## V.4 Reeb Graphs

The structure of a continuous function can sometimes be made explicit by plotting the evolution of the components of the level set. This leads to the concept of the Reeb graph of the function. It has applications in medical imaging and other areas of science and engineering.

**Iso-surface extraction.** The practical motivation for studying Reeb graphs is the extraction of iso-surfaces for three-dimensional density data. In topological lingo, the density data is a continuous function, $f : [0,1]^3 \to \mathbb{R}$, and an iso-surface is a level set, $f^{-1}(t)$. If $f$ is smooth and $t$ is a regular value then the level set is a 2-manifold, possibly with boundary. Similarly, if $f$ is generic PL and $t$ is not the value of a PL critical point then the level set is a 2-manifold, again possibly with boundary. Figure V.12 illustrates this fact for a generic PL function on the unit square. Assuming we enter a triangle at a boundary

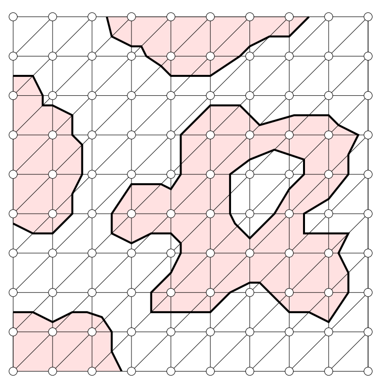

Figure V.12: The level set of a generic PL function on a triangulation of the unit square. The superlevel set is white and the sublevel set is shaded.

point $x$ with $f(x) = t$, there is a unique other boundary point $y$ with $f(y) = t$ where we exit the triangle. We draw the line segment from $x$ to $y$ as part of the level set and repeat the construction by entering the next triangle at $y$. The procedure is similar for a PL function on the unit cube except that we use a graph search algorithm to collect the triangular and quadrangular surface pieces we get for the tetrahedra.

Given a first point on the level set, it easy to trace out the component that contains it. But to be sure we did not not miss any of the other components it seems we need to check every edge of the triangulation. The desire to avoid this costly computation leads to the introduction of the contour tree, which is a data structure that can be queried for initial points on components of the level set without checking the entire triangulation. It is based on the concept of a Reeb graph, which we discuss next.

**Quotient topology and space.** We begin by introducing a basic construction of topological spaces. Suppose we have an equivalence relation, $\sim$, defined on a topological space $\mathbb{X}$. Let $\mathbb{X}_\sim$ be the set of equivalence classes and let $\psi : \mathbb{X} \to \mathbb{X}_\sim$ map each point $x$ to its equivalence class.

DEFINITION. The *quotient topology* of $\mathbb{X}_\sim$ consists of all subsets $U \subseteq \mathbb{X}_\sim$ whose preimages, $\psi^{-1}(U)$, are open in $\mathbb{X}$. The set $\mathbb{X}_\sim$ together with the quotient topology is the *quotient space* defined by $\sim$.

We have seen examples of this construction before, one being the torus obtained by gluing opposite sides of a square. In this case, the equivalence classes are individual points in the interior, pairs of points on the edges, and the quadruple of points at the corners of the square. We construct another example.

DEFINITION. Let $f : \mathbb{X} \to \mathbb{R}$ be continuous and call a component of a level set a *contour*. Two points $x, y \in \mathbb{X}$ 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 $R(f) = \mathbb{X}_\sim$, is the quotient space defined by this equivalence relation.

By construction, the Reeb graph has a point for each contour and the connection is provided by the map $\psi : \mathbb{X} \to R(f)$. Letting $\pi : R(f) \to \mathbb{R}$ be defined such that $f(x) = \pi(\psi(x))$ we can construct the level set by going backward, from the real line to the Reeb graph to the topological space. Given $t \in \mathbb{R}$ we 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 $t$.

Besides using the Reeb graph as a data structure to accelerate the extraction of level sets, we may hope to learn something about the function or the topological space on which the function is defined. Even though the Reeb graph loses a lot of the original topological structure there are some things that can be said. We have a continuous surjection, $\psi : \mathbb{X} \to R(f)$, which maps components to components. Furthermore, a loop in $\mathbb{X}$ that maps to a loop in $R(f)$ is not contractible and two loops in $\mathbb{X}$ 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,

$$\begin{aligned}\beta_0(R(f)) &= \beta_0(\mathbb{X});\\ \beta_1(R(f)) &\leq \beta_1(\mathbb{X}).\end{aligned}$$

Hence, if $\mathbb{X}$ is connected and simply connected then the Reeb graph is a tree, independent of the function $f$.

