Visualization of Discrete Gradient Construction

Attila Gyulassy
SCI Institute
University of Utah
72 S Central Campus Drive,
Room 3750
Salt Lake City, UT 84112
jediati@sci.utah.edu

Joshua A. Levine
SCI Institute
University of Utah
72 S Central Campus Drive,
Room 3750
Salt Lake City, UT 84112
jlevine@sci.utah.edu

Valerio Pascucci
SCI Institute
University of Utah
72 S Central Campus Drive,
Room 3750
Salt Lake City, UT 84112
pascucci@sci.utah.edu

ABSTRACT

This video presents a visualization of a recent algorithm to compute discrete gradient fields on regular cell complexes [3]. Discrete gradient fields are used in practical methods that robustly translate smooth Morse theory to combinatorial domains. We describe the stages of the algorithm, highlighting both its simplicity and generality.

Categories and Subject Descriptors

F.2.2 [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms and Problems—*Geometrical problems and computations*; I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—*Curve, surface, solid, and object representations*

General Terms

Algorithms, Theory

Keywords

Scalar field topology, discrete gradient fields, discrete Morse theory, regular cell complexes

1. BACKGROUND

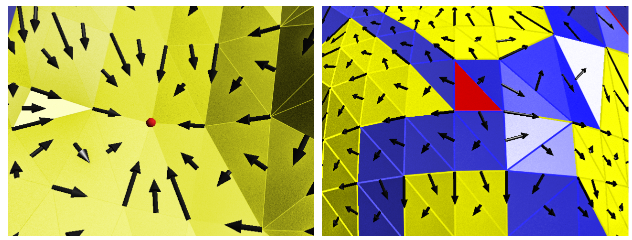
Practical techniques for Morse theory-based analysis often utilize a discretization of the domain and function. Such algorithms have proven to be effective tools for feature extraction from complex scientific data. The only known, scalable algorithm for computing volumetric Morse-Smale complexes [3] is based on discrete Morse theory, and the enabling ingredient is the construction of a discrete gradient vector field from sampled data.

The following is a brief overview of discrete Morse theory, due to Forman [2]. A d-cell is a topological space that is homeomorphic to a d-ball $B^d = \{x \in \mathbb{E}^d : |x| \leq 1\}$. A cell α with dimension d is denoted $\alpha^{(d)}$. For cells α and β, we write $\alpha < \beta$ to mean that α is a *face* of β and β is a *co-face* of α. When $dim(\alpha) = dim(\beta) - 1$, we say α is a *facet* of β, and β is a *co-facet* of α. The *boundary operator* ∂ maps a cell to its facets.

A d-cell is *regular* if (a) its closure is homeomorphic to a d-ball and its boundary to a d-sphere, and (b) its boundary is a union of regular cells. We say K is a *regular cell complex* if all its cells are regular. When the underlying space $|K|$ of K is a topological space $\mathbb{M}$, we say that K is a mesh representation of $\mathbb{M}$. A function $f\colon K \to \mathbb{R}$ that assigns scalar values to every cell of K is a *discrete Morse function* if for every $\alpha^{(d)} \in K$, the sets:

$$L = \{\beta^{(d+1)} > \alpha^{(d)} : f(\beta) \leq f(\alpha)\}$$
$$U = \{\gamma^{(d-1)} < \alpha^{(d)} : f(\gamma) \geq f(\alpha)\}$$

satisfy $|L| \leq 1$ and $|U| \leq 1$. A cell $\alpha^{(d)}$ is critical if $|L| = 0$ and $|U| = 0$.



(a) (b)

Figure 1: Discrete vector arrows around a (a) minima, or critical 0-cell shown as a red sphere and (b) maxima or critical 2-cell shown as a red triangle.

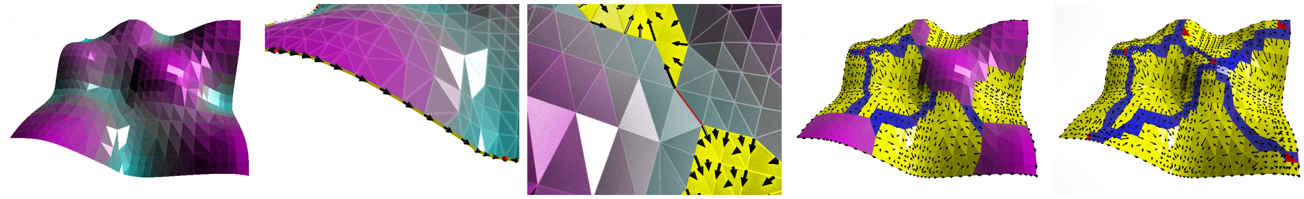
A *vector* in the discrete sense is a pair of cells (α, β) such that $\alpha^{(d)} < \beta^{(d+1)}$ and $f(\beta) \leq f(\alpha)$. We say that an *arrow* points from $\alpha^{(d)}$ to $\beta^{(d+1)}$. Intuitively, this vector simulates a direction of flow from α to β. A *discrete vector field* V on K is a collection of vectors $\{\alpha^{(d)} < \beta^{(d+1)}\}$ such that each cell of K is in at most one vector of V (discrete vectors are visualized in Figure 1). Given a discrete vector field V on K, a V-*path* is a sequence of cells

