Morse Theory on Meshes

submitted by Niloy J. Mitra*

9th December 2002

1 Introduction

In this report, we discuss two papers that deal with computing Morse function on triangulated manifolds.

Axen [1] gives an algorithm for computing Morse function on a triangulated manifold of arbitrary dimension but it not practical because of its space requirement. Hence, he describes an $O(n\alpha(n))$ algorithm for computing critical points and their Morse indices for a 2-manifold.

Edelsbrunner et al. [2] deals with compact 2-manifolds without any boundary. The paper describes how to derive a Morse complex for such a manifold and also talks about how to simply the complex by canceling pairs of critical points in order of increasing persistence.

2 Morse Functions on Triangulated Manifolds[1]

A smooth differentiable function defined over a smooth manifold is called a *Morse function* if all its critical points (points where the gradient vanishes) are isolated. Such a definition only applies to the continuous world where gradient is continuous over a smooth manifold. However, for a triangulated mesh, Morse theory does not apply directly. This paper talks about how to extend Morse theory so that it is applicable to triangulated meshes which are homeomorphic to manifolds.

The input is assumed to be K, a k-dimensional simplicial complex embedded in some $\mathbb{R}^n$. A height function f is defined over this complex as $f : \mathbb{R}^n \to \mathbb{R}$ which assigns a scalar value to all points on K. For any pair of vertices $u, v \in K$, it is assumed that $f(u) \neq f(v)$. For such a f, it has been shown that f is indeed a Morse function given that K is a complex homeomorphic to a manifold.

The steps involved in the algorithm for computing such a f is as follows:

- compute the k^{th} barycentric subdivision $sd^{(k)}K$ of K.
- for an $u \in K$, compute $d(u, v)$ as the minimum number of edges from u to v for all $v \in sd^{(k)}K$.

*niloy@stanford.edu

- for every simplex σ, let $d(u, \sigma) = \min d(u, v)$ where v is a any vertex in σ.
- *wave* $W_{sd^{(k)}K}(i)$ represents the sub-complex with *all vertices* at the same distance i from u.
- $S_{sd^{(k)}K}(i)$ denotes all the simplices with $d(u, \sigma) = i$, but $\sigma \notin W_{sd^{(k)}K}(i)$.
- let d_j be the distance function from vertex u_j on a component of wave $W_{sd^{(k)}K}(i)$ defined by $d_{j-1}(u_{j-1}, v)$, where $d_0 = d$.
- since there can be multiple components of a wave, compute a unique component number $c_{j-1}(i, v), 0 \leq c_{j-1}(i, v) \leq C-1$, for each component of wave i which is associated with each v in the component
- Finally for every $v \in K$, f is defined as:

$$f(v) = \epsilon^0 d_0(u_0, v) - \epsilon^1 c_0(d_0(u_0, v), v) \ldots - \epsilon^{2k-1} c_{k-1}(d_{k-1}(u_{k-1}, v), v) \quad (1)$$

for some small $\epsilon > 0$. Once the Morse function is defined over the mesh as above, critical points and their Morse indices can be found by local neighborhood analysis of each vertex as in the case of 2-manifolds. Such an algorithm using barycentric coordinates involves a growth in data size by a factor $((k+1)!)^k$ and hence is not a practical algorithm.

In a search for a practical algorithm, the author has designed an algorithm tailored only to 2-manifold. In this approach, instead of computing the general function f, only the critical points with their Morse indices are computed from W_k and S_k. This works because of the claim that "*all critical points of f are in the wave sub-complex W_k or in the grounded simplices of S_k*" where *grounded simplices* refer to the simplices of $S_K(i)$ which have *all* vertices in $W_K(i)$. This algorithm runs in $O(n\alpha(n))$, where $\alpha(n)$ denotes the inverse Ackermann function.

2.1 Comments

Unlike the Betti numbers, which are a global property of the manifold, critical points and their Morse indices are local properties and hence, can be evaluated locally. As noted earlier, the storage requirement for the algorithm for k-manifold is huge rendering the algorithm impractical. Details about why equation (1) is indeed a Morse function is not given in this paper. So this paper is more like an algorithm description rather than containing proofs as to why this is true. The subdivision process helps to make the distance to vertex into a Morse function.

3 Hierarchical Morse Complexes for PL 2-Manifolds[2]

This paper deals with Morse complexes for piecewise linear (PL) 2-manifold. The algorithm first computes a complex having the same combinatorial description as the Morse complex, then derives the Morse complex by applying local transformations.

Finally, the Morse complex is simplified via local transformations to generate a hierarchy of Morse complexes. The algorithm uses the *simulation of differentiability* (SoD) to ensure that computed complex of the PL manifold have same structural form as those in the smooth case.

