Introduction to Reeb Graphs and Contour Trees

Lecture 15

Scribed by: ABHISEK KUNDU

Sometimes we are interested in the topology of smooth functions as a means to analyze and visualize intrinsic properties of geometric models and scientific data. Creation and destruction of components of the level set could be an important visual technique for this purpose. Reeb graphs are obtained by contracting the connected components of the level sets to points. They could be useful visual analytical tool because they express the connectivity of level sets.

DEFINITION: (*Level Set*)

A *level set* of a real-valued function f of n variables is a set of the form:

$\{ (X_1,...,X_n) \mid f(X_1,...,X_n) = c\}$

where c is a constant. In other words, it is the set where the function takes on a given constant value.

When the number of variables is two, this is a *level curve* or *contour line*, if it is three a *level surface*, and for higher values of n the level set is a *level hypersurface.*

Sublevel set: of f is a set of the form

$\{ (X_1,...,X_n) \mid f(X_1,...,X_n) \leq c\}$

Contour: at a given value c, contour is the connected component of level set X_c [Figure 1]

Equivalence of points: two points x and y are equivalent if they belong to the same contour of X_c for some given constant c. [Figure 1]

Quotient topology and space:

Let us denote ~ to be an equivalence relation defined on a topological space X. Let, $\utilde{X}$ be the set of equivalence classes and let $\psi : X \rightarrow \utilde{X}$ map each point x to its equivalence class

DEFINITION: *Quotient topology* of $\utilde{X}$ consists of all subsets U belonging to $\utilde{X}$ whose preimages, $\psi^{-1}(U)$, are open in X. The set $\utilde{X}$ together with the quotient topology is the *quotient space* defined by ~.

DEFINITION: *Reeb Graph*

Let f : X → R be continuous function and call a component of a level set a *contour*. Two points x and y ϵ X equivalent if they belong to the same component of $f^{-1}(t)$ with $t = f(x) = f(y)$. The *Reeb graph* of f , denoted as $R(f) = X_{\sim}$, is the quotient space defined by this equivalent relation.

The Reeb graph has a point for each contour and the connection is provided by the map

$$\psi : X \rightarrow R(f)$$

that is, continuous collapsing of each contour of f . [Figure 1]

Let, $\pi : R(f) \rightarrow \mathrm{R}$ be defined such that $f(x) = \pi(\psi(x))$; one can construct the level sets by going backward, from the real line to the Reeb graph to the topological space. Given t ϵ R, one can get $\pi^{-1}(t)$, a collection of points in $R(f)$, and $\psi^{-1}(\pi^{-1}(t))$, the corresponding collection of contours that make up the level set defined by t.

Reeb graph is a very useful data structure to accelerate the extraction of level sets, and though Reeb graph loses a lot of original topological structure we can say something about the function or the topological space on which the function is defined.

The function $\psi : X \rightarrow R(f)$ is a continuous surjection, which maps components to components. Furthermore, a loop in $R(f)$ cannot be contracted to a point and two loops in X that map to different loops in $R(f)$ are not homologous. It follows that the number of components is preserved and the number of loops cannot increase; therefore from homology relations:

$$\beta_0 (R(f(X))) = \beta_0 (\mathrm{X}) \quad \text{and}$$

$$\beta_1 (R(f(X))) \leq \beta_1 (\mathrm{X})$$

The inequality ≤ is due to the fact that old homology classes may die in Reeb graph: a non-trivial cycle may get killed in Reeb graph

Contour tree: if X is connected and simple then the Reeb graph is a tree, independent of the function f , that is, $R(f(X)$ is a tree called contour tree.

But, the inverse is not true. If X is an open cylinder and f is the height function, then the Reeb graph is a line.

Reeb graphs for Morse functions defined on a manifold:

Reeb graphs can reveal more information about the structure of X if X = M is a manifold of dimension $d \geq 2$ and $f: M \rightarrow R$ is a Morse function, shown in Figure 1 and Figure 2.

Each point $u \in R(f)$ is the image of a contour in M. We call u a node of the Reeb graph if $\psi^{-1}(u)$ contains a critical point or, equivalently if u is the image of a critical points under ψ. From definition of Morse function, the critical points have distinct functional values, that implies a bijection between the critical points of f and the nodes of $R(f)$. Rest of the Reeb graph is partitioned into arcs or edges connecting the nodes. A minima starts a contour and therefore corresponds to degree 1 node. An index 1 saddle that merges to contours into one corresponds to a degree 3 node. Similarly, a maxima corresponds to a degree 1 node and an index $d-1$ saddle splits a contour into two corresponds to a degree 3 node. All other critical points correspond to nodes of degree 2. [figure 2]

Reeb graph is a one-dimensional toplogical space with points on arcs being individually meaningful objects. There is no preferred way to draw the graph in the plane or space.

