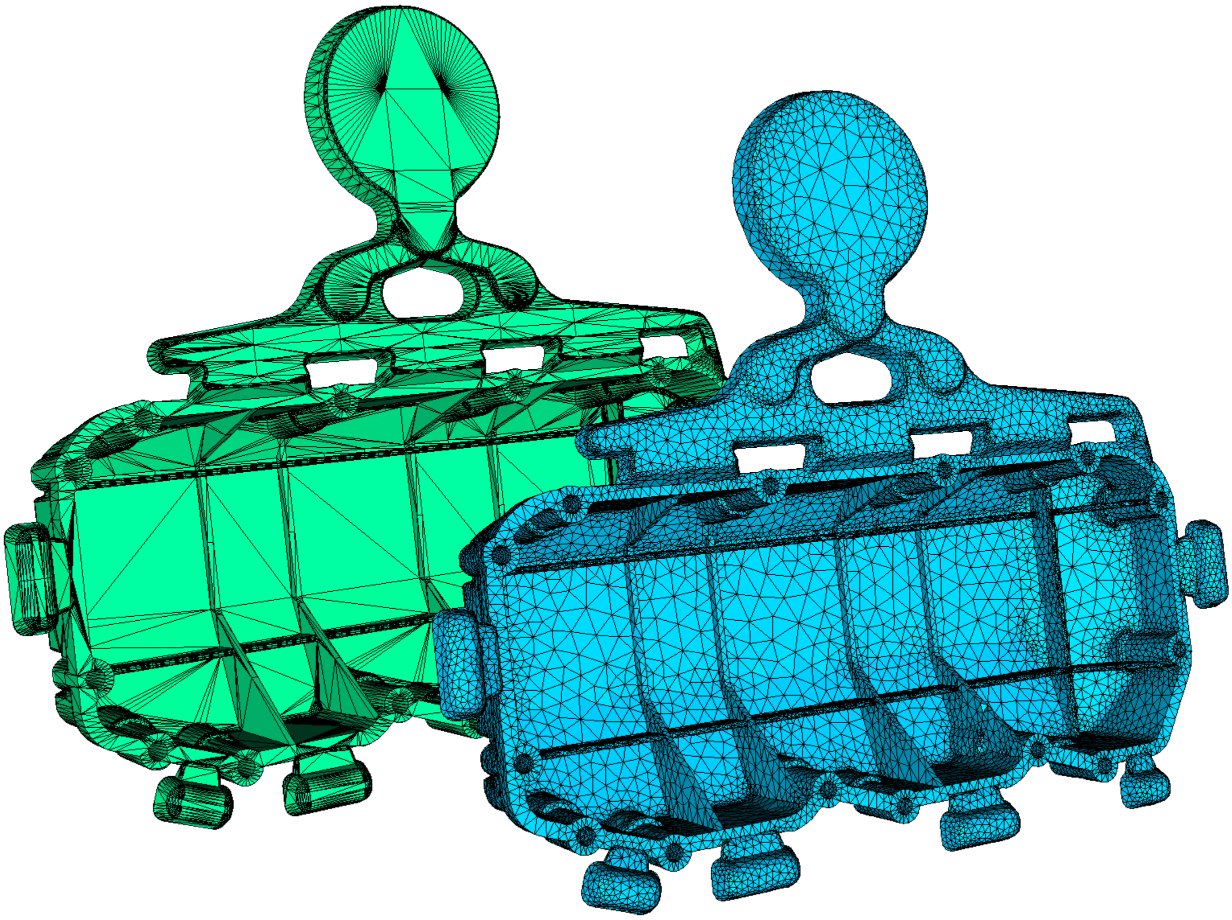
An introduction to

YAMS V3

A fast, reliable, high quality surface remeshing tool



For further information, contact:

Mark Loriot

DISTENE

Pôle Ter@tec - BARD-1, Domaine du Grand Rué, 91680 Bruyères le Châtel FRANCE

Tel: +33 (0) 1 69 26 62 10, Fax : +33 (0) 1 69 26 90 33

mail:mark.loriot@distene.com

A. Introduction

A.1. Context

Although being largely used worldwide, surface meshing is still a topic of major concern for the industrial world. Today, one can easily get a surface triangulation of any object, but this triangulation is often far from being suitable across a wide range of applications.

The use of “discrete” geometries as an alternative to native “analytical” CAD geometries has become widely used through the last decade, as this is the easiest, although crude, way of exchanging geometry data. But more important, as 3D numerical simulation becomes used widely in new areas like computer aided medicine or gaming, or as it addresses more and more comples geometries, it has to face the challenge of “incorrect” geometry data. The use of reconstruction algorithms like wrappers or 3D imaging to triangulation is required, and this generates a whole new set of meshing problems. At the same time, the need to exchange geometries or to visualize them as efficiently as possible has also risen

A major concern is now the ability to get a suitable mesh out of these poorly accurate or very dense geometries. This new mesh has to meet the constraints imposed by hardware resources (graphics or computing), robustness of the solver, compliance to geometry, communication bandwidth, etc.

Yams was designed for this purpose.

A.2. What is Yams ?

Yams is a fully automatic adaptive surface **remeshing** component which allows one to obtain a governed surface triangular mesh from an arbitrary given surface triangulation.

Yams is the “Swiss Army knife” for surface remeshing. It can indeed be used for very different tasks such as : decimating the triangulation while keeping the geometry to obtain coarser meshes for visualisation purposes or computational applications, optimizing the quality of elements to make them better suited to CAE computations, enriching the mesh to make it closer to the geometry, etc.

It is governed by the geometric surface properties. The initial surface triangulation can be any triangulation, for instance coming from an STL or DXF file, a finite-element type mesh or any surface coming from a reconstruction algorithm. The only requirement for the input data is to be orientable (eg, Mœbius rings are not orientable) and it has to contain triangles and/or quadrilaterals only. In particular, Yams **is capable of processing non-manifold** industrial models.

As all components of the MeshGems suite, Yams is command line driven (with command line options), and can be used entirely in batch mode.

