Discrete dynamical system framework for construction of connections between critical regions in lattice height data

Madjid Allili*, David Corriveau, Sara Derivière, Tomasz Kaczynski*, and Anik Trahan

January 23, 2007

Abstract

We define a new mathematical model for the topological study of lattice height data. A discrete multivalued dynamical system framework is used to establish discrete analogies of a Morse function, its gradient field, and its stable and unstable manifolds in order to interpret functions numerically given on finite sets of pixels. We present efficient algorithms detecting critical components of a height function f and displaying connections between them by means of a graph, called the *Morse connections graph* whose nodes represent the critical components of f and edges show the existence of connecting trajectories between nodes. This graph encodes efficiently the topological structure of the data and makes it easy to manipulate for subsequent processing.

1 Introduction

Height fields are used to represent data in many application areas such as geographic information systems (GISs) where the height corresponds to the elevation of a terrain, in imaging with pixel intensity values as the height, and more generally in scientific visualization where height functions can be

*supported by the NSERC of Canada discovery grant

arbitrary and do not necessarily measure height. In each case, the goal remains to extract features that enhance our understanding of the measured data and allow us to represent the data in a structure that is easy to manipulate for subsequent processing. Researchers in these areas have extensively studied methods of extracting the height field features such as critical points (peaks, pits, passes), and ridge and ravine lines from height field data over a discrete rectangular lattice. Once the critical points are extracted, different techniques are used to analyze relationships between them and trace structures such as ridge and ravine lines, watersheds, and level contour changes with respect to the height. The relations among critical points are usually represented in what is known as topological graphs that partially capture the topological structure of isolines and allow to handle the critical point information in an abstract way. Among well known topological graphs there are the *Reeb graph* [21, 6] and the *contour tree* [2].

In this paper, we introduce a new mathematical model derived from the Morse theory for the study of lattice height data. The classical Morse theory relies on strong assumptions of smoothness, that is, C^∞ or, at least, C^2 differentiability of studied functions, and on isolation and non-degeneracy of its critical points. In many applications of that theory to imaging science, one spends a lot of effort on forcing those assumptions by local deformation of data, so to adjust the finite input to the theory, with the aim at validating practical implementations. Among most systematic studies of that kind one could point the work of Edelsbrunner *et al.* [4] who presented an algorithm constructing a hierarchy of Morse-Smale complexes that decompose a piecewise-linear 2-manifold. There is also an extensive work in artificial vision related to the watershed model, see [12, 20, 9]. That model uses ideas borrowed from the fields of geomorphology and hydrology where the main concern in the last century and the beginning of the present is the study of the drainage patterns of landscapes, directly related to their ridges and valleys. A 2-dimensional image can be viewed as the sampling of a landscape and after smoothing operations, classical differential techniques are applied to extract the topographic structure of the image consisting of critical points (maxima, minima, and saddles) and their regions of influence called districts or more specifically hills for maxima and basins for the minima. Watersheds are defined as slopelines separating basins, and watercourses as slopelines separating hills, which are a subset of the separatrices that are paths between saddle points and the extrema. In contrary to the previous approaches, we decided to directly work on a simple discrete model. Note that our discretization of the Morse function does not correspond to the discrete Morse theory introduced by Forman [7]. More precisely, we define a discrete multivalued dynamical system framework to establish discrete analogies of a Morse function, its gradient field, and its stable and unstable manifolds in order to interpret functions numerically given on finite sets of pixels. In our approach, the arguments are not points in the Euclidean plane but pixels, geometrically interpreted as unit-size squares. The discrete multivalued dynamical system (an analogy of a flow) is generated by a map sending a pixel to a set of pixels (an analogy of a gradient field).

While Morse theory requires that only isolated points can be critical, general data sets may contain entire regions that are critical. So, we extend the concept of critical point to that of a *critical component*. This concept is a discrete analogy of critical regions introduced in [24] in the context of continuous functions. Roughly speaking, any given position or component of the lattice data set can be classified as critical or regular by considering a small neighborhood around it and counting the number of regions with larger values as well as the number of regions with smaller values than the value at the considered location. The advantage of considering critical components is the ability to handle general data, but the drawback is that the well-known Euler formula is not necessarily satisfied. The Euler formula, also known as the Mountaineer's equation wax first introduced by Maxwell [14], based on purely empirical observations about terrains. Maxwell proposed the following relation between peaks, pits, and passes: $pits - passes + peaks = 2$, for a terrain equivalent to a closed surface. This formula was later proved by Morse [18] using differential topology. It is used (sometimes enforced) in a number of works about digital elevation models (DEM) [22] as a criterion of correctness of the extracted critical points and a guaranty that the methods do not incur spurious critical points. However strong assumptions about the non-degeneracy and isolation of critical points are made. In addition, the DEM model is put on the top of a sphere and the bottom pit of the sphere is taken into account in the Euler formula. Conventional methods of extracting critical points directly from a raster based DEM [8] in which the classification of a raster point depends on the relative elevations of its eight direct neighbors do not maintain the Euler-Poincare formula. Our method deals directly with the raster data and is also based on the eight neighbor classification. So, it is not surprising that it does not preserve the Euler number formula. However, using the concept of critical components allows us to get rid of most if not all unwanted and non essential critical points extracted originally with the eight neighbor method.

We associate the analogous concepts of stable and unstable manifolds with the detected critical components by determining trajectories of the discrete dynamical system through each pixel and labeling it accordingly to the critical components connected by the trajectories in forward and backward times. We define two critical components to be connected if there exists an upward or downward trajectory connecting them.

At this point we should emphasize an important consequence of using the discrete model rather than the classical Morse theory. In the classical approach, we have a smooth gradient field ∇f that represents the direction of the steepest ascent. The direction of the steepest descent is the opposite, $-\nabla f$. This is not the case in the discrete dynamics on pixels. As a consequence, the trajectory of the steepest descent passing through a given pixel (if there is a unique one) is not necessarily equal to the trajectory of the steepest ascent run in the backward time. Thus we need to use the concept of discrete multivalued semidynamical system (DMSS), that is, a discrete-time, point-to-set, and time-non-invertible dynamical system and we need to distinguish two types of connections between critical components: those realized by upward (ascending) trajectories and those realized by the downward (descending) ones. The DMSS model allows to deal naturally with a variety of data sets. It allows to construct efficient algorithms detecting and classifying critical components of a function f and displaying connections between them by means of a graph, called the *Morse connections graph* (see [3]) whose nodes represent the critical components of f and edges show the existence of a connecting trajectory between nodes. A novelty with respect to the Morse connections graph studied in [3] is, first, that vertices correspond now to critical components, and second, that we have two types of connections, the upward and backward ones.

This graph structure is extremely well suitable to synthesize the regions of topological interest of the lattice height data and the relationships between them. Readers familiar with the Reeb graph [21, 6] will observe many similarities. The Reeb graph considers the equi-height contours of the height data and records any change occurring in their topology as the height increases. It tracks the number of contours in each cross-section of the height data and records any time when a contour appears, disappears or when any two contours merge or split. These changes in the topology of the contours are related to the presence of critical points of f. To compare the two graphs, the nodes of the Morse connections graph (critical points of f) and the nodes of the Reeb graph are in fact the same. The difference will be in the way the arcs between the nodes are established. For the Reeb graph, an arc is created between two nodes when the associated loops belong to the same connected component. For the Morse connections graph, an arc is created when there exists a connecting trajectory between two critical components. The Morse connections graph provides more information than the Reeb graph. In addition to encoding the regions where the changes of the topology of the isolines of the height field data occur, the Morse connections graph gives directly the detail about the ridge and ravine lines and the region of influence (watersheds) of each critical component. One can also add the information about the type of connection occurring between two given nodes as well as the information about the stable and unstable manifolds associated to the nodes.

