Visualizing Forman's discrete vector field

THOMAS LEWINER, HÉLIO LOPES AND GEOVAN TAVARES

Department of Mathematics — Pontifícia Universidade Católica — Rio de Janeiro — Brazil
{tomlew, lopes, tavares}@mat.puc--rio.br.

Abstract. Morse theory has been considered to be a powerful tool in its applications to computational topology, computer graphics and geometric modeling. Forman introduced a discrete version of it, which is purely combinatorial. This opens Morse theory applications to a much larger scope.

The main objective of this work is to illustrate Forman's theory. We intend to use some of Forman's concepts to visually analyze the topology of an object. We present an algorithm to build a discrete gradient vector field on a cell complex as defined in Forman's theory.

Keywords: ***Morse Theory. Forman Theory. Vector Field Visualization. Computational Topology.***

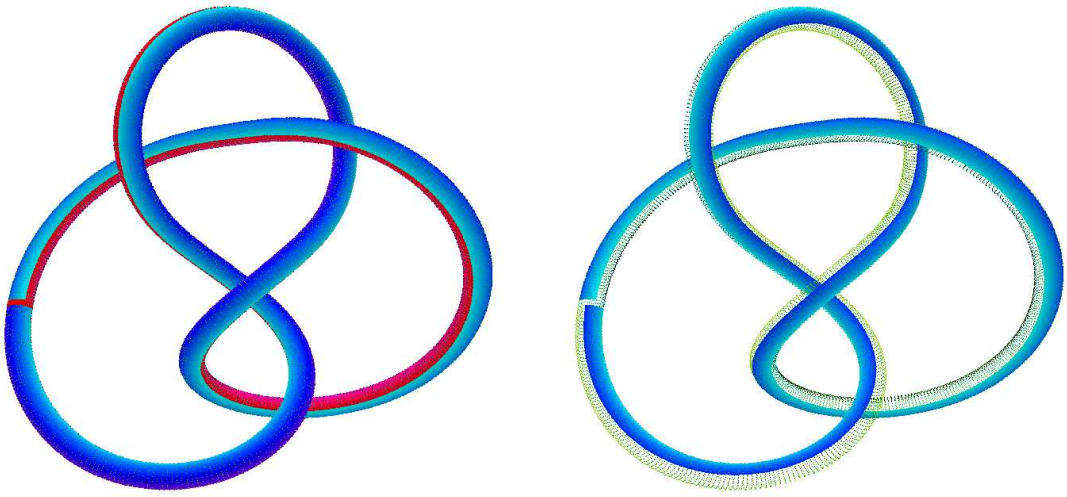


Figure 1: *The gradient vector field on a figure eight knot model [20].*

1 Introduction

Morse theory is a fundamental tool for investigating the topology of smooth manifolds. Particularly for computer graphics, many applications have been devised [9, 27, 18, 19]. Also in the new field of computational topology [7, 29], Morse theory has been used to devise topology based algorithms and data structures [8, 23]. The aim of this work is to visualize a similar tool for discrete structures (see Figure 1).

Morse proved that the topology of a manifold is very closely related to the critical points of a real smooth map defined on it [26]. Morse theory is one of the most powerful tools to understand the topology of a manifold.

Preprint MAT. 02/02, communicated on March 10^{th}, 2002 to the Department of Mathematics, Pontifícia Universidade Católica — Rio de Janeiro, Brazil. The corresponding work was published in Visualization and Mathematics III, pp. 95–112. Springer, 2002..

The recent results in Morse theory by Forman [11, 12] extended several aspects of this fundamental tool to cell complexes. This theory has already been used in a more theoretical context [4, 5]. The main goal of this work is to visually investigate topological aspects of a geometric or an abstract model, by using this theory.

The paper is organized as follows. In section 2 *Basic concepts*, we will briefly introduce the notion of cell complex, define discrete gradient vector field and its critical elements as defined in Forman's theory, and state a very nice result of Forman on homotopy. In section 3 *Hypergraphs and Hypertrees* we will need some definitions of hypergraph theory, which are slightly different from the classical ones [3]. In section 4 *Algorithm*, we will introduce our algorithm to build those gradient fields, trying to reach optimality. This algorithm is proven to give a minimal number of critical cells for the case of 2-manifolds [21]. Reaching the minimum in the general

case is MAX SNP hard [10]. However, our algorithm shows to give a reasonable number of critical cells in quadratic time. We will illustrate some applications to visualization in the last section.

2 Basic concepts

This section aims to give familiarity with Forman's theory. For a given cell complex, discrete Morse theory as introduced by Forman can be built on a class of discrete gradient vector field and its critical elements. We will define those notions in the following paragraphs.

Similarly to the classical Morse Theory, Forman proved that the topology of a cell complex is related to its critical elements in a very strong way. More precisely, a cell complex with a discrete gradient vector field V is homotopy equivalent to a complex composed of only the critical elements of V.

For a complete presentation of Forman's theory and its application, see [11, 12, 13, 14].

(a) Cell Complexes

A *cell complex* is, roughly speaking, a generalization of the structures used to represent solid models: it is a consistent collection of cells (vertices, edges,...).

More formally, a *cell* $\alpha^{(p)}$ of dimension p is a set homeomorphic to the p-ball $\{x \in \mathbb{R}^p : \|x\| \leq 1\}$. When the dimension p of the cell is obvious, we will simply denote α instead of $\alpha^{(p)}$.

A *cell complex* K of dimension n is a collection of p-cells, $0 \leq p \leq n$, such that every intersection of the closure of two cells of K is also a cell of K. A complete introduction to cell complexes can be found in [25].