$$\alpha_0^{(d)}, \beta_0^{(d+1)}, \alpha_1^{(d)}, \beta_1^{(d+1)}, \alpha_2^{(d)}, \ldots, \beta_r^{(d+1)}, \alpha_{r+1}^{(d)}$$

such that for each $i = 0,\ldots, r$, the vector $\{\alpha^{(d)} < \beta^{(d+1)}\} \in V$, and $\beta_i^{(d+1)} > \alpha_{i+1}^{(d)} \neq \alpha_i^{(d)}$. A V-path is the discrete equivalent of a streamline in a smooth vector field. A discrete vector field in which all V-paths are monotonic with respect to the discrete Morse function, and does not contain any loops is a *discrete gradient field*. This pairing of cells is equivalent to defining the sets L and U for each cell. Consequentially, critical cells of the discrete Morse function are equivalent to unpaired cells in the discrete gradient field.

A discrete gradient field unambiguously defines flow-based structures; the descending manifold of a critical cell α is the

SCG'11, June 13–15, 2011, Paris, France.
ACM 978-1-4503-0682-9/11/06.



(a) (b) (c) (d) (e)

Figure 2: Assigning gradient arrows on a terrain (a). Scalar values (height) are encoded from cyan (low values) to magenta (high values). (b) Boundary cells are paired first. (c) Pairing interior cells finds a saddle (red edge). (d) As pairing continues, a maxima is identified (red triangle). (e) Gradient construction is complete. Ascending 1-manifolds shown as blue cells.

sum of V-paths starting at α, and the ascending manifold is the sum of V-paths ending at α. These structures can be queried to extract user-defined features. This video presents some of the specific details regarding the construction of the discrete gradient field using a variant of the algorithm presented in [3, 4].

2. ALGORITHM OVERVIEW

The algorithm presented in [3] creates gradient arrows by assigning cells to pairs, ordered first based on dimension, then on function value. This enables certain optimizations for gridded data, however, it makes the construction more difficult to understand intuitively in the case of regular cell complexes. Here we present a variant on the algorithm (see Algorithm 1) that assigns gradient arrows in a single loop acting on a priority queue P, ordered both on function value (low-to-high) and dimension (high-to-low). Any ties in this ordering are resolved using simulation of simplicity [1, 4].

Algorithm 1 Construct Discrete Gradient.

Assign function values to all cells in K
Initialize P with potential minima.
while P is not empty **do**
  $\sigma = P$.pop()
  if unassigned(σ) **then**
    if unassignedFacets(σ) is empty **then**
      setCritical(σ)
    else
      $\sigma' =$ pair(σ)
    end if
    for each cofacet τ of σ, σ' **do**
      if $|$unassignedFacets(τ)$| = 1$ **then**
        P.push(τ)
      end if
    end for
  end if
end while

The pseudocode for creating a discrete gradient vector field uses the following functions:

1. unassigned(σ) returns true iff a cell is not yet marked critical or paired;
2. unassignedFacets(σ) returns the set $\{\rho_0, \ldots, \rho_k\}$ of unassigned facets of σ;
3. setCritical(σ) marks σ as a critical cell; and
4. pair(σ) assigned σ as paired along with its sole unpaired facet σ', which it returns.

Algorithm 1 is first run on boundary cells, then on the interior, to simulate mirrored boundary conditions. As in [3], each cell is first given a function value that is the maximum of its vertices. Next, 0-cells with function value lower than any neighbor, the potential minima, are inserted into P. We iteratively remove the head of the queue and either pair it in a gradient vector, or assign it critical, in either case adding to P the co-facets of the newly assigned cells that have exactly one unassigned facet. A visual overview of the stages of the algorithm is shown in Figure 2.

3. CREATING THE VIDEO

Our software was implemented in C++, producing still frame visualizations of the gradient field in OpenGL for each stage of the construction.

Our video is intended to be a self-contained introduction to many of these concepts. It first discusses some additional background on discrete Morse theory and scalar field analysis, gives an overview of the steps of the algorithm, and finally shows some additional examples of gradient field construction. The same algorithm is an integral step towards computing discrete Morse-Smale complexes.

4. ACKNOWLEDGMENTS

This work was supported by National Science Foundation awards IIS-0904631, IIS-0906379, and CCF-0702817. This work was also performed under the auspices of the U.S. Department of Energy by the University of Utah under contracts DE-SC0001922 and DE-FC02-06ER25781.

5. REFERENCES

[1] H. Edelsbrunner and E. P. Mücke. Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms. In *Symposium on Computational Geometry*, pages 118–133, 1988.

[2] R. Forman. A user's guide to discrete Morse theory. In *Proc. of the 2001 Internat. Conf. on Formal Power Series and Algebraic Combinatorics, A special volume of Advances in Applied Mathematics*, page 48, 2001.

[3] A. Gyulassy, P.-T. Bremer, B. Hamann, and V. Pascucci. A practical approach to Morse-Smale complex computation: Scalability and generality. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1619–1626, 2008.

[4] A. Gyulassy, P.-T. Bremer, B. Hamann, and V. Pascucci. Practical considerations in Morse-Smale complex computation. In *Topological Methods in Visualization (TopoInVis)*, 2009.