The paper is organized as follows. In Section 2 we present the mathematical model: we start from the background on the classical Morse theory, we next introduce the concept of critical components for discrete data, and then discuss the multivalued discrete semidynamical system pertaining to the definition of the Morse connections graph. In Section 3, we give details on the algorithms used to compute the Morse connections graph and graphically present results of several experiments. Section 4 concludes the paper.

2 Mathematical model

2.1 Background from the classical Morse theory

The classical Morse theory for smooth functions is an important tool to investigate the topology of manifolds. This theory shows that the topology depends on the critical points (and their connections) of certain smooth functions defined on the manifold.

To be more explicit, let M be a smooth manifold and $f : M \to \mathbb{R}$ a smooth function. A point $p \in M$ is *critical* if the gradient ∇f vanishes at p and f is called a *Morse function* if all of its critical points p are *non degenerate*, i.e. if the determinant of the *Hessian* of f given by

$$\det H_f = \det \left(\partial^2_{x_i x_j} f \right)$$

does not vanish at p.

Given a Morse function f, the index of any critical point p, denoted by $\lambda(p)$, is the number of negative eigenvalues of the Hessian $H_f(p)$. This number corresponds to the dimension of the stable manifold

$$W^s(p) = \left\{ q \in M \mid \lim_{t \to \infty} \varphi(q,t) = p \right\},$$

where $\varphi : M \times \mathbb{R} \to M$ denotes the (positive) gradient flow associated with f on M. By contrast, the unstable manifold of p is

$$W^u(p) = \left\{ q \in M \mid \lim_{t \to -\infty} \varphi(q,t) = p \right\},$$

whose dimension is exactly $\dim(M) - \lambda(p)$. When f is interpreted as a height field, the unstable manifold of a critical point p corresponds to the *hill* of p and its stable manifold is the *basin of attraction* of p, that is, the set of points having p as the sink. The notions of stable and unstable manifolds play a very important role in understanding the interplay between the geometry of a manifold and dynamics of a Morse function. Indeed, let us consider the flow φ associated to a smooth Morse function f. Every regular (non-critical) point of f lies on a uniquely defined trajectory. This trajectory is said to *start* at a critical point p when, going backward in time, the limit of the points on the trajectory as $t \to -\infty$ is p. Similarly, the trajectory *ends* at a critical point q when, going forward in time, the limit as $t \to +\infty$ is q. Let us say that for a given regular point r, its trajectory starts and ends at the critical points p and q. Clearly, the point r belongs to the unstable manifold of p, $W^u(p)$, since its trajectory gets away of p as the t parameter of the flow increases. Similarly, the point r belongs to the stable manifold of q, $W^s(q)$, since its trajectory approaches q as the t parameter of the flow increases. Thus we may say that this trajectory is *connecting* the critical points p and q. Moreover, for any two distinct critical points p and q of a Morse function f, we have the following property

$$W^u(p) \cap W^u(q) = \emptyset \text{ and } W^s(p) \cap W^s(q) = \emptyset. \tag{1}$$

In addition, the stable (or unstable) manifolds of critical points in M form a covering of the whole M, that is, if $\{p_1, \ldots, p_k\}$ represents the set of all critical points of a Morse function $f : M \to \mathbb{R}$,

$$\bigcup_{i=1}^{k} W^s(p_i) = M \text{ and} \tag{2}$$

$$\bigcup_{i=1}^{k} W^u(p_i) = M. \tag{3}$$

Finally, we discuss the role played by level sets. We denote by M_t the submanifold of M, called the lower level set of f, defined by

$$M_t := \{p \in M \mid \ f(p) \leq t\}.$$

One of main ideas of the Morse theory is to study the change of the shape of M_t as t increases. Indeed, each time t passes through a critical value, topological classification of M_t changes accordingly to the Morse index of the corresponding critical point. For more details on this theory, see [13].

2.2 Critical components

In practice, the values of the Morse function f are only known on a set of points that corresponds to a finite set of pixels, denoted by X. Thus, $f : X \to \mathbb{R}$ is a discrete function. We interpret pixels $x \in X$ as unit squares of the form $x = [k, k+1] \times [l, l+1]$, k, l integers, for a chosen grid scale. Given any $A \subset X$ the *support* of A is the set $|A| \subset \mathbb{R}^2$ given by

$$|A| = \bigcup A.$$

Note that, while A is a finite set of pixels, $|A|$ has the continuum of points: it is the corresponding closed region in the plane. Thus the association $A \mapsto |A|$ provides the passage from combinatorics to geometry.

Definition 2.1 Let x and y be pixels in X. We say that x and y are *adjacent* or *0-adjacent*, denoted by $x0y$, if $x \cap y$ contains a vertex. They are called *edge adjacent* or *1-adjacent*, denoted by $x1y$, if $x \cap y$ contains an edge.

A set of pixels $A \subset X$ is called *connected* if, for any $x, y \in A$, there exists a sequence of pixels in A $x_1 = x,\ x_2, \cdots,\ x_n = y$ such that x_i and x_{i+1} are 0-adjacent, $i = 1, \cdots, n-1$. A set $A \subset X$ is called *edge-connected* or 1-*connected* if, for any $x, y \in A$, there exists a sequence of pixels in A $x_1 = x,\ x_2, \cdots,\ x_n = y$ such that x_i and x_{i+1} are 1-adjacent, $i = 1, \cdots, n-1$.

A *connected component* of A is a maximal connected subset of A. The number of connected components of A is denoted by $cc(A)$. Similarly, an *edge-connected component* of A is a maximal edge-connected subset of A and the number of edge-connected components of A is denoted by $cc_1(A)$.

Note that "1-adjacent" implies "0-adjacent" and "1-connected" implies "0-connected". Moreover, a set of pixels A is connected if and only if its support $|A|$ is connected in the topological sense. The set of pixels A is 1-connected if and only the interior of its support $\mathrm{Int}\,|A|$ is a connected open region in the plane.

$$\begin{array}{cc} \mathrm{x} & \mathrm{y} \\ \mathrm{z} & \mathrm{t} \end{array} \qquad\qquad \begin{array}{ccc} 0 & \mathbf{1} & 0 \\ \mathbf{1} & 0 & \mathbf{1} \\ 0 & \mathbf{1} & 0 \end{array}$$

(a) (b)

Figure 1: (a) Adjacency relations: $x1y$, $x1z$, $y1t$, $z1t$, $x0t$, $y0z$; (b) The set A of pixels with the value 1 is connected but it is not edge-connected and $cc_1(A) = 4$.