A p-cell $\alpha^{(p)}$ is a *face* of a q-cell $\beta^{(q)}$ ($p < q$) if $\alpha \in \beta$. We will use the notation $\alpha^{(p)} \prec \beta^{(q)}$, and say that α and β are *incident*.

In this paper, we will only consider finite cell complexes, i.e. complexes with a finite number of cells.

(b) Forman's discrete gradient vector fields

Forman's theory relies on admissible functions on a cell complex, or equivalently their gradient vector field. We chose here to introduce the theory from the second point of view, although our construction of a vector field can be done in the same way to define discrete Morse function.

Definition 1 (Combinatorial vector field) *A* combinatorial vector field V *defined on a cell complex K is a disjoint collection of pairs $\{\alpha^{(p)}, \beta^{(p+1)}\}$ of incident cells : $\alpha^{(p)} \prec \beta^{(p+1)}$.*

For such pairs, $V(\alpha) = \beta$ and $V(\beta) = 0$. If a cell σ does not belong to any pair, then $V(\sigma) = 0$.

We will represent this paring with an arrow from $\alpha^{(p)}$ to $\beta^{(p+1)}$.

A *non-trivial closed V-path* is an alternate sequence of r p- and $(p+1)$-cells $\alpha_0, \beta_0, \ldots, \alpha_r, \beta_r, \alpha_{r+1} = \alpha_0$ satisfying :

$$V(\alpha_i^{(p)}) = \beta_i^{(p+1)} \quad \text{and} \quad \beta_i^{(p+1)} \prec \alpha_{i+1}^{(p)} \neq \alpha_i^{(p)}.$$

Definition 2 (Discrete gradient vector field) *A combinatorial vector field V will be called a* discrete gradient vector field *if there is no non-trivial closed V-path.*

(c) A simple example

In the example of Figure 2, the discrete gradient vector field V is represented by arrows, from a cell of the complex to its image by V: from an edge to a face, and from a vertex to an edge.

The corresponding Hasse diagram (Figure 3) represents every cell by one node. The faces (2-cells) are aligned on top rank, the edges (1-cells) on the middle one and the vertices (0-cells) on the bottom rank. A link between two nodes symbolizes that the corresponding cells are incident. We linked by a red line paired cells. Blue lines represent the incidences we selected for the spanning tree, as we will do in the algorithm (see section 4 *Algorithm*.

The Hasse diagram are drawn as a directed graph with the AT&T software named `GraphViz - dot` [16].

(d) Critical cells

Morse proved that the topology of a manifold is related to its critical elements. Forman gave an analogous result, with the following definition for the critical cells.

Definition 3 (Critical Cells) *A cell α is* critical *if it is not paired with any other cell, i.e.:*

$$V(\alpha) = 0 \qquad \text{and} \qquad \alpha \notin Im(V)$$

In the example of Figure 2, critical cells are drawn in red: there is one critical vertex and one critical edge. In the Hasse diagram of Figure 3, red nodes represent those critical cells.

The number of critical cells is not a topological invariant, as it depends on the discrete gradient vector field defined. For example, an empty discrete vector field (i.e. no cells are paired) would have all its cells critical. Our algorithm is proven to give a minimal number of critical cells for the case of 2-manifolds [21]. Reaching the minimum in the general case is MAX SNP hard [10]. However, our algorithm gives a reasonable number of critical cells in quadratic time.

(e) Homotopy properties

Forman proved that a cell complex with a discrete gradient vector field V is *homotopy equivalent* to a complex built with exactly one cell for each critical element of V. In the example of Figure 2, there is one critical vertex and one critical edge: the corresponding complex has the homotopy of a circle.

Homotopy equivalence means continuous deformation (see [1]), and Forman gave an explicit way of doing this deformation.

From homotopy theory, we know there is exactly one critical vertex (1-cell) per connected component of the complex.

The corresponding work was published in Visualization and Mathematics III, pp. 95–112. Springer, 2002..

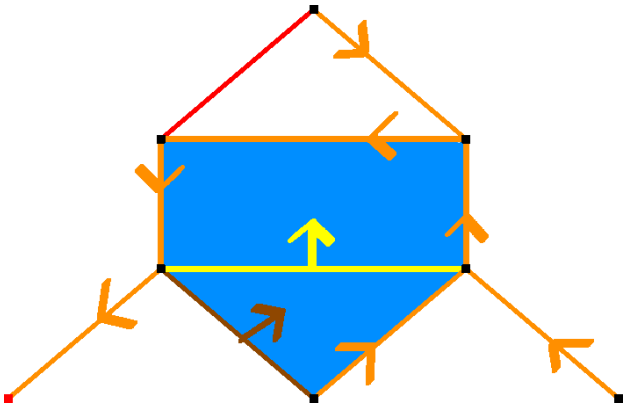


Figure 2: *A cell complex with its discrete gradient field.*

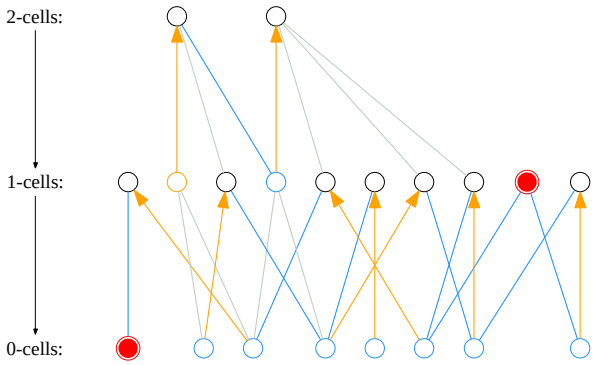


Figure 3: *The Hasse diagram with the pairing.*

Starting from that vertex, we can follow the gradient to go from one cell to the incident ones, and from those to their paired cells, and so on as on Figure 4. Critical cells are not paired, so this route stops at those, and forks at regular cells.

