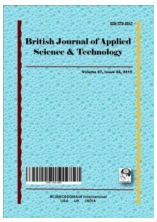


British Journal of Applied Science & Technology

21(1): 1-14, 2017; Article no.BJAST.32827
ISSN: 2231-0843, NLM ID: 101664541

Geometry and Topology-based Segmentation of 2-Manifold Triangular Meshes in R^3

Stella Orozco[1], Arno Formella[1], Carlos A. Cadavid[2], Oscar Ruiz - Salguero[2*] and Maria Osorno[2]

[1] *Departamento de Informatica, Universidade de Vigo, Spain.*
[2] *Laboratorio de CAD/CAM/CAE, Universidad EAFIT, Colombia.*

Authors' contributions

The work was carried out in collaboration between all authors. Authors CAC and MO undertook the theoretical part of spectra of meshes. Authors MO and ORS conducted the implementation of the methods. Author AF coordinated the whole project. All authors read and approved the final manuscript.

Article Information

DOI: 10.9734/BJAST/2017/32827
Editor(s):
(1) Samir Kumar Bandyopadhyay, Department of Computer Science and Engineering, University of Calcutta, India.
Reviewers:
(1) Sara Gonizzi Barsanti, Politecnico di Milano, Italy.
(2) Ravi Koirala, Kathmandu University, Nepal.
(3) Yu Zhang, USA.
(4) Rajagopal Gayathri, Anna University, India.
Complete Peer review History: http://www.sciencedomain.org/review-history/19175

Short Research Article

Received: 17th March 2017
Accepted: 14th May 2017
Published: 23rd May 2017

ABSTRACT

This manuscript reports a geometrical and a topological methods to segment a closed triangular 2-manifold mesh $M \subset R^3$. The mesh M does not self-intersect) and has no border (i.e. *watertight*. Geometrical and topological segmentation methods require a Boundary Representation (BRep) from M. Building the BRep for M uniforms the triangle orientations, and makes explicit triangle and edge - counter edge adjacency. In the context of Reverse Engineering, the sub-meshes produced by the segmentation are subsequently used to fit parametric surfaces, which are in turn trimmed by the sub-mesh boundaries (forming FACEs). A Full Parametric Boundary Representation requires a seamless set of FACEs, to build watertight SHELLs. The fitting of parametric surfaces to the triangular sub-meshes (i.e. sub-mesh parameterization) requires quasi-developable sub-meshes.

**Corresponding author: E-mail: oruiz@eafit.edu.co;*

As a result, our geometric segmentation places 2 neighboring triangles in the same sub-mesh if their dihedral angle is $\pi \pm \eta$ for a small η (angle between their triangle normal vectors is a small η angle). On the other hand, our topological segmentation heuristic classifies triangles in a common sub-mesh if the value of the First eigenfunction of the triangulation graph Laplacian in these triangles falls in the same bin of a histogram formed with the eigenfunction values. The segmentation will obviously depend on the histogram bin distribution. The data sets processed indicate that geometrical segmentation is more convenient for mechanical parts with analytical surfaces. Conversely, topological segmentation works better for organic or artistic shapes. Future work is needed on the tuning of both the dihedral threshold η (for geometrical segmentation) and on the bin distributions of the eigenfunction (for topological segmentation).

Keywords: Computational geometry 3D mesh segmentation; spectral graph theory; boundary representation (BRep).

Glossary

M	:	Triangular Mesh in R^3. $M = \{t_1, t_2, ...\}$, with t_i= i-th triangle in M
∂M	:	Boundary of a mesh M. ∂M is a closed LOOP in R^3.
S	:	Partition or Segmentation of M
G	:	Connectivity Graph of M with $G = (V, E)$, V: vertices, E: edges.
W	:	Adjacency matrix of graph G.
D	:	Degree matrix of graph G.
K	:	Laplacian matrix of graph G.
U	:	Eigenvectors of graph Laplacian K.
Λ	:	Eigenvalues of graph Laplacian K.
bin	:	Given a sequence $a_0 < a_1 < a_2 < ... < a_f$, with $a_i \in R$, the bins are the intervals $[a_i, a_{i+1}]$.
η	:	Threshold of the Dihedral angle between two adjacent triangles in M.
BRep	:	Boundary Representation (BODY, LUMPs, SHELLs, FACEs, LOOPs, EDGEs, VERTEX) of a solid object in R^3. The parametric surfaces that carry the FACEs of a BRep are usually smooth (C^1, C^2, unless the model specifically requires a flat FACE).
Tr-BRep	:	BRep whose FACEs are exclusively triangles. In this case, the FACEs have no holes.
FACE	:	A connected region on a parametric surface $S \subset R^3$ (BRep context).
LOOP	:	Closed piecewise smooth curve in R^3 (BRep context).
BFS	:	Breadth First Search.
CCW	:	counter-clockwise.
RE	:	Reverse Engineering.

1 INTRODUCTION

This manuscript presents two implementations (geometrical and topological) for the segmentation of a 2-manifold watertight triangular Tr-Brep.