Definition 2.2 The *distance* between two adjacent pixels is measured by the distance between their centers. More explicitly, given $x, y \in X$ two adjacent pixels:

$$dist(x, y) = \begin{cases} \sqrt{2} & \text{if } x \cap y \text{ is a vertex,} \\ 1 & \text{if } x \cap y \text{ is an edge.} \end{cases}$$

The *directional derivative* of f at x in the direction of an adjacent pixel y is

$$\partial_y f(x) := \frac{f(y) - f(x)}{dist(x, y)}.$$

Definition 2.3 Let X be a set of pixels. The natural concept of a *neighborhood* of a pixel x is the wrap of x, denoted by $wrap(x)$, that consists of x together with all adjacent pixels. The *boundary* of $x \in X$, $bd(x)$, is defined as:

$$bd(x) = wrap(x) \setminus \{x\}.$$

We denote by $\overline{wrap}\,(x)$ the set of all pixels in $wrap(x)$ with higher values of f than x, that is

$$\overline{wrap}\,(x) := \{y \in bd(x) \mid \partial_y f(x) > 0\}\,,$$

and by $\underline{wrap}\ (x)$ the set of all pixels in $wrap(x)$ with lower values of f than x, that is

$$\underline{wrap}\ (x) := \{y \in bd(x) \mid \partial_y f(x) < 0\}\,.$$

We define the set $wrap_{\,0}(x)$ by:

$$wrap_{\,0}(x) := \{y \in\ bd(x) \mid \partial_y f(x) = 0\}\,.$$

Hence, $|bd(x)|$ represents a square composed by 8 pixels around x whereas $|wrap(x)|$ is the full square of 9 pixels. If A is a subset of X, the notions of *wrap* and *boundary* are trivially extended to $wrap(A)$ and $bd(A)$. The extension of $\underline{wrap}$, $\overline{wrap}$, and $wrap_{\,0}$ to sets is more delicate:

$$\overline{wrap}\ (A) := \{y \in bd(A) \mid f(y) > f(x) \text{ for all } x \in A \cap bd(y)\}\,,$$

$$\underline{wrap}\ (A) := \{y \in bd(A) \mid f(y) < f(x) \text{ for all } x \in A \cap bd(y)\}\,.$$

$$wrap_{\,0}(A) := bd(A) \setminus \big(\overline{wrap}\ (A) \cup \underline{wrap}\ (A)\big)$$

Using these definitions, we introduce various types of pixels and critical components.

Definition 2.4 Given a function $f : X \to \mathbb{R}$, a pixel x is

regular	$\Longleftrightarrow$	$\overline{wrap}\ (x)$ and $\underline{wrap}\ (x)$ are non-empty and 1-connected;
minimum	$\Longleftrightarrow$	$\overline{wrap}\ (x) = bd(x)$;
maximum	$\Longleftrightarrow$	$\underline{wrap}\ (x) = bd(x)$;
k-saddle	$\Longleftrightarrow$	$wrap_{\,0}(x) = \emptyset$ and $cc_1(\overline{wrap}\ (x)) = k+1,\ k \geq 1$.

A pixel x is called *unclassified* if none of the above holds.

Remark that a pixel is unclassified if and only if it is non-regular and $wrap_0(x) \neq \emptyset$.

Note that an equivalent condition for a minimum at x is that $\underline{wrap}\ (x) = wrap_{\,0}(x) = \emptyset$. Similarly the condition for a maximum is that $\overline{wrap}\ (x) = wrap_{\,0}(x) = \emptyset$. A k-saddle is a generalization of so called *monkey saddle* considered in the smooth theory. It is easy to check that if x is a k-saddle then edge-connected components of $\overline{wrap}\ (x)$ separate $\underline{wrap}\ (x)$, so we also have $cc_1(\underline{wrap}\ \ (x)) = k+1$. Geometrically, a k-saddle pixel has $k+1$ ascending directions and the same number of descending ones. Since $bd(x)$

contains 8 pixels, k-saddle pixels may occur for $k \leq 3$. Many cases where a pixel x is at the junction of diagonal isolines crossing diagonally would not be detected as a saddle, because then $wrap_0(x) \neq \emptyset$. This saddle would only be grasped by the next definition of a critical component and, at this moment, it is unclassified. Unclassified pixels also include so called *flat points* or *slabs* where $wrap_0(x) = bd(x)$, non-isolated minimum pixels, and non-isolated maximum pixels.

We would like to define a critical component as connected components of pixels with the same values of f as a given non-regular pixel, however, several problems are encountered:

1. Two saddles with distinct values may be adjacent. This is illustrated by Fig. 2-(a) where the center pixels with values 1 and 3 are both saddles;

2. Adjacent non-regular pixels with the same value of f may form together a connected level component which acts on a surrounding area as a regular pixel. This is shown on Fig. 2-(b) where the two middle pixels with the value 0 are unclassified: The one on the left could be described as a "non-isolated saddle", and the one on the right as a "non-isolated maximum". This also shows why do we require $wrap_0(x) = \emptyset$ in the definitions of saddle, maximum and minimum pixels;

3. By considering the connected component of a saddle or unclassified pixel, we may obtain a saddle together with all its merging isolines. But the remaining pixels on the isolines may be regular and we don't want to count them as critical, see Fig. 2-(c).

Definition 2.5 Let X be a finite set of pixels. Given $\mathcal{C} \subset X$, we adopt the following notation for the numbers of edge connected components:

$$n_p = cc_1(\overline{wrap}\,(\mathcal{C})),\ n_n = cc_1(\underline{wrap}\,(\mathcal{C})),\ n_z = cc_1(wrap_0(\mathcal{C})).$$

Consider a function $f : X \to \mathbb{R}$ and a non-regular pixel $x \in X$. The component $\mathcal{C}(x) = \mathcal{C}$ is defined and constructed by the following algorithm:

Algorithm 2.6 Detecting and classifying critical components in $\mathbb{R}^2$

$\mathcal{C} := \{x\}$

while $y \in bd(\mathcal{C}) \cap X$ *is non-regular*

												0	1	2	1	0
2	0	4	0		1	1	1	1				-1	0	1	0	-1
2	**1**	**3**	4		-1	**0**	**0**	1				-2	-1	**0**	-1	-2
0	4	0	4		-1	1	1	1				-1	0	1	0	-1
												0	1	2	1	0
	(a)					(b)								(c)		

Figure 2: (a) Adjacent center pixels with values 1 and 3 are both saddles and they form a component which is not a level set of f but it has a property of a 4-saddle; (b) Adjacent non-regular pixels with the value 0. The one on the left is of a "non-isolated saddle" type, the one on the right is of a "non-isolated maximum" type; (c) The pixel 0 in the center is unclassified, but the component $\{0\}$ is a saddle, whose value is the same as on its crossing isolines.

  $\mathcal{C} := C \cup \{y\}$
  compute $bd(\mathcal{C})$
endwhile
do
  if $n_p = n_n = 1$ **then** $\mathcal{C}$ *is a regular component* **break**
    else if $n_p = 0$ **then** $\mathcal{C}$ *is a maximum component* **break**
    else if $n_n = 0$ **then** $\mathcal{C}$ *is a minimum component* **break**
    else $\mathcal{C}$ *is a saddle component* **break**
  endif
break

A set $\mathcal{C}$ is a *critical component* if it is either a minimum, maximum or a saddle component of a non-regular pixel. A pixel $x \in X$ is called a *critical pixel* if $\mathcal{C}(x)$ is a critical component.

Remark 2.7 Unlike in the case of a k-saddle pixel, a saddle component $\mathcal{C}$ may fail the equality $n_p = n_n$ because it may have a non-trivial topology of $\mathcal{C}$, see Fig. 3. This figure also shows that by considering a critical component as single object (a vertex in the Morse connections graph) we may loose account of some topological information.

-1	-1	1	1	1	-1	-1
-1	**0**	**0**	**0**	**0**	**0**	-1
1	**0**	1	1	1	**0**	1
1	**0**	1	2	1	**0**	1
1	**0**	1	1	1	**0**	1
-1	**0**	**0**	**0**	**0**	**0**	-1
-1	-1	1	1	1	-1	-1

Figure 3: The critical component $\mathcal{C}$ composed by the pixels with the value 0 value form a saddle component with $n_p = 5$ and $n_n = 4$. Topologically, $\mathcal{C}$ is equivalent to a circle.

2.3 Dynamical systems

We adapt several concepts of the theory of dynamical systems to describe the dynamics induced by a discrete function $f : X \to X$. As previously, X stands for a finite set of pixels and $\mathbb{Z}$ stands for the set of all integers. We introduce a generalization of a flow $\varphi : X \times \mathbb{R} \to X$. The variable $n \in \mathbb{Z}$ appearing below is interpreted as the (discrete) time.