Forman proved that the inverse routes, without the critical parts, are deformation retracts, so do not alter the homotopy. The routes that end with a critical cell can also be retracted as above if we glue back the critical cell on the remaining cells. Thus, the critical cells represent the modification of the homotopy in this route.

On Figure 4 we see this route at different steps. This corresponds to cutting a differentiable manifold at different heights, as in classical Morse theory [26], although Forman's theory is completely independent of the geometry of the complex.

3 Hypergraphs and Hypertrees

We need here to generalize the notion of graphs. For example, in the triangulation of a solid, an edge can be incident to more than one face. Thus the graph whose nodes are the triangles of the triangulation, and whose links joins triangles that share an edge would not be an ordinary graph: such links can have more than two end nodes.

(a) Simply oriented hypergraphs

We use here a slightly different structure of hypertrees than the classical ones. A complete introduction to hypergraphs can be found in [3] for more details.

Definition 4 (Hypergraph) *A* hypergraph *is a pair* (N, L). N *is the set of nodes. The elements of* L *are family of nodes, and are called* hyperlinks.

We will classify hyperlinks into the *regular hyperlinks* (or shortly, *link*), which join two distinct nodes as in ordinary graphs, and the *non-regular hyperlinks*, which loop on one node or join three nodes or more.

We will give a hypergraph a simple orientation by distinguish one node in every non-regular hyperlink. We will call that node the *source node* of the hyperlink lk and write n_{lk}. The other nodes of lk will be called destination nodes of lk. A node can be the source of at most one link. The regular hyperlinks are not necessarily oriented.

(b) Illustration

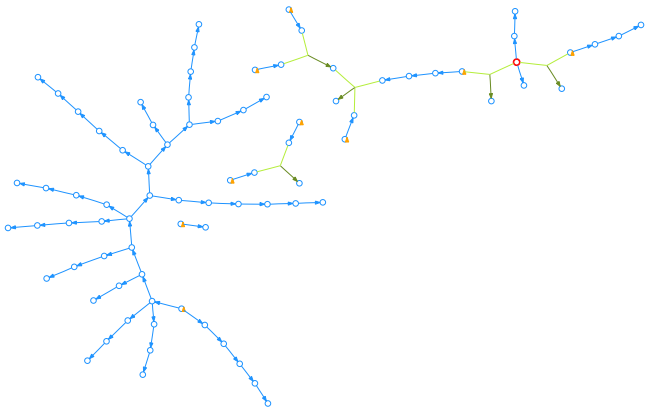


Figure 5: *A part of the dual hypertree resulting while processing a solid 2-sphere.*

The graph of Figure 5 represents a simply oriented hypergraph. Every regular link (in blue) has not yet a meaningful orientation (it represents the gradient vector field).

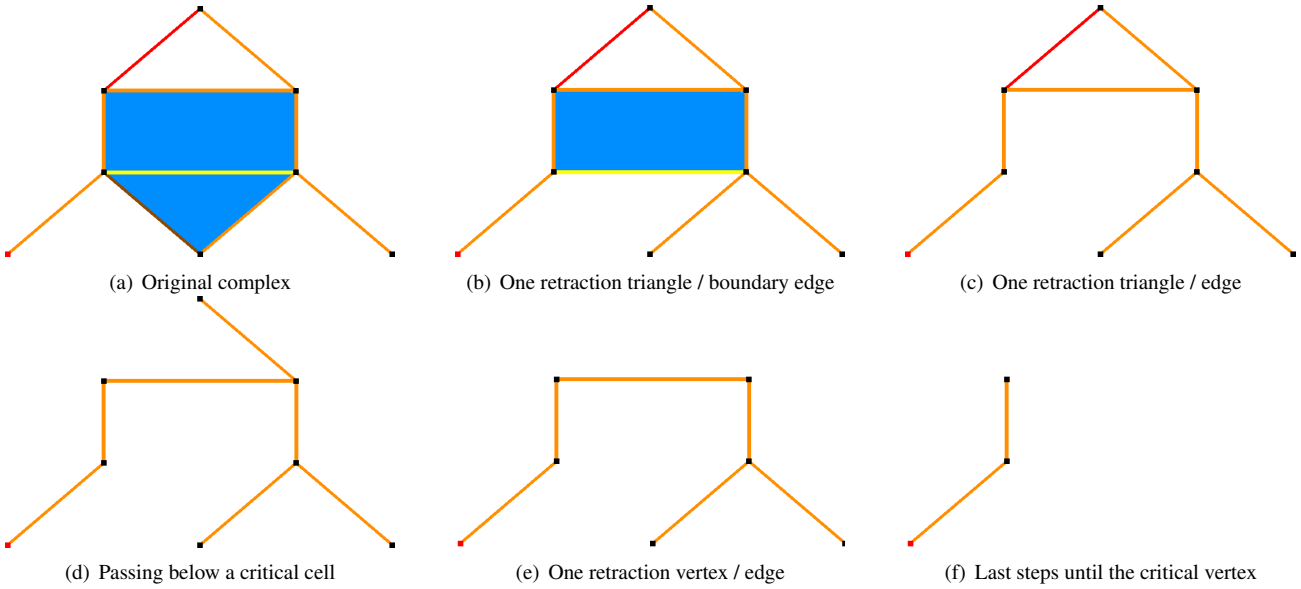
The non-regular hyperlinks are of two kinds: those incident to only one node (boundary links in orange), and those incident to more than two nodes (in green). In both cases, exactly one node is the origin of the non-regular hyperlink.

