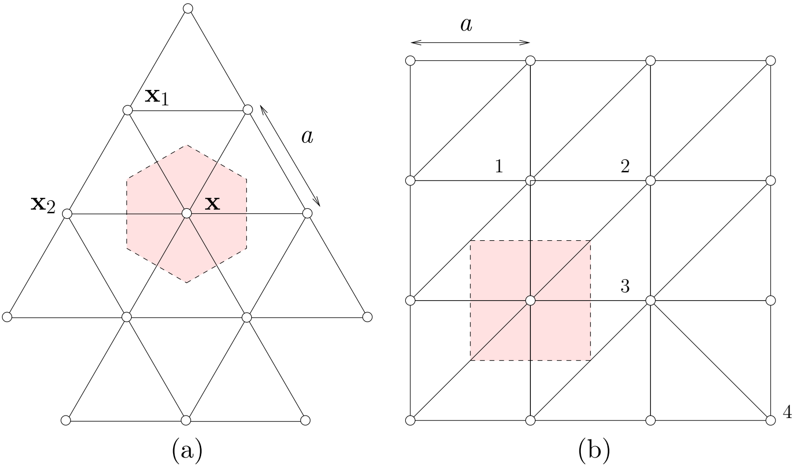
Anisotropic mesh:



Anisotropic metric

- Build an anisotropic triangular mesh
- Reimannian metric is used to measure distances
- Here, principal directions of the curvature tensor are used

Indirect approach: an obvious problem



Voronoi cells of one vertex that belongs either to mesh of a equilateral triangles (a) or of right triangles (b).

Indirect approach: an obvious problem

- Consider a uniform mesh of R^2 made of equilateral triangles of size a.
- The Voronoi cell relative to each vertex of this mesh is an hexagon of area $a^2\sqrt{3}/2$. The number $2/(a^2\sqrt{3})$, is the number of points per unit of surface of this mesh.
- Assume now a uniform mesh of R^2 made of squares of size a.
- The Voronoi cell relative to each vertex of this mesh is a square of area a^2.
- This means that filling R^2 with equilateral triangles requires $2/\sqrt{3}$ times more vertices than filling the same space with squares.
- So, even though it is always possible to build a mesh made of quadrangles by recombining triangles, a good triangular mesh made of equilateral triangles contains about $2/\sqrt{3}$ times too many to make a good quadrilateral mesh.

Indirect approach: an obvious problem

- Consider a uniform mesh of R^2 made of equilateral triangles of size a.
- The Voronoi cell relative to each vertex of this mesh is an hexagon of area $a^2\sqrt{3}/2$. The number $2/(a^2\sqrt{3})$, is the number of points per unit of surface of this mesh.
- Assume now a uniform mesh of R^2 made of squares of size a.
- The Voronoi cell relative to each vertex of this mesh is a square of area a^2.
- This means that filling R^2 with equilateral triangles requires $2/\sqrt{3}$ times more vertices than filling the same space with squares.
- So, even though it is always possible to build a mesh made of quadrangles by recombining triangles, a good triangular mesh made of equilateral triangles contains about $2/\sqrt{3}$ times too many to make a good quadrilateral mesh.

Indirect approach: an obvious problem

- Consider a uniform mesh of R^2 made of equilateral triangles of size a.
- The Voronoi cell relative to each vertex of this mesh is an hexagon of area $a^2\sqrt{3}/2$. The number $2/(a^2\sqrt{3})$, is the number of points per unit of surface of this mesh.
- Assume now a uniform mesh of R^2 made of squares of size a.
- The Voronoi cell relative to each vertex of this mesh is a square of area a^2.
- This means that filling R^2 with equilateral triangles requires $2/\sqrt{3}$ times more vertices than filling the same space with squares.
- So, even though it is always possible to build a mesh made of quadrangles by recombining triangles, a good triangular mesh made of equilateral triangles contains about $2/\sqrt{3}$ times too many to make a good quadrilateral mesh.