Definition 2.8 A map $\Phi : X \times \mathbb{Z} \rightrightarrows X$ whose values are non empty subsets of X is called a *discrete multivalued dynamical system (DMDS)* on X if the following conditions are satisfied (see [11]):

1. For all $x \in X$, $\Phi(x, 0) = \{x\}$;
2. For all $n, m \in \mathbb{Z}$ with $nm > 0$ and all $x \in X$;
   $\Phi(\Phi(x, n), m) = \Phi(x, n + m)$;
3. For all $x, y \in X$, $y \in \Phi(x, -1) \iff x \in \Phi(x, 1)$.

A set-valued map $\Phi : X \times \mathbb{Z} \rightrightarrows X$ is called a *discrete multivalued semi-dynamical system (DMSS)* if it satisfies (1) to (3) but the condition $\Phi(x, n) \neq \emptyset$ is only assumed for $n > 0$.

The map $\mathcal{F} = \Phi(\cdot, 1) : X \rightrightarrows X$ is called the *generator* of the system. Conversely, given any multivalued map $\mathcal{F} : X \rightrightarrows X$, we can define the corresponding DMSS $\Phi : X \times \mathbb{Z} \rightrightarrows X$ by two-sided induction. First, we put

$\Phi(x,0) := \{x\}$. For $n = 1$, we let

$$\Phi(x,1) := \mathcal{F}(x) \text{ and } \Phi(x,-1) := \mathcal{F}^{-1}(x),$$

where $\mathcal{F}^{-1}(x) := \{y \in X \mid x \in \mathcal{F}(y)\}$. For $n > 1$ we let

$$\Phi(x,n) := \mathcal{F}^n(x) := \mathcal{F}\left(\mathcal{F}^{n-1}(x)\right)$$

and

$$\Phi(x,-n) := \mathcal{F}^{-n}(x) := \mathcal{F}^{-1}\left(\mathcal{F}^{-(n-1)}(x)\right).$$

Thus $\mathcal{F}$ is the generator of Φ. Moreover, if $\mathcal{F}^{-1}$ has non empty values, Φ becomes a DMDS. A generator $\mathcal{F}$ of a DMDS is a discrete analogy of a vector field generating a flow. In texts on topological DMDS one assumes that X is a locally compact space, and $\mathcal{F}$ and $\mathcal{F}^{-1}$ are upper semi-continuous with compact values but we consider DMDS on discrete sets so we do not need those assumptions, and the only property distinguishing generators of DMDS and DMSS in our case is $\mathcal{F}^{-1}(x) \neq \emptyset$ for all $x \in X$.

Definition 2.9 Let I be an interval in $\mathbb{Z}$, that is, intersection of a real interval (bounded or not) with $\mathbb{Z}$, and let $\mathcal{F}$ be a generator of a DMSS. Consider $\{x_n\}_{n\in I} \subset X$. The sequence $(x_n)_{n\in I}$ is a *trajectory* of $\mathcal{F}$, if

$$x_{n+1} \in \mathcal{F}(x_n)$$

for all $n \in I$ with $(n, n+1) \subset I$. If $x \in X$, $0 \in I$, and $x_0 = x$, we say that $(x_n)_{n\in I}$ is a *trajectory of* x_0 or a *trajectory passing through* x_0 with respect to $\mathcal{F}$. When $I = \mathbb{Z}$, it is called a *full* trajectory.

Given a critical pixel x, we consider the decomposition of the support of $\overline{wrap}\ (x)$ to k connected components and that of $\underline{wrap}\ (x)$ to l connected components. This corresponds to the decomposition of pixels:

$$\overline{wrap}\ (x) = \bigcup_{i=1}^{k} \overline{wrap}\ _i(x)$$

and

$$\underline{wrap}\ (x) = \bigcup_{i=1}^{l} \underline{wrap}\ _i(x)$$

We put

$$m_i(x) = \min_{y \in \underline{wrap}\ _i(x)} \partial_y f(x) \quad \text{and} \quad M_i(x) = \max_{y \in \overline{wrap}\ _i(x)} \partial_y f(x).$$

Definition 2.10 Given $f : X \to \mathbb{R}$, the *upward system* is the DMSS generated by the map $\mathcal{F}_+ : \mathcal{X}^d \rightrightarrows \mathbb{R}$ defined on pixels $x \in \mathcal{X}^d$ as follows. Let first

$$\mathcal{F}_+^*(x) = \bigcup_{i=1}^{k} \{y \in \overline{wrap}_{\,i}(x) \mid \partial_y f(x) = M_i(x)\}\,.$$

If x is a regular pixel, we put $\mathcal{F}_+(x) = \mathcal{F}_+^*(x)$. If x is a critical pixel and $\mathcal{C}$ its critical component, we put $\mathcal{F}_+(x) = \mathcal{C} \cup \mathcal{F}_+^*(x)$.

The *downward system* is generated by the map $\mathcal{F}_- : X \rightrightarrows \mathbb{R}$ defined on pixels $x \in X$ as follows. Let first

$$\mathcal{F}_-^*(x) = \bigcup_{i=1}^{l} \left\{y \in \underline{wrap}_{\,i}(x) \mid \partial_y f(x) = m_i(x)\right\}.$$

If x is a regular pixel, we put $\mathcal{F}_-(x) = \mathcal{F}_-^*(x)$. If x is a critical pixel and $\mathcal{C}$ its critical component, we put $\mathcal{F}_-(x) = \mathcal{C} \cup \mathcal{F}_-^*(x)$.

Note that the inverses of $\mathcal{F}_+$ and $\mathcal{F}_-$ may have empty values (and these two maps are not mutual inverses), hence $\mathcal{F}_+$ and $\mathcal{F}_-$ generate a DMSS, not necessarily a DMDS.

Remark 2.11 Let $\mathcal{F}$ be either $\mathcal{F}_+$ or $\mathcal{F}_-$. Note that a pixel $x \in X$ is critical if and only if it is a *fixed point* of $\mathcal{F}$, that is $x \in \mathcal{F}(x)$.

2.4 Stable, unstable manifolds, and Morse connections graph

We shall define the analogies of stable and unstable manifolds discussed in the subsection 2.1, for discrete multivalued semidynamical systems. In general, we will not be able to get the property (1) nor (3), however, the covering property (2) for the stable manifolds of both $\mathcal{F}_+$ and $\mathcal{F}_-$ will still hold and that shall permit us obtaining at least a partial information on the topology of the system.

Definition 2.12 Let $\mathcal{F} : X \rightrightarrows X$ be a generator of a DMSS. The *stable* and *unstable manifolds* of a point $x \in X$ relatively to $\mathcal{F}$ are defined by

$$\begin{aligned} W^u(x, \mathcal{F}) &:= \textstyle\bigcup_{n\geq 1} \mathcal{F}^n(x); \\ W^s(x, \mathcal{F}) &:= \textstyle\bigcup_{n\geq 1} \mathcal{F}^{-n}(x). \end{aligned}$$

If $\mathcal{F}$ is clear from the context, we will just write $W^u(x)$, $W^u(x)$. The stable and unstable manifolds of a critical component P are defined by

$$\begin{array}{ll} W^u(P,\mathcal{F}) & := \bigcup_{x\in P} W^u(x); \\ W^s(P,\mathcal{F}) & := \bigcup_{x\in P} W^s(x). \end{array}$$

From now on, we suppose that X is a finite set of pixels and $\mathcal{F}$ stands either for $\mathcal{F}_+$ or $\mathcal{F}_-$ given in Definition 2.10. Then, the orbit of every trajectory $\{x_n\}_{n\in\mathbb{Z}}$ is a finite set. Moreover, the values of $f(x_n)$ are either increasing or decreasing depending if we choose $\mathcal{F}_+$ or $\mathcal{F}_-$. Thus, every trajectory becomes eventually constant as $t\to\pm\infty$ and it is enough to consider finite trajectories : $\{x_n\}_{n\in I}$ where I is a finite interval in $\mathbb{Z}$. By a shift of indices, we may assume that $I=\{0,1,\dots,n_0\}$ for some $n_0>0$.