Once the remeshing process is completed, another surface mesh is output for which the element density and the mesh quality was controlled by the user through the input data and runtime options. The geometric surface properties may include additional entities and information that will be used to control and drive the surface remeshing (such as for example : constrained entities, corners, ridges, normals, etc.).

A.3. What Yams is not…

Yams is **not** capable of meshing NURBS or b-splines or any analytical definition of a geometry. It needs as input an existing triangulation of surfaces (even if it is of poor quality, such as discrete geometries described using the STL format) in order to create suitable meshes.

Yams is not a mesh healing tool either. Although Yams is capable of correcting minor errors such as for example, multiple identical facets, or of improving the mesh quality, Yams makes the assumption that the given mesh should be considered as an adequate representation of the surface. In particular, Yams will not correct holes, non conforming meshes, self-intersecting edges or facets etc. On the contrary, all these characteristics will be considered as being representative of the geometry, and therefore kept as such.

However, the MeshGems suite offers other components for these needs, such as:

- **Cleaner**, which heals surfaces by removing self–intersections, gaps, holes, non-conformities, badly shaped elements, etc., even if this can alter locally and in a limited way the geometry
- **BLSURF,** which is capable of generating high quality surfaces from analytical CAD geometries

B. Yams V3.2 : a surface remeshing tool

B.1. Features

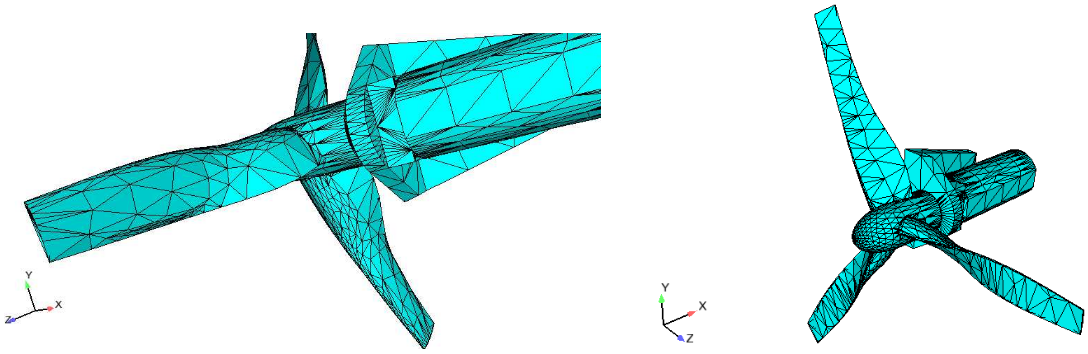
The input files of previous versions of Yams are fully compatible with this new release. All previous features are kept in this latest release, that is:

- Mesh smoothing and sandpapering,
- Mesh decimation, to reduce the number of triangles,
- Mesh enrichment, to increase the density, using curvature detection and approximation, generating both quadrangular and triangular meshing,
- Mesh adaptation,
- Triangular and quadrangular meshing,
- Isotropic or anisotropic remeshing,

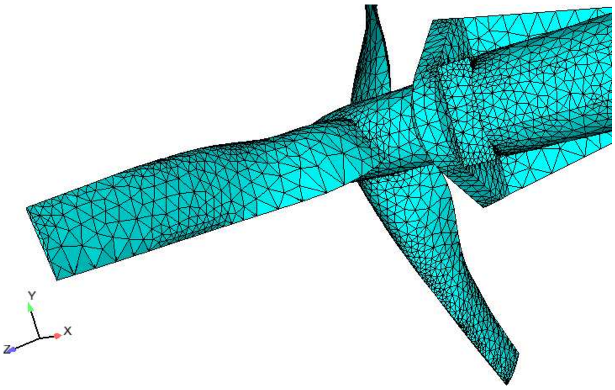
B.2. What's new with respect to the previous version ?

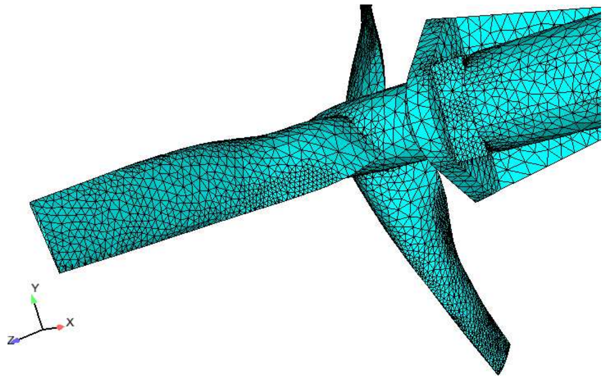
B.2.1. Volume proximity detection and control

The release 3.2 of Yams offers a new outstanding capability: volume proximity detection. This feature allows Yams to remesh the geometry while taking into account the fact that surfaces might be close to one another in the 3D space without being so in a topological sense.
This can ensure that the worst aspect ratio of corresponding tetrahedra in the volume is of much better quality, but also that self-intersections are avoided during the surface remeshing phase in thin and curved areas.



Original triangulation, prior to Yams remeshing



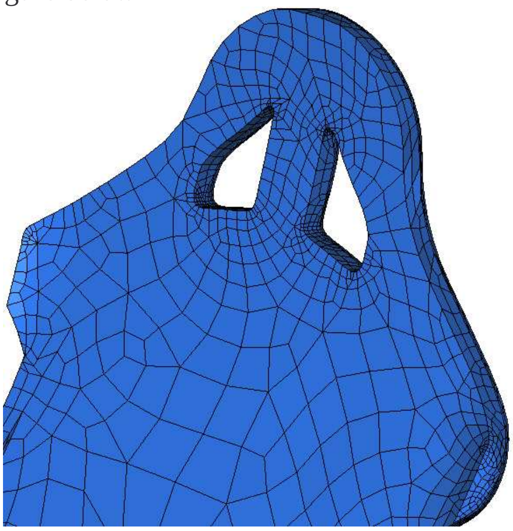


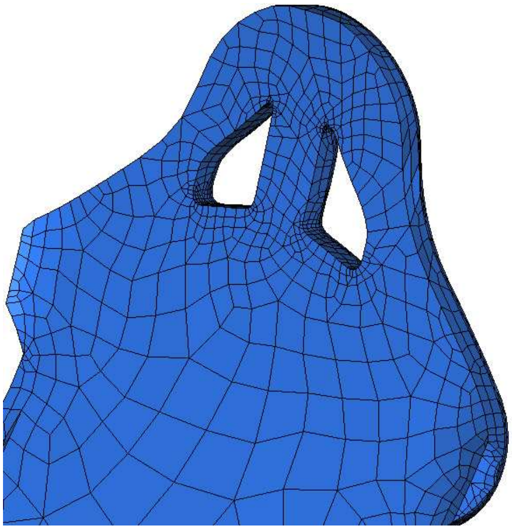
Resulting mesh after Yams' remeshing, without volume proximity taken into account. The size is governed by the curvature, error estimates and gradation only.

