Generating Discrete Morse Functions from Point Data

Henry King, Kevin Knudson, and Neža Mramor

CONTENTS

If K is a finite simplicial complex and h is an injective map from the vertices of K to $\mathbb{R}$, we show how to extend h to a discrete Morse function in the sense of Forman [Forman 02] in a reasonably efficient manner so that the resulting discrete Morse function mirrors the large-scale behavior of h. A concrete algorithm is given for the case where K is a subcomplex of $\mathbb{R}^3$.

1. INTRODUCTION

When modelling physical phenomena, one often begins with a collection of point data. These may be sample points of a terrain or a collection of temperature values in a region in 3-space. If one knows in advance that these points arise from a smooth function $f : D \to \mathbb{R}$ defined on some space D, then standard techniques allow one to analyze the gradient paths, decompose the space into a Morse-Smale complex, etc. However, the space D is often sampled, and one has only the values of f at the sample points. Making the sample more dense does not allow one to use smooth methods to analyze the qualitative behavior of f.

This problem was addressed in the case of PL 2-manifolds by Edelsbrunner et al. in [Edelsbrunner et al. 03b], where the authors construct a complex whose combinatorial form matches that of the Morse-Smale complex, and, therefore, allows one to understand the qualitative behavior of f. This approach leads to fast algorithms for visualization of, for example, geographic terrains. It uses piecewise linear functions on the complex and extends the notion of index of a critical point to this setting.

Extending the methods of [Edelsbrunner et al. 03b] to higher dimensions is a challenge, however. Instead, it is natural to ask if one could use the discrete Morse theory of R. Forman [Forman 98, Forman 02] to model the qualitative behavior of f. Giving the values of f on a sample in D produces a subcomplex of the 0-skeleton of the region. One would like to extend this to a discrete

Morse function on the entire space in such a way that the associated gradient vector field approximates that of f.

In this paper, we present a method for doing this that is motivated by the techniques developed in [Edelsbrunner et al. 03b]. Let K be a finite simplicial complex and let h be an injective map from the vertices of K to $\mathbb{R}$. We will extend h to a discrete Morse function on K (in the sense of [Forman 02]) in an efficient manner. That is, the algorithm has reasonable running time for K encountered in practice and produces relatively few critical simplices, so that, for example, it is efficient to compute the homology of $|K|$ using the critical simplices.

We also incorporate persistence, so that pairs of critical simplices are cancelled if they are connected by just one gradient path and their values differ by less than the persistence p (details about persistence may be found in [Edelsbrunner et al. 02]). We allow $p = \infty$, so it is possible to cancel as much as we can. The resulting discrete Morse function mirrors the behavior of h on a scale greater than the persistence p. For example, critical vertices are at local minima of h, although not every local minimum of h gives rise to a critical vertex; indeed, if it is connected to an index-1 saddle whose h value is within p, then it probably will not. Likewise if, say K is a surface, then critical 2-simplices are adjacent to local maxima of h, although if a local maximum is connected to a saddle whose h value is within p, it might not give rise to a critical 2-simplex.

Other algorithms for constructing optimal Morse functions exist, for example, in [Lewiner et al. 03] and [Hersh 05]. The approach in [Lewiner et al. 03] differs from ours. While we start with a given function on the vertices of the simplicial complex and construct a discrete Morse function, which, up to a certain error given by the persistence, models the given data, the algorithm in [Lewiner et al. 03] starts with only a simplicial complex and constructs a discrete Morse function that is as close as possible to an optimal one. Thus, it can be effectively used for computing homology. The algorithm generally produces fewer critical simplices than our algorithm with persistence set to ∞. It cannot be used for analyzing specific Morse functions, though. In [Hersh 05], the author begins with a given discrete Morse function on a complex and addresses the question of when several gradient paths may be cancelled simultaneously to further reduce the complex. The primary applications are to poset order complexes.

In Section 2, we review the rudiments of discrete Morse theory. Section 3 outlines our algorithm for defining the discrete Morse function extending $h : K_0 \to \mathbb{R}$. We prove that the result is indeed a discrete Morse function on K. In Section 5, we incorporate persistence to minimize the number of critical simplices. An implementation of our algorithm is discussed in Section 6.

1.1   Notation

We let K_i denote the set of i-simplices of K. If an i-simplex σ has vertices $v_0, v_1, \ldots, v_i$, then we may write $\sigma = [v_0 \; v_1 \; \ldots \; v_i]$. If τ is a face of σ, we write $\tau < \sigma$. If $\sigma = [v_0 \; v_1 \; \ldots \; v_i]$, then we define $\max h(\sigma) = \max_{0 \leq j \leq i}\{h(v_j)\}$. If $\sigma \in K_i$ and $\tau \in K_j$ are disjoint simplices, then $\sigma * \tau$ is either undefined or the $(i+j+1)$-simplex whose vertices are the union of the vertices of σ and τ. For example, if v and w are vertices, then $v * w$ is undefined if $[v \; w]$ is not an edge in K_1 and otherwise $v * w = [v \; w]$. If $v \in K_0$ is a vertex, then the link of v is the simplicial complex L whose simplices are all τ so that $v * \tau$ is defined. Following [Edelsbrunner et al. 03a], we define the lower link of v to be the maximal subcomplex of the link of v so that all its vertices have h value less than that of v. In other words, the simplices of the lower link of v are all simplices τ of K, so that $v * \tau$ is defined and $\max h(\tau) < h(v)$.

2.   DISCRETE MORSE THEORY

In this section, we review the basics of Forman's discrete Morse theory [Forman 02]. Let M be a simplicial complex. A typical p-simplex will be denoted by $\alpha^{(p)}$.