Proposition 2.13 *Here alternative formulas for stable and unstable manifolds in terms of a DMSS $\Phi : X\times\mathbb{Z}\rightrightarrows X$ and its trajectories.*

$$\begin{array}{rcl} W^u(x,\mathcal{F}) & = & \bigcup_{n\geq 1}\Phi(x,n)=\Phi(x,\mathbb{N}) \\ & = & \bigcup\{\{x_n\}_{n\in I} \mid (x_n) \textit{ is a trajectory issued from } x, \\ & & \quad I \textit{ interval in } \mathbb{N}\}; \\ W^s(x,\mathcal{F}) & = & \{y \mid \exists n:\ x\in\Phi(y,n)\}; \\ & = & \bigcup\{\{x_n\}_{n\in I} \mid (x_n) \textit{ is a trajectory issued from } y \textit{ ending at } x, \\ & & \quad I \textit{ interval in } \mathbb{N}\}. \end{array}$$

Proposition 2.14 Let $\mathcal{F} : X\rightrightarrows X$ be a generator of a DMSS, p and q be pixels such that $W^u(p)\cap W^s(q)\neq\emptyset$. Then, there exists a connecting trajectory from p to q.

Proof: Let x be an intersection point: $x\in W^u(p)\cap W^s(q)$. The proof is given in two steps: we first show that there is a trajectory from p to x, and next that there exists a trajectory from x to q.

- As $x\in W^u(p)=\bigcup_{n\geq 1}\Phi(p,n)$, there exists $n_0\geq 1$ such that

$$x\in\Phi(p,n_0)=\mathcal{F}^{n_0}(p)=\mathcal{F}^{n_0-1}\left(\mathcal{F}(p)\right).$$

Put $x_0:=p$. Since $x\in\mathcal{F}^{n_0-1}\left(\mathcal{F}(p)\right)$, there exists $x_1\in\mathcal{F}(p)$ such that $x\in\mathcal{F}^{n_0-1}(x_1)=\mathcal{F}^{n_0-2}\left(\mathcal{F}(x_1)\right)$. Next, $x\in\mathcal{F}^{n_0-2}\left(\mathcal{F}(x_1)\right)$, so there exists

$x_2 \in \mathcal{F}(x_1)$ such that $x \in \mathcal{F}^{n_0-2}(x_2) = \mathcal{F}^{n_0-3}(\mathcal{F}(x_2))$. Using this argument, we obtain by induction the sequence

$$x_1 \in \mathcal{F}(p),\ x_2 \in \mathcal{F}(x_1),\ x_3 \in \mathcal{F}(x_2),\ \ldots,\ x_{n_0-1} \in \mathcal{F}(x_{n_0-2}),$$

such that $x \in \mathcal{F}(x_{n_0-1})$. Put $x_{n_0} = x$. By Definition 2.9, the sequence

$$(p, x_1, x_2, x_3, \ldots, x_{n_0-1}, x)$$

is a trajectory from p to x.

• Next, $x \in W^s(q)$, hence by definition of the stable manifold, there exists $n_1 \geq 1$ such that

$$q \in \Phi(x, n_1) = \mathcal{F}^{n_1}(x) = \mathcal{F}^{n_1-1}(\mathcal{F}(x)).$$

Now, $q \in \mathcal{F}^{n_1-1}(\mathcal{F}(x))$. Let n_0 be the index identified in the first step so that $x = x_{n_0}$. By the same arguments as the first step, we find a sequence of points

$$x_{n_0+1} \in \mathcal{F}(x),\ x_{n_0+2} \in \mathcal{F}(x_{n_0+1}),\ \ldots,\ x_{n_0+n_1-1} \in \mathcal{F}(x_{n_0+n_1-2}),$$

such that

$$q \in \mathcal{F}^{n_1-n_1+1}(x_{n_0+n_1-1}) = \mathcal{F}(x_{n_0+n_1-1}).$$

Thus, $(x, x_{n_0+1}, x_{n_0+2}, \ldots, x_{n_0+n_1-1}, q)$ is a trajectory from x to q.

By combining these two steps, we obtain a trajectory

$$(p, x_1, x_2, x_3, \ldots, x_{n_0-1}, x, x_{n_0+1}, x_{n_0+2}, \ldots, x_{n_0+n_1-1}, q)$$

from p to q. □

By the definition of stable and unstable manifolds of critical components, this result immediately generalizes as follows:

Corollary 2.15 Let $\mathcal{F} : X \rightrightarrows X$ be a generator of a DMSS, P and Q be critical components such that $W^u(P) \cap W^s(Q) \neq \emptyset$. Then there exists a connecting trajectory from P to Q, in the sense, that it connects a point in P to a point in Q.

Definition 2.16 Let P and Q be two critical components of $f : X \to \mathbb{R}$. We say there is an *upward connection* from P to Q, denoted $P \nearrow Q$, if

$$W^u(P, \mathcal{F}_+) \cap W^s(Q, \mathcal{F}_+) \neq \emptyset.$$

There is a *downward connection* from P to Q, denoted $P \searrow Q$, if

$$W^u(P, \mathcal{F}_-) \cap W^s(Q, \mathcal{F}_-) \neq \emptyset.$$

By the virtue of Corollary 2.15, these conditions are equivalent to the existence of a trajectory connecting a point in P to a point in Q respectively to the upward and downward map.

We show now that the covering property (2), for X instead of M, by the stable manifolds is satisfied for the upward and for the downward systems defined in Def. 2.10:

Theorem 2.17 *Let* $\{P_1, \ldots ; P_k\}$ *be the set of all critical components of* $f : X \to \mathbb{R}$. *Then*

$$\bigcup_{i=1}^{k} W^s(P_i, \mathcal{F}_+) = X = \bigcup_{i=1}^{k} W^s(P_i, \mathcal{F}_-).$$

PROOF: We only give the proof for the case of $\mathcal{F}_+$ because the case of $\mathcal{F}_-$ goes by the same arguments. Let x be a point in X. If x is critical, then it belongs to its stable manifold. Suppose that x is regular. Then, let $y_1 \in bd(x)$ be a pixel of maximum value such that $\partial_{y_1} f(x) > 0$ (else, x is a critical maximum). Then, $y_1 \in \mathcal{F}(x)$.