Resulting mesh after Yams' remeshing, with volume proximity also taken into account. Here, a size of surface triangles corresponding to the local width of the geometry was requested, inducing higher density in thin areas. One can see that wide areas are not denser.

B.2.2. Quadrangular remeshing improved

This new version also improves the quality of the quads generated, as can be shown on the figure below





100% quad remeshing with high gradation from triangular mesh, previous release of Yams (3.1)

100% quad remeshing with high gradation from triangular mesh, previous release of Yams (3.2)

C. What are Yams' typical applications?

Yams is a generic surface remeshing tool. As such, it is capable of coarsening, enriching or adapting an existing triangulation. In all cases, Yams uses the curvature to place the points (the points are not moved linearly on the surface, but on the approximated curved surface).

We highlight below some of its features.

C.1. Surface mesh enrichment

In some cases, it may be desirable to enrich a given surface mesh, for instance to improve the geometric approximation of the surface, or to get a suitable mesh for computations out of a low cell count triangulation. Yams uses the given triangulation as a representation of the geometry and creates new points in regions where the discretization is too far from the geometry or not adequate for computations.

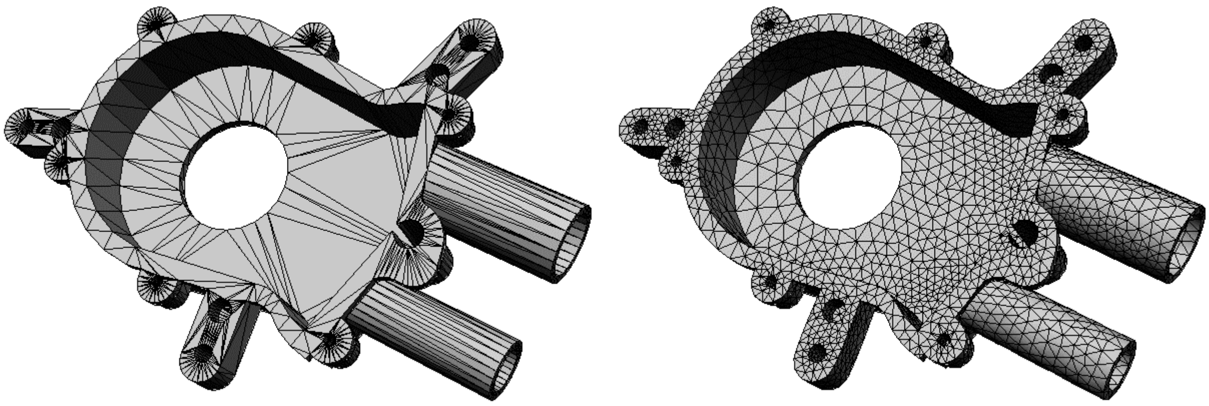


Figure 1: Example of mesh enrichment for computational purposes. Left: original mesh, right: refined mesh (courtesy of UGS)

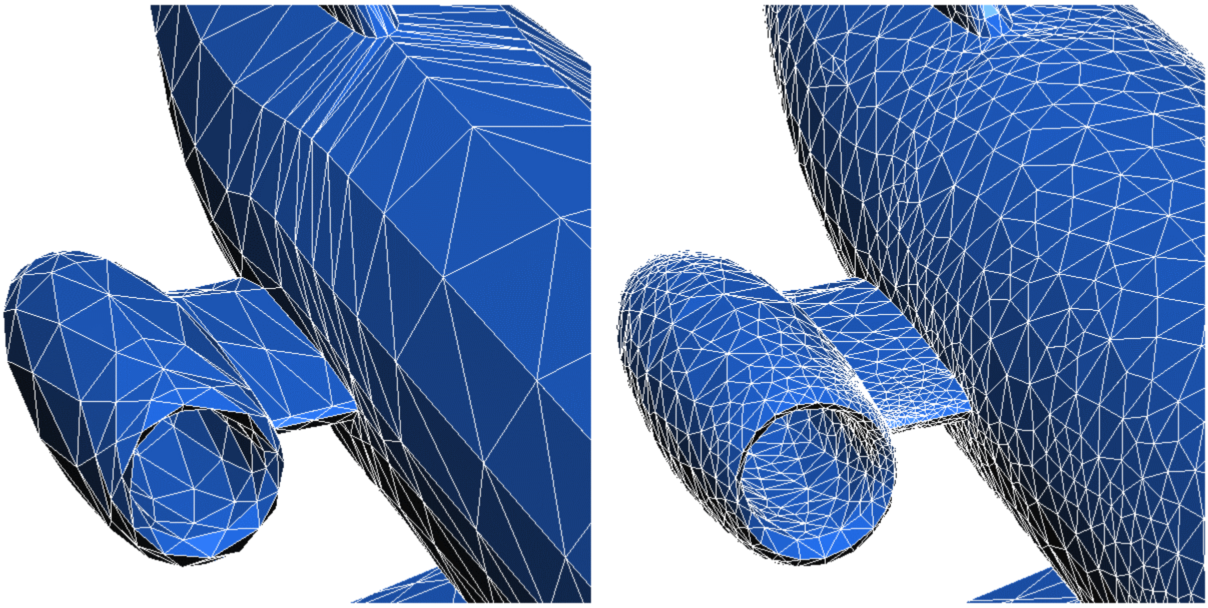


Figure 2: Example of geometric mesh enrichment (geometry courtesy of Dassault Aviation)

C.2. Surface mesh coarsening

One of the most important features of Yams is the ability to reduce the number of mesh entities while controlling:

- the quality of the geometric approximation,
- the surface properties (curvatures, ridges, etc.),
- the global mesh quality.