Definition 2.1. A function $f : M \longrightarrow \mathbb{R}$ is a *discrete Morse function*, if for every $\alpha^{(p)} \in M$, the following two conditions hold:

1. $\#\{\beta^{(p+1)} > \alpha | f(\beta) \leq f(\alpha)\} \leq 1$,

2. $\#\{\gamma^{(p-1)} < \alpha | f(\gamma) \geq f(\alpha)\} \leq 1$.

Essentially, discrete Morse functions are functions on M that increase with the dimension of the simplices; that is, the values on all but at most one face of α must be smaller than the value on α itself. Discrete Morse functions exist. Indeed, the simplest (and most trivial) example is the following. If σ is a simplex of M, we define $f : M \longrightarrow \mathbb{R}$ by

$$f(\sigma) = \dim \sigma.$$

Note that Conditions (1) and (2) in the definition of a discrete Morse function are exclusive. That is, if one of the sets has cardinality one, the other is empty. To see this, assume that $\sigma^{(p)} \in M$ has a coface $\tau^{(p+1)}$ such that

$f(\tau) \leq f(\sigma)$ and a face $v^{(p-1)}$ such that $f(v) \geq f(\sigma)$. Let $\sigma'^{(p)}$ be a different face of τ that also contains v. Then the value $f(\sigma')$ must be bigger than $f(v)$, since v already has a coface with smaller value, and similarly the value of τ must be bigger than $f(\sigma')$, so

$$f(\tau) \leq f(\sigma) \leq f(v) < f(\sigma') < f(\tau),$$

which is not possible.

It follows that, unless $\dim K = 1$, the function maxh is definitely not a discrete Morse function. However, Theorem 3.4 says it can be approximated by a discrete Morse function.

Smooth Morse theory depends heavily on the notion of critical point; the discrete theory does as well.

Definition 2.2. Let $f : M \longrightarrow \mathbb{R}$ be a discrete Morse function. A simplex $\alpha^{(p)}$ is *critical* if the following two conditions hold:

1. $\#\{\beta^{(p+1)} > \alpha | f(\beta) \leq f(\alpha)\} = 0$,

2. $\#\{\gamma^{(p-1)} < \alpha | f(\gamma) \geq f(\alpha)\} = 0$.

A simplex that is not critical is called *regular*.

For example, if $f : M \longrightarrow \mathbb{R}$ is given by $f(\sigma) = \dim \sigma$, then every cell is critical. An example of a discrete Morse function on the torus is given in Figure 1.

The critical simplices are the vertex $f^{-1}(0)$, the two edges $f^{-1}(42)$, $f^{-1}(44)$, and the 2-simplex $f^{-1}(86)$. This

FIGURE 1. A discrete Morse function on the torus.

is in keeping with the main theorem of discrete Morse theory.

Theorem 2.3. [Forman 02, Theorem 2.5] *Suppose M is a simplicial complex with a discrete Morse function. Then, M is homotopy equivalent to a CW-complex with exactly one cell of dimension p for each critical simplex of dimension p.*

As the above example shows, constructing discrete Morse functions is rather involved, especially if the complex has many simplices. Recall that in smooth Morse theory, one often works with the gradient vector field of a Morse function. This encapsulates the qualitative behavior of the function so that one need not even know the values of the function itself. This is possible in the discrete setting, and it forms the basis for our algorithm.

The fundamental observation is this: regular simplices occur in pairs. In the example shown in Figure 1, the vertex $f^{-1}(50)$ is regular since it has an adjacent edge $(f^{-1}(25))$ with a lower value. Similarly, the edge $f^{-1}(54)$ is regular since it is an edge of a triangle with a lower value. The triangle $f^{-1}(27)$ is regular, as it has an edge with a higher value. We indicate these pairings by drawing an arrow from the vertex $f^{-1}(50)$ to the edge $f^{-1}(25)$ and an arrow from $f^{-1}(54)$ to $f^{-1}(27)$. It is useful to think of these arrows as a representation of the simplicial collapses needed to prove Theorem 2.3.

For an arbitrary simplicial complex with a discrete Morse function f, we draw the arrows as follows. If $\alpha^{(p)}$ is a regular simplex with $\beta^{(p+1)} > \alpha$ satisfying $f(\beta) \leq f(\alpha)$, then we draw an arrow from α to β. Figure 2 shows the arrows for the function in Figure 1.

It is easy to see that every simplex α satisfies exactly one of the following:

1. α is the tail of exactly one arrow;

2. α is the head of exactly one arrow;

3. α is neither the head nor the tail of an arrow.