(c) Hypertrees

Definition 5 (Regular components) *The* regular components *of a hypergraph* (N, L) *are the connected components of the ordinary graph* (N, R)*, where* R *is the set of regular hyperlinks.*

An hypercircuit in a simply oriented hypergraph is a sequence of hyperlinks $lk_1, lk_2, \ldots, lk_r$ where :

– lk_i and lk_{i+1} share a node, with the convention $lk_{r+1} = lk_1 : lk_i \cap lk_{i+1} \neq \emptyset$.



(a) Original complex (b) One retraction triangle / boundary edge (c) One retraction triangle / edge

(d) Passing below a critical cell (e) One retraction vertex / edge (f) Last steps until the critical vertex

Figure 4: *The inverse route of the gradient of last example*

– if lk_{i+1} is a non-regular hyperlink, the shared node of lk_i and lk_{i+1} is the source node of lk_{i+1} : $lk_i \cap lk_{i+1} = n_{lk_i}$

Definition 6 (Hypertree) *We will say that a simply oriented hypergraph (N,L) is a* hypertree *if the 3 conditions below are satisfied :*

1. *Every regular component of* (N, L) *is an ordinary tree.*
2. *There is at most one source node in each regular component.*
3. (N, L) *has no hypercircuit.*

(d) Example

On Figure 6 for example, we can see different regular component in blue. They are isolated or connected by a non-regular hyperlink (in green). Those hyperlinks in green form a kind of tree, respecting the above definition 6.

The regular component only have one source node. This source node is the one incident to a boundary link (orange loops) or to a non-regular hyperlink (on the dark green arrow side).

4 Algorithm

In this section we will introduce our algorithm to define a discrete gradient vector field for a given cell complex. This algorithm's validity and analysis will be published elsewhere.

The algorithm is optimal for surfaces [21], in the sense that it minimizes the number of critical cells. But the general case has been proven to be MAX SNP hard, i.e. any polynomial approximation can be arbitrarily far from the optimal. However, our algorithm shows to give a reasonable number of critical cells.

(a) Outline

Let us consider a finite cell complex K of dimension n. The algorithm consists in the following steps :

1. In the first step, we select all n-cells, with some incident $(n-1)$-cells, as explained in section 4(d) *Selecting cells of the dual hypertree*. The algorithm optimality relies on this step, and its complexity is quadratic in the worst case. Elsewhere it has a linear complexity.

2. We then define the vector field for the selected cells as presented in section 4(c) *First steps: construction on dual hypertrees*. The cells of K not selected in the last step form again a complex K'. As every n-cell is selected during the first step, K' has dimension at most $n-1$.

3. So we repeat those steps until the unselected cells form a complex of dimension 1, i.e. a graph. At last, we build the vector field on that graph as explained in section 4(b) *Last step: construction on graphs*.

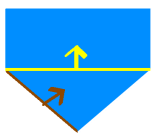


Figure 7: *First step : selecting faces and edges in a spanning tree fashion.*

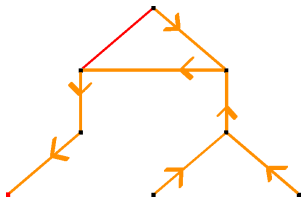


Figure 8: *Last step : processing the remaining vertex/edge graph.*

Working again on the example of Figure 2, we see on Figure 7 and Figure 8 the two steps of the algorithm. During the

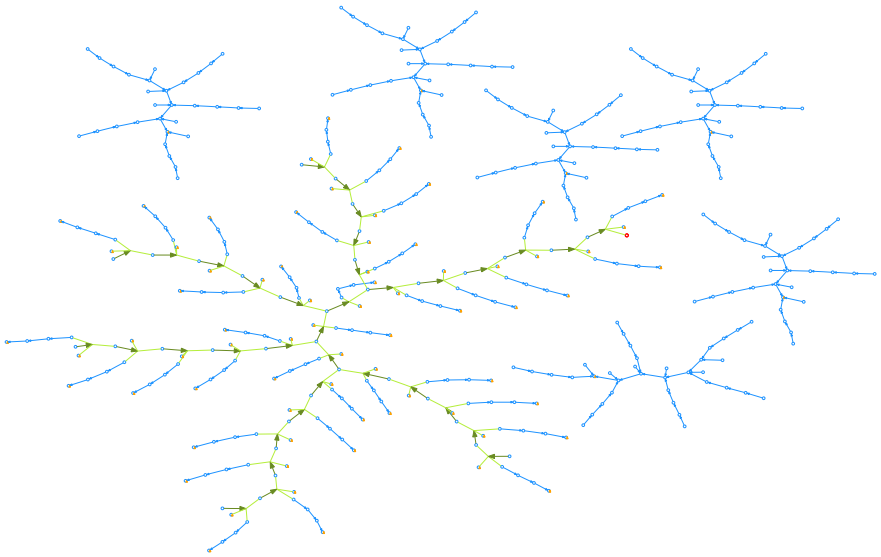


Figure 6: *The dual hypertree resulting while processing a model of $S^2 \times S^1$.*

first step, the vector field is defined on a dual tree containing all faces (see section 4(c) *First steps: construction on dual hypertrees* and section 4(d) *Selecting cells of the dual hypertree*). The unpaired vertices and edges form another cell complex, actually an ordinary graph. During the second and last step, the vector field is defined on it.