C.2.1. Decimation for computations (curvature-controlled decimation)

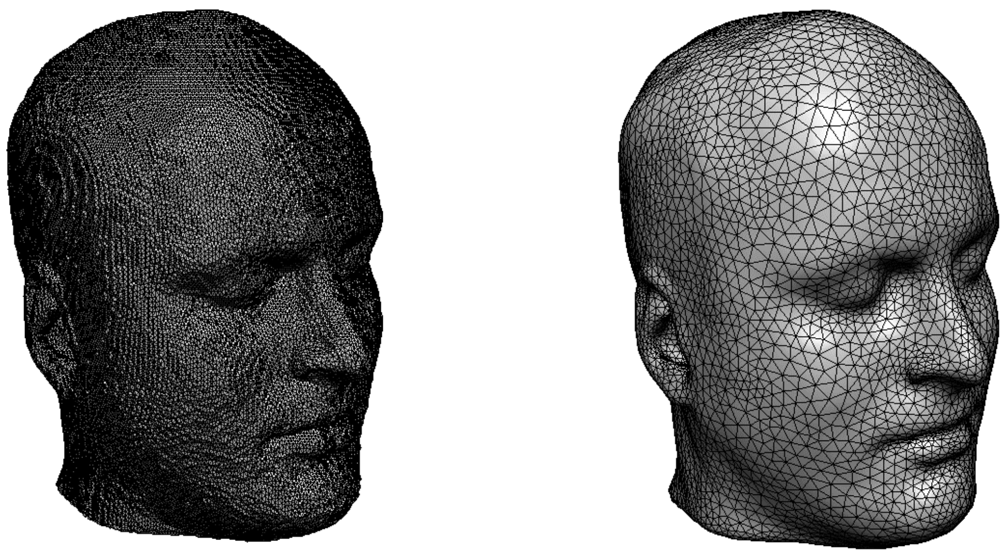
The original surface triangulation can be simplified according to the intrinsic size map, automatically computed by the program. The goal is to get a smaller mesh, with a good geometric approximation, in which the element shapes and sizes are controlled. Some tolerance parameters (angular deviation and/or tolerance parameters) enable the user to control the chordal deviation and the surface smoothness (see D.3)

Typically, this option triggers the creation of finite element (isotropic) meshes in which the density of the elements is related to the local curvatures of the surface. The size gradation, eg how the density propagates from one element to its neighbours, can also be controlled by the user.

Table reports some results for this type of mesh simplification procedure; *# points* and *# triangles* represent the number of points and triangles, Q_{max} denotes the quality[1] of the worst element, *time* gives the elapsed time including I/O on a Intel Xeon X5650 @2.66GHz PC. In this table, the first line of each example indicates the initial values and the second line the final results.

Test cases		*# points*	*# triangles*	Q_{max}	time (sec.)
Parab2	Original	17,956	35,378	2.78	-
	Yams	241	421	1.81	0.3
Cyl-original	Original	40,538	81,152	19.28	-
	Yams	17,968	36,012	19.28	1.46
B2	Original	43,076	86,176	4.19	-
	Yams	26,188	52,400	4.89	2.2
RCL005-small	Original	2,168,275	4,336,550	653.8	-
	Yams	277,057	554,114	37.4	59.0
Ipnd_raceway	Original	6,452,115	12,904,250	44.0	-
	Yams	647,567	1,295,154	45.5	159.8

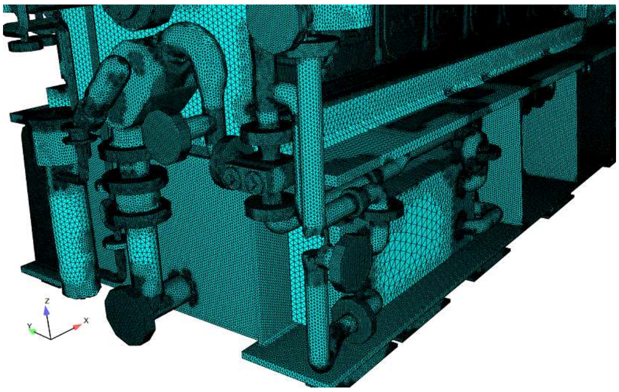
Table 1: Several statistics about *controlled mesh simplification*



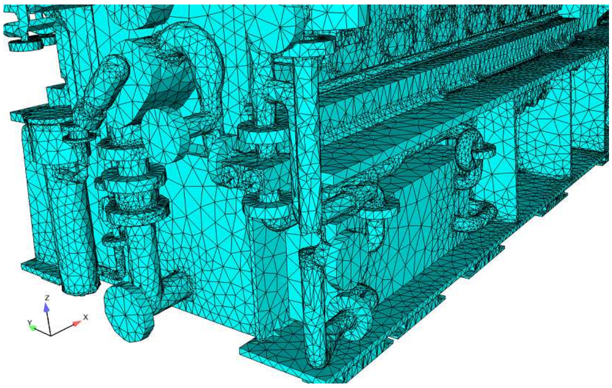
original mesh (output of a 3D scanning reconstruction)   Coarsened mesh (Yams)

Figure 3: Examples of finite element geometric mesh simplification.
Left: original mesh, right: simplified mesh