A simplex is critical if and only if it satisfies Condition (3) above. These arrows can be thought of as the gradient vector field of the Morse function. A better point of view is the following. A discrete vector field V can be thought of as a collection of pairs $\{\alpha^{(p)}, \beta^{(p+1)}\}$ of simplices, where a pair α, β is in V if and only if $\alpha < \beta$ and $f(\beta) \leq f(\alpha)$.
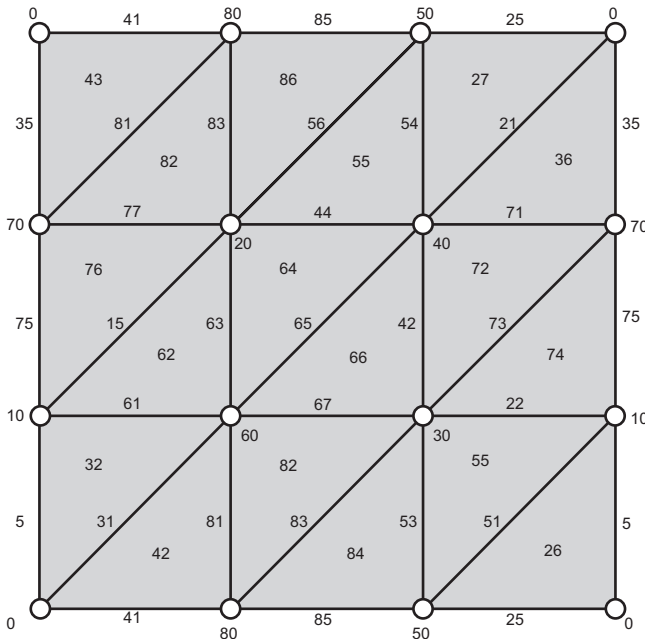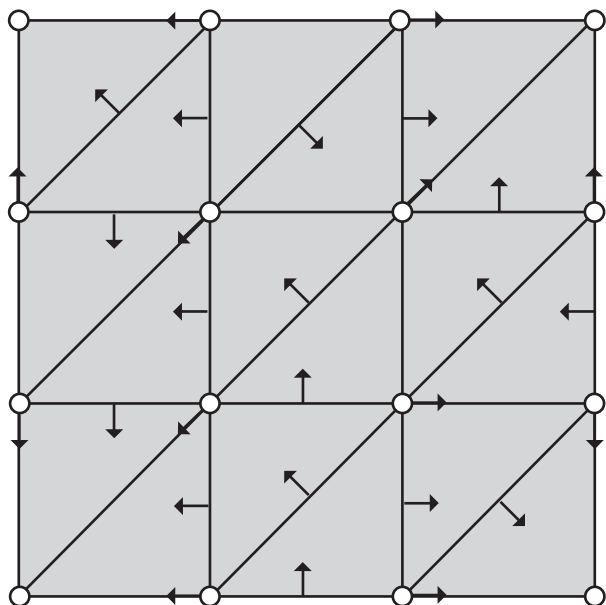
FIGURE 2. A gradient vector field on the torus.

Definition 2.4. A *discrete vector field* V on M is a collection of pairs $\{\alpha^{(p)}, \beta^{(p+1)}\}$ of simplices of M with $\alpha < \beta$ such that each simplex is in at most one pair of V.

Of course, one would like to know if a discrete vector field V is the gradient field of some discrete Morse function on M. The following definition is crucial.

Definition 2.5. Let V be a discrete vector field on M. A *V-path* is a sequence of simplices

$$\alpha_0^{(p)}, \beta_0^{(p+1)}, \alpha_1^{(p)}, \beta_1^{(p+1)}, \ldots, \beta_r^{(p+1)}, \alpha_{r+1}^{(p)}$$

such that for each $i = 0, \ldots, r$, $\{\alpha_r, \beta_r\} \in V$ and $\beta_i > \alpha_{i+1} \neq \alpha_i$. Such a path is a *nontrivial closed path* if $r \geq 0$ and $\alpha_0 = \alpha_{r+1}$.

Forman proved the following two theorems.

Theorem 2.6. [Forman 02, Theorem 3.4] *Suppose V is the gradient vector field of a discrete Morse function f. Then, a sequence of simplices is a V-path if and only if $\alpha_i < \beta_i > \alpha_{i+1}$ for $i = 0, \ldots, r$, and*

$$f(\alpha_0) \geq f(\beta_0) > f(\alpha_1) \geq f(\beta_1) > \cdots$$
$$\geq f(\beta_r) > f(\alpha_{r+1}).$$

In particular, if V is a gradient vector field, then there are no nontrivial closed V-paths. The converse is true as well.

Theorem 2.7. [Forman 02, Theorem 3.5] *A discrete vector field V is the gradient vector field of a discrete Morse function if and only if there are no nontrivial closed V-paths.*

A convenient combinatorial description of vector fields may be given in terms of the Hasse diagram of M. This is the directed graph whose vertices are the simplices of M and whose edges are given by the face relations in M (i.e., there is an edge from β to α if and only if α is a codimension-one face of β). Given a vector field V, modify the Hasse diagram in the following manner. If $\{\alpha < \beta\} \in V$, then reverse the orientation of the edge between α and β. A V-path is then a directed path in the modified graph.

Theorem 2.8. [Forman 02, Theorem 6.2] *There are no nontrivial closed V-paths if and only if there are no nontrivial closed directed paths in the modified Hasse diagram.*

We shall use the following notation for a gradient path $\sigma_1, \sigma_2, \ldots, \sigma_k$:

$$\sigma_1 \to \sigma_2 \to \cdots \to \sigma_k.$$

In Figure 2, using the value of the function on a simplex to denote the simplex (the function is injective), we have the following paths (for example):

$$85 \to 84 \to 83 \to 82 \to 67 \to 66 \to 65$$
$$\to 64 \to 63 \to 62 \to 61 \to 32 \to 5$$

and

$$20 \to 15 \to 10 \to 5 \to 0.$$

3.   THE ALGORITHM

As far as computation is concerned, finding an extension of h to a discrete Morse function is equivalent to finding the modified Hasse diagram [Forman 02, Theorem 3.5], so this is what we will do instead.

We will describe a procedure `Extract`(K, h, p) that takes as input a finite simplicial complex K, an injective $h \colon K_0 \to \mathbb{R}$, and a persistence $p \geq 0$. Its output is:

1. three lists A, B, C of simplices of K so that each simplex of K is on exactly one list;

2. a one-to-one onto map $r \colon B \to A$ so that $r(\sigma)$ is a codimension-one face of σ.