(b) Last step: construction on graphs

An ordinary *graph* is a pair (N, L). N is a set which elements are called *nodes*. L is a family of pairs of nodes (i.e. duplicated edges are allowed), whose elements are called *links*. Such a graph is an ordinary *tree* if it is connected and contains no cycle.

We know from the topology of a graph that any graph is homotopy equivalent to a node with loops. From the vector field point of view, we have to pair every node except one, leaving the loops unpaired.

We build a spanning tree (with any of the classical methods) of the nodes of the graph. All the links which are not in the tree will remain unpaired, and thus be critical. We then choose a root node r for the tree, which will also be left unpaired. Then, every link $\{r, s\}$ incident to r will be paired with their other end node s. We repeat the process on all those nodes m, and so on until the leaves are reached. Finally, we have paired all the nodes and links of the spanning tree, except r.

As the tree contains no cycle, the resulting vector field is admissible as a discrete gradient vector field. On Figure 9, we can visualize the vector field on a graph resulting of the process of a model of Poincare's homological sphere [17]. There are two cycles in the graph, which remains unpaired and critical edges. The root node of the graph also remains unpaired, and is critical. Those graphs are drawn with the AT&T software named `GraphViz - neato` [15].

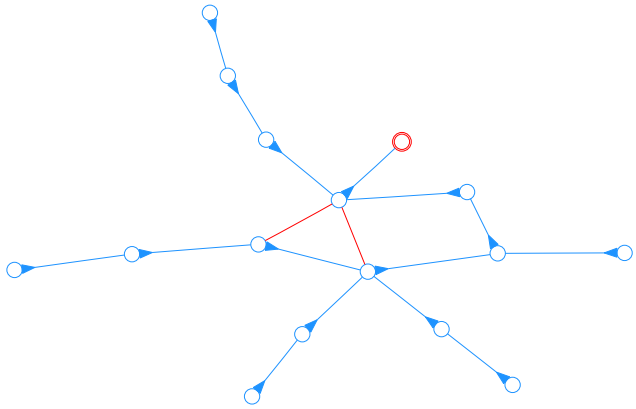


Figure 9: *The graph remaining after processing Hachimori's model of Poincare's homological sphere [17].*

(c) First steps: construction on dual hypertrees

As an extension of the case of a vertex spanning tree, we will define the vector field on dual hypertrees. A vertex spanning tree of a complex K has its nodes representing the vertices of K and its links representing some edges of K.

We will now consider a dual hypertree (N, L) extracted from K. Its nodes will represent the p-cells of K. A hyperlink representing a $(p-1)$-cell σ of K will join the nodes corresponding to the p-cells incident to σ. On the contrary of the previous case, the vector field will pair a node to a hyperlink. In the next section, we will present a procedure to choose the hyperlinks in such a way that (N, L) will be a hypertree.

We will process regular component per regular component. As (N, L) does not have any hypercircuit (definition 6 condition 3), there is at least one component that has no destination node. If this component has a source node (and there

is at most one by definition 6 condition 2), we will denote it r. In the other case, r will denote an arbitrary node.

The component is an ordinary tree (definition 6 condition 1). Thus we can pair each leaf node with its unique incident link. The unpaired elements of the component form again an ordinary tree, and we repeat the pairing until there is only the root node r left.

If the component had a source node, we pair that source node with its non-regular hyperlink. In the other case, we leave the root r unpaired and it will remain critical.

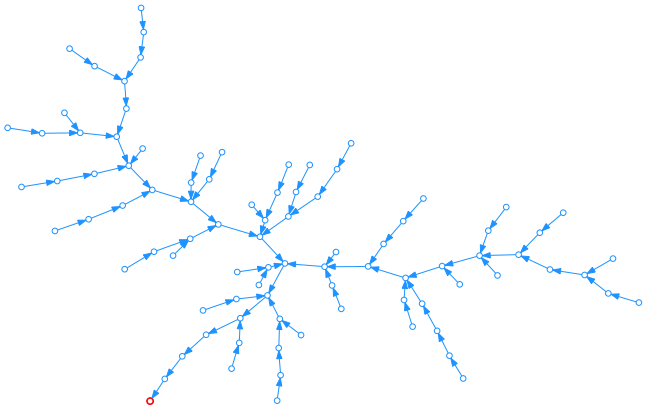


Figure 10: *A spanning tree tetrahedra/triangles processing Hachimori's model of Poincare's homological sphere [17].*

For example on Figure 10, the tree is processed from the leaves to the root, which is critical (in red).

As the hypertree does not contain hypercircuit, the resulting vector field will be admissible as a discrete gradient vector field.

The unpaired elements of (N, L) form again a dual hypertree, and we repeat the process above to pair all the hypertree except possibly the critical root nodes.

(d) Selecting cells of the dual hypertree

We use a greedy algorithm to select cells of the dual hypertrees. As in the construction of a spanning tree, we maintain an auxiliary structure that assigns to each cell its component number. This structure is similar to the union/find structure of [6].

The dual hypertree must contain all the cells of maximal dimension, say p, which will be represented by nodes. It must also contain some of the cells of dimension $(p-1)$ represented by hyperlinks.

