Geodesic Meshing of Closed Surfaces

Christophe Geuzaine*　　Jean-François Remacle†

Abstract

We propose a new meshing strategy for closed surfaces, based on the computation of exact discrete geodesics. The resulting meshes are discretely isogeometric to the underlying surface, and their coarseness is only limited by the surface topology. This makes them perfect candidates for creating polynomial finite elements or Bézier or NURBS patches suitable for high-order numerical solvers.

1 Introduction

A major bottleneck of high-order numerical methods on complex geometries is the construction of meshes with sufficiently large elements, on which high-order approximations of the solution can be efficiently constructed. This bottleneck is a key reason why most finite element tools used in engineering practice today are still based on relatively old and inaccurate low order (typically, second order) discretizations, even though the advantages of higher order methods are well-known [1]: they provide faster grid convergence and they increase arithmetic intensity, making them amenable to highly parallelized implementations on both CPUs and GPUs.

High-order meshing on complex geometries is still an open problem for two main reasons. First, most high-order meshing approaches are indirect: they require a valid straight-sided mesh first, before attempting to curve it to better match the underlying geometry. On the one hand this intrinsically makes it impossible to create very coarse meshes, as the straight-sided mesh has to be sufficiently fine in the first place - typically to avoid a self-intersecting boundary mesh before proceeding to the tetrahedralization of the volume. On the other hand, the curving procedure also typically leads to some invalid (self-intersecting, non-positive) high-order elements, requiring costly untangling procedures that have proved hard to make robust [12, 4, 11]. The second obstacle to high-order meshing is the use of CAD geometries. Generating large elements inevitably leads to the necessity of cross-patch meshing (with elements having nodes on more than one CAD patch), negating much of the advantage of CAD-based meshing.

However it must be noted that any CAD representation can easily be transformed into a triangulation that has an arbitrary precision i.e. that is arbitrary close (e.g. in term of Hausdorff distance) to the CAD model. (The opposite is not true: parametrizing arbitrary complex triangulations into regular patches is not easy at all by all means [2].) And in addition, the field of engineering analysis has extended to areas where no CAD blueprint exists, most importantly in material and biomedical sciences. The raw input geometrical data in these areas are point clouds or voxels obtained through imaging techniques such as tomography, which are converted to triangulations after segmentation.

Triangulations can thus be considered as the universal input for mesh generation, which is the point of view we adopt in this work: we assume that the geometry we consider is given by a watertight triangulation $\mathcal{T}$. Our goal is to build an arbitrary coarse triangulation T of $\mathcal{T}$ that is *isogeometric* to $\mathcal{T}$, by exploiting the fast calculation of discrete geodesics on $\mathcal{T}$ [3]. The quality of the isogeometric triangulation can be improved by geodesic Delaunay edge flips, and refined by geodesic Delaunay refinement if desired. The final isogeometric triangulation can then be e.g. fitted to a polynomial high-order mesh to be amenable to classical high-order finite element solvers; or to Bézier or NURBS patches for use in isogeometric methods [5].

The approach we are proposing today applies only to closed surfaces, regardless of their genus.

2 Discrete Geodesic Triangulation

In differential geometry, a geodesic is a curve representing in some sense the shortest path between two points in a surface. Assume a manifold triangulated surface $\mathcal{T}$ with n vertices. The MMP algorithm [7] can be used to compute the exact straightest distance between a vertex to all other $n-1$ vertices of $\mathcal{T}$. While its worst complexity is $\mathcal{O}(n^2)$, it can in practice be lowered using smart pruning techniques [13]. See [3] for a recent review paper on the topic of numerical geodesics.

Figure 1 shows the image of a human vertebra. The image has been clipped to show on one of its sides the geometric triangulation $\mathcal{T}$ and the other side shows the calculation of the geodesic distance between any source point s that lies on the surface (s being not necessarily a

*Université de Liège, Institut Montefiore, 4000 Liège, Belgium.

†Université catholique de Louvain, Institute of Mechanics, Materials and Civil Engineering, 1348 Louvain-la-Neuve, Belgium.