1.1 Problem Specification

- **Given:** A closed connected triangular 2-manifold M in R^3. M is the C^0-continuous approximation ($M \approx \partial B$) or tessellation of the boundary (∂B) of a solid $B \subset R^3$

- **Goal:** A partition S of M, such that: $S = \{M_1, M_2, ..., M_w\}$ is a set of sub-meshes such that $M_i \cap M_j = \Phi$ and $\bigcup_{i=1}^{w} M_i = M$. Therefore, S covers the entire triangulation M.

1.2 Context

A triangular mesh M usually is the clean, processed result of digitizing a 3D surface or 3D object B and adding neighboring information to such a point sample. After a meshing process, M is a 2-manifold, which means that all its local neighborhoods are isomorphic to the 2D unit disk. It is desirable that the triangles of M be as equilateral as possible.

For Reverse Engineering purposes, it is additionally convenient to have homogeneously sized triangles over the mesh M. RE requires at some stage to have a parametric, as opposed to a triangular Boundary Representation of M. In such BReps, FACEs are mounted on smooth parametric surfaces, and may have internal holes, thus producing internal LOOPs, in addition to the external LOOP of the FACE. In triangular Tr-BReps, all FACEs are triangular and the have no holes. The segmentation of the mesh M is a requisite for reaching a full BRep for M. In turn, a parametric BRep is required for re-design, manufacturing, finite element analysis, among other processes.

The need for parametric BReps influences the mesh segmentation, as follows. Opposite heuristics compete to define the size of a sub-mesh M_i: (1) few triangles per sub-mesh lead to BReps with high FACE fragmentation, but the parameterizations are easier and closer to isometries. (2) large sub-meshes are more likely to have no parameterization. RE segments a mesh M, attempts parameteri - zation of the sub-meshes M_i, and re-arranges the segmentation if the parameterization for M_i does not exist or is a significant departure from isometric parameterizations. Actual usage of the resulting Parametric BReps is by no means straightforward. The sole import of BReps into Finite Element Analysis or CAM packages is full of obstacles, due to the semantic and numeric limitations of either the BRep itself or of the standard (IGES, STEP, SAT) used to convey the BRep. A B-rep may look correct in the CAD stage, but its shortcomings surface at the CAM or FEA stages, thus forcing to re-take the mesh segmentation / parameterization steps. This process is, at the present time, highly interactive, and time - consuming. The process of point sampling, mesh cleaning, mesh segmentation and parameterization, BRep creation, exporting to FEA, FEA mesh creation takes weeks of highly - trained user interaction.

RE presents applications ranging from design, virtually test, produce and modify the designs of challenging devices or shapes (e.g. bones, medical devises, sculptures, mechanical components, and Mandelbrot - like actual objects, such as filigree jewelry [1]).

An important notation observation for this manuscript is that *Tr-BRep* is a Boundary Representation for objects whose FACEs are flat and triangular. On the other hand, (Full) BReps are the ones whose FACEs are mounted on flat or smooth parametric surfaces (i.e. the usual meaning for BRep term).

2 LITERATURE REVIEW

Mesh segmentation is usually analyzed with two taxonomies: (1) part vs. surface , and (2) topology vs. geometry approaches. Part vs. Surface (survey in [2]) taxonomy establishes 2 categories: (a) part-type, which uses mesh's volumetric attributes (concavity, convexity, etc.), and (b) surface-type, which is based on attributes such as dihedral angles, planarity, curvature, etc. This taxonomy relies on feature identification, which is a very vulnerable approximation (Ref. [3]). Because of this reason, we prefer to use the taxonomy (2) above. Thus, we survey and implement two methods: Dihedral Angle and Graph Laplacian segmentation.

2.1 Dihedral Segmentation

Ref. [3] reports difficulty in recognizing depression / protrussion features in a 3D mesh and therefore proposes a hybrid segmentation algorithm that use the dihedral angle for the sharp

feature edges, and the shape of the triangles to identify the non sharp feature edges. The importance of such feature edges is that they are used in the identification of the boundaries of a region and thus they separate the interest region from the surrounding neighborhoods.

Ref. [4] presents 3 steps: (1) Extraction of *analytic* geometric primitives, (2) intersection between primitives in (1) resulting in LOOPs or wires bounding the BReps FACEs, and (3) BRep model creation. The first step is enhanced with a pre-segmentation stage based on the dihedral angle between pairs of triangles. The pre-segmentation step improves the precision therefore, the resultant geometric primitives are closer to real world objects. Ref. [3] only works with planes, spheres and cylinders. In ou approach, we prefer not to make pre-judgement on the nature of the feature underlying a sub-mesh since in many cases the feature is not an analytic one.

2.2 Graph Laplacian Segmentation

The survey [5] expresses that geometrical (e.g. dihedral angle) segmentation of meshes is a particular case of spectral mesh processing. This particular case is stated if the connectivity matrix of the mesh graph is replaced by a dihedral angle connectivity matrix. The manuscript starts with a arbitrary A operator (matrix) that relates vertices i and j of the mesh. An eigen-decomposition $A = U * \Lambda * U^T$ follows, which finds the eigenvectors U and eigenvalues Λ of A. The mesh can be approximated with the k leading eigenvectors $\tilde{V} = U_k * U_k^T * V$. k produces a $(k-1)$ - Dimensional approximation of V.