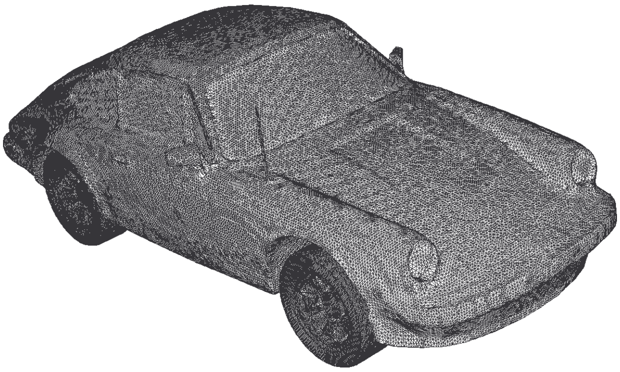
[1] The quality of an element K used here is computed as $Q_K = \alpha \frac{h_{\max}}{\rho_k}$, with $h_{\max}$ being K's longest edge, ρ_k the *inradius* of K, and α is a normalisation coefficient chosen so that the quality of the equilateral triangle is 1



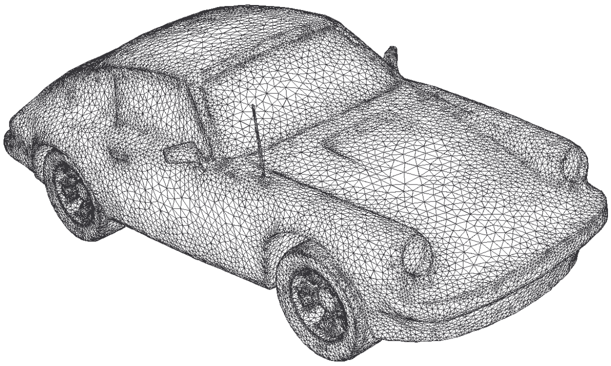
Mak case, original mesh (this is the output of a 3D wrapper)



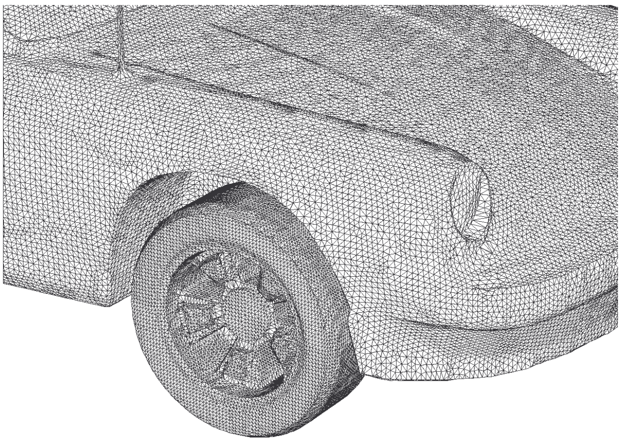
Corresponding coarsened mesh by Yams : 95% reduction in 50s



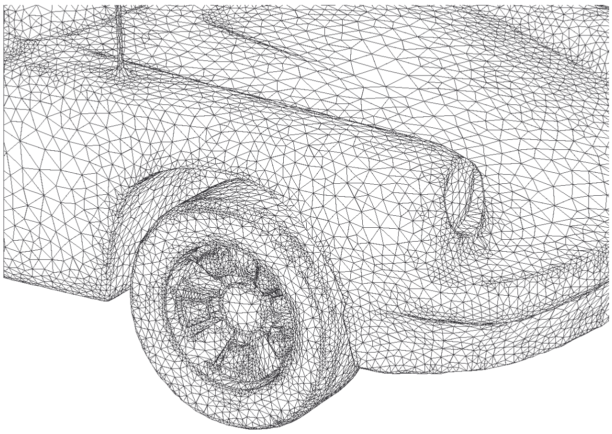
Original mesh : 154,941 vertices, 309,862 triangles



Decimated mesh : 46,720 vertices, 93,420 triangles



Original mesh : zoom near front wheel



Decimated mesh : zoom near front wheel

Figure 4: Examples of finite element geometric mesh simplification.
Left: original mesh, right: simplified mesh

C.2.2. Decimation for visualization (distance controlled decimation)

Yams also allows the user to generate a surface approximation that is guaranteed to deviate from the initial surface by no more than a user-specified amount while preserving the topology. The tolerance can be seen as a *protective envelope* surrounding the original triangulation.

This approach usually results in geometric meshes, in which the geometric approximation is well preserved, while keeping the number of triangles very low.

The following table reports some results obtained with Yams with default options for this type of mesh simplification: *#points* and *#triangles* stand for the number of points and elements respectively, *elapsed* indicates the elapsed time including I/O on a Intel Xeon X5650 @2.66GHz. In this table, the first (resp. second) line of each example corresponds to the initial (resp. final) mesh.

Test cases	#points	#triangles	elapsed (sec.)
B2	43,076 4,408	86,176 8,840	- 0.7
Happy (cf Figure 5)	542,549 49,489	1,085,477 99,361	- 10.2
RCL005 (cf Figure 9)	2,790,189	5,580,378	- 38
Ipnd_raceway (cf Figure 7)	6,452,115 560,904	12,904,250 1,121,828	- 86