Indirect approach: an obvious problem

- Consider a uniform mesh of R^2 made of equilateral triangles of size a.
- The Voronoi cell relative to each vertex of this mesh is an hexagon of area $a^2\sqrt{3}/2$. The number $2/(a^2\sqrt{3})$, is the number of points per unit of surface of this mesh.
- Assume now a uniform mesh of R^2 made of squares of size a.
- The Voronoi cell relative to each vertex of this mesh is a square of area a^2.
- This means that filling R^2 with equilateral triangles requires $2/\sqrt{3}$ times more vertices than filling the same space with squares.
- So, even though it is always possible to build a mesh made of quadrangles by recombining triangles, a good triangular mesh made of equilateral triangles contains about $2/\sqrt{3}$ times too many to make a good quadrilateral mesh.

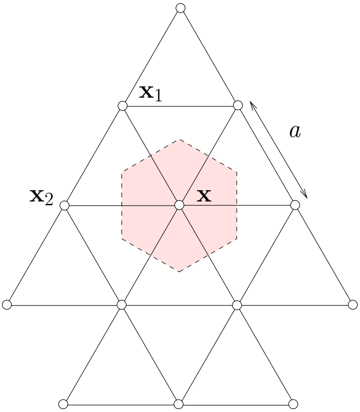
Indirect approach: an obvious problem

- Consider a uniform mesh of R^2 made of equilateral triangles of size a.
- The Voronoi cell relative to each vertex of this mesh is an hexagon of area $a^2\sqrt{3}/2$. The number $2/(a^2\sqrt{3})$, is the number of points per unit of surface of this mesh.
- Assume now a uniform mesh of R^2 made of squares of size a.
- The Voronoi cell relative to each vertex of this mesh is a square of area a^2.
- This means that filling R^2 with equilateral triangles requires $2/\sqrt{3}$ times more vertices than filling the same space with squares.
- So, even though it is always possible to build a mesh made of quadrangles by recombining triangles, a good triangular mesh made of equilateral triangles contains about $2/\sqrt{3}$ times too many to make a good quadrilateral mesh.

Indirect approach: an obvious problem

- Consider a uniform mesh of R^2 made of equilateral triangles of size a.
- The Voronoi cell relative to each vertex of this mesh is an hexagon of area $a^2\sqrt{3}/2$. The number $2/(a^2\sqrt{3})$, is the number of points per unit of surface of this mesh.
- Assume now a uniform mesh of R^2 made of squares of size a.
- The Voronoi cell relative to each vertex of this mesh is a square of area a^2.
- This means that filling R^2 with equilateral triangles requires $2/\sqrt{3}$ times more vertices than filling the same space with squares.
- So, even though it is always possible to build a mesh made of quadrangles by recombining triangles, a good triangular mesh made of equilateral triangles contains about $2/\sqrt{3}$ times too many to make a good quadrilateral mesh.

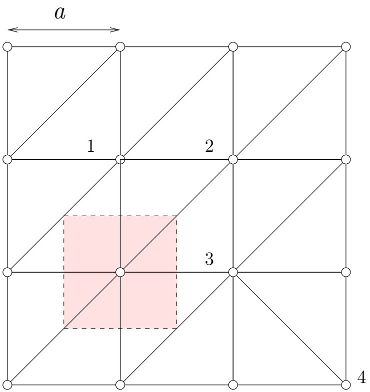
Distances and Norms



- The equilateral mesh has all its edges of size a in the Euclidian norm.
- The mesh made of right triangles has "long" edges of size $a\sqrt{2}$.
- Find out a way to measure distances in a way that all edges of the right triangles are of size a.
- One could think of using standard metric techniques. Yet, this is not the right way to go: edges at 45 degree and -45 degree should have the same size.
- Use the L^∞-norm distance:

$$\begin{aligned} \mathbf{x}_2 - \mathbf{x}_{1\,\infty} &= \lim_{p\to\infty} \mathbf{x}_2 - \mathbf{x}_{1\,p} \\ &= \max\left(|x_2 - x_1|, |y_2 - y_1|\right) \end{aligned}$$