**Reeb graphs of Morse functions.** More can be said if $\mathbb{X} = \mathbb{M}$ is a manifold of dimension $d \geq 2$ and $f : \mathbb{M} \to \mathbb{R}$ is a Morse function, like in Figure V.13. Recall that each point $u \in R(f)$ is the image of a contour in $\mathbb{M}$. We call $u$ a

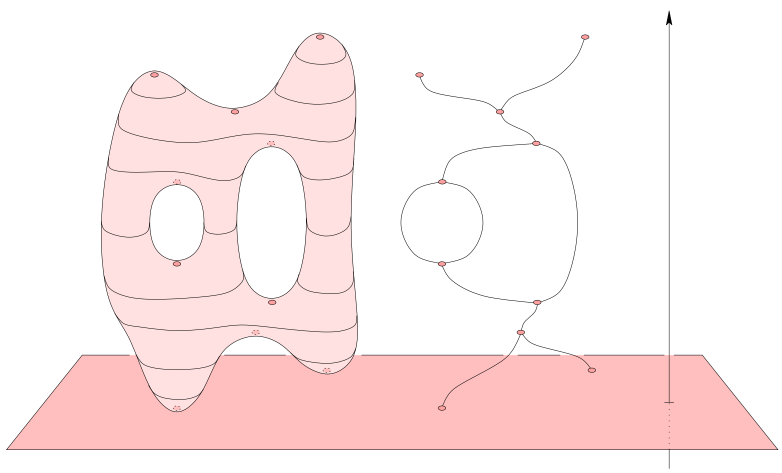

Figure V.13: Level sets of the 2-manifold map to points on the real line and components of the level sets map to points of the Reeb graph.

*node* of the Reeb graph if $\psi^{-1}(u)$ contains a critical point or, equivalently, if $u$ is the image of a critical point under $\psi$. By definition of Morse function, the critical points have distinct function values, which implies a bijection between the critical points of $f$ and the nodes of $R(f)$. The rest of the Reeb graph is partitioned into *arcs* connecting the nodes. A minimum starts a contour and therefore corresponds to a degree 1 node. An index 1 saddle that merges to contours into one corresponds to a degree 3 node. Symmetrically, a maximum corresponds to a degree 1 node and an index $d-1$ saddle that splits a contour into two corresponds to a degree 3 node. All other critical points correspond to nodes of degree 2.

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

**Orientable 2-manifolds.** If $d = 2$ and $\mathbb{M}$ 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 to pin down 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 $e = (n_1 + 3n_3)/2$ and the number of loops is $1 + e - (n_1 + n_3)$.

Loop Lemma A. The Reeb graph of a Morse function on a connected, orientable 2-manifold of genus $g$ has $g$ loops.

Proof. Suppose first that the original Reeb graph has no loop. It is a tree with $n_1 = n_3 + 2$ degree 1 nodes. Writing $c_i$ for the number of critical points of index $i$ we have $n_1 = c_0 + c_2$ and $n_3 = c_1$. The last strong Morse inequality implies $\chi = c_0 - c_1 + c_2 = n_1 - n_3 = 2$, which is the Euler characteristic of the sphere.

Suppose second that the original Reeb graph has at least one loop. We collapse degree 1 nodes and merge arcs across degree 2 nodes which get eliminated in the process. For example, the Reeb graph in Figure V.13 simplifies this way to two degree 3 nodes connected to each other by three arcs. Both operations preserve the homotopy type and therefore the number of loops. Let $m_3$ be the number of remaining degree 3 nodes and note that it is even because $3m_3$ is twice the number of remaining arcs. Using the Euler-Poincaré Theorem for graphs we get $\chi = m_3 - 3m_3/2 = \beta_0 - \beta_1$. The graph is connected which implies that the number of loops is $\beta_1 = m_3/2 + 1$. We have $c_1$ degree 3 nodes in the original Reeb graph and for each minimum and maximum we collapse one degree 1 node removing a degree 3 node in the process. Using the last strong Morse inequality we get $m_3 = c_1 - (c_0 + c_2) = -\chi = 2g - 2$. The number of loops is therefore $\beta_1 = (2g - 2)/2 + 1 = g$, as claimed. ⊔⊓

**Non-orientable 2-manifolds.** The situation for non-orientable 2-manifolds is more complicated as the number of loops in the Reeb graph is not longer independent of the function. To determine tight upper and lower bounds, we make use of the doubling operation that turns a non-orientable 2-manifold, $\mathbb{N}$, into an orientable 2-manifold, $\mathbb{M}$. In the process every vertex, edge, and triangle of a triangulation of $\mathbb{N}$ gets duplicated implying $\chi(\mathbb{M}) = 2\chi(\mathbb{N})$. It follows that the doubling process decreases the genus by 1. Table V.2 lists the first few cases of this correspondence along with the genus and the Euler characteristic of each manifold. Let now $f : \mathbb{N} \to \mathbb{R}$ be a Morse function. In contrast to