This output is meant to encode the relevant data from a discrete Morse function, in particular, its modified Hasse diagram. The simplices in C will be critical, and r gives the usual pairing of the noncritical simplices. Our discrete Morse function will have a larger value on $r(\sigma)$ than on σ (but necessarily smaller values on all other codimension-one faces of σ).

We find it convenient to consider subgraphs R_i of the modified Hasse diagram for $i = 1, \ldots, \dim K$. There is a one-to-one correspondence between directed paths in R_i connecting critical i- and $(i-1)$-simplices and Forman's gradient paths between those simplices. Let K_i denote the set of i-simplices of K. Let $A_i = A \cap K_i$, $B_i = B \cap K_i$, and $C_i = C \cap K_i$. The vertices of R_i are of two types: the $(i-1)$-simplices in $A_{i-1} \cup C_{i-1}$ and the i-simplices in $C_i \cup B_i$. Put an edge between any i-simplex and each of its faces in $A_{i-1} \cup C_{i-1}$. These edges are directed from the simplex to its face with one exception. If $\sigma \in B_i$, then the edge between $r(\sigma)$ and σ is directed from $r(\sigma)$ to σ.

The initial vertices in R_i are of two types: i-simplices in C_i and $(i-1)$-simplices $\sigma \in A_{i-1}$ whose only coface in $C_i \cup B_i$ is $r^{-1}(\sigma)$. A prime example of the second type would be a noncritcal edge on the boundary of a surface.

There are, likewise, two types of terminal vertices in R_i. There are $(i-1)$-simplices in C_{i-1}. There are i-simplices τ all of whose $(i-1)$-faces are in B_{i-1} except one face $r(\tau)$.

It is convenient to divide the algorithm `Extract` into two parts. `ExtractRaw`(K, h) produces a preliminary A, B, C, and r without doing any cancellation. Then the routine `ExtractCancel`(K, h, p, j) does all the persistence cancellation of critical j- and $(j-1)$-simplices by modifying A, B, C, and r. So the main outline of the algorithm is:

`Extract`(K, h, p)

- `ExtractRaw`(K, h)

- for $j = 1$ to $\dim K$

    - `ExtractCancel`(K, h, p, j)

- end for

`ExtractRaw`(K, h)

- Initialize A, B, C to be empty.

- foreach $v \in K_0$

    - let $K' =$ the lower link of v

    - if K' is empty then add v to C % local min

    - else

        * add v to A;
        * let $h' \colon K'_0 \to \mathbb{R}$ be the restriction of h or, alternatively, use the definition given in the text below;
        * `Extract`(K', h', ∞) and let A', B', C', r' denote the resulting partition of the simplices of K';
        * find the $w_0 \in C'_0$ so that $h'(w_0)$ is the smallest; add $[v, w_0]$ to B and define $r([v, w_0]) = v$;
        * for each $\sigma \in C' - w_0$ add $v * \sigma$ to C;
        * for each $\sigma \in B'$ add $v * \sigma$ to B, add $v * r'(\sigma)$ to A, and define $r(v * \sigma) = v * r'(\sigma)$.

    - end if

- continue foreach

The algorithm `ExtractRaw` works inductively by using `Extract` on the links of vertices. If we have a metric on K so that we can measure the lengths of edges, there is an alternative definition of h' in the lower link of v with the property that the vertex with the minimum value of h' more closely approximates the direction of steepest decrease of h. In the alternative definition, we set $h'(w) = (h(w) - h(v))/\ell([v, w])$, where $\ell([v, w])$ is the length of the edge $[v, w]$. This might result in duplicate values of h', in which case we perturb h' slightly; it is natural to break such ties using the value $h(w)$.

A naive version of persistence cancelling follows. In practice, you would probably wait to cancel until you found the pair $\tau \in C_{j-1}, \sigma \in C_j$ connected by exactly one gradient path so that $\mathrm{maxh}(\sigma) - \mathrm{maxh}(\tau)$ is minimized. Also, for example, if K is a subcomplex of a j-dimensional manifold, then it is more efficient to find all gradient paths that end at a $\tau \in C_{j-1}$ (since there are at most two of them), so you would implement it differently.

`ExtractCancel`(K, h, p, j)

- foreach $\sigma \in C_j$

    - find all gradient paths $\sigma = \sigma_{i1} \to \sigma_{i2} \to \ldots \to \sigma_{i\ell_i} \in C_{j-1}$ with $\mathrm{maxh}(\sigma_{i\ell_i}) > \mathrm{maxh}(\sigma) - p$;
    - foreach i, if $\sigma_{i\ell_i}$ does not equal any other $\sigma_{j\ell_j}$, let $m_i = \mathrm{maxh}(\sigma_{i\ell_i})$;
    - if at least one m_i is defined, then

        * pick a j so that $m_j = \min\{m_i\}$;

∗ Cancel$(K, h, \sigma_{j\ell_j}, \sigma, j)$.

– end if

• continue foreach

The algorithm Cancel(K, h, σ, τ, j) works if $\sigma \in C_{j-1}$, $\tau \in C_j$, and there is only one gradient path from τ to σ. This cancels the two critical simplices by reversing the arrows in the gradient path.
Cancel(K, h, σ, τ, j)

• find the unique gradient path $\tau = \tau_1 \to \sigma_1 \to \tau_2 \to \sigma_2 \to \cdots \to \sigma_k = \sigma$ (thus $\sigma_i = r(\tau_{i+1})$, σ_i is a face of τ_i, and $\sigma_i \neq \sigma_{i+1}$);

• delete σ and τ from C, add τ to B, and add σ to A;

• for $i = 1$ to k, redefine $r(\tau_i) = \sigma_i$.