If y_1 is a critical point, $x \in W^s(y_1)$, else let $y_2 \in bd(y_1)$ be a pixel of maximum value such that $\partial_{y_2} f(y_1) > 0$. We get

$$y_2 \in \mathcal{F}(y_1) \subset \mathcal{F}(\mathcal{F}(x)) = \mathcal{F}^2(x).$$

If y_2 is critical, $x \in W^s(y_2) \subset \bigcup_{i=1}^k W^s(P_i)$, else let $y_3 \in bd(y_2)$ be a pixel of maximum value such that $\partial_{y_3} f(y_2) > 0$. Then, by induction, as X is a finite set, there exists $m < \infty$ such that

$$\begin{cases} y_1 \in \mathcal{F}_+(x), \\ y_l \in \mathcal{F}_+(y_{l-1}), \quad \forall l = 2, \ldots, m, \\ y_m \text{ is a critical point.} \end{cases}$$

We have $y_m \in \mathcal{F}_+^m(x) = \Phi(x, m)$. By Definition 2.12, this implies that

$$x \in W^s(y_m) \subset \bigcup_{i=1}^{k} W^s(P_i).$$

Since x is arbitrary, this show that every pixel in X is in $W^s(P_i)$ for some critical component P_i. $\square$

We can now define the Morse Connection Graph as follows:

Definition 2.18 The Morse Connections Graph $MCG_f = (V_f, E_f)$ is a graph whose vertices (or nodes) V_f and edges E_f are defined as follows:

$$\begin{aligned} V_f &= \{\text{critical components of } f\}\,; \\ E_f &= \{(P_i, P_j) \in V_f \times V_f \mid P_i \nearrow P_j \text{ or } P_i \searrow P_j\} \end{aligned}$$

Equivalently, (P_i, P_j) is an edge of the graph if

$$W^u(P_i, \mathcal{F}_+) \cap W^s(P_j, \mathcal{F}_+) \neq \emptyset \text{ or } W^u(P_i, \mathcal{F}_-) \cap W^s(P_j, \mathcal{F}_-) \neq \emptyset.$$

Remark 2.19 An additional information about the dynamics on X can be extracted by distinguishing two types of edges in Morse connections graph: The *upward connection edges* and the *downward connection edges* respectively defined by

$$E_f^+ = \{(P_i, P_j) \in V_f \times V_f \mid P_i \nearrow P_j\}$$

and

$$E_f^- = \{(P_i, P_j) \in V_f \times V_f \mid P_i \searrow P_j\}\,.$$

In the graphics presented in the next section we will display those edges, respectively, by dotted and dashed lines, while a connection both upward and downward will be exhibited by a solid line edge.

Computing the Morse connections graph requires finding efficient ways to determine the critical components and their connections, that is, their stable and unstable manifolds. The next section provides a discussion of algorithms and presentation of experimental results.

3 Computational results

3.1 Implementation details

In this section, we explain the implementation of the algorithm that builds the discrete multivalued dynamical system and its associated Morse connections graph. We consider an image, which can be first smoothed using a gaussian function to decrease the noise as needed. We assign the value of the grey-level to each pixel of this image. These values define a discrete function $f : X \to \mathbb{R}$ which our algorithm takes as input. The following data members are defined for pixels and components:

Pixel, component:

- `downFlow, upFlow`: sets of next pixels by iterating one step in downward direction and upward direction of the flow respectively.
- `invDownFlow, invUpFlow`: sets of adjacent pixels whose `downFlow` and `upFlow` contain the given pixel/component respectively.
- `downSetWu, downSetWs`: sets of critical pixels whose stable and unstable manifolds contain the given pixel/component in downward direction.
- `upSetWu, upSetWs`: sets of critical pixels whose stable and unstable manifolds contain the given pixel/component in upward direction.

For visualization and comparison purposes we present our algorithm in two stages. The first stage is the algorithm `ComputeMCGpixels` which computes the Morse Connections Graph by taking account of each critical pixel instead of its critical component in the Definitions 2.16 and 2.18. This algorithm gives the final result in case when critical pixels are isolated, that is, all critical components are singletons. When this is not the case, the output visualization will produce many vertices clustered together with multiple edges and a rather incomprehensible information. The second stage is the algorithm `ComputeMCGcomponents` which groups the critical pixels in components to form a new vertex of MCG and updates accordingly the information on edges. In visualization, the new vertex will be shown at the center of the critical component.

`ComputeMCGpixels()`

1. `Initialization of pixels`;

2. `Compute stable and unstable manifolds`;

3. `Determine Connections`.

`ComputeMCGcomponents()`

1. `Initialization of components`;

2. `Update stable and unstable manifolds`;

3. `Update Connections`.

Step 1 of `ComputeMCGpixels()`: This step initializes data members of all pixels. For each pixel $x \in X$, we compute the directional derivatives $\partial_y f(x)$ for all $y \in bd(x)$. If $\partial_y f(x) < 0$, then $y \in \underline{wrap}\,(x)$, if $\partial_y f(x) > 0$, then $y \in \overline{wrap}\,(x)$, and if $\partial_y f(x) = 0$, then $y \in wrap_{\,0}(x)$. Then, we determine all the 1-connected components of $\underline{wrap}\,(x)$ and $\overline{wrap}\,(x)$. For each connected component of $\underline{wrap}\,(x)$, we determine the pixel $\underline{y}$ with the minimal value of the directional derivative, and for each connected components of $\overline{wrap}\,(x)$, we determine the pixel $\overline{y}$ with the maximal value of $\partial_{\overline{y}}\, x$. Pixels with maximal value are included in the `upFlow` of x, and those with minimal value are included in the `downFlow` of x. We include x in `invUpFlow` of $\overline{y}$ and in `invDownFlow` of $\underline{y}$.

Then, pixels types are determined as follows:

- if x has only 1 pixel in its `upFlow`, and 1 pixel in `downFlow`, then x is a regular point,

- else if x has only 1 pixel in its `upFlow`, `downFlow` is empty and $n_z = 0$, then x is a minimum,

- else if x has only 1 pixel in its `downflow`, `upFlow` is empty and $n_z = 0$, then x is a maximum,

- else if there is more than 1 pixel in `upFlow` or `downFlow` and $n_z = 0$, then x is a saddle point,

- else x is unclassified.

Moreover, `downSetWu`, `downSetWs`, `upSetWu` and `upSetWs` are initialized to the empty set. When a non-regular pixel is found, we insert it in its `downSetWu`, `downSetWs`, `upSetWu` and `upSetWs`.

Pixels on the boundary are used to classify adjacent pixels but are not classified (as minimum, maximum, saddle, unclassified or regular), therefore no component can reach the boundary.

Step 2 of `ComputeMCGpixels()`: In this step, we compute separately the manifolds (stable and unstable) for the two cases: downward and upward directions.

(a) By decreasing values of the pixels, if x is a regular, a saddle or an unclassified pixel, we compute the unstable manifold of x by

$$\texttt{x.downSetWu} = \bigcup_{y \in \texttt{x.downFlow}} \texttt{y.downSetWu}.$$

For the stable manifolds, we proceed by increasing values.

(b) We proceed in a similar manner to calculate the stable and unstable manifolds of the upward flow.

Step 3 of `ComputeMCGpixels()`: Finally, we establish connections between non-regular pixels. We compute three Morse connections graphs according to the upward, downward directions and the union of both.

(a) We construct the downward connections graph. This is easily done in linear time by scanning all pixels x and connecting `downSetWu` of x to `downSetWs` of x which correspond to the non-empty intersections of stable and unstable manifolds of non-regular pixels in downward direction.

(b) Same as in (a) but using the upward flow.

Step 1 of `ComputeMCGcomponents()`: Using Algo 2.6, we compute the components of the systems. Then, for each component, we determine `downflow, upFlow, invDownFlow` and `invUpFlow` by following the procedure described in the first part of ComputeMCGpixels().

Step 2 of `ComputeMCGcomponents()`: To compute the stable and unstable manifolds of a non-regular component, we use the information obtained in the Step 2 of `ComputeMCGpixels()`: Let P(x) be a function that transforms a non regular pixel identifier into the identifier of the component to which it belongs. Then, the stable manifold for the downward flow of a critical component C is given by $\texttt{C.downSetWu} = \bigcup_{x \in C} P(\texttt{x.downSetWu})$. Similar for `C.downSetWs, C.upSetWu` and `C.upSetWs`.

Step 3 of `ComputeMCGcomponents()`: This step is similar to Step 3 for pixels but it considers components instead of pixels.