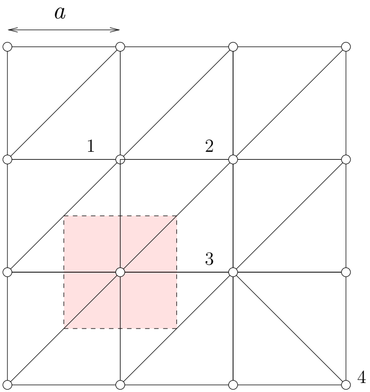
Distances and Norms



- The equilateral mesh has all its edges of size a in the Euclidian norm.
- The mesh made of right triangles has "long" edges of size $a\sqrt{2}$.
- Find out a way to measure distances in a way that all edges of the right triangles are of size a.
- One could think of using standard metric techniques. Yet, this is not the right way to go: edges at 45 degree and -45 degree should have the same size.
- Use the L^∞-norm distance:

$$\begin{aligned} \mathbf{x}_2 - \mathbf{x}_{1\,\infty} &= \lim_{p\to\infty} \mathbf{x}_2 - \mathbf{x}_{1\,p} \\ &= \max\left(|x_2 - x_1|, |y_2 - y_1|\right) \end{aligned}$$

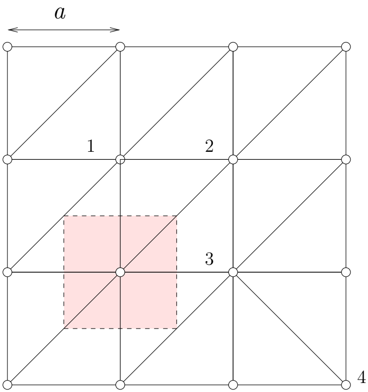
Distances and Norms



- The equilateral mesh has all its edges of size a in the Euclidian norm.
- The mesh made of right triangles has "long" edges of size $a\sqrt{2}$.
- Find out a way to measure distances in a way that all edges of the right triangles are of size a.
- One could think of using standard metric techniques. Yet, this is not the right way to go: edges at 45 degree and -45 degree should have the same size.
- Use the L^∞-norm distance:

$$\begin{aligned}\mathbf{x}_2 - \mathbf{x}_{1\infty} &= \lim_{p\to\infty} \mathbf{x}_2 - \mathbf{x}_{1p} \\ &= \max\left(|x_2 - x_1|, |y_2 - y_1|\right)\end{aligned}$$

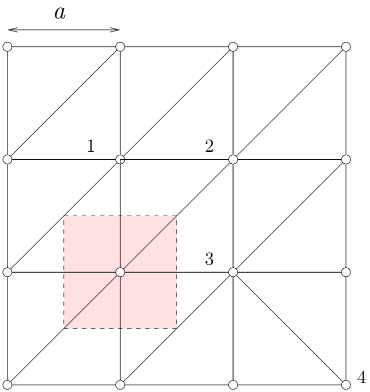
Distances and Norms



- The equilateral mesh has all its edges of size a in the Euclidian norm.
- The mesh made of right triangles has "long" edges of size $a\sqrt{2}$.
- Find out a way to measure distances in a way that all edges of the right triangles are of size a.
- One could think of using standard metric techniques. Yet, this is not the right way to go: edges at 45 degree and -45 degree should have the same size.
- Use the L^∞-norm distance:

$$
\begin{aligned}
\mathbf{x}_2 - \mathbf{x}_{1\infty} &= \lim_{p\to\infty} \mathbf{x}_2 - \mathbf{x}_{1p} \\
&= \max\left(|x_2 - x_1|, |y_2 - y_1|\right)
\end{aligned}
$$