Ref. [6] bisects the graph mesh representation into balanced parts having approximately the same number of vertices, by using a minimum number of edges (cut size) to split the mesh. The Fiedler vector of the spectrum of the graph Laplacian is obtained. The sign of each entry in the Fiedler Vector serves as criterion to partition the mesh. Another alternative is to use eigenvectors 2nd and 3rd, not only the 2nd (i.e. Fiedler) of the graph Laplacian. For our purposes, however,we require to split the mesh in more than 2 parts.

Ref. [7] uses Geometric and Laplacian spectral analysis for mesh segmentation. The geometric operator M measures the curvatures at vertices $i$and j of the edge (i, j). The Laplacian operator L is usual one, determined by the mesh connectivity. In either case (M or L), the mesh is projected in 2D via the first 3 eigenvectors. The outer contour of the 2D projection is extracted, and its extremes determine the limbs of the object. However, the segmentation is basically manual, by using the mentioned M and L indicators.

Ref. [8] uses spectral graph theory to approximate the eigenvalues - eigenfunctions of the Laplace-Beltrami operator of a compact Riemannian manifold. This reference runs experiments showing that the eigenvector of the first nonzero eigenvalue is a Morse function with minimal number of critical points for the given manifold. In the present article, we use the Laplace operator for the Topology - based segmentation.

An important characteristic of all the discussed methods is the need for manual input to use the spectral decomposition to materialize the segmentation.

2.3 Theory vs. Practice in Reverse Engineering

The actual mesh segmentation practice in Reverse Engineering (RE) presents significant departures from the numerous published theoretical approximations. Fig. 1 presents the main processes, all of which are currently user - intensive. Assuming as input a point sample, the processing follows: (1) Addition of connectivity, to generate a manifold, faithful-to-object triangular mesh. (2) Re-meshing to ensure quality measures (quasi - equilateral triangles, curvature - sensitive size, etc.), (3) Triangular B-Rep generation (consistency of triangle orientation, neighborhood and border information, SHELL, LUMP, BODY information), (4) Triangular BRep segmentation. (5) Mesh parameterization to form trimmed FACEs. (6) Promotion of FACE set to full BRep. (7) Export of

BRep to CAD or FEA software and usage (e.g. FEA meshing and analysis). All these processes use intensive user input in actual engineering practice. In addition, it is normal that, for example, step 7 above fails, causing iteration in any or all of 1 to 6 processes, until FEA meshing or Tool Path can be successfully completed. In particular, Fig. 2 presents a mesh segmentation proposed by Geomagic software, which still required (as with all comparable commercial tools) intensive user cleaning, to achieve correct patch fairing and manifold conditions. Such segmentations are completely different from the ones proposed in literature. It is not in the capacity or goal of this manuscript to research why the academic meshing is not applied in the real practice. Such a departure also indicates that mesh segmentation is far away of being a closed topic.

Fig. 2 shows that current industrial mesh partition for RE basically (1) uses a highly granulated triangle set (i.e. small triangles), (2) approximately rectangular regions are formed with the small triangles sets. (3) cylindrical or conical FACEs are avoided, since a discontinuity in the parameter $\theta = 2\pi$ impedes a (bijective) parameterization. Therefore, geometrical or topological segmentations that produce cylinders, cones or similar non-bijections are avoided. Each case study in Fig. 2 takes approx 150 hours for the process displayed in Fig. 1.

2.4 Conclusions of the Literature Review

The contrast between the theoretical articles and the industrial practice in Mesh Segmentation (for RE) shows that, whichever the reason, theoretical spectral segmentation is not industrially applied. The results of segmentation by commercial cutting-edge software present sub-meshes much smaller as compared with the ones prescribed by spectral methods.

We do not intend in this manuscript to explain the distance between spectral methods and industrial practice. Instead, we seek to evaluate the application of geometrical dihedral angle vs. topological (spectral) methods with respect to the piece at hand.

3 METHODOLOGY

Fig. 3 shows the flowchart of the BRep construction and segmentation processes. The process begins with the BRep construction which consists basically in correcting the mesh triangle orientation and the topology reconstruction. The next step is the construction of the structures that hold the segmented data set and, the final step (Laplacian and Dihedral segmentation) is the application of the intermediate results to actually segment the mesh.