3.2 Experimentation on image data

In this section, we illustrate examples of Morse connections graphs. In the following tests, our height field f is defined as the grey-level of pixels in the image. The first example (Fig 4) shows the particular case where there are only isolated critical pixels. Thus the notion of components is not useful. In the other examples (Fig 5 and 6), there are many adjacent critical (or unclassified) pixels. Then, the notion of components is more appropriate. Indeed, the Morse connections graph obtained without considering components is not accurate in the sense that there are adjacent nodes in the graph that do not correspond to a critical region. These nodes disappear when we consider the Morse connections graph obtained in taking into account the components.

4 Conclusion

The main direction of the future work on this topic is to generalize the construction of critical components and of the Morse connections graph to higher dimensions and to other types of meshes than cubical grids studied here, such as simplicial triangulations. These generalizations are needed, in particular, for the qualitative geometric study of iso-surfaces and iso-volumes. In higher dimensional spaces, the numbers of connected components in upper and lower wraps of pixels are not sufficient to characterize their criticality, thus more advanced topological tools such as homotopy or homology are required. The continuation of our work in this direction is in progress.

Our program appeared very efficient in implementations performed so far because of the two-dimensional setting. However, the computational complexity usually increases significantly with dimension. Then, some variations of the formula for the multivalued generator of a semi-dynamical system in Definition 2.10 may be introduced for improving its capacity of handling more general data, if needed. For example, one may replace the exact value of the maximum M_i by an interval of type $[(1-\epsilon)M_i, M_i]$ (the same for the minimum m_i), where ϵ is an admissible error bound. One may also attempt to use random dynamical systems approaches.

Among our future goals is to incorporate the notion of persistence studied by [5, 4, 25]. This notion aims at making the constructions robust with respect to changes in the resolution scale while the global topology is preserved.

A mathematically challenging goal is to study generalizations of the Euler formula in the context of our work. Our experiments test positively when the isolation and non-degeneracy assumptions hold and when there are no critical pixels on the boundary of the region. But the Euler formula cannot hold in presence of multiple saddles and critical components. In fact, a critical component itself may have an Euler characteristics different than that of a point. As we mentioned in the introduction, the standard way of dealing with this problem is forcing the assumptions of the classical theory by manipulating with data so to achieve the isolation and non-degeneracy of critical points. Our philosophy is to keep the original data but we search for a generalization of the Euler formula that would hold for the degenerate case. We hope that this can be achieved by incorporating ideas from the Conley index theory (see [17, 15]) but we have not found yet a formula that would be satisfactory from the point of view of implementation.

References

[1] M. Allili, D. Corriveau, D. Ziou: Morse Homology Descriptor for Shape Characterization. 17th International Conference on Pattern Recognition **4** (2004) 27–30.

[2] H. Carr, J. Snoeyink, U. Axen: Computing contour trees in all dimensions. In Proc. 11th Sympos. Discrete Alg.(2000), 918–926.

[3] D. Corriveau, M. Allili, D. Ziou: Morse Connections Graph for Shape Representation, In Lecture Notes in Computer Science 3708, Springer-Verlag 2005, 219–226.

[4] H. Edelsbrunner, J. Harer, A.J Zomorodian: Hierarchical Morse-Smale Complexes for Piecewise Linear 2-Manifolds. In AMC Symposium on Computational Geometry, Discrete & Compu. Geom. **30** (2003) 1, 87–107.

[5] H. Edelsbrunner, D. Letscher, A.J. Zomorodian: Topological Persistence and Simplification. Discrete & Compu. Geom. **28** (2002) 4, 511–533.

[6] A. Fomenko, T. Kunii: Topological Modeling for Visualization. Springer-Verlag, Tokyo 1997.

[7] R. Forman: Morse theory for cell complexes. Advances in Math. 134 (1998), 90-145.

[8] R.J. Fowler, J.J. Little: Automatic extraction of irregular network digital terrain models. Computer Graphics, 13 (1979) 199–207.

[9] L. D. Griffin and A. C. F. Colchester: Superficial and deep structure in linear diffusion scale space: Isophotes, critical points and separatrices. Image and Vision Computing, 13:543-557, 1995.

[10] T. Kaczynski, K. Mischaikow, M. Mrozek: Computational Homology. Appl. Math. Sci. Series **157**, Springer-Verlag, New York 2004.

[11] T. Kaczynski, M. Mrozek: Conley Index for Discrete Multivalued Dynamical Systems. Topology and Its Appl. **65** (1995) 83–96.

[12] J.J Koenderink, A.J van Doorn: The structure of two-dimensional scalar fields with applications to vision. Biol. Cybern., Vol 33 (1979) 151–158.

[13] Y. Matsumoto: An Introduction to Morse Theory. Mathematical Monographs **208**, Amer. Math. Soc. 1997.

[14] J. C. Maxwell: On hills and dales. The London, Edinburg and Dublin Philosophical Magazine and Journal of Science, (1870) 40:421-425.

[15] C.K. McCord: On the Hopf Index and the Conley Index. Transactions of the American Mathematical Society, Vol. 313, No. 2 (1989) 853-860.

[16] J. Milnor: Morse Theory. Princeton University Press, Princeton, 1969.

[17] K. Mischaikow: The Conley Index Theory: A Brief Introduction. Banach Center Publications **47** (1999).

[18] M. Morse: Relations between the critical points of a real function on n independent variables. Transactions of the American Mathematical Society, 27 (1925) 345-396.

[19] J.R Munkres: Elements of Algebraic Topology. Addison-Wesley, 1984.

[20] Lee R. Nackman: Two-dimensional critical point configuration graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence, 6:442-450, 1984.

[21] Y. Shinagawa, T. Kunii: Constructing a Reeb Graph Automatically from Cross Sections. IEEE Comp. Graph. and Appl. **11** (1991) 44–51.

[22] S. Takahashi, T. Ikeda, Y. Shinagawa, T.L. Kunii, M. Ueda: Algorithms for Extracting Correct Critical Points and Constructing Topological Graphs from Discrete Geographical Elevation Data. The International Journal of the Eurographics Association, 14 (3), (1995) C-181- C-192.

[23] R. Thom: Sur une partition en cellules associée à une fonction sur une variété. Comptes Rendus de l'Académie des Sciences **228** (1949) 973–975.

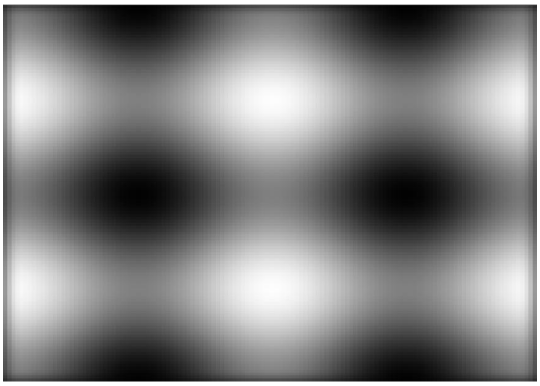
[24] G.H Weber, G. Scheuermann, B. Hamann: *Detecting critical regions in scalar fields.* IEEE TCVG Symposium on Visualization (2003) 1-11.

[25] A.J. Zomorodian, G. Carlsson: Computing Persistent Homology. Discrete & Compu. Geom. **33** (2005) 2, 249–274.