To reach optimality, we will try to select the maximum number of hyperlinks into the hypertree. For example, adding the hyperlink of the left side of Figure 11 allows us to pair it with the node on the left. Thus, there will be less critical (unpaired) nodes.

First, for each regular hyperlink, we test whether it loops inside a component or it joins two components. In the latter case, we select it for the hypertree and (lazily) update the auxiliary structure. At the end of this step, the connected

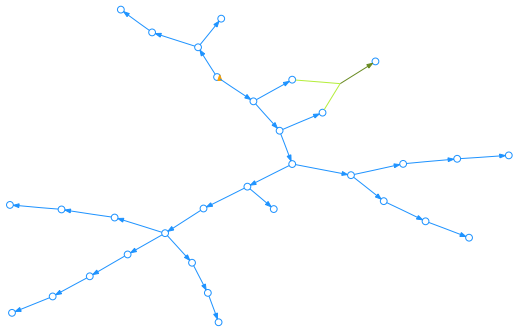


Figure 11: *Detail of a hyperlink insertion in the dual hypertree appearing with a solid torus model.*

components of the tree are the regular components of the final hypertree.

Then we process every *boundary cell*, i.e. a $(p-1)$-cell that is incident to only one p-cell in the complex. We add to the hypertree at most one boundary cell per regular component (to respect definition 6 condition 2). A regular component without boundary cell will be said *deficient* and in the other case *completed*.

Among the non-regular hyperlinks, we give priority to the boundary cell: it is the cheapest way to ensure the resulting hypergraph will not contain hypercircuit.

We finally process the non-regular hyperlinks. For each non-regular hyperlink lk left, we test if it is incident to a deficient component. We require the selected deficient components to have only one node of lk. We also check whether lk is incident to at least one completed component. In that case, we add lk to the hypertree and consider the deficient component as completed. As this has changed the configuration of deficient and connected component, we test again all non-regular hyperlinks.

As long as there are deficient components, we try until we cannot add non-regular hyperlinks. This is the bottleneck point where the algorithm is, in the worst case, quadratic in the number of those non-regular hyperlink.

5 Applications

(a) Visualizing the gradient field of a geometric model

In differentiable Morse theory, the gradient vector field can be obtained with a Morse function by a derivative computation. For height functions, this leads to a very simple geometric interpretation of the vector field.

A similar result can be obtained by a purely combinatorial way, using Forman's theory, as in Figure 12. The relation of classical Morse theory to geometry does not stand *as is* in the discrete theory. For example, the discrete gradient vector field can be disconnected from the geometry, even for simple models as for a sphere (see Figure 13). This gives a real power of Forman's theory: all the above figures has been done without any geometrical test.

The corresponding work was published in Visualization and Mathematics III, pp. 95–112. Springer, 2002..

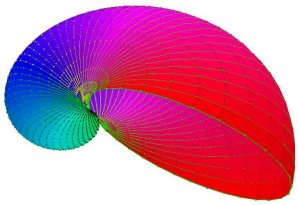


Figure 12: *A discrete gradient vector field on a shelf model.*

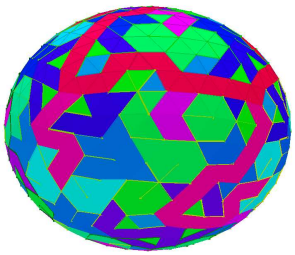


Figure 13: *A discrete gradient vector field on a simple 2-sphere.*

As mentioned in section 2(e) *Homotopy properties*, there is a natural way to go along the discrete gradient vector field: beginning with the critical vertex, following to the incident edge and their paired vertices, then continuing with a boundary edge to the faces and their incident edges.

The colors of the figure mark this route, following increasing the Hue component of the HSV decomposition: in green are the first visited faces, the route continue on with the blue and then purple faces, until the red ones.

Considering a classical Morse gradient field and following it in the same way, we would see cells of the same height drawn with the same color. The ordering of color (by the hue) gives the ordering of the heights. The color of the figures can be interpreted as a height function.

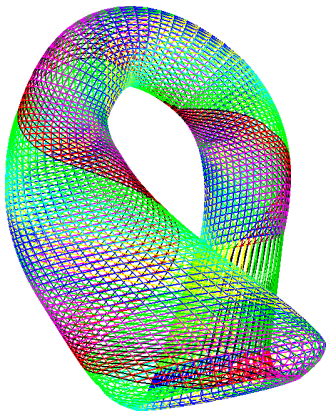


Figure 14: *The gradient vector field on edges/vertices of Klein bottle model.*

The discrete gradient vector field is also a powerful tool to understand the structure of a model. As above, edges drawn with the same color are at the same height. For example on Figure 14 (model from the Math&Media Lab., created by Sinésio Pesco), we dawned the edges of a Klein bottle model to see the auto-intersection. Looking at the green edges for example, we see clearly a Möbius strip spiraling along the bottle. The discrete gradient vector field gives a more intuitive sense of the non-orientability of the Klein bottle.

(b) Visualizing the structure of an abstract complex

Many topological objects appear without a geometric model, or with a model in higher dimensions. Those structures are quite difficult to understand without visualization. Forman's theory points out a topologically consistent way of choosing significant cells. Those cells can be outlined in a Hasse diagram, as done on Figure 15 and Figure 16.