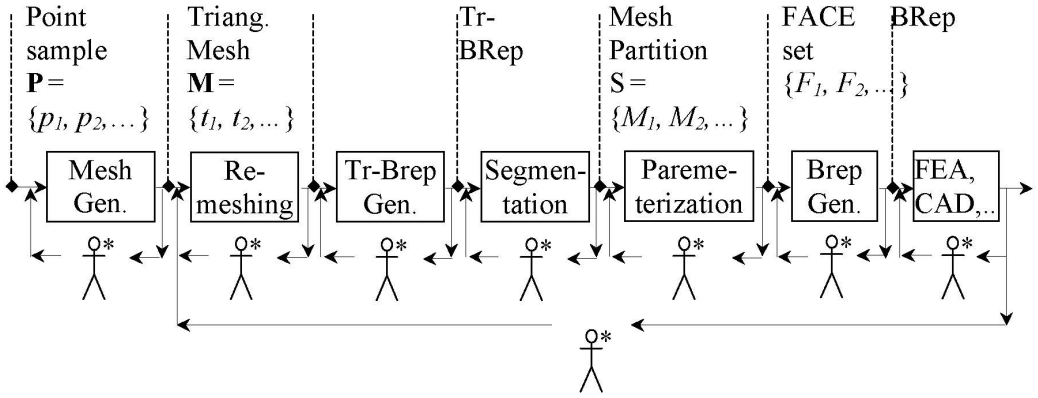
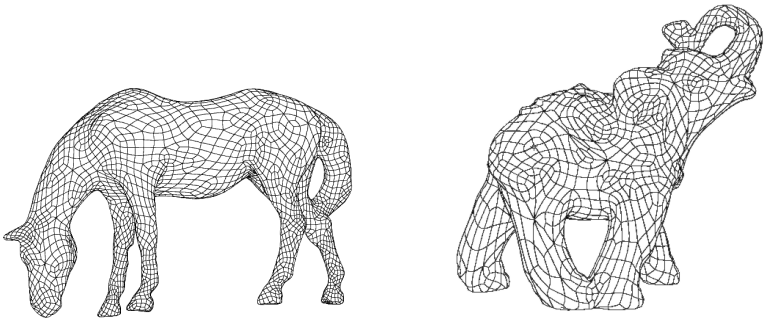


Fig. 1. Current user role in reverse engineering



(a) Grazing Horse Data Set ([9]). (b) Elephant Data Set ([10]).

Fig. 2. Actual industrial segmentation with geomagic ™ ([9], [10])

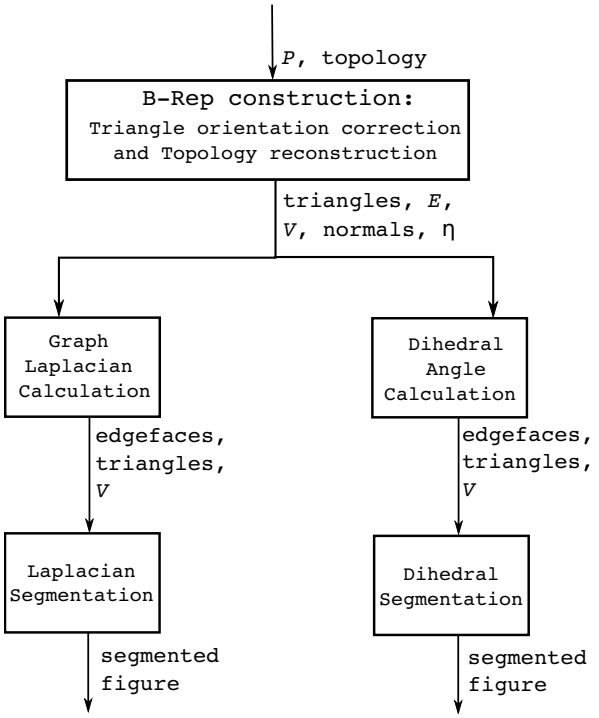


Fig. 3. Synthesis of the BRep construction and segmentation processes

3.1 BRep Construction

To perform a segmentation the surface must be a 2-manifold M embedded in E^3. Informally, this means for all point $p \in M$ there exists a small enough ball $B(p,r)$ around p such that $B(p,r) \cap M$ is isomorphic to the 2D disk S^2. When M is not completely closed, there are points (called boundary or border points) at the edge of M for which the disc D is homeomorphic to half of S^2 (half - disk). In such case it is said that M is a 2-manifold with border, embedded in E^3 ([11]).

The simplest representation for M ia an undirected graph $G = (V, E)$ with V = set of

mesh vertices and $E \subseteq V \times V$ = set of edges $e = (v_i v_j)$.

In our segmentation process, M is promoted to be a Boundary Representation (Fig. 4), therefore having consistent FACE orientation, information of neighboring triangles, mesh border, vertex incidente set, etc. This is a Boundary Representation having exclusively triangular FACEs (Tr-BRep). As in the general BRep, the FACEs have orientation. If the SHELL is closed, each EDGE receives 2 FACEs and the EDGEs in all FACES are in counter-clockwise (CCW) sense with respect to the outwards normal vector. This fact causes that the same segment $(v_i v_j)$ be traversed in opposite directions in the two FACEs in which it appears. If the SHELL is open, the outward normal direction is ambiguous. However, all the FACEs still have consistent normal vector orientation. In an open SHELL, the border EDGEs are precisely the ones that receive only one FACE. For the current discussion we will make no difference (as in usual BRep) between a VERTEX and its position $p = (x, y, z)$. Likewise, we will refer to the EDGE $(v_i v_j)$ or $(v_j v_i)$ without using the usual additional notation (EDGE , CO-EDGE). Fig. 5 shows the Cat data set before and after the BRep construction.