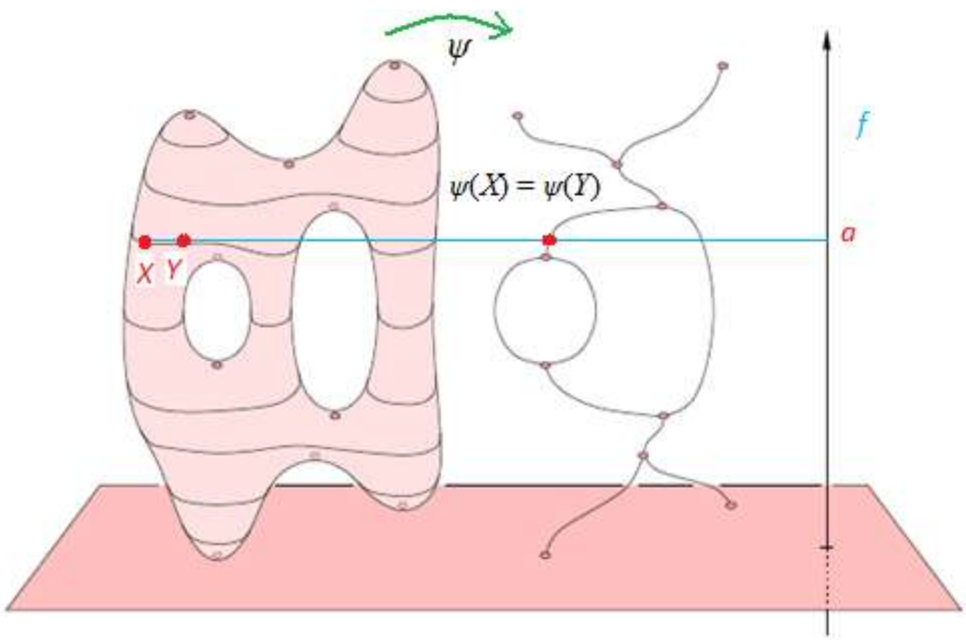


Figure 1: Level sets of the 2-manifold map to points on the real line and components of the level sets map to points of the Reeb graph. ψ is the surjection and X and Y are equivalent points because they belong to the same contour for given constant a . Consequently, they are mapped to a single point on Reeb graph where $\psi(X) = \psi(Y)$. [figure modification is based on the figure by *Edelsbrunner 2006*]

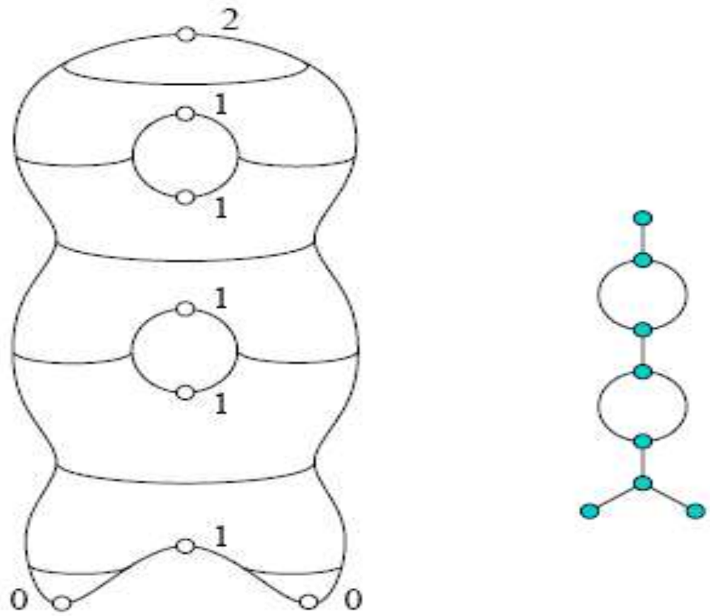


Figure 2: left: double torus with critical points along with the index numbers; right: corresponding Reeb graph [figure taken from: Cole-McLaughlin K *et al*, “Loops in Reeb Graphs of 2-Manifolds”, 2004]

<u>Orientable 2-manifolds and Reeb graph</u>:

If $d = 2$ and M is orientable then every saddle either merges two contours into one or it splits a contour into two. Either way it corresponds to a degree 3 node in the Reeb graph. We use this information to compute the number of loops in the Reeb graph.