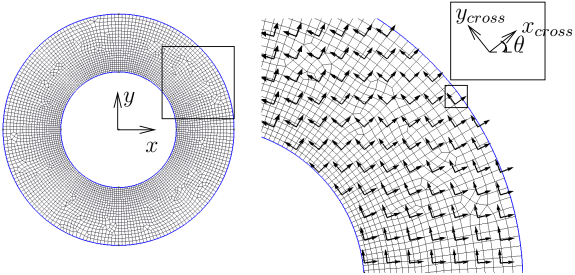
Distances and Norms



- The equilateral mesh has all its edges of size a in the Euclidian norm.
- The mesh made of right triangles has "long" edges of size $a\sqrt{2}$.
- Find out a way to measure distances in a way that all edges of the right triangles are of size a.
- One could think of using standard metric techniques. Yet, this is not the right way to go: edges at 45 degree and -45 degree should have the same size.
- Use the L^∞-norm distance:

$$\begin{aligned}\mathbf{x}_2 - \mathbf{x}_{1\infty} &= \lim_{p\to\infty} \mathbf{x}_2 - \mathbf{x}_{1p} \\ &= \max\left(|x_2 - x_1|, |y_2 - y_1|\right)\end{aligned}$$

Cross Fields

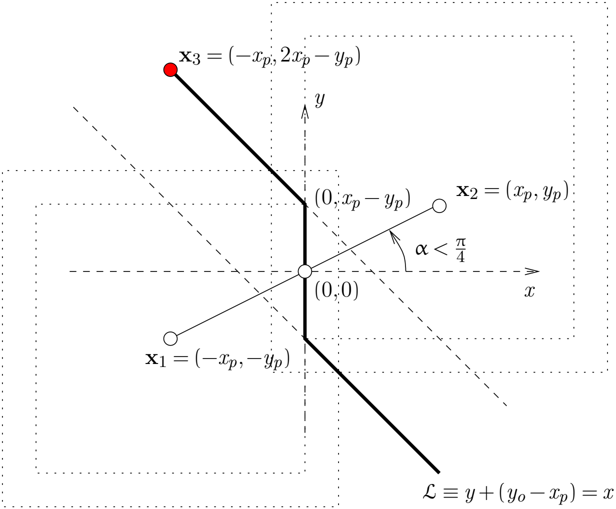


- The L^∞ norm is not invariant by rotation $\rightarrow$ quads should be oriented
- A cross field (in 2D) is a scalar field $\theta(\mathbf{u})$ that gives the orientations of a local system of axis at point $\mathbf{u}$ in the parameter plane.
- The local value of the cross field $\theta(u, v)$ is defined only up to rotations by $\pi/2$, we choose to propagate

$$\alpha(u, v) = a(u, v) + ib(u, v) = e^{4i\theta(u,v)}$$

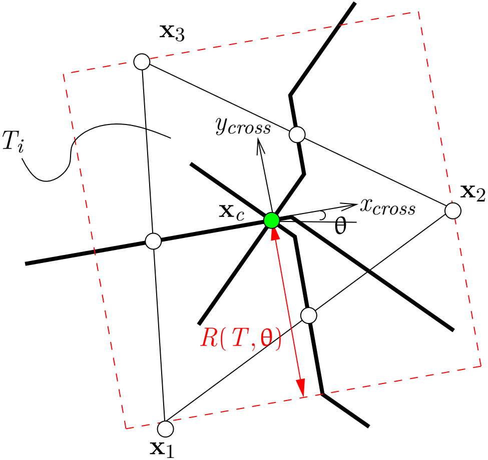
through a harmonic map.

Perpendicular bisectors in the L^∞-norm



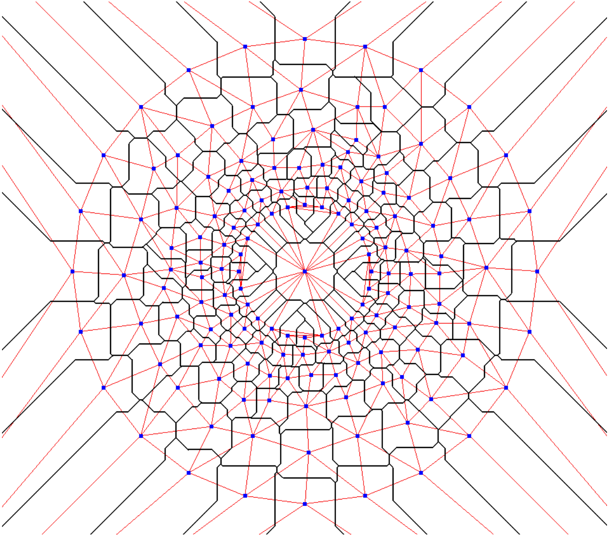
Bisector of two points $\mathbf{x}_1 = (-x_p, -y_p)$ and $\mathbf{x}_2 = (x_p, y_p)$ using the L^∞-norm.