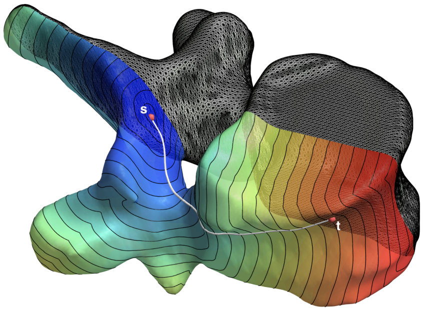


Figure 1: Discrete geodesic between two points of a triangulation $\mathcal{T}$ of a human vertebra.

vertex of $\mathcal{T}$) and all other points. Knowing this distance field $d(s)$ it is possible to start from any point $t \in \mathcal{T}$ and return to s along the geodesic between s and t through smart backtracking. This calculation is very fast and is illustrated on Figure 1 by the white curve.

We define a *discrete geodesic triangulation* T as a set of N_t triangles whose edges are exact discrete geodesics on $\mathcal{T}$. This geodesic triangulation T is compatible with the surface $\mathcal{T}$, i.e., it has the same topology as $\mathcal{T}$ and all its points lie exactly on $\mathcal{T}$.

2.1 Validity In order to assess the validity of T, consider a vertex p_i of T and all its n_i adjacent edges e_{ij_k}, $k = 1, 2, \dots, n_i$ that are *topologically ordered* i.e. two consecutive vertices j_k and j_{k+1} form a triangle $p_i, p_{j_k}, p_{j_{k+1}}$ of T. We define the normal n_i at point p_i as follows:

- if p_i lies inside a triangle of $\mathcal{T}$, n_i is the normal to the triangle in which p_i lies;
- if p_i lies on an edge of $\mathcal{T}$, then n_i is the average of the normals of the triangles adjacent to this edge;
- if p_i lies on a vertex of $\mathcal{T}$, then n_i is the average of the normals of the triangles adjacent to that vertex.

We then define t_{i,j_k} as the tangent vector to the discrete geodesics and g_{i,j_k} at point p_i.

A way of guaranteeing that a planar triangulation is valid is to look at each node in the triangulation and check that the edge e_{ij_k} lies geometrically between its neighbors $e_{ij_{k-1}}$ and $e_{ij_{k+1}}$. We can extend this to geodesic triangulations by checking that the topological order of the edges starting from a node corresponds to their geometric order, i.e. that they are oriented anti-clockwise (the surface normal is assumed to be the exterior normal). If all vertices are well-oriented in the sense just defined above, then the geodesic triangulation forms a regular partition of the surface i.e. it exactly (iso-geometrically) covers the surface without holes or overlaps. This is true because, if two geometrically successive edges adjacent to a node are the edges of a triangle and the geodesics do not intersect, then no geodesic intersects another.

2.2 Local Modifications Local mesh modification operators can be adapted to the geodesic framework:

- split operators, such as edge split (adding a node to an edge and dividing two adjacent triangles in two) or triangle split (dividing a triangle in three);
- coarsening operators such as edge collapse or small polygon reconnection (SPR);
- the relaxation operator, i.e. the edge flip.

The only condition to be met to authorize one of these local mesh modifications is that the new vertices must have counter-clockwise adjacent geodesic edges. Note that all meshes obtained by refining or coarsening these meshes are iso-geometric, i.e. they all represent exactly the same geometric object – namely $\mathcal{T}$.