Theorem 3.1.   *The A, B, C, and r produced by* ExtractRaw *have the property that there are no directed loops in the resulting modified Hasse diagram.*

Proof: Note that $\mathrm{maxh}(r(\sigma)) = \mathrm{maxh}(\sigma)$, since in the algorithm v is the vertex in all mentioned simplices with highest value of h. Also, if σ is a face of τ, then $\mathrm{maxh}(\sigma) \leq \mathrm{maxh}(\tau)$. Consequently, maxh is nonincreasing along any directed path in the modified Hasse diagram. So, maxh must be constant on any directed loop. Let $\sigma_0 \to \sigma_1 \to \cdots \to \sigma_k = \sigma_0$ be a directed loop in the modified Hasse diagram. Let v be the unique vertex so $h(v) = \mathrm{maxh}(\sigma_j)$ for all j. Then, v is a vertex of each σ_j so $\sigma_j = v * \tau_j$ for simplices τ_j in the lower link of v. But then, $\tau_0 \to \tau_1 \to \cdots \to \tau_k$ is a directed loop in the modified Hasse diagram of the lower link of v, which is impossible by induction on dimension and the following theorem.   □

Theorem 3.2.   *The* Cancel *algorithm does not produce directed loops. Hence, the A, B, C, and r produced by* Extract *have the property that there are no directed loops in the resulting modified Hasse diagram.*

Proof: (see [Forman 02, Theorem 9.1]) Suppose we have an A, B, C, and r so the resulting modified Hasse diagram has no directed loops. Suppose $\tau, \sigma \in C$ are joined by a unique gradient path. Assume that after performing Cancel(K, h, σ, τ, j) we end up with A', B', C', and r' so that the resulting modified Hasse diagram has a directed loop α. This was not previously a directed loop, and hence, a portion of it (say γ) must coincide with a segment (say η) of the gradient path. But then we can

construct a different gradient path from τ to σ by replacing η by $\alpha - \gamma$. This violates the condition that there be only one gradient path from τ to σ.   □

The previous theorems allow us to actually construct a discrete Morse function on K, so we can use all the machinery of discrete Morse theory. We use the following lemma in our three-dimensional implementation of the algorithm to reduce the number of gradient paths between critical simplices after cancellation.

Lemma 3.3.   *The A, B, C, r produced by* ExtractRaw(K, h, p) *have the property that there are no $\sigma \in C_i$, $i > 0$ so that all codimension-one faces of σ are in B_{i-1}, and this property persists under cancellation.*

Proof: Assume that all $(i-1)$-faces $\tau_0, \ldots, \tau_i$ of $\sigma^{(i)}$ belong to B_{i-1}. Then, there exist $(i-2)$-faces $v_0, \ldots, v_i$ of σ such that $r(\tau_j) = v_j$ for all $j = 0, \ldots, i$. Each v_j belongs to precisely two $(i-1)$-faces of σ, so there is a directed path

$$v_0 \to \tau_0 \to v_{i_1} \to \tau_{i_1} \to \cdots \to v_{i_j} \to \tau_{i_j}$$

such that τ_{i_ℓ} and $\tau_{i_{\ell+1}}$ are the only $(i-1)$-faces of σ having v_{i_ℓ} as a face. Let τ_k be the remaining $(i-1)$-coface of v_0. Then,

$$\tau_k \to v_0 \to \tau_0 \to v_{i_1} \to \cdots \to v_k \to \tau_k$$

is a directed loop. By Theorem 3.2 this is not possible.   □

In practice, it is not necessary to actually construct a discrete Morse function, since what one really needs is the modified Hasse diagram. However, the whole point of our algorithm is that a given function on the vertices of K gives rise to a discrete Morse function which somehow has the same behavior. In particular:

Theorem 3.4.   *There is an extension of h to a discrete Morse function h' with the same modified Hasse diagram as that produced by* ExtractRaw*. Moreover, given $\epsilon > 0$, we may choose such an h' so that $|h'(\tau) - \mathrm{maxh}(\tau)| \leq \epsilon$ for any simplex τ.*

Proof: We may suppose that $3\epsilon < |h(v) - h(w)|$ for all vertices $v \neq w$. If K' is the lower link of a vertex v, we may, by Theorems 2.7 and 3.1, find a discrete Morse function g_v on K'. After a linear scaling, we may suppose the range of g_v is in the interval $(h(v), h(v) + \epsilon)$. Let w_0 be the vertex of K' that minimizes h. By Lemma 3.5, we

know that w_0 is critical in K'. Now, define h' on the lower star of v by $h'([v\ w_0]) = h(v) - \epsilon$ and $h'(v * \tau) = g_v(\tau)$ for any simplex $\tau \neq w_0$ of K'.

We claim that if τ is a codimension-one face of σ and $h'(\tau) \geq h'(\sigma)$, then $\sigma \in B$ and $\tau = r(\sigma)$. Conversely, we claim that $h'(r(\sigma)) \geq h'(\sigma)$ for all $\sigma \in B$. The first claim implies that h' is a discrete Morse function, and the second then implies that the modified Hasse diagram for this Morse function coincides with that produced by `ExtractRaw`.