Circumcenter, circumradius and circumcircle



Circumcenter $\mathbf{x}_c$ of a triangle using the L^{∞}-norm and circumcircle (red dotted square).

Delaunay and Voronoï

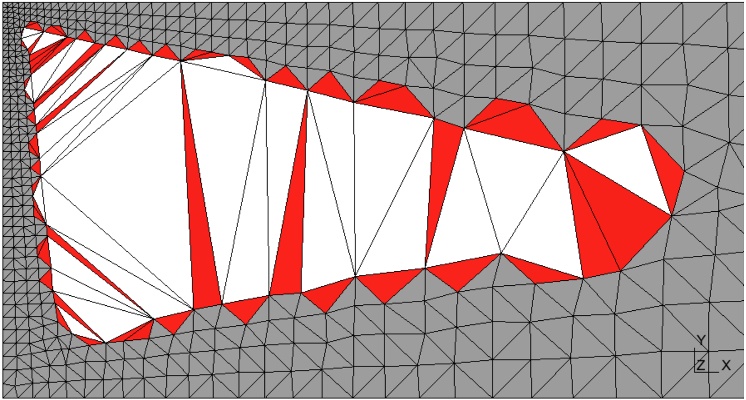


Voronoi diagram (dark lines) and Delaunay triangulation, both in the L^∞ norm.

A frontal-Delaunay mesher in the L^∞-norm

The frontal-delaunay technique is based on

Rebay S. Efficient unstructured mesh generation by means of delaunay triangulation and bowyer-watson algorithm. *Journal of Computational Physics* 1993; **106**(1):125–138.



A frontal-Delaunay mesher in the L^∞-norm

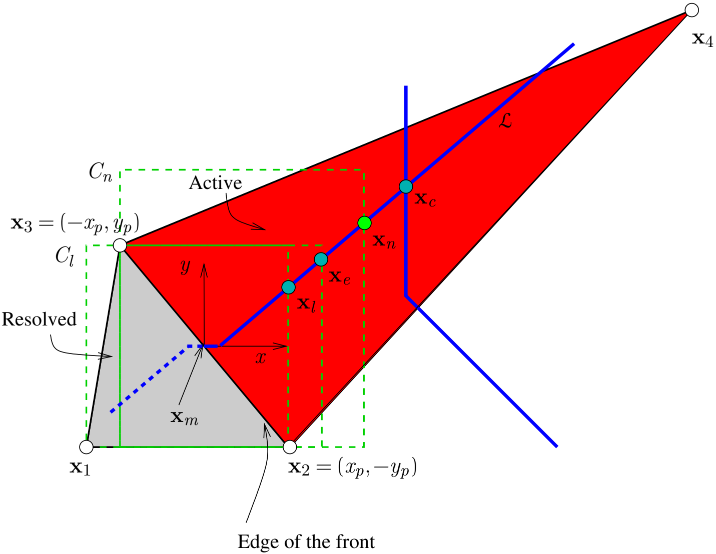
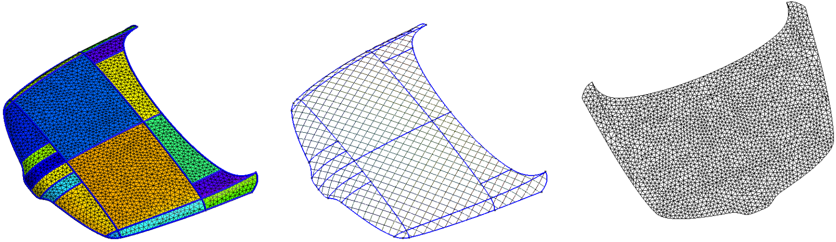


Illustration of the point insertion algorithm.