Let n_i be the number of nodes with degree i. For orientable 2-manifolds only n_1 and n_3 are non-zero. The number of arcs is $e = (n_1 + 3n_3)/2$

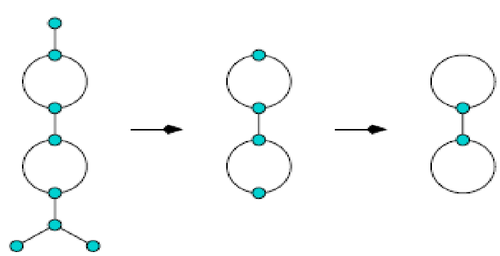


Figure 3: from left: Original Reeb graph of the double torus in Figure 2, the graph after collapsing, and the graph after merging arcs [figure taken from: Cole-McLaughlin K *et al*, “Loops in Reeb Graphs of 2-Manifolds”, 2004]

LEMMA: The Reeb graph of a Morse function on a connected, orientable 2-manifold of genus g has g loops.

Proof: When the original Reeb graph has no loop it is a tree with $n_1 + n_3 - 1$ edges because there are total $n_1 + n_3$ nodes in the graph. Let c_i denote the number of critical points of index i we have $n_1 = c_0 + c_2$ and $n_3 = c_1$. Therefore, total number of loops in the Reeb graph is:

Total number of edges – (total number of nodes – 1)

$$= (n_1 + 3n_3)/2 - (n_1 + n_3 - 1) = (n_3 - n_1)/2 + 1 \qquad \text{(Eq 1)}$$

Now using $n_1 = c_0 + c_2$ and $n_3 = c_1$, we have Eq 1

$$(n_3 - n_1)/2 + 1 = (c_1 - c_0 - c_2)/2 + 1 \qquad \text{(Eq 2)}$$

Now Euler's characteristics of M equals $c_0 - c_1 + c_2 = n_1 - n_3 = 2 - 2g$

Combining with Eq 2 we have the number of loops in Reeb graph as: $-(2 - 2g)/2 + 1 = g$

□

LEMMA: The Reeb graph of a Morse function on a connected, non-orientable 2-manifold of genus g has at most $g/2$ loops.

APPLICATIONS:

1. Handle Removal: [Wood *et al* , "Removing Excess Topology From Isosurfaces", 2004]

   Reeb graph can be used to detect and remove handles from a geometric shape in order to perform topological simplification. Basic steps followed by Woods *et al* are:

   - sweep through the volume to *encode* the topology in an augmented Reeb graph
   - *isolate* handles using the augmented Reeb graph
   - for each handle found, *measure* its size
   - if the size is sufficiently small, *remove* the handle

Figure 4: Two ways to remove a handle: left: collapsing the handle, right: pinching the Handle [Wood *et al* , "Removing Excess Topology From Isosurfaces", 2004]

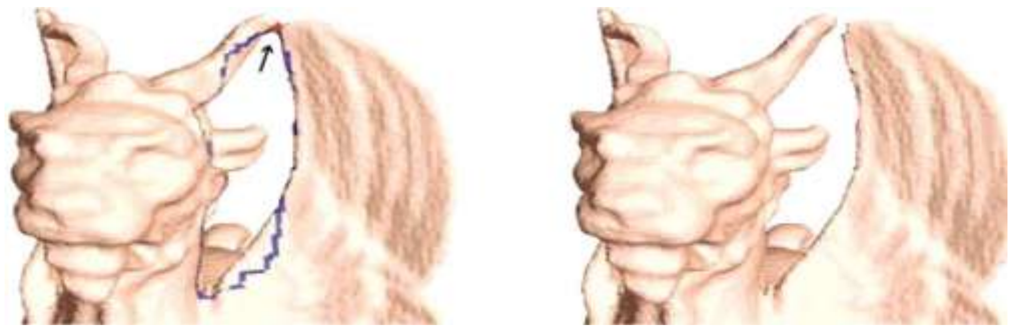


Figure 5: Feline mesh: left: Reeb loop shown in blue, cross loop shown in red, right: after collapsing the cross loop of the handle
[Wood *et al* , "Removing Excess Topology From Isosurfaces", 2004]

2. <u>Skeletonize a shape</u>:

[Biasotti *et* al, "Reeb graphs for shape analysis and applications", 2008]

Extended Reeb Graph (ERG) algorithm:

- This algorithm decompose manifold M into a set of regions, each region is defined a regular or a critical area according to the number and the value of *f* along its boundary components.