Let us prove the claims. Let v be the maximal vertex of σ so $h(v) = \mathrm{maxh}(\sigma)$. If $\sigma \neq v$, let σ' be the simplex so $\sigma = v * \sigma'$. To prove the first claim, let w be the maximal vertex of τ. If $w \neq v$, then $h'(\tau) \geq h'(\sigma) \geq h(v) - \epsilon \geq h(w) + 2\epsilon \geq h'(\tau) + \epsilon$, a contradiction; so, in fact, $w = v$, i.e., τ is in the lower star of v. Suppose first that $\tau = v$. Then, the only possibility is $\sigma = [v\ w_0]$, so $r(\sigma) = \tau$. Now, suppose that $\tau \neq v$. Then, there is a simplex τ' in the lower link of v so that $\tau = v * \tau'$. So, $g_v(\tau') \geq g_v(\sigma')$, and thus, there is only one possibility for τ' and, in fact, $\tau = r(\sigma)$. Now, let us prove the second claim. It holds if $r(\sigma) = v$, since then $\sigma = [v\ w_0]$ and $h'(\sigma) = h(v) - \epsilon$. But if $r(\sigma) \neq v$, then $r(\sigma) = v * r'(\sigma')$, so $h'(r(\sigma)) = g_v(r'(\sigma')) \geq g_v(\sigma') = h'(\sigma)$. □

Of course, a similar result would not hold for the output of `Extract`. Recall the smooth case where cancelling pairs of critical points requires changing the values of the function. The same thing holds for discrete Morse functions.

Lemma 3.5. *If v is the vertex of K at which h attains its minimum, then* `Extract` *will make v a critical vertex.*

Proof: `ExtractRaw` will make v critical because it is a local minimum. But then, `ExtractCancel` will never cancel v. This is because any critical 1-simplex τ is the start of exactly two gradient paths. So, if τ is connected to v by a single gradient path, it must be connected to some other vertex w by a single gradient path. Since $\mathrm{maxh}(\tau) - h(w) < \mathrm{maxh}(\tau) - h(v)$, `ExtractCancel` would cancel τ with w instead of with v, if it cancelled at all. □

4. AN EXAMPLE

Let K be the torus with the function $h : K_0 \rightarrow \mathbb{R}$ given as in Figure 1. We shall denote each vertex by its h-value. Implementing the algorithm `Extract`(K, h, ∞) by hand yields the following. We proceed through the vertices of K beginning at 0. At each stage, we show the partition

A, B, C. The elements of A and B are listed so that $r(\tau_i) = \sigma_i$ for $\tau_i \in B$ and $\sigma_i \in A$.

0   : $A = \emptyset, B = \emptyset, C = \{0\}$

10 : $A = \{10\}, B = \{[10, 0]\}, C = \{0\}$

20 : $A = \{10, 20\}, B = \{[10, 0], [20, 10]\}, C = \{0\}$

30 : $A = \{10, 20, 30\}, B = \{[10, 0], [20, 10], [30, 10]\},$
        $C = \{0\}$

40 : $A = \{10, 20, 30, 40\},$
        $B = \{[10, 0], [20, 10], [30, 10], [40, 0]\},$
        $C = \{0, [40, 20], [40, 30]\}$

50 : $A = \{10, 20, 30, 40, 50, [50, 10], [50, 20], [50, 30], [50, 40]\},$
        $B = \{[10, 0], [20, 10], [30, 10], [40, 0], [50, 0], [50, 10, 0],$
        $\quad [50, 40, 20], [50, 30, 10], [50, 40, 10]\},$
        $C = \{0, [40, 20], [40, 30]\}$

60 : $A = \{10, 20, 30, 40, 50, 60, [50, 10], [50, 20], [50, 30],$
        $\quad [50, 40], [60, 10], [60, 20], [60, 30], [60, 40]\},$
        $B = \{[10, 0], [20, 10], [30, 10], [40, 0], [50, 0], [60, 0],$
        $\quad [50, 10, 0], [50, 40, 20], [50, 30, 10], [50, 40, 10],$
        $\quad [60, 10, 0], [60, 20, 10], [60, 40, 30], [60, 40, 20]\},$
        $C = \{0, [40, 20], [40, 30]\}$

70 : $A = \{10, 20, 30, 40, 50, 60, 70, [50, 10], [50, 20], [50, 30],$
        $\quad [50, 40], [60, 10], [60, 20], [60, 30], [60, 40],$
        $\quad [70, 20], [70, 10], [70, 40], [70, 30]\},$
        $B = \{[10, 0], [20, 10], [30, 10], [40, 0], [50, 0], [60, 0],$
        $\quad [70, 0], [50, 10, 0], [50, 40, 20], [50, 30, 10],$
        $\quad [50, 40, 10], [60, 10, 0], [60, 20, 10], [60, 40, 30],$
        $\quad [60, 40, 20], [70, 20, 10], [70, 30, 10], [70, 40, 0],$
        $\quad [70, 40, 30]\},$
        $C = \{0, [40, 20], [40, 30]\}$

80 : $A = \{10, 20, 30, 40, 50, 60, 70, 80, [50, 10], [50, 20],$
        $\quad [50, 30], [50, 40], [60, 10], [60, 20], [60, 30],$
        $\quad [60, 40], [70, 20], [70, 10], [70, 40], [70, 30],$
        $\quad [80, 20], [80, 30], [80, 50], [80, 60], [80, 70]\},$
        $B = \{[10, 0], [20, 10], [30, 10], [40, 0], [50, 0], [60, 0],$
        $\quad [70, 0], [80, 0], [50, 10, 0], [50, 40, 20], [50, 30, 10],$
        $\quad [50, 40, 10], [60, 10, 0], [60, 20, 10], [60, 40, 30],$
        $\quad [60, 40, 20], [70, 20, 10], [70, 30, 10], [70, 40, 0]$
        $\quad [70, 40, 30], [80, 70, 20], [80, 50, 30], [80, 50, 20],$
        $\quad [80, 60, 0][80, 70, 0]\},$
        $C = \{0, [40, 20], [40, 30], [80, 60, 30]\}$