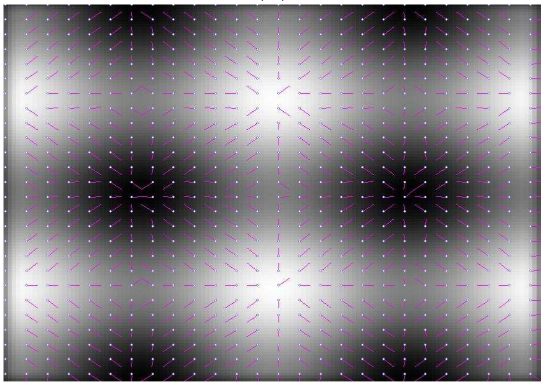
Department of Computer Science
Bishop's University
Sherbrooke (Québec), Canada J1M 1Z7
mallili@ubishops.ca, anik.trahan@usherbrooke.ca

Département d'informatique
Université de Sherbrooke,
Sherbrooke (Québec), Canada J1K 2R1
david.corriveau@usherbrooke.ca

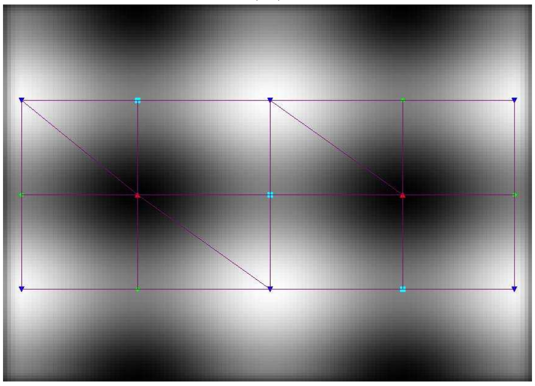
Département de mathématiques
Université de Sherbrooke,
Sherbrooke (Québec), Canada J1K 2R1
{sara.deriviere, tomasz.kaczynski}@usherbrooke.ca



(a)

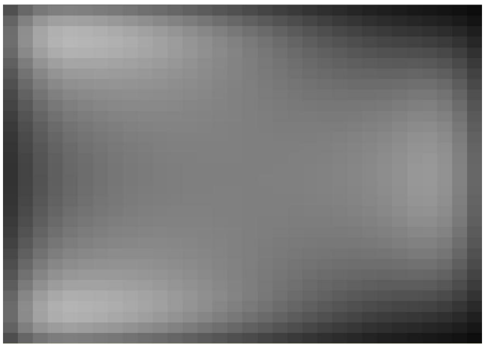


(b)

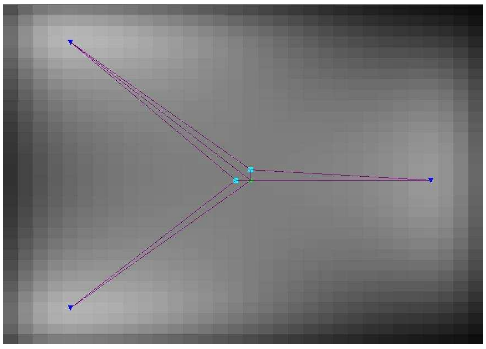


(c)

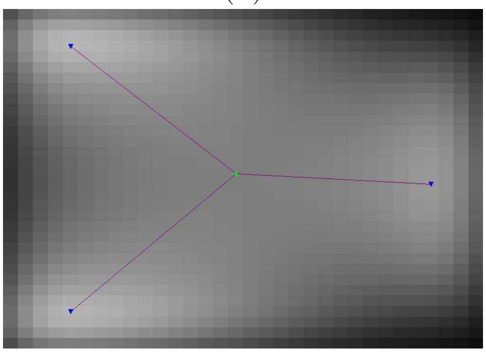
Figure 4: (a) An image of the height field given by $f(x, y) = cos(x) - cos(y)$. (b) Its upward flow displayed on some pixels. (c) The associated Morse connections graph. Green diamonds correspond to saddles, blue triangles are maxima, red triangles are minima and cyan squares are unclassified pixels.



(a)

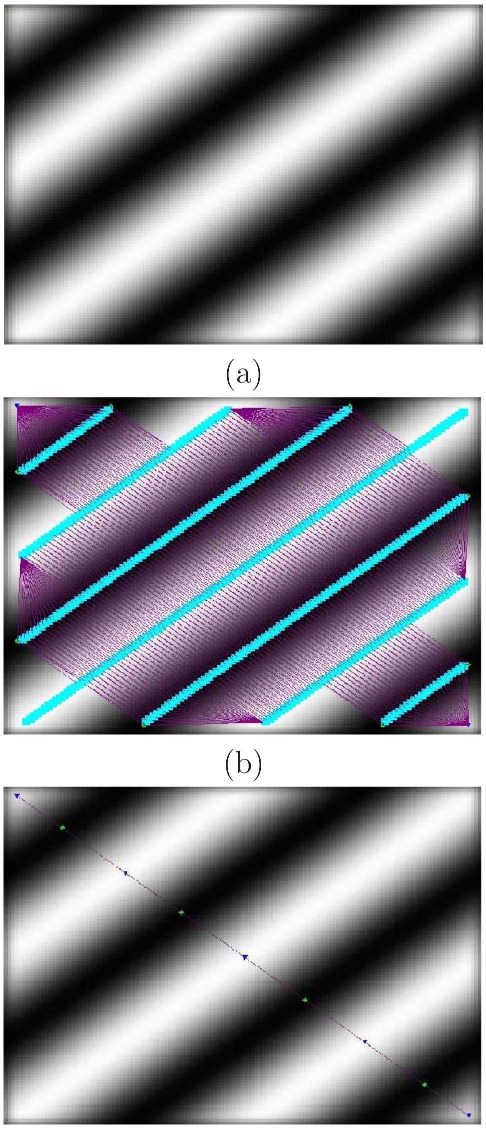


(b)



(c)

Figure 5: (a) An image of a monkey saddle of the height field given by $f(x, y) = x^3 - 3xy^2$. (b) Its Morse connections graph based on pixels. (c) Its Morse connections graph based on components. Green diamonds correspond to saddles, blue triangles are maxima, red triangles are minima and cyan squares are unclassified pixels.



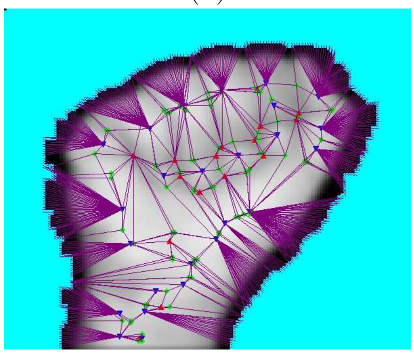
(a)

(b)

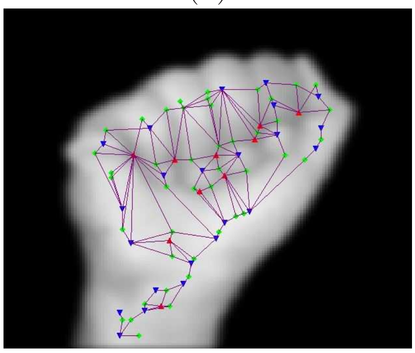
(c)

Figure 6: (a) An image of the height field given by $f(x, y) = cos(x + y)$. (b) Its Morse connections graph based on pixels. Multiple vertices clustered around the contour of the hand represent unclassified pixels. (c) Its Morse connections graph based on components: each critical component consists of a flat line which represents a plateau for the function. The corresponding vertex of MCG is shown at the center of the component. Green diamonds correspond to saddles, blue triangles are maxima, red triangles are minima and cyan squares are unclassified pixels.

(a)



(b)



(c)

Figure 7: (a) Input image depicting a hand gesture on uniform background. (b) Its Morse connections graph based on pixels. (c) Its Morse connections graph based on components. Green diamonds correspond to saddles, blue triangles are maxima, red triangles are minima and cyan squares are unclassified pixels.