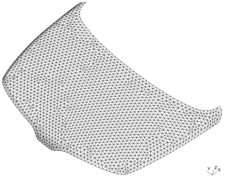
Car Hood



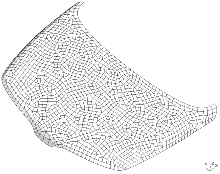
(a) (b) (c)

Conformal reparametrization of a car hood. Figure (a) shows the CAD decomposition of the hood into 18 Bézier surfaces and the initial conforming mesh. Figure (b) shows the result of the conformal reparametrization i.e. isolines of u and v on the whole hood. Figure (c) shows the projection of the initial mesh on the parameter plane $\mathbf{u}$.

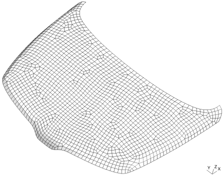
Car Hood



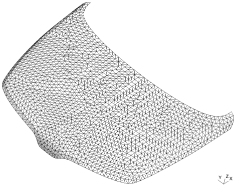
(F)



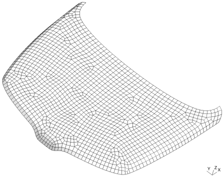
(FB)



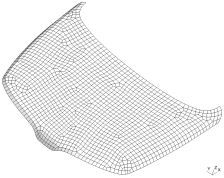
(FLB)



(D)



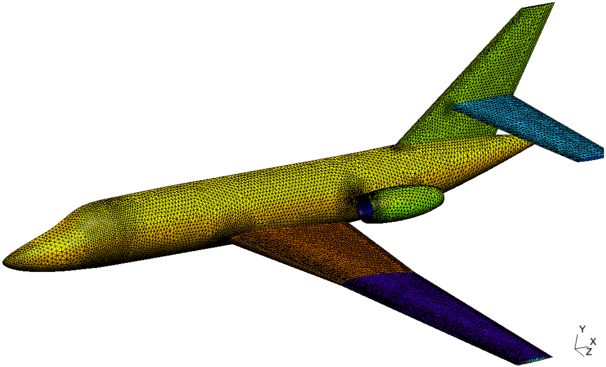
(DB)



(DLB)

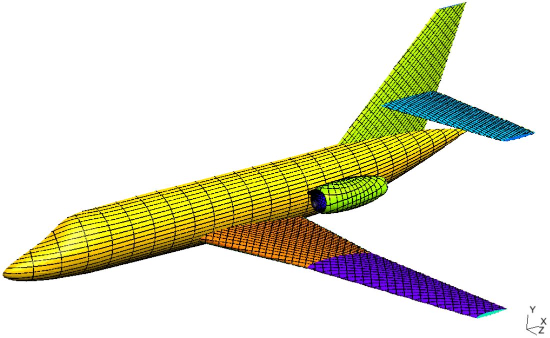
	N	$\bar{\eta}$	η_w	$\tau(\%)$	$d_4(\%)$	$d_{\max}$
(FB)	1726	0.77	0.37	86	71	6
(FLB)	1744	0.86	0.39	87	81	5
(DB)	1527	0.90	0.37	93	81	5
(DLB)	1543	0.91	0.43	94	83	5