Table 2: Several statistics about *geometrical mesh simplification*

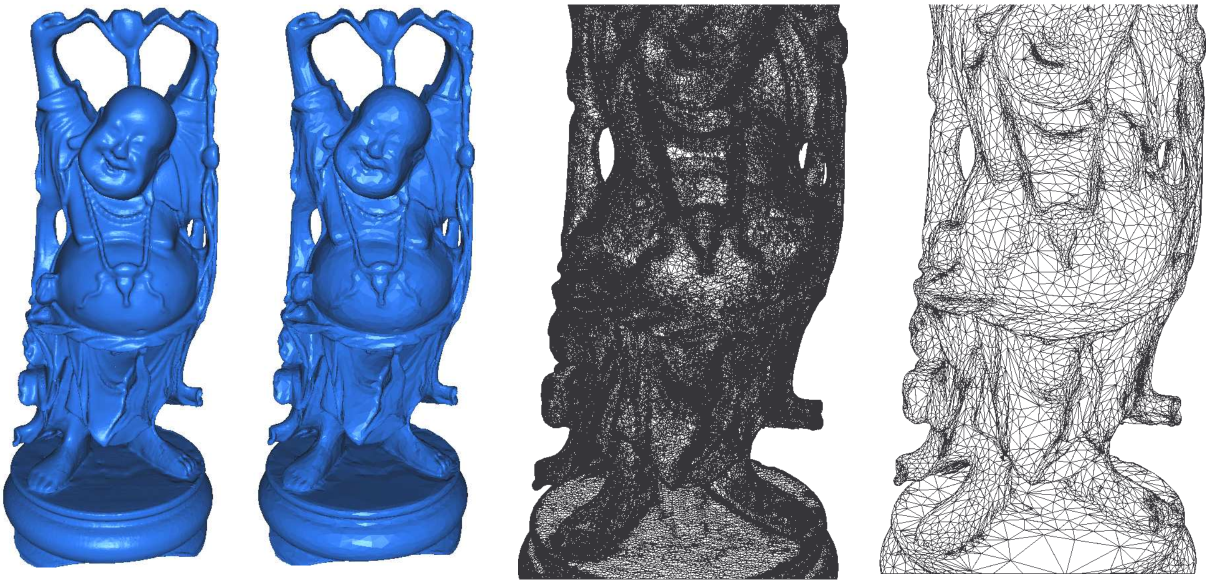


Figure 5: Example of geometric mesh simplification for visualisation purposes.
LEFT: original geometry and decimated geometry, RIGHT: original mesh (542,549 vertices) and corresponding decimated mesh (73,896 vertices) - geometry courtesy of INRIA-

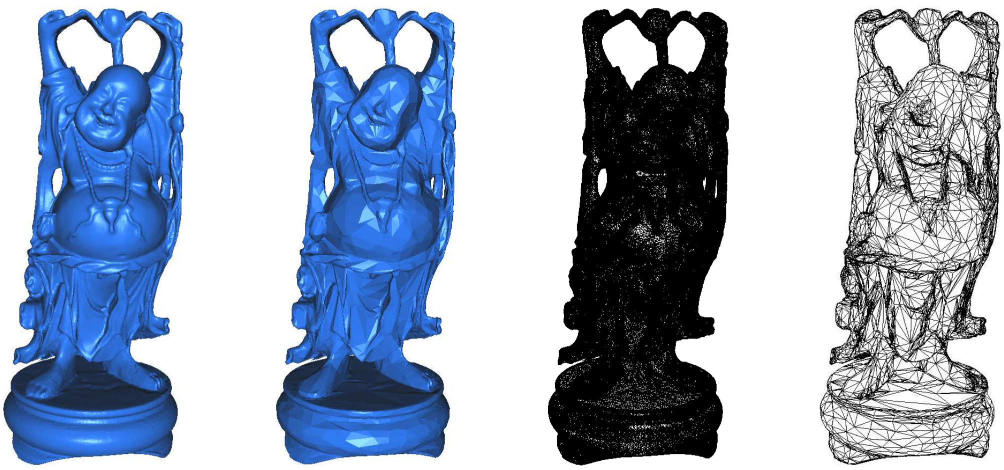


Figure 6: Example of mesh simplification, **requested to 20,000 nodes**. LEFT: original geometry and corresponding simplified 20,000 node geometry, RIGHT: original mesh and decimated mesh.

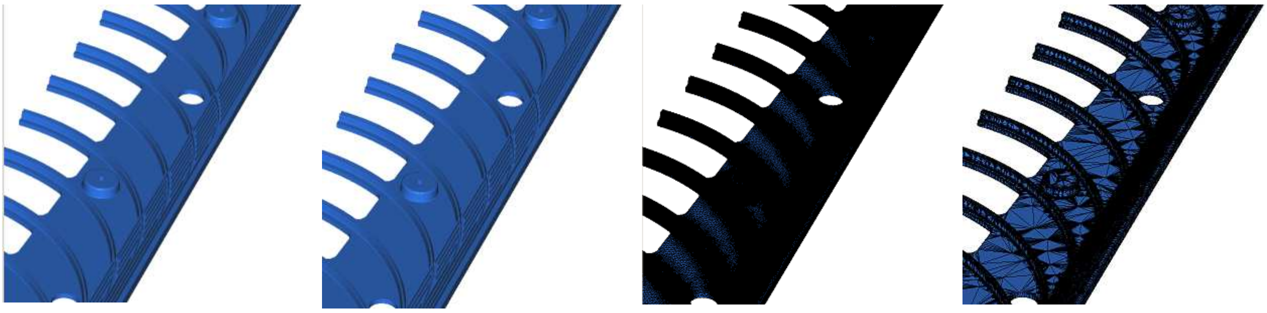


Figure 7: Example of mesh simplification (Ipnd_raceway case). LEFT: original geometry and corresponding simplified geometry, RIGHT: original mesh and decimated mesh

C.3. Surface mesh optimization

Yams allows one to improve the quality (in terms of aspect ratio) of any given surface triangulation, while preserving the geometric approximation. Indeed, most of the numerical simulation software packages (especially based on finite elements/volumes techniques) are very sensitive to the quality of the meshes, to some extent, depending on the solver used. The quality of the surface mesh may also affect the quality of the 3D mesh if a 3D tetrahedral mesh generation algorithm is to be used.