Figure 2 shows a geodesic triangulation of the vertebra that has been generated in four steps:

1. The STL mesh $\mathcal{T}$ of 168,144 triangles seen of Figure 1 is coarsened using the decimation algorithm proposed by Schroeder et. al. [9]. The mesh T with 320 triangles that is generated has all its points on the initial triangulation and its topology is guaranteed to be the same as the original fine STL. The coarse straight sided mesh can be seen on the left of Figure 2.
2. Every edge of the coarse mesh has been transformed onto a discrete geodesic. Here, we sort every edge of the coarse mesh in lexicographic order. We compute every edge that starts at node p_i using one single instance of the geodesic distance computation. Exact distance computation is stopped when every edge that has its initial vertex equal to p_i is reached by the Breadth first search algorithm. This allows to dramatically reduce the number of calls to this rather expensive algorithm which is also run in parallel using OpenMP.
3. The meshes shown in the middle and right part of Figure 2 have been untangled and enhanced by local geodesic edge flips. We flip an edge if the minimal angle between the geodesics is maximized, as with the classic Delaunay flip [10].
4. The geometric mesh T is finally intersected by those geodesics and isogeometric triangles are finally generated. In Figure 2, we have used a greedy coloring algorithm to clearly visualize the geodesic triangles.

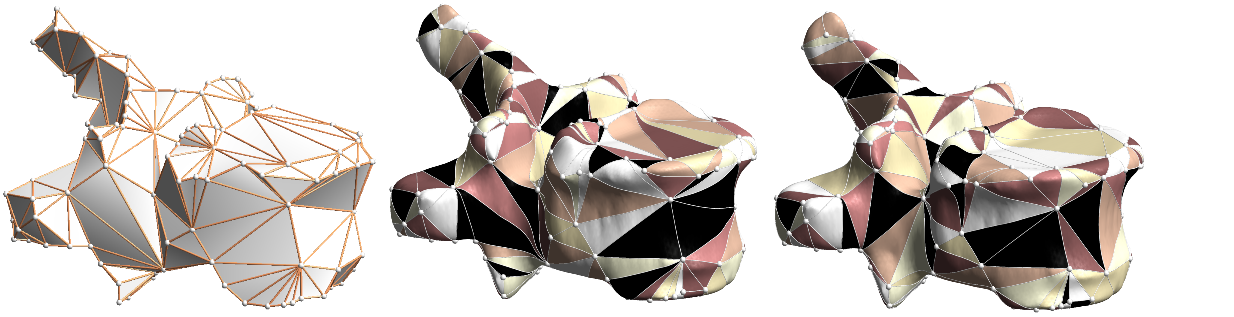


Figure 2: On left, a coarse mesh of the vertebra; in the middle, the discrete geodesic triangulation; on the right the discrete geodesic triangulation that has been optimized using geodesic flips.

2.3 Refinement It is also possible to refine/adapt the mesh using classical techniques such as the longest-edge bisection [8] or a Delaunay refinement. Figure 3 shows an example of a mandible with a coarse mesh and a mesh refined by the longest-edge bisection algorithm. Note that both coarse and fine meshes are perfectly isogeometric.

3 Examples

Figure 4 shows a variety of examples, with fine geometric triangulations ranging from a few triangles for the truncated cube to about 4 million triangles for the brain. Geodesic meshes are generated in just a few minutes on a standard laptop using a maximum number of threads.

4 Conclusion and Perspectives

The geodescic triangulations introduced in this work constitute a new kind of surface meshes, discretely isogeometric to the underlying surface. The main ingredient for their construction is the computation of exact discrete geodesics.

One first use of geodesic triangulations will be to replace geodesic edges and isogeometric triangles by standard polynomial-based finite elements, or Bézier or NURBS patches. For that we plan to propose an approach that is adaptive i.e. that guarantees that the high order polynomial mesh or Bézier/NURBS patches will be i) valid [6, 14] and ii) at a controlled L^∞ distance from the “true geometry” i.e. the isogeometric triangles.