Falcon



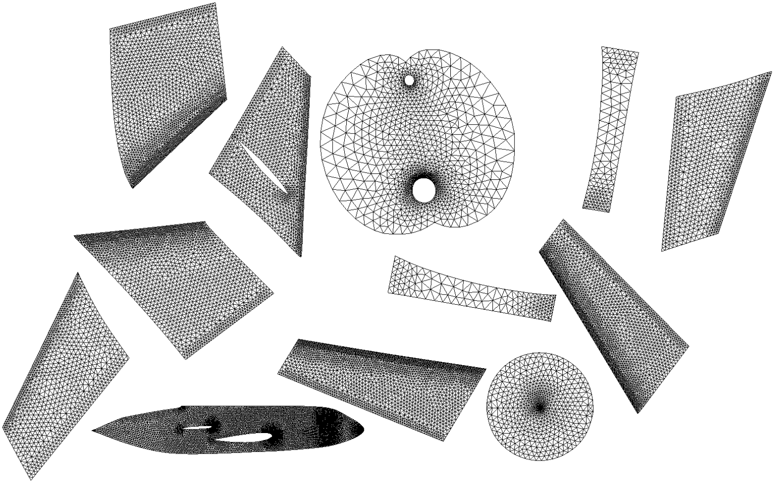
Surface triangular mesh of a Falcon aircraft (left) and contour lines of the conformal parametrizations (right). Colors are indicative of the different surfaces of the model.

Falcon



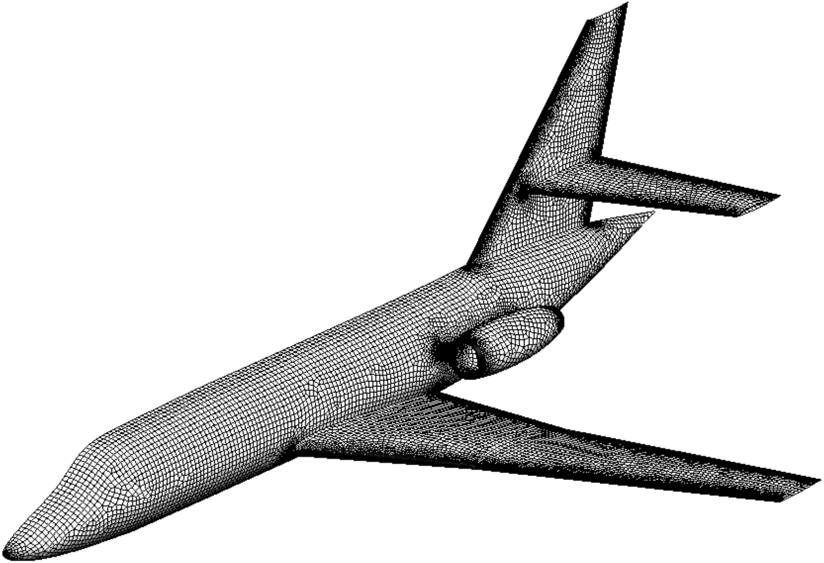
Surface triangular mesh of a Falcon aircraft (left) and contour lines of the conformal parametrizations (right). Colors are indicative of the different surfaces of the model.

Falcon

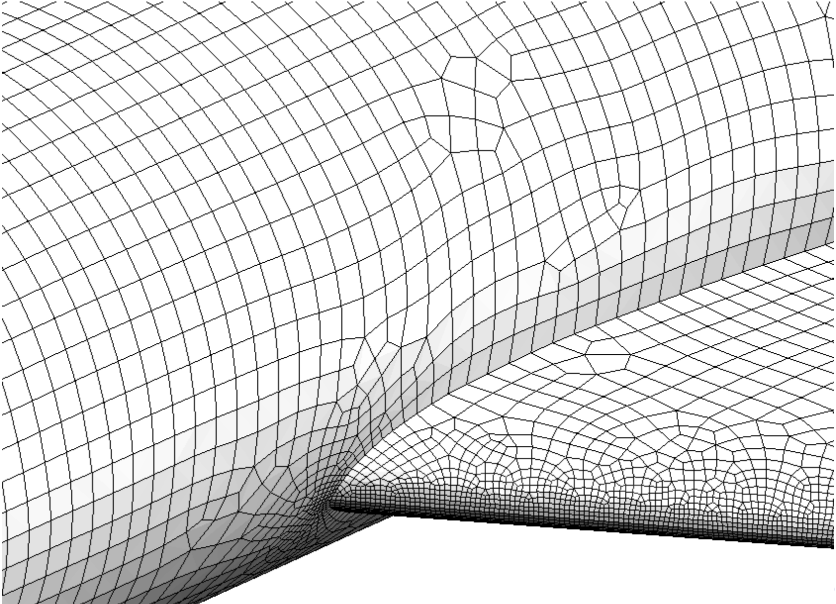


Parametrizations of the surfaces of the Falcon aircraft in the $\{u, v\}$ plane.