3.2 Dihedral Segmentation

3.2.1 Dihedral angle equivalence relation

Consider the relation $r_\eta(t_i, t_j)$ as satisfied if there exists a path of triangles in M departing from triangle t_i and reaching triangle t_j so that the normal vectors of 2 triangles sharing an edge form an angle smaller than η. This is an equivalence relation because: (i) $r_\eta(t_i, t_i)$ trivially holds, (ii) $r_\eta(t_i, t_j)$ implies $r_\eta(t_j, t_i)$, and (iii) $r_\eta(t_i, t_j)$ and $r_\eta(t_j, t_k)$ implies $r_\eta(t_i, t_k)$. This equivalence relation $r_\eta()$ applied in the form of transitive closure generates a partition or segmentation of the mesh M, as per section 1.1. This partition is calculated with the Breadth - First Search (BFS) algorithm using the relation $r_\eta()$. Figs. 6(f) and 6(g) show the M after the segmentation process for different values of η. Notice in Fig. 6(f) that η being too small is an excessively strong criterion, defining practically flat sub-meshes, and causing over-segmentation.

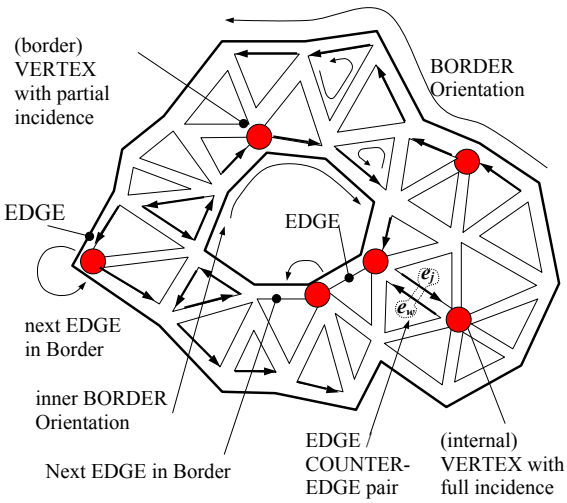
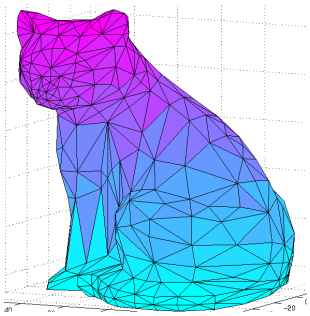
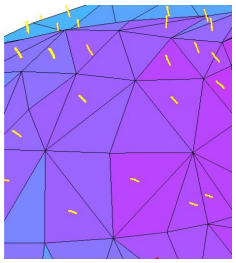


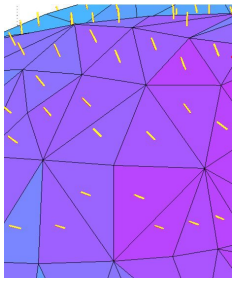
Fig. 4. Triangular Mesh M [12]. Neighboring triangles sharing VERTICES and EDGEs are depicted separated only for illustration purposes



(c) Cat Data Set.

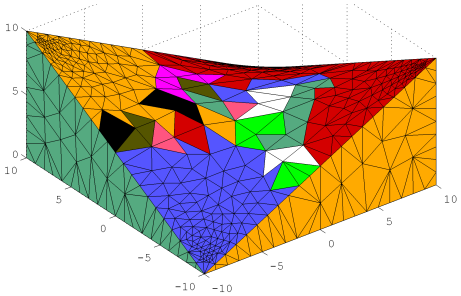


(d) Raw, un-oriented Mesh.

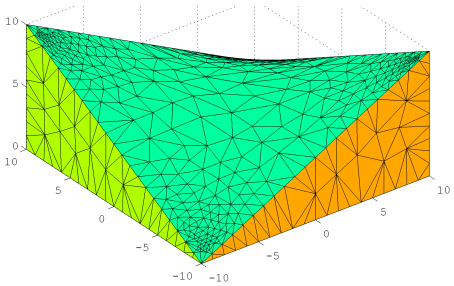


(e) Oriented Mesh.

Fig. 5. Cat Data Set (downloaded from [13]) before and after BRep Construction. FACE degradee colors are proportional to the z coordinate and have no segmentation semantics

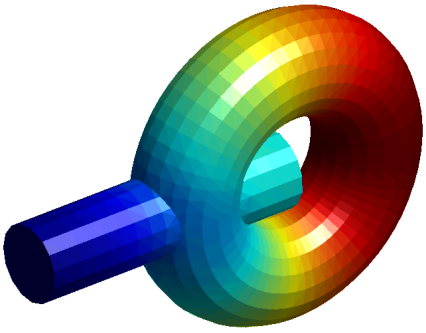


(f) Threshold η = 5 deg. causes over-segmentation. Repeated colors are caused by narrow palette, and not by segmentation strategy.