- Critical areas are furthered classified as maximum, minimum and saddle areas and correspond to nodes of the graph.

- The arcs between nodes are detected through an expansion process of the critical areas, which tracks the evolution of the isocontours.

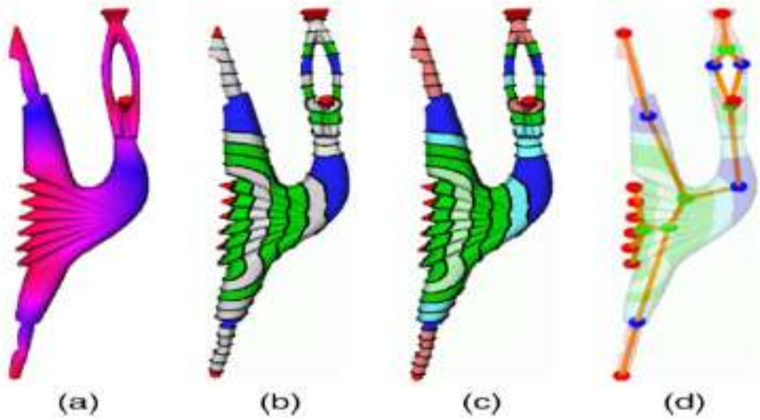


Figure 6: (a) evolution of the values of a function f on a surface model, blue values representing minima and read values representing maxima; (b) recognition of critical areas: minima (blue), maxima (red), and saddles (green); (c) the expansion process; (d) the resulting ERG configuration
(Figure taken from: Biasotti *et* al, “Reeb graphs for shape analysis and applications”, 2008)

3. <u>Shape matching:</u>

(Hilaga *et* al, “Topology matching for Fully Automatic Similarity Estimation of 3D Shapes”, 2001)

In this paper authors proposed a normalized integral of geodesic distance (μ_n) as the continuous function for topology matching. This function is particularly useful because it is resistant to the type of deformation shown in the following figure (Figure 7). This is due to the fact that the deformation does not drastically change geodesic distance on the surface.

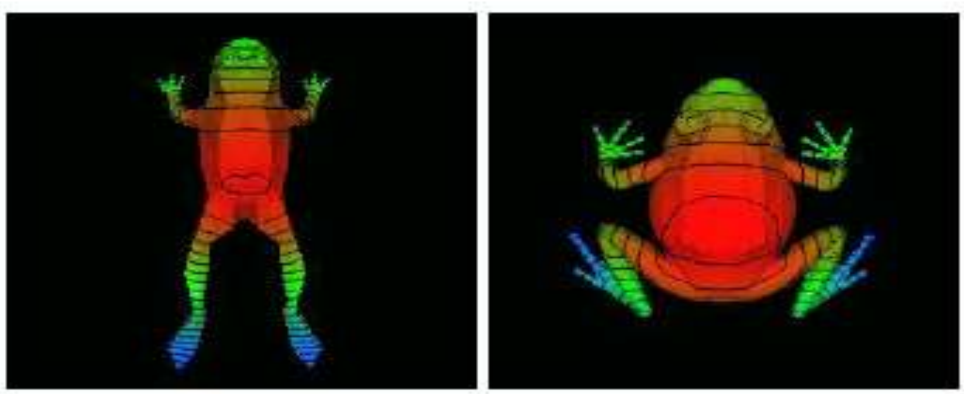


Figure 7: example of a deformed shape showing distribution of the function μ_n

Figure taken from: Hilaga *et* al, “Topology matching for Fully Automatic Similarity Estimation of 3D Shapes”, 2001

Simplification of Contour Tree and Reeb Graph:

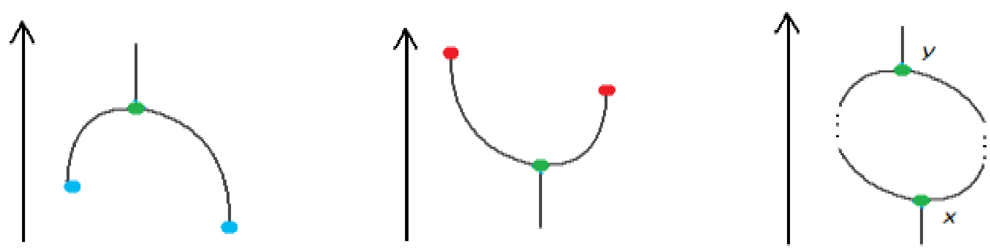
Let M be a connected, orientable 2-manifold and f : M → R be a Morse function. A point x ϵ M is *critical* if the derivative of f at x is zero, and it is *non-degenerate* if the Hessian at the point is invertible. There are three types of critical points for Morse functions on M: *minima, saddle*, and *maxima*. They are distinguished by the number of negative eigenvalues of the Hessian.