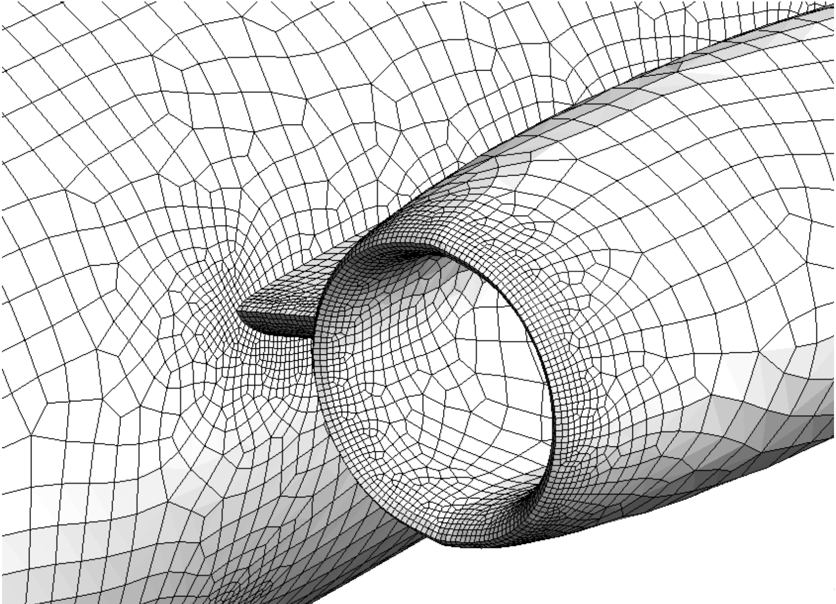
Falcon



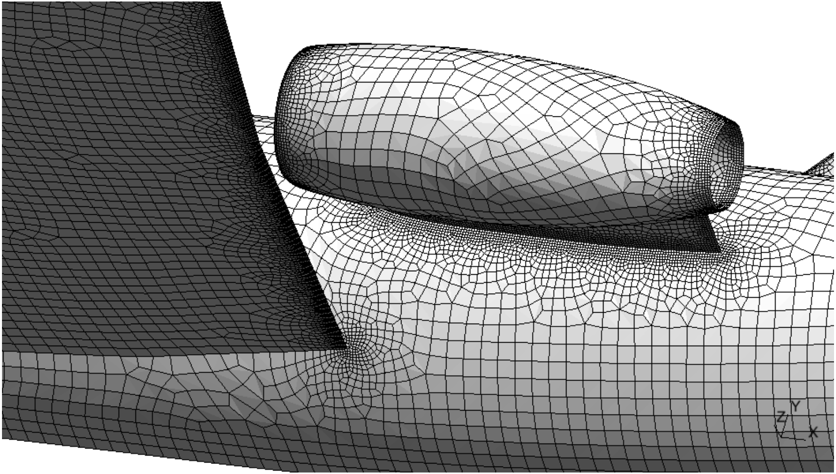
Falcon



Falcon



Falcon



Research and developpement in Gmsh

New stuff in Gmsh

- Remeshing surfaces with parametrizations
- High order/curvilinear meshes
- Quad and Hex meshing

References

J.-F. Remacle et. al.
Geometrical Validity of Curvilinear Finite Elements, IJNME, In preparation, 2011.

Outline

Roadmap for Gmsh 3.0

- Complete re-build of the GUI
  - Use Qt (a first demo version should be available in 2011)
  - More flexibility : reconfigurable GUI (OnleLab project), true plugins, remote control through sockets.
- Meshing capabilities enhanced
  - Curvilinear meshes ($p = 2$ and $p = 3$ in 2011)
  - Hex dominant meshing (DomHex project)
  - Boundary layers (already ok for surface meshes), pyramids (even for high order)
- Python API has to be cleaned and documented !