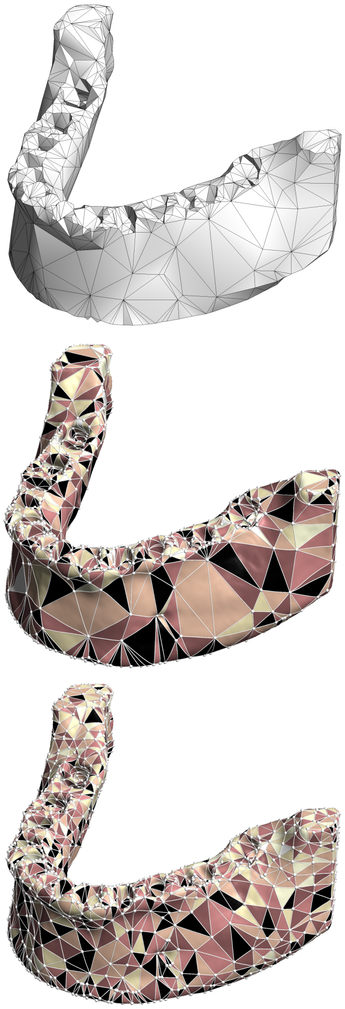
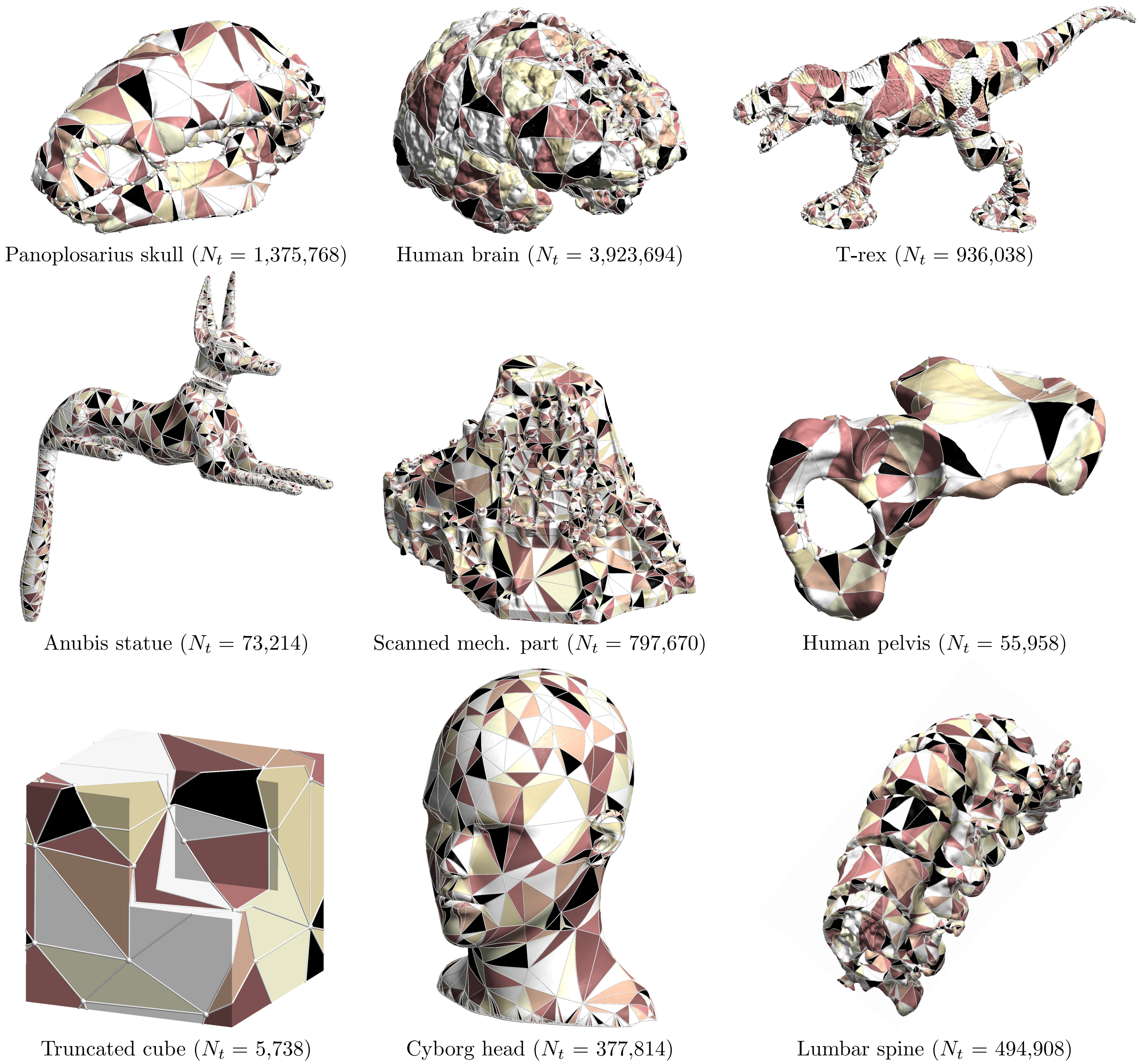