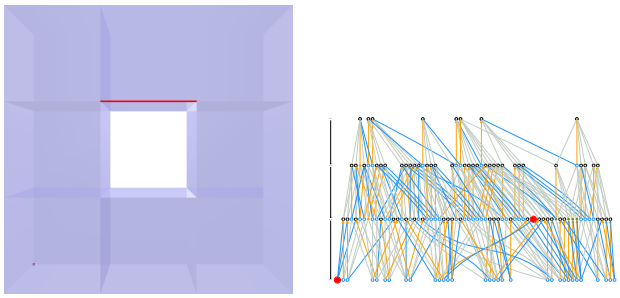


Figure 16: *A ring made of 8 cubes and its Hasse diagram.*

The Hasse diagram represents every cell by one node. Red nodes represent critical cells. Non-regular hyperlinks of the hypertrees are drawn in green. The cells of same dimension are displayed on the same row. The rows are ordered decreasingly on the dimension.

A link between two nodes symbolizes that the corresponding cells are incident one to the other. We linked by a red line paired cells. Blue lines represent the incidences we selected for the hypertrees and the final graph, as we did in the algorithm (see section 4(d) *Selecting cells of the dual hypertree*).

(c) Topologically controlling a deformation

Morse theory studies the topology of an object by its critical points. Another way to analyze it is provided by the handlebody theory [26, 22]. This theory constructs an object by successively attaching handles to a disc. The addition of a critical point corresponds to a handle attachment. Forman provides a similar result as introduces in section 2(e) *Homotopy properties*. This allows to describe a complex as glueing a few number of cells (the critical ones) *without any geometrical test*.

For example Figure 17 represents a decomposition of a torus with 25600 cells in 4 critical cells: 1 face, 2 edges and 1 vertex. This result can be interpreted as follow. The first cell removed is the critical face, leading to a punctured torus (a red face in the meridian on Figure 17(a)). Then, the torus deformation along decreasing height (color) until it reaches one of the two critical edges, leading to Figure 17(b). Repeating the deformation until the second critical edge is reached, we get Figure 17(c), which is a disk. We continue retracting until having reduced all the faces (Figure 17(d)), and continue reducing the remaining tree until reaching the critical vertex.

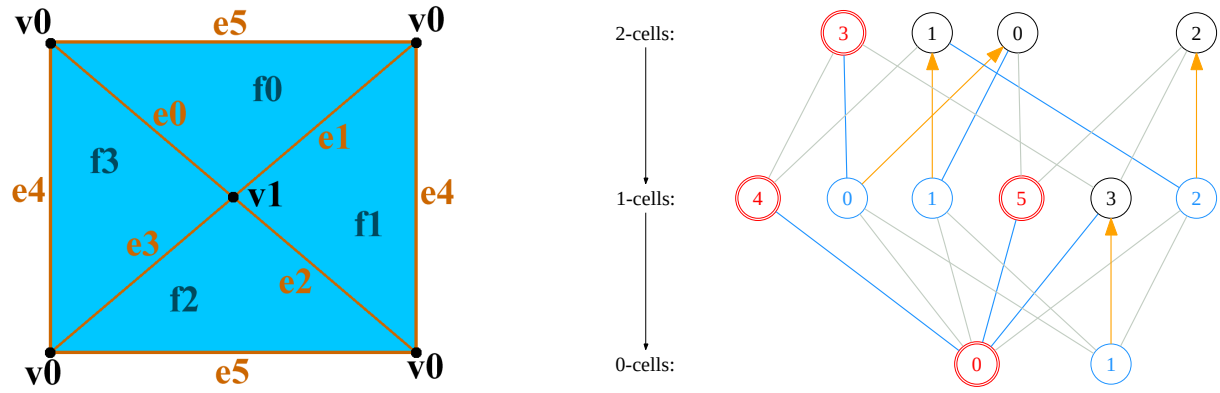


Figure 15: *A non-PL torus and its Hasse diagram.*

6 Future Works

We intended by this work to illustrate Forman's theory, and to use some of its concepts to visually analyze the topology of an object. We presented an explicit construction of a discrete gradient vector field. With this fundamental tool, we provided various ways of using it to visually extract topological information of a given combinatorial structure.

An important application of this work to computer graphics would be in the field of geometric compression. The algorithm Grow&Fold of A. Szymczak and J. Rossignac [28] could be justified and enhanced by our algorithm to minimize the number of so-called "glue faces" in order to achieve a better encoding. This work has been done optimally for the case of surfaces with handles in [24].

We plan to continue this work in three directions. Firstly, we intend to apply the auditory method designed by Axen and Edelsbrunner in [2], together with Forman's tools we provided. This would give a more sensitive way of studying higher-dimensional cell complexes. Secondly, we will try to develop graphical tools to capture as much as possible the topology 3-manifolds, where very hard mathematical problems remain unsolved. Finally, we look forward to produce a topologically consistent morphing based on mapping directly the discrete gradient field between two objects of the same homotopy type.

References

[1] M. A. Armstrong. *Basic topology*. McGraw–Hill, London, 1979.

[2] U. Axen and H. Edelsbrunner. Auditory Morse analysis of triangulated manifolds. In H.-C. Hege and K. Polthier, editors, *Mathematical Visualization*, pages 223–236. Springer, Berlin, 1998.

[3] C. Berge. *Graphes et hypergraphes*. Dunod, Paris, 1970.

[4] M. K. Chari. On discrete Morse functions and combinatorial decompositions. *Discrete Mathematics*, 217:101–113, 2000.

[5] M. K. Chari and M. Joswig. Discrete Morse complexes. In *ACCOTA: Workshop on Combinatorial and Computational Aspects of Optimization, Topology and Algebra*, San Crist, Dec. 2002.