Since the input to the algorithm is a mesh, the natural neighborhood for a point u is its *star*, which is defined as $St\,u = \{\sigma \in K \mid u \leq \sigma\}$ consisting of all the edges and triangles sharing u as a vertex. Since we can assume (and simulate using SoD) vertices having different heights, we have unique minimum and maximum vertices for each complex. Since height of each vertex is assumed to be distinct, lower and upper stars of u can be defined as,

$$\begin{aligned} \underline{St}\,u &= \{\sigma \in St\,u \mid h(v) \leq h(u),\, v \leq \sigma\} \\ \overline{St}\,u &= \{\sigma \in St\,u \mid h(v) \geq h(v),\, v \leq \sigma\} \end{aligned}$$

Based on $St\,u$, $\overline{St}\,u$, $\underline{St}\,u$ we can classify a vertex u as maximum, minimum, regular or saddle. In case of a k-fold saddle, a simple recursive algorithm which splits wedges of $\underline{St}\,u$, can be used to break the multiple saddle into k simple saddles. The important thing is that though the splitting process may be ambiguous but it is usually sufficient to pick any arbitrary splitting or unfolding order.

The concept of *integral lines* from the smooth 2-manifold does not extend directly to a PL function. So the paper proposes to use monotonic curves which never cross. When these curves merge or fork they are considered to maintain an infinitesimal separation without crossing each other. *Junctions* are used to represent these merging and forking.

The algorithm begins by constructing a quasi Morse complex. All the vertices are classified and the steepest edge in each wedge is determined. Each path starts in its own wedge and follows a sequence of steepest edges till it hits a minumum, maximum, a saddle or intersects a previously traced path at a regular point. Quad-edge structure has been used to store the complex defined by these paths.

In the second stage, paths are extended and re-routed infinitesimally close to already existing paths in an effort to reduce junctions. Vertices are processed in a sequence to prevent cyclic dependencies. First ascending paths are extended in order of increasing height and then descending paths are extended in order of decreasing height.. Paths are extended by duplication and concatenation without creating crossing. Finally in the third stage, multiple saddles are unfolded into simple saddles by splitting wedges.

In the second phase of the algorithm, a few local operations transform the quasi Morse complex to a Morse complex. The primitive operation is to apply a *handle slide* which flips between the quadrangulations of a octagon. The decision about applying a handle flip is based on rerouting the interior paths following the direction of steepest ascent. SoD approach is used for the rerouting. The algorithm applies handle slides in the order of decreasing height , where the *height* is defined of an octagon is the height of the lower saddle of the middle quadrangle.

The n vertices of K are sorted in order of increasing height to get the sequence $u^1, u^2, \ldots, u^n$, $h(u^i) < h(u^j)$ for all $1 \leq i < j \leq n$. Let K^j be the sub-complex of K consisting of the j lower stars, $K^j = \cup_{i \leq i \leq j} \underline{St}\,u^i$. These K^j define a filtration of K with corresponding Betti numbers as β_0^j, β_1^j, β_2^j. Using the fact that the Betti numbers can only change at a minimum, maximum or a saddle points (but not at regular points), the Betti numbers for K^{j+1} can be computed from those of K^j and using the information about the type of u^{j+1} along with how its lower star connects to K^j.

Persistence of critical point pairs is defined and are used for manifold simplification. A critical point is called *positive* if it creates and *negative* if it destroys. Every negative saddle is paired with its preceding positive minimum and every negative maximum with a preceding positive saddle. Such a process is possible since all multi-fold saddles have been transformed to simple saddles. Now *persistence* of a critical point a is defined to be the absolute height difference between a and its paired critical point. The pairing of critical points and their persistence computation is done using the persistence algorithm as proposed by the Edelsbrunner et al in [3]. The complexity of the algorithm is $O(n\alpha(n))$.

For simplification, the paper introduces a cancellation operation that removes critical points depending on persistence. In the one-dimensional case, the critical points can appear as alternating maximum and minimum. They are paired and canceled depending on increasing persistence. In case of meshes, critical points are paired as described before. They are also removed in a similar persistence order. The contraction of a critical point pair pulls the points to the minimum (or maximum) of the quadrangle, and the minimum (or maximum) also inherits the connects of the canceled pairs. The algorithm requires the critical point pairs to share an edge before they can be contracted. This is ensured by the *Adjacency Lemma* which states that for every positive i, the i-th critical point pair ordered according to persistence forms an arc in the complex formed after cancellation of the first $i - 1$ pairs.

3.1 Comments

The results apply only to meshes from 2-manifolds without any boundary. The fact that splitting a multiple-saddle in any order still leads to an unfolded manifold is important for the success of the algorithm. After the first phase, the rest of the algorithm assumes an unfolded manifold since any k-saddle has been decomposed to k simple saddles by that point. The adjacency lemma is very important for the success of the simplification algorithm since only adjacent control points can be cancelled. Some experimental results have also been provided to show the correct working of the algorithm.

References

[1] U. Axen, "Computing Morse Functions on Triangulated Manifolds". In Proc. SODA, 1999.

[2] H. Edelsbrunner, J. Harer, A. Zomorodian, "Hierarchical Morse Complexes for Piecewise Linear 2-Manifolds". In Proc. Symposium on Computational Geometry, 2001.

[3] H. Edelsbrunner, D. Letscher, A, Zomorodian, “Topological Persistence and Simplification”. In Proc. 41^{st} Ann. IEEE Sympos. Found COmput. Sci. (2000), pp 454-463.