Figure 3: Decimated mesh (top), valid and optimized geodesic mesh using geodesic Delaunay flips (middle) and mesh after shortest edge bissection refinement (bottom).



Panoplosarius skull (N_t = 1,375,768) Human brain (N_t = 3,923,694) T-rex (N_t = 936,038)

Anubis statue (N_t = 73,214) Scanned mech. part (N_t = 797,670) Human pelvis (N_t = 55,958)

Truncated cube (N_t = 5,738) Cyborg head (N_t = 377,814) Lumbar spine (N_t = 494,908)

Figure 4: Various examples of geodesic meshes.

References

[1] T. J. Barth and H. Deconinck, *High-order methods for computational physics*, vol. 9, Springer Science & Business Media, 2003.

[2] P.-A. Beaufort, C. Geuzaine, and J.-F. Remacle, *Automatic surface mesh generation for discrete models–a complete and automatic pipeline based on reparametrization*, Journal of Computational Physics, 417 (2020), p. 109575.

[3] K. Crane, M. Livesu, E. Puppo, and Y. Qin, *A survey of algorithms for geodesic paths and distances*, arXiv preprint arXiv:2007.10430, (2020).

[4] C. Dobrzynski and G. El Jannoun, *High order mesh untangling for complex curved geometries*, PhD thesis, INRIA Bordeaux, équipe CARDAMOM, 2017.

[5] T. J. Hughes, J. A. Cottrell, and Y. Bazilevs, *Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement*, Computer methods in applied mechanics and engineering, 194 (2005), pp. 4135–4195.

[6] A. Johnen, J.-F. Remacle, and C. Geuzaine, *Geometrical validity of curvilinear finite elements*, Journal of Computational Physics, 233 (2013), pp. 359–372.

[7] J. S. Mitchell, D. M. Mount, and C. H. Papadimitriou, *The discrete geodesic problem*, SIAM Journal on Computing, 16 (1987), pp. 647–668.

[8] M.-C. Rivara, *Mesh refinement processes based on the generalized bisection of simplices*, SIAM Journal on Numerical Analysis, 21 (1984), pp. 604–613.

[9] W. J. Schroeder, J. A. Zarge, and W. E. Lorensen, *Decimation of triangle meshes*, in Proceedings of the 19th annual conference on Computer graphics and interactive techniques, 1992, pp. 65–70.

[10] N. Sharp, Y. Soliman, and K. Crane, *Navigating intrinsic triangulations*, ACM Transactions on Graphics (TOG), 38 (2019), pp. 1–16.

[11] M. Stees, M. Dotzel, and S. M. Shontz, *Untangling high-order meshes based on signed angles*, Proceedings of the 28th International Meshing Roundtable, (2020).

[12] T. Toulorge, C. Geuzaine, J.-F. Remacle, and J. Lambrechts, *Robust untangling of curvilinear meshes*, Journal of Computational Physics, 254 (2013), pp. 8–26.

[13] C. Xu, T. Y. Wang, Y.-J. Liu, L. Liu, and Y. He, *Fast wavefront propagation (FWP) for computing exact geodesic distances on meshes*, IEEE transactions on visualization and computer graphics, 21 (2015), pp. 822–834.

[14] X.-Y. Zhao and C.-G. Zhu, *Injectivity conditions of rational Bézier surfaces*, Computers & Graphics, 51 (2015), pp. 17–25.