[6] C. J. A. Delfinado and H. Edelsbrunner. An incremental algorithm for Betti numbers of simplicial complexes. In *Symposium on Computational Geometry*, pages 232–239. ACM, 1993.

[7] T. K. Dey, H. Edelsbrunner and S. Guha. Computational topology. In B. Chazelle, J. E. Goodman and R. Pollack, editors, *Advances in Discrete and Computational Geometry*, volume 223 of *Contemporary mathematics*, pages 109–143. AMS, 1999.

[8] T. K. Dey and S. Guha. Algorithms for manifolds and simplicial complexes in euclidean 3–space. In *Symposium on Theory of Computing*, pages 398–407, 2001.

[9] H. Edelsbrunner, J. L. Harer and A. Zomorodian. Hierarchical Morse complexes for piecewise linear 2–manifolds. In *Symposium on Computational Geometry*, pages 70–79. ACM, 2001.

[10] Ö. Eğecioğlu and T. F. Gonzalez. A computationally intractable problem on simplicial complexes. *Computational Geometry*, 6:85–98, 1996.

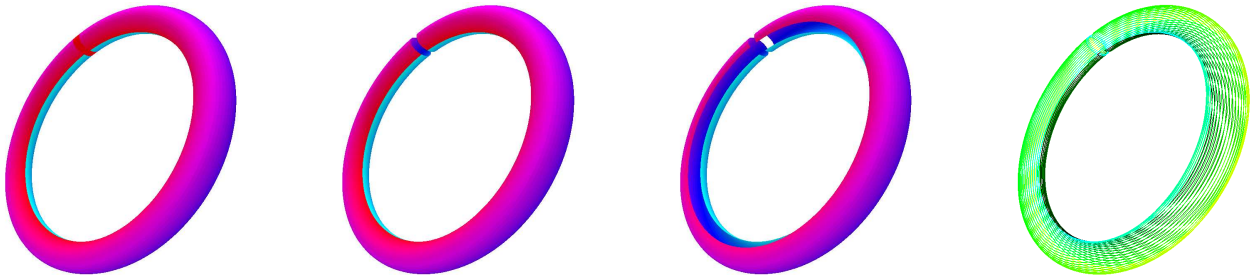
[11] R. Forman. A discrete Morse theory for cell complexes. In S. T. Yau, editor, *Geometry, Topology and Physics for Raoul Bott*, pages 112–115. IP, 1995.

[12] R. Forman. Morse theory for cell complexes. *Advances in Mathematics*, 134:90–145, 1998.

[13] R. Forman. Some applications of combinatorial differential topology. In *Sullivan Fest*, 2001.

[14] R. Forman. A user guide to discrete Morse theory. In *Séminaire Lotharingien de Combinatoire*, volume 48, 2001.

The corresponding work was published in Visualization and Mathematics III, pp. 95–112. Springer, 2002..



(a) The gradient vector field.

(b) First step: opening the torus along its meridian.

(c) Second step: opening the torus along its equator.

(d) Further steps continue on edges and vertices.

Figure 17: *A decomposition of a torus.*

[15] S. C. North. Neato user's guide. Technical report, AT&T Bell Laboratories, Murray Hill, NJ, 1992.

[16] E. Koutsofios and S. C. North. Drawing graphs with dot. Technical report, AT&T Bell Laboratories, Murray Hill, NJ, 1993.

[17] M. Hachimori. Simplicial complex library. infos-hako.sk.tsukuba.ac.jp/~hachi.

[18] J. C. Hart. Morse theory for implicit surface modeling. In H.-C. Hege and K. Polthier, editors, *Visualization and Mathematics*, pages 257–268, Heidelberg, 1998. Springer.

[19] J. C. Hart. Computational topology for shape modeling. In *Shape Modeling International*, pages 36–45, Japan, 1999. IEEE.

[20] R. Scharein. Knot–plot. www.pims.math.ca/knotplot/.

[21] T. Lewiner, H. Lopes and G. Tavares. Optimal discrete Morse functions for 2–manifolds. *Computational Geometry*, 26(3):221–233, 2003.

[22] H. Lopes. Algorithm to build and unbuild 2 and 3 dimensional manifolds. PhD thesis, *Department of Mathematics, PUC–Rio*, 1996. Advised by Geovan Tavares.

[23] H. Lopes and G. Tavares. Structural operators for modeling 3–manifolds. In C. Hoffman and W. Bronsvort, editors, *Solid Modeling and Applications*, pages 10–18. ACM, 1997.

[24] H. Lopes, J. Rossignac, A. Safonova, A. Szymczak and G. Tavares. Edgebreaker: a simple compression for surfaces with handles. In C. Hoffman and W. Bronsvort, editors, *Solid Modeling and Applications*, pages 289–296, Saarbrücken, Germany, 2002. ACM.

[25] A. T. Lundell and S. Weingram. *The topology of CW–complexes*. Van Nostrand Reinhold, New York, 1969.

[26] J. W. Milnor. *Morse theory*. Number 51 in *Annals of Mathematics Study*. Princeton University Press, 1963.

[27] Y. Shinagawa, T. L. Kunii and Y. L. Kergosien. Surface coding based on Morse theory. *Computer Graphics and Applications*, 11:66–78, 1991.

[28] A. Szymczak and J. Rossignac. Grow & Fold: compressing the connectivity of tetrahedral meshes. *Computer–Aided Design*, 32(8/9):527–538, 2000.

[29] G. Vegter. Computational topology. In J. E. Goodman and J. O'Rourke, editors, *Handbook of Discrete and Computational Geometry*, pages 517–536. CRC Press, 1997.