The `ExtractCancel` algorithm is not used until one gets to vertex 50 and becomes increasingly complicated as the vertex values increase. In fact, for vertex 80, two
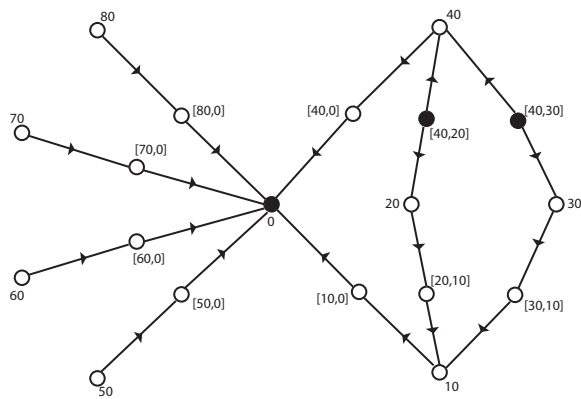
FIGURE 3. The graph R_1.

pairs of critical simplices in the lower link of 80 must be cancelled.

Note also that the critical 2-simplex produced by the algorithm is $[80, 60, 30]$, while in Figure 1, the critical 2-simplex is $[80, 50, 20]$. That is not surprising, since the algorithm tends to find critical simplices that maximize the average value of their vertices. For $[80, 60, 30]$, this average is $170/3$, while for $[80, 50, 20]$, it is 50. Note, however, that the value on the critical 2-simplex itself must be greater than the value of h at each of its vertices. Still, the critical 2-simplex is adjacent to the maximum of h on K_0, as one would expect. Also, the critical edges are the same as those in Figure 1.

The graphs R_1 and R_2 are shown in Figures 3 and 4, respectively. The critical simplices are represented by solid dots in the graphs.
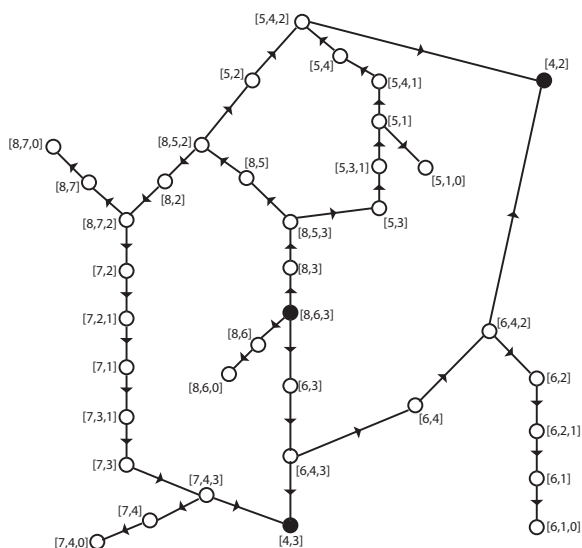

FIGURE 4. The graph R_2 (with function values scaled by 1/10).

5.   INFINITE PERSISTENCE

Applying `Extract` with infinite persistence in low dimensions yields the minimum possible number of critical simplices for any discrete Morse function. In particular, we have the following result:

Theorem 5.1. *Suppose K has dimension less than 2 or K has dimension 2 and is a subcomplex of a two-dimensional manifold. If one applies `Extract` with infinite persistence, then the number of critical i-simplices is the rank of $H_i(|K|; \mathbb{Z}/2\mathbb{Z})$ and, hence, must be minimal.*

Proof: By Lemma 5.3, this result is true for $i = 0$ and 2. But the Euler characteristic $\chi(|K|)$ is both the alternating sum of the ranks of $H_i(|K|; \mathbb{Z}/2\mathbb{Z})$ and (by Theorem 5.4) the alternating sum of the number of critical simplices of each dimension. So, the result must be true for $i = 1$. $\qquad\square$

Corollary 5.2. *Suppose $|K|$ is a manifold of dimension $n \leq 3$ and suppose for every vertex v either*

1. *the lower link of v is empty (local min),*

2. *the lower link of v deformation retracts to a $(k-1)$-sphere (local max if $k = n$; otherwise an index-k saddle), or*

3. *the lower link of v deformation retracts to a point (regular point).*

Then, when we perform `ExtractRaw`, we will obtain exactly one critical simplex for each vertex that is not a regular point. Each local minimum will be a critical vertex. Each local maximum will be in a critical n-simplex. Each index-k saddle will be in a critical k-simplex.

Proof: If the lower link of v is empty, `ExtractRaw` will designate v a critical 0-simplex. If the lower link of v deformation retracts to a point, then by Theorem 5.1, when we apply `Extract` to the lower link of v, we will get just one critical simplex, a critical vertex, which does not produce any critical simplices of K. If the lower link of v deformation retracts to a $(k-1)$-sphere, then, when we apply `Extract` to the lower link of v, we will get just two critical simplices, one a critical vertex and the other a critical $(k-1)$-simplex, which just produce a critical k-simplex of K. $\qquad\square$

Lemma 5.3. *If you apply `Extract` to K with infinite persistence, then there will be exactly one critical 0-simplex*

in each connected component of $|K|$. If K is a subcomplex of an n-dimensional manifold, then there will be exactly one critical n-simplex in each component of $|K|$, which is itself an n-manifold without boundary; in particular, the number of critical n-simplices is the rank of $H_n(|K|; \mathbb{Z}/2\mathbb{Z})$.