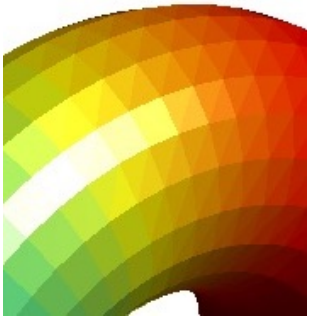


(g) Threshold η = 20 deg. achieves a correct segmentation.

Fig. 6. Ramp Data Set. Dihedric segmentation with varying threshold angle η



(h) Full View.



(i) Detail.

Fig. 7. Hook Data Set. Fiedler Vector - based Coloring

3.3 Graph Laplacian Segmentation

In this work, Combinatorial Graph Laplacian is the spectral method applied to a mesh M. The Tr-BRep contains the information of the Intrinsic Geometry of the Graph (i.e. independent of the embedding in $\mathbb{R}^3$). This method follows the steps in [5]:

(a) Build the adjacency matrix *W* of M

$$W_{ij} = \begin{cases} 1, & if\ (i,j) \in E, \\ 0, & otherwise. \end{cases} \tag{3.1}$$

(b) Find the degree matrix D

$$D_{ij} = \begin{cases} d_i = |N(i)|, & if\ i = j, \\ 0, & otherwise. \end{cases} \tag{3.2}$$

where the set of neighbors of a vertex i in G is denoted by $N(i)$ and d_i denotes the degree of vertex i. W and D are $m \times m$ matrices, where $m = |V|$.

(c) Build the graph Laplacian matrix K

$$K = D - W \tag{3.3}$$

(d) Calculate the Eigen - decomposition of *K*:

$$K = U\,\Lambda\,U^T \tag{3.4}$$

(e) Find the Fiedler vector (eigenvector corresponding to the smallest non zero eigenvalue of Eq. 3.4). The Fiedler vector is considered as the optimal mapping between the real line and the main axis along the object ([14] p. 7). Fig. 7 shows on manifold M a linear mapping between the Fiedler vector and a color palette. The average color value over the 3 vertices of each triangle is used as color for the triangle (Fig. 7(i)).

(f) Segment the Fiedler vector (and 2-manifold M) as follows.

(1) Transform geometry from 3D to 2D by the Laplacian operator. The coordinates of the mesh vertices are represented by V, which are also a linear combination of the eigenvectors of K. V and be projected onto the subspace spanned by the k leading eigenvectors of K. The projection is $\tilde{V}$:

$$\tilde{V} = U_{1\ldots k}\,U_{1\ldots k}^T\,V \tag{3.5}$$

where $U_{1\ldots k}$ contains the first k columns of U. Choosing $k = 3$, we obtain a planar shape, since the first eigenvector is constant [7].

(2) Segment the 2D figure. Apply (for example) the red segmentation polygon in Fig. 8(j) by using the segmentability of the 3D figure transformed to the plane.

(3) Segment the 3D mesh. The 2D partition in Fig. 8(j) determines the 3D segmentation in Fig. 8(k). Analog 2D partitions (e.g. with 4 regions) engender the 3D segmentation in Fig. 8(l).

4 RESULTS

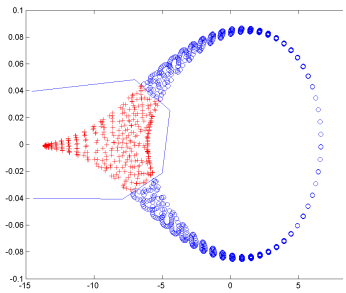
The Ramp dataset shows the effect of the threshold estblished in the dihedral angle partition. A dihedral angle $\eta \to 0$ deg represents a very strong condition, as basically excludes curved faces. For example, $\eta = 5$ deg causes an over-segmentation of the upper part of the Ramp dataset. A higher value of η (20 deg) eliminates the over-segmentation, rendering intuitive results for the workpice.

Fig. 9 shows results of the dihedral segmentation. These mechanical part meshes are correctly segmented into meaningful sub-meshes. These sub-meshes represent in turn correct FACEs in a BRep. Dihedral angle partition works correctly whenever the continuity among the object FACEs is C^0. If such a continuity is C^1 or superior, the dihedral angle partition loses effectiveness as the angle tuning becomes more demanding.

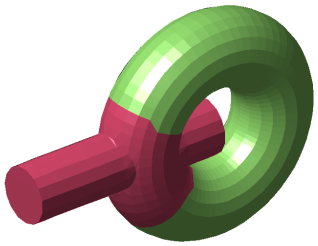
The Hook dataset (Fig. 8) was mapped to 2D (Fig. 8(j)) by using the first 3 eigenvectors of the Mesh Laplacian spectrum. A manual bisection was performed as displayed in Fig. 8(j). The 3D effect of such a bisection appears on the 3D mesh M in Fig. 8(k). The 2D projection in Fig. 8(j) can be segmented onto more subsets. Such a partitions would produce a 3D mesh segmentation such as the one shown in Fig. 8(l).

Fig. 10 shows two organic figures (Elephant, Horse) segmented with the graph Laplacian.