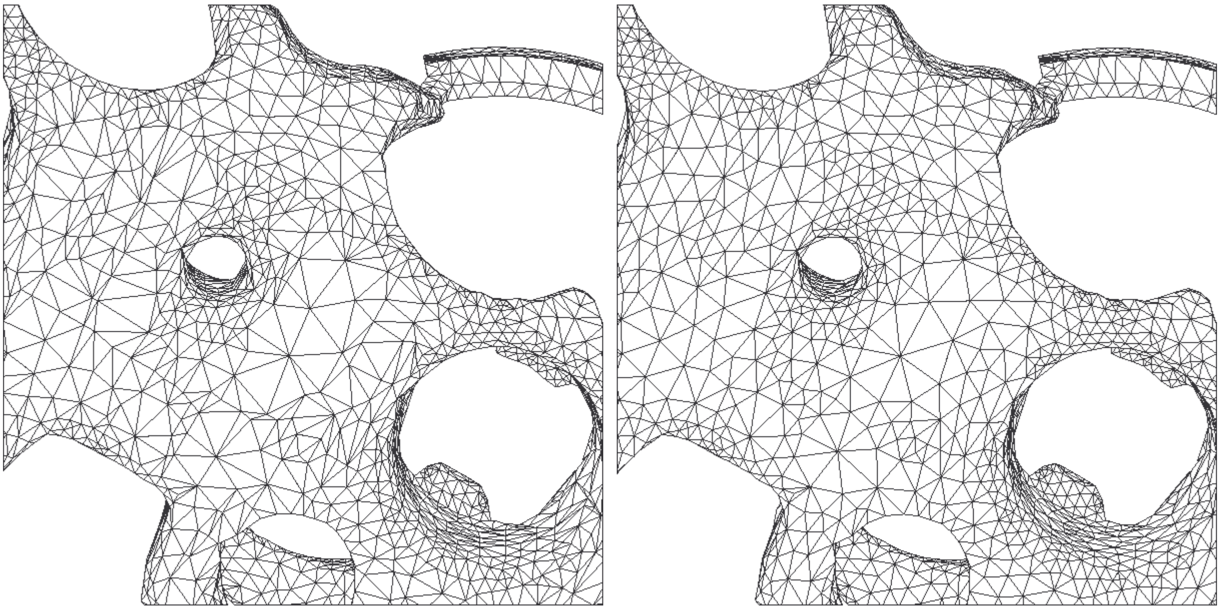
A straightforward way of optimizing a mesh is to keep the number of mesh entities unchanged and simply make *local* modifications so as to improve the quality. In this case, geometric operations (node relocation) and topological changes (edge flipping) are involved.

In any case, the number of elements and points are most of the time kept by the algorithm.

The following Table shows the result of one optimization pass applied on some examples; *#triangles* stands for the number of mesh triangles, Q_{max}, and $Q_{avg\ 2}$ represent respectively the worst and the average quality. *Elapsed time* stands for the total time taken including file I/O on a Intel Xeon X5650 @2.66GHz. The number of points and elements remains unchanged by the optimisation process for all these cases.

Test cases	#triangles	Q_{max}		Q_{avg}		Elapsed time (s) (incl. I/O)
		Before	After	Before	After	
2strokeK	23,216	31.1	1,9	1.27	1.16	0.28
Cyl_original (cf Figure 8)	81,152	19.3	19.3	1.65	1.39	1.5
Scalph18	195,692	3.65	2.77	1.26	1.10	2.38
11755-pump_body	474,210	3.25	2.11	1.06	1.05	3.9
RCL005-small	4,336,550	653.8	18.92	1.08	1.07	64.7
Car-BH	4,808,886	17.6	17.6	1.3	1.27	43.4

Table 3: Quality improvement on various test cases



original mesh optimized mesh

Figure 8: Example of mesh optimization in three dimensions (Cyl_original case, geometry courtesy of IFP)

D. Technical description of Yams

D.1. Input and output Files

The **input** surface geometry is described by a surface triangulation (using the *.mesh* format, a simple open format used by all components of the MeshGems suite). The input must contain at least the list of vertices and their coordinates, and the topology (list of elements) of the surface mesh. The basic input described above may also be conveniently enriched by **geometry-based** additional information such as corners, edges, ridges, required entities, all of which must be preserved in the final mesh.

The input surface triangulation may be non-conformal and must be orientable.

Apart from the surface triangulation, one can also input a metric map which is to be used as a basis for remeshing.

Remark: the input triangulation is used as geometry reference and therefore there is no link to a CAD system.

The resulting **output** mesh uses the .mesh format. The surface remeshing algorithm *preserves the attributes* specified in the input file (vertex, edge and face attributes). These attributes are usually used to specify additional information, such as for example boundary conditions.

D.2. Speed

Depending on the options, the size of the input triangulation and the hardware, one can reasonably expect to complete Yams on a PC in:

- a few seconds to simplify or optimise the quality of a mesh of several tens of thousands of elements (typical speeds are around 5 million triangles processed per minute on a Xeon X5650 @2.66GHz). This speed is scalable
- a few minutes to enrich a mesh of several hundreds of thousands of elements

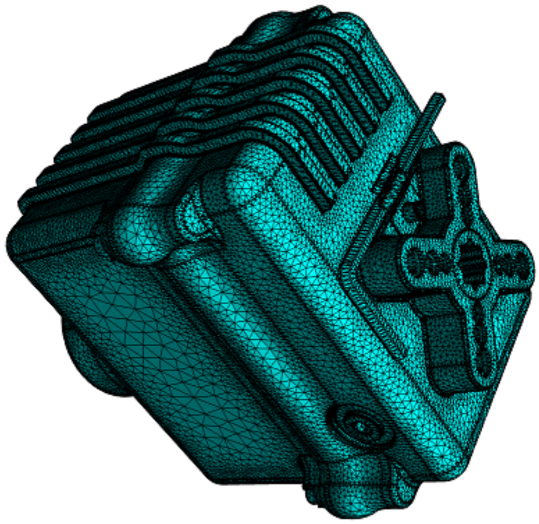


Figure 9: Example of coarsened mesh (RCL005)

D.3. Geometry Control parameters

In addition to various control options to optimise, coarsen or enrich, additional parameters can be specified in order to characterise better the expected surface properties, such as for example:

- *Chordal deviation tolerance* : This field enables the user to control the accuracy of the piecewise linear approximation of the surface. This allows the user to specify the maximal chordal deviation ξ_{max}. If the chordal deviation ξ (see Figure 10: geometry accuracy control parameters) is smaller than ξ_{max}, it means that, following that criterion, it is acceptable to move the considered point on the average plane.