| $\chi(\mathbb{N})$ | $g(\mathbb{N})$ | $\mathbb{N}$ | $\mathbb{M}$ | $g(\mathbb{M})$ | $\chi(\mathbb{M})$ |
|---|---|---|---|---|---|
| 1 | 1 | $\mathbb{P}^2$ | $\mathbb{S}^2$ | 0 | 2 |
| 0 | 2 | $\mathbb{P}^2\#\mathbb{P}^2$ | $\mathbb{T}^2$ | 1 | 0 |
| $-1$ | 3 | $\mathbb{P}^2\#\mathbb{P}^2\#\mathbb{P}$ | $\mathbb{T}^2\#\mathbb{T}^2$ | 2 | $-2$ |
| ... | ... | ... | ... | ... | ... |

Table V.2: Doubling turns the non-orientable 2-manifold on the left into the orientable 2-manifold on the right.

the orientable case, $R(f)$ can also have nodes of degree 2. They correspond to saddles whose contours reverse orientation; we have to traverse them twice to return to the same point with the same orientation. The existence of degree 2 nodes complicates matters and we no longer have a predictable number of loops.

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

Proof. Let $f : \mathbb{N} \to \mathbb{R}$ be the Morse function on the non-orientable 2-manifold and $f_0 : \mathbb{M} \to \mathbb{R}$ the function obtained by doubling. Contours of $f$ that do not contain critical points lift to two contours of $f_0$. It follows that $R(f_0)$ has twice the number of arcs of $R(f)$. Similarly, a contour that maps to a degree 1 or 3 node in $R(f)$ lifts to two disjoint copies giving rise to two nodes in $R(f_0)$. Finally, a contour that maps to a degree 2 node in $R(f)$ lifts to a single contour, giving rise to a single node incident to four arcs in $R(f_0)$.

Letting $e$ be the number of arcs and $n_i$ the number of degree $i$ nodes in $R(f)$, we have $2e$ arcs and $2n_1 + n_2 + 2n_3$ nodes in $R(f_0)$. The number of loops in $R(f)$ is $\beta_1(R(f)) = 1 + e - n_1 - n_2 - n_3$. The number of loops in $R(f_0)$ is therefore

$$\begin{aligned}\beta_1(R(f_0)) &= 1 + 2e - 2n_1 - n_2 - 2n_3 \\ &= 2\beta_1(R(f)) - 1 + n_2.\end{aligned}$$

The critical points of $f_0$ are non-degenerate but they share function values in pairs. A small perturbation that does not affect the structure of the Reeb graph suffices to turn $f_0$ into a Morse function. By the Loop Lemma A, the number of loops in $R(f_0)$ is $g - 1$ and we get $\beta_1(R(f)) = (g - n_2)/2$. Since $n_2 \geq 0$ this implies the claimed upper bound. ⊔⊓

The number of saddles with orientation reversing contours can be anywhere between zero and $g$ which implies that the upper bound is tight and any smaller non-negative number of loops can be achieved.

**Bibliographic notes.** The most common method for extracting iso-surfaces from density data is the marching cube algorithm due to Lorensen and Cline [3]. As the name suggests, it works with a cube complex rather than a triangulation. The portion of the iso-surface within a single cube can be complicated and the implementation of the algorithm requires some care. The idea of speeding up the iso-surface extraction with a contour tree is more recent [5]. This tree is really the Reeb graph of a PL function on a cube, which has no loops. The definition of the Reeb graph itself is much older [4]. The analysis of the number of loops is taken from a relatively recent source [2]. This paper also gives an algorithm that constructs the Reeb graph of a PL function on a triangulated 2-manifold in time $O(n \log n)$, where $n$ is the number of edges in the triangulation. If there are no loops then the running time can be reduced to $O(n\alpha(n))$ [1].

[1] H. Carr, J. Snoeyink and U. Axen. Computing contour trees in all dimensions. *Comput. Geom. Theory Appl.* **24** (2002), 75–94.

[2] K. Cole-McLaughlin, H. Edelsbrunner, J. Harer, V. Natarajan and V. Pascucci. Loops in Reeb graphs of 2-manifolds. *Discrete Comput. Geom.* **32** (2004), 231–244.

[3] W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. *Comput. Graphics* **21**, Proc. siggraph, 1987, 163–169.

[4] G. Reeb. Sur les points singuliers d'une forme de Pfaff complèment intégrable ou d'une fonction numérique. *Comptes Rendus de L'Académie ses Séances, Paris* **222** (1946), 847–849.

[5] M. van Kreveld, R. van Oostrum, C. L. Bajaj, V. Pascucci and D. R. Schikore. Contour trees and small seed sets for isosurface traversal. *In* "Proc. 13th Ann. Sympos. Comput. Geom., 1997", 212–220.