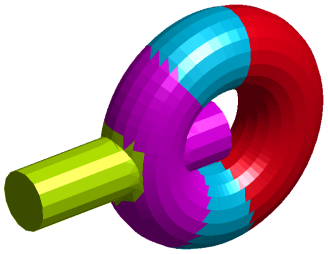
Fig. 10(p) shows the Elephant data set colored according to the Fiedler vector (eigenvector U_2). Fig. 10(q) displays the Elephant segmentation based on the bins of a 4 bin histogram of the Fiedler vector. Fig. 10(r) shows the Elephant Data Set partitioned via the manual selection of the segmentation polygon (as explained in section 3.3).



(j) Manual 2D Segmentation enforcing 2 regions.

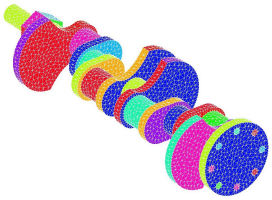


(k) 3D Segmentation Using 2D Polygon in Fig. 8(a)

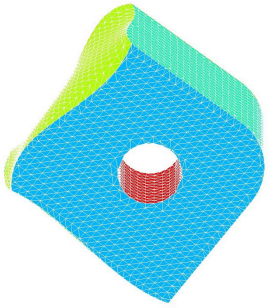


(l) 3D Segmentation Using 4 regions in 2D.

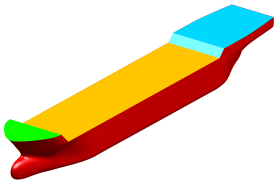
Fig. 8. Hook data set segmented using the Fiedler Eigen-vector



(m) Crankshaft. Dihedral angle η = 40 deg.



(n) Jello Block. Dihedral angle η = 40 deg.



(o) Ship Hull. Dihedral angle η = 40 deg.

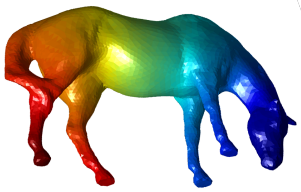
Fig. 9. Dihedral Segmentation for Crankshaft, Jello Block and Ship Hull Datasets

In the case of the Horse dataset the results show the same process: Fig. 10(s) is colored according to the Fiedler vector. In Fig. (10(t)), the bins of the histogram are used for the segmentation. Notice that partitions are not connected. Both rear hoofs belong to the same segment while the front hoofs and the head are in another (disconnected) partition. Fig. 10(u) shows the Horse segmented via the 2D segmentation polygon: notice that the body of the Horse (contrary to 10(t)) is modeled in a single partition.

(p) Laplacian Fiedler vector.

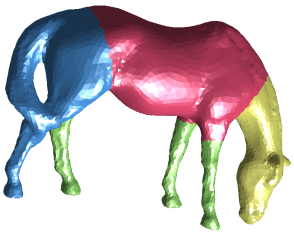
(q) 4-bin Histogram of Fiedler vector.

(r) 3D Result of 2D Manual Segmentation.



(s) Laplacian Fiedler vector.

(t) 4-bin Histogram of Fiedler vector.



(u) 3D Result of 2D Manual Segmentation.

Fig. 10. Horse and Elephant Data Sets ([9], [10])

Table 1. Data set statistics

Data Set	Dihedral Threshold (degrees)	Num. Triangles in Mesh	Num. of sub-meshes obtained	Incorrectly classified triangles
Jello Block	40	6208	9	0
Ramp	5	1226	6	57
Ramp	40	1226	6	0
Crankshaft	40	13356	64	0
Ship Hull	40	45274	11	4

Table 2. Time complexity Estimation. Left: Spectral Method (m: number of Vertices in M). Right: Dihedral Method (n: number of triangles in M). The number of triangles n and vertices m of mesh M usually hold a linear relation (i.e. $O(m^3) = O(n^3)$)

Spectral Method			Dihedral Method	
Sub - Task	Cost		Sub - Task	Cost
Tr-BRep. Construction	$O(n^3)$		Tr-BRep. Construction	$O(n^3)$
Construction of Adjacency Matrix W	$O(m^2)$		Partition of M under Equivalence relation $r_\eta()$	$O(n^3)$
Construction of Degree Matrix D	$O(m^2)$			
Singular Value Decomposition	$O(m^3)$			
Laplacian Map of M to $\mathbb{R}^2$	$O(m^3)$			
Partition of Fiedler Vector using $\mathbb{R}^2$	$O(m)$			
TOTAL:	$O(m^3)$		TOTAL:	$O(n^3)$

5 CONCLUSIONS

Table 1 shows that, although additional tests and data sets are required, Dihedral Angle segmentation seems to work in reasonable way when mechanical - part C^0 - continuous meshes are segmented. The dihedral angle is in no manner a unique tuning parameter, but certainly captures the fact that in mechanical parts different functionality accompanies different parametric surfaces. In organich shapes, however, the dihedral angle loses segmenting power.

Reverse Engineering requires for re-design the full Boundary Representation of the piece. In BRep, a parametric surface may carry several FACEs, but the contrary is not allowed: A FACE cannot encompass several parametric surfaces. Therefore, in mesh segmentation it is preferable to under-estimate than to over-estimate the sub-meshes.