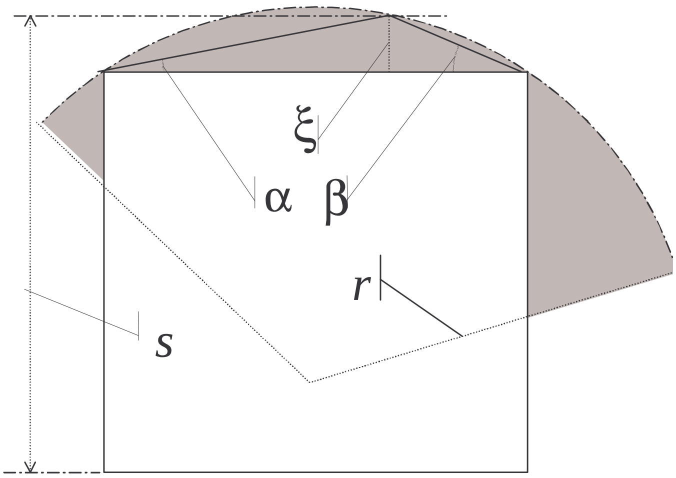


Figure 10: geometry accuracy control parameters

- *Geometric approximation*: This field enables the user to control the accuracy of the piecewise linear approximation of the surface. This controls the maximal chordal deviation ε_{max} relatively to the curvature. If the chordal deviation ξ (see Figure 10: geometry accuracy control parameters) is smaller than $\varepsilon_{max} \times r$, it means that, following that criterion, it is acceptable to move the considered point on the average plane.
- *Threshold ridge angle*: this parameter specifies the angular value for the detection of ridges and corners. This option is provided as a simple (but crude) way of addressing ridges if they were not previously identified in the input geometry
- *Element size propagation*: This parameter enables the user to control the element size variation in the triangulation
- Controlling size at vertices : with these options, one can control with great precision the minimal and maximal size of the mesh elements

D.4. Platforms

The Yams software is available on all major platforms:

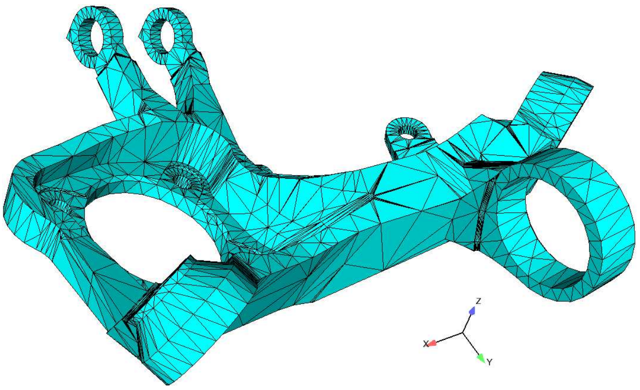
- Windows (2000, XP, Vista, Seven, 32 bit or 64 bit operating systems)
- Linux (32 bit and 64 bit operating systems)
- Mac OS
- Many legacy Unix systems (32 bit and 64bit operating systems)

E. Ongoing R&D

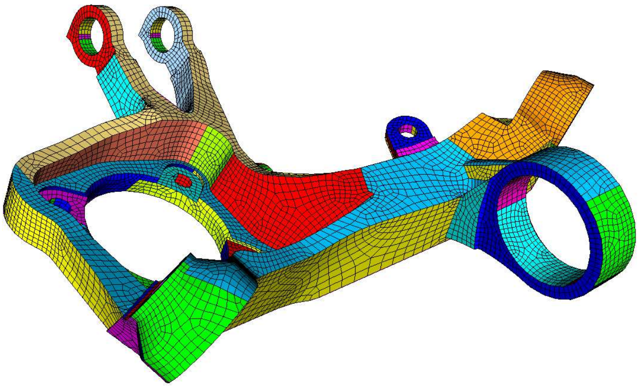
Several topics of interest are currently being investigated for surface remeshing. These include:

- Optimising the code for better performance,
- Extending quadrangular remeshing capabilities,
- Introducing completely new remeshing approaches, to provide optimal results depending on the targets expected.

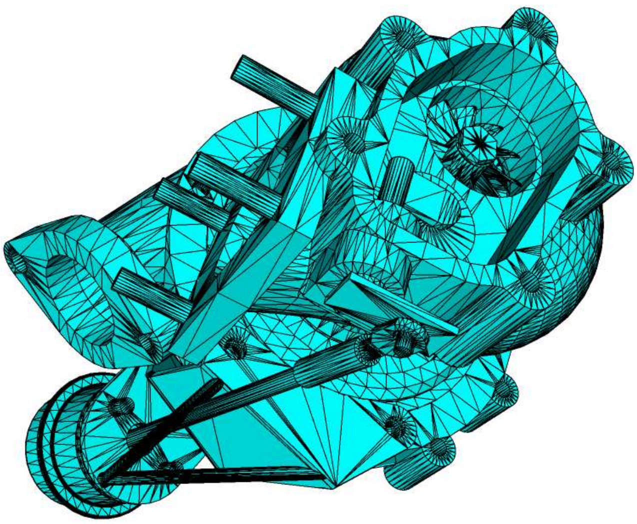
  This last topic is illustrated below, where a binary STL file -without attributes- is remeshed into pure quads, triangles or mixed tri/quad with a new remeshing technique, with automatic part identification.



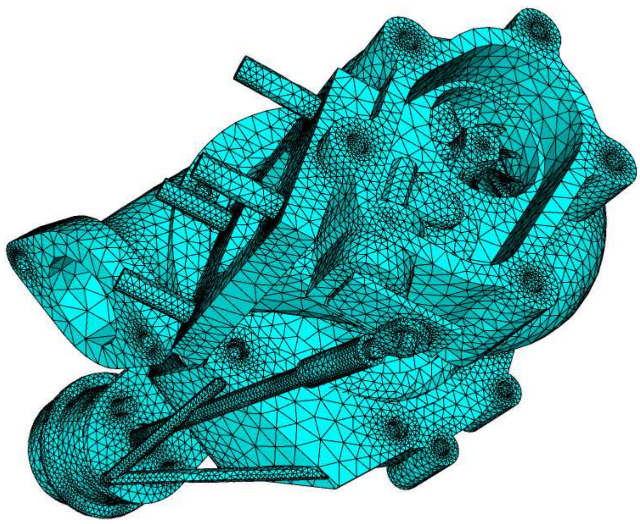
Original geometry and triangulation (STL)



Remeshed geometry (quads)



Original geometry and triangulation (STL)



Remeshed geometry (tri)

Figure 11: future alternative remeshing approach