Proof: Each critical 1-simplex is the start of only two gradient paths. These paths can not end up at different critical 0-simplices, since `Extract` would have cancelled one of them. So, both paths end up at the same critical 0-simplex, which means that $\partial_1 = 0$ in Theorem 5.4, so $H_0(|K|; \mathbb{Z}/2\mathbb{Z}) = D_0$ has rank equal to the number of critical 0-simplices; i.e., there is exactly one critical 0-simplex in each component of $|K|$.

Now, suppose K is a subcomplex of an n-dimensional manifold. Then each $(n-1)$-simplex is a face of at most two n-simplices. Consequently, each critical $(n-1)$-simplex is the end of at most two gradient paths. So again, $\partial_n = 0$ and $H_n(|K|; \mathbb{Z}/2\mathbb{Z}) = D_n$. $\square$

The following result is the $\mathbb{Z}/2\mathbb{Z}$ version of Theorem 7.1 of [Forman 02].

Theorem 5.4. *The homology $H_*(|K|; \mathbb{Z}/2\mathbb{Z})$ can be computed from a chain complex D_*, where $D_i = \{\sum_{\sigma \in C_i} g_\sigma \sigma \mid g_\sigma \in \mathbb{Z}/2\mathbb{Z}\}$, and the boundary $\partial_i \colon D_i \to D_{i-1}$ is determined by $\partial_i(\sigma) = \sum_{\tau \in C_{i-1}} g_\tau \tau$, where g_τ is the number of gradient paths from σ to τ.*

6. IMPLEMENTATION

An implementation of this algorithm for K a subcomplex of a three-dimensional manifold is available at http://www.math.umd.edu/~hck/MorseExtract.html. It is fairly fast in practice; for example, for K a triangulation of the Klein bottle cross the circle with about 12,000,000 simplices, it takes less than 30 seconds to generate a discrete Morse function and cancel all pairs of critical points with small persistence.

Software for generating simplicial complexes with random vertices in $\mathbb{R}^3$ and function values on the vertices and an application for viewing the results of the algorithm interactively are also available. We tested the algorithm on several triangulations of different 3-manifolds; for example, $K \times S^1$, $(S^1)^3$, $(S^1)^2$ cross the interval, and their disjoint unions, and with various functions on the vertices. We also ran the algorithm on a simplicial complex obtained by computing the Delaunay triangulation on a natural data set containing meteorological data measured at 30,793 points, describing the ozone levels as a function of other meteorological data (temperature, humidity, and pressure). If the persistence is set to a higher level, the resulting values on the simplicial complex after cancellation represent a smoother function reflecting the behavior of the ozone values on a greater scale.

With persistence set to ∞, the algorithm can be effectively used for computing homology. If the total number of simplices is not too big, as many critical simplices as possible are cancelled. For example, on natural data with 200 vertices and about 4,200 simplices, all critical simplices except the minimum were cancelled. On a data set with 30,793 vertices and 200,887 simplices, the original set of critical simplices (283 vertices, 2,391 edges, 2,530 triangles, and 421 tetrahedra) obtained by setting persistence to 0 was reduced to zero critical tetrahedra, one critical vertex (the minimum), and 124 critical edges and triangles. On triangulations of 3-manifolds, a maximal number of critical simplices was cancelled as long as the number of simplices was not more than a few thousand. For example, on a disjoint union of two tori with 180 vertices and altogether about 5,000 simplices, the algorithm produced the minimal possible number of critical simplices, while on a triangulation of the torus with 270 vertices and a total of about 7,500 simplices, one critical vertex, six critical 1- and 2-handles, and one critical tetrahedron remain after the cancellation, i.e., three pairs of critical 1- and 2-handles remain uncancelled.

REFERENCES

[Edelsbrunner and Natarajan 04] H. Edelsbrunner and V. Natarajan. "Simplification of Three-Dimensional Density Maps." *IEEE Trans. Visual. Comput. Graphics* 10 (2004), 587–597.

[Edelsbrunner et al. 02] H. Edelsbrunner, D. Letscher, and A. Zomorodian. "Topological Persistence and Simplification." *Discrete Comput. Geom.* 28 (2002), 511–533.

[Edelsbrunner et al. 03a] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. "Morse-Smale Complexes for Piecewise Linear 3-Manifolds." In *Proc. 19th Ann. Sympos. Comput. Geom. 2003*, pp. 361–370. New York: ACM Press, 2003.

[Edelsbrunner et al. 03b] H. Edelsbrunner, J. Harer, and A. Zomorodian. "Hierarchical Morse-Smale Complexes for Piecewise Linear 2-Manifolds." *Discrete Comput. Geom.* 30 (2003), 87–107.

[Forman 98] R. Forman. "Morse Theory for Cell Complexes." *Adv. Math.* 134 (1998), 90–145.

[Forman 02] R. Forman. "A User's Guide to Discrete Morse Theory." *Sém. Lothar. Combin.* 48 (2002), Art. B48c, 35pp.

[Hersh 05] P. Hersh. "On Optimizing Discrete Morse Functions." *Adv. Appl. Math.* 35 (2005), 294–322.

[Lewiner et al. 03] T. Lewiner, H. Lopes, and G. Tavares. "Toward Optimality in Discrete Morse Theory." *Experimental Math.* 12 (2003), 271–285.

Henry King, Department of Mathematics, University of Maryland, College Park, MD 20742 (hck@math.umd.edu)

Kevin Knudson, Department of Mathematics and Statistics, Mississippi State University, Mississippi State, MS 39762 (knudson@math.msstate.edu)

Neža Mramor, Department of Computer and Information Science and Institute of Mathematics, Physics, and Mechanics, University of Ljubljana, Slovenia (neza.mramor@fri.uni-lj.si)