When we sweep M in the direction of increasing function value, proceeding along a level set of closed curves, the swept portion of M is called the sublevel set, and is denoted as:

$$M_a = \{x \in M \mid f(x) \le a\}.$$

The sublevel set changes the topology whenever the level set passes through a critical point.

A *component* of M_a starts at a minimum and ends when it merges with another, older component at a saddle. A *hole* in the 2-manifold starts at a saddle and ends when it is closed off at a maxima. Each saddle either merges two components or starts a new hole, but not both. The traditional way to simplify the topology is to pair up the critical point that starts a component or a hole with the critical point that ends it. The method pairs all critical points except for the first minima, last maxima, and the 2g saddles (g *upfork* and g *downfork* saddles) [Figure 8] when the sweep is complete. Here g is the genus of M. Total of 2 + 2g unpaired critical points remaining in this approach of pairing.



(a) (b) (c)

Figure 8: saddle points are shown in green, minima in blue, and maxima in red (a) *upfork* saddle, (b) *downfork* saddle, (c) *loop saddle*. (a) and (b) represent *tree saddles* that appear in contour trees because contour tree is cycle-free. In (c) x is the lo-point and y is the hi-point for the cycle shown

Extended Persistence approach: [Agarwal *et al* , "Extreme Elevation on 2-Manifold", 2004]

In this approach we pair the remaining minima with remaining maxima, and the remaining g upfork saddles with g downfork saddles in a way that reflects how they introduce a cycle during sweep. Each cycle has a unique lowest (*lo-point*) and a unique highest point (*hi-point*). There is a one-to-one correspondence between lo-points and upfork saddles from the remaining saddle set, and thereby we have exactly g lo-points and g hi-points. We pair lo-point x with the lowest hi-point y that spans a cycle with x, and this x is the highest lo-point that spans a cycle with y [Figure 8]. This implies that each lo-point and each hi-point belongs to exactly one pair, giving a total of g pairs between upfork and downfork saddles [Figure 9].

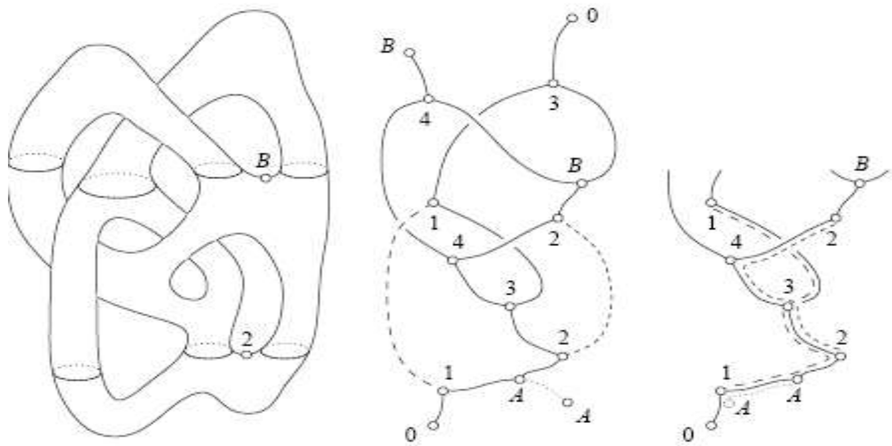


Figure 9: The Reeb graph in the middle is obtained from the 2-manifold on the left. The labels indicate the pairing. On the right the tree representing the Reeb graph
[figure taken from Agarwal *et al* , “Extreme Elevation on 2-Manifold”, 2004]

REFERENCES:

[1] Online course notes by Herbert Edelsbruner on computational topology
http://www.cs.duke.edu/courses/fall06/cps296.1/Lectures/sec-V-4.pdf

[2] Cole-McLaughlin K *et al,* “Loops in Reeb Graphs of 2-Manifolds”, 2004

[3] Wood *et al* , “Removing Excess Topology From Isosurfaces”, 2004

[4] Biasotti *et* al, “Reeb graphs for shape analysis and applications”, 2008

[5] Hilaga *et* al, “Topology matching for Fully Automatic Similarity Estimation of 3D Shapes”, 2001

[6] Agarwal *et al* , “Extreme Elevation on 2-Manifold”, 2004