The capacity to parameterize the sub-meshes places similar priority: sub - meshes with large area are more difficult to parameterize.

For Dihedral Angle segmentation, it is therefore not convenient to specify a large η dihedral angle. Obviously, what *large* means is matter of user tuning. From our tests, we see that η should not surpass 20 degrees. On the other hand, a very η angle near 0 is not convenient, because it will cause that the BRep be very similar to the Tr-BRep (i.e. to favor planes -triangles- as carries surfaces for FACEs). Dihedral angle segmentation is more convenient for mechanical parts, in which the continuity between FACEs be C^0 and not C^1 or superior. Inter - FACEs C^0 continuity allows to use the dihedral angle criterion for a clear cut among the sub-meshes.

Spectral - based segmentation is more convenient where dihedral angle loses power to bound the sub-mesh. Such cases occur when the mesh smoothly evolves all over its extent. In these cases, dihedral angle would extract only one sub - mesh (i.e., the original mesh itself).

Our results using spectral segmentation are (obviously) not complete, but they are in the right direction, as they do not cause an over-segmentation. The obvious step ahead is to further segment the current sub-meshes, in order to achieve sub - mesh parameterization. Our current spectral results still allow for further segmentation.

The following aspects must be kept in mind when formulating the future work: (1) both, the scientific geometrical and spectral approaches to mesh segmentation require of massive user interaction. The theoretical foundations are certanly sound, but all algorithms discussed require a tuning. (2) Industrial mesh segmentation is at this time completely different from the prescribed theoretical algorithms.

Therefore, a necessary step is to assess why the theoretical and industrial mesh segmentations differ so much, and to assess whether it is worth to approximate the two positions and how to do so.

Computing Time Expenses. The methods presented were decomposed in sub-tasks, executed on diverse hardware. Also, the execution times are obviously dependent on the hardware power. Therefore, the time count would not give a fair comparison of the two alternatives. To address this issue, we have analyzed the computational complexity of the Spectral and Dihedral methods (Table 2). The time complexities assigned to the sub - tasks assume no special optimization and are the usual ones for them. The overall time complexity of the Spectral and Dihedral methods is equivalent ($O(n^3)$ or $O(m^3)$), since the number or Triangles n and Vertices m of the mesh M keep a linear relation.

ACKNOWLEDGEMENTS

The data sets used in this work were provided by the CAD CAM CAE Laboratory at Universidad EAFIT, Colombia.The authors thank the High Performance Computing Facility APOLO of U. EAFIT. In particular, special thanks are in order for Dr. Eng. Juan Lalinde and Eng. Juan David Pineda for their valuable help in using such services.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

References

[1] Stamati V, Antonopoulos G, Azariadis Ph, Fudos I. A parametric feature-based approach to reconstructing traditional filigree jewelry. Computer-Aided Design. 2011;43(12):1814–1828.

[2] Shamir A. A survey on Mesh Segmentation Techniques. Computer Graphics Forum. 2008;27(6):1539-1556.

[3] Sunil VB, Pande SS. Automatic recognition of features from freeform surface CAD models. Computer-Aided Design. 2008;40(4):502-517.

[4] Beniere R, Subsol G, Gesquiere G, Le Breton F, Puech W. A comprehensive process of reverse engineering from 3D meshes to CAD models. Computer-Aided Design. 2013;45:1382-1393.

[5] Zhang H, van Kaick O, Dyer R. Spectral mesh processing. Computer Graphics Forum. 2010;29(6):1865-1894.

[6] Gotsman C. On graph partitioning, spectral analysis, and digital mesh processing. In: In Proc. Intl. Conf. Shape Modeling and Applications. 2003;1865-1894.

[7] Liu R, Zhang H. Mesh Segmentation via spectral embedding and contour analysis. Eurographics. 2007;26(3).

[8] Cadavid C, Osorno MC, Ruiz O. On the critical point structure of eigenfunctions belonging to the first nonzero eigenvalue of a genus two closed hyperbolic surface. Science Journal of Physics. 2012;2012:1-8.

[9] Toro D, Ruiz-Salguero O. Reverse engineering of grazing horse data set. U. EAFIT. Technical Report. U. EAFIT; 2014.

[10] Puerta-Ochoa J, Ruiz-Salguero O. Reverse engineering of indian elephant data set. U. EAFIT. Technical Report. U. EAFIT; 2014.

[11] Spivak MD. Calculus on manifolds: A modern approach to classical theorems of advanced calculus. Westview Press; 1971.

[12] Ruiz O, Neugebauer P. Topologically consistent partial surface reconstruction from range pictures. IASTED Conference on Computer Graphics and Imaging (CGIM); 2000.

[13] 3D Cafe Free Models. Cat Dataset; 2003. Available: http://www.3dcafe.com/

[14] Reuter M. Hierarchical shape segmentation and registration via topological features of laplace-beltrami eigenfunctions. Intl. Journal of Computer Vision. 2010;89(2):287-308.

Peer-review history:
The peer review history for this paper can be accessed here:
http://sciencedomain.org/review-history/19175