Membrane parallelism for discrete Morse theory applied to digital images

Raúl Reina-Molina · Daniel Díaz-Pernil · Pedro Real · Ainhoa Berciano

Abstract In this paper, we propose a bio-inspired membrane computational framework for constructing discrete Morse complexes for binary digital images. Our approach is based on the discrete Morse theory and we work with cubical complexes. As example, a parallel algorithm for computing homology groups of binary 3D digital images is designed.

Keywords Discrete Morse theory · Digital Imagery · Homology groups · Membrane Computing

1 Introduction

Nowadays, to determine in a fast and accurate way topology-related information of technologically relevant mathematical structures has become a very important question. Within the context of Digital Imagery, computational topology has acquired a significant role in fields of applications like computer vision, digital image processing or medical imaging. Working with cubical cell complex representations of digital

R. Reina-Molina (✉) · D. Díaz-Pernil · P. Real
CATAM Research Group, Department of Applied Mathematics I, University of Seville,
Seville, Spain
e-mail: raureimol@alum.us.es

D. Díaz-Pernil
e-mail: sbdani@us.es

P. Real
e-mail: real@us.es

A. Berciano
Department of Didactic of Mathematics and Experimental Sciences,
University of the Basque Country, Bilbao, Spain
e-mail: ainhoa.berciano@ehu.es

images, we rewrite here a non-negligent part of the computational machinery underlying in discrete Morse theory [13], using bio-inspired models of Membrane Computing area [25]. Discrete Morse theory (DMT) deals with the combinatorial homotopy analysis of cell complexes using discrete Morse functions or, equivalently, discrete vector fields. Membrane Computing deals with distributed and (massively) parallel computing models inspired from cellular membranes and locally processing multisets of symbol objects. This paper is a foundational effort to properly combine the computational power of bio-inspired computing with the detailed topological analysis of cubical complexes that DMT provides us. In this way, we implement a membrane computing framework for constructing "simplified" digital images with regard to some special topologically consistent properties mainly related with homology groups.

As a result of the inherent parallelism involved in the definition of digital images and due to the current improvement in parallel computational architectures, several studies are devoted to parallel implementation of sequential algorithms in order to take advantage of parallel hardware.

The development of parallel algorithms is not a trivial question and involves an adequate management of task's concurrency and a performance, memory efficiency and scalability trade-off. Natural Computing has been presented as a productive research field that can develop no ordinary computational models inspired by Nature that can solve several difficult problems in an elegant and simple way. Here, to maintain a strong link between this computational framework and algebraic topology tools is a priority. In this way, we obtain a topological description of the digital objects which allows us to understand, efficiently compute and handle advanced topological information of kD digital objects and images [2,8]. For this purpose, we present a new methodology to develop algorithms in topological analysis of digital images. Within this bioinspired setting, our approach mainly focus on performance more than on memory efficiency or scalability.

This paper is the first one using membrane computing for DMT techniques. Various DMT-based approaches to compute discrete functions defined on cells of a cell complex have been already used. Reininghaus et al. [29,30] apply DMT methods to analyze vector fields. Bauer et al. [1] computed simplified two-dimensional scalar functions while ensuring that the input function is modified by no more than a threshold d and all surviving critical point pairs have persistence greater than $2d$. Cazals et al. [7] and Lewiner et al. [17] successfully employed Forman's discrete Morse theory to compute Morse complexes of piecewise-linear functions and show applications to segmentation, visualization, and mesh compression. Gyulassy et al. [15] also used a DMT-based formulation to develop an efficient algorithm for computing Morse (more concretely, Morse-Smale) complexes of large 3D data. Robins et al. [30] proposed an algorithm to compute the Morse complex of 2D and 3D gray-scale digital images modeled as discrete functions on cubical complexes. The algorithm computes the Morse complex with provable guarantees on its correctness with respect to the critical cells.

In [20], an algorithm for computing homology using a recursive tree-based technique for generating Morse complexes is established. Peterka et al. [27] introduced a set of building blocks for implementing parallel algorithms, which leverage high performance computing clusters. In particular, they discuss a parallel implementation of the discrete Morse theory based on the algorithm proposed by Gyulassy et al. [15].

Günther et al. [14] design a memory efficient algorithm to compute the Morse–Smale complex for 3D data and use the complex to compute persistent homology groups. The discrete gradient field is computed using a parallel variant of the method proposed by Robins et al. followed by an efficient computation of the boundary map representing the Morse complex. Finally, the parallel algorithm for computing the discrete gradient field given in [22] is based on a novel description of the discrete Morse function followed by a two-step algorithm computing the cells of the Morse–Smale complex. The algorithms are implemented using a hybrid multi-core implementation.

The papers [6,9,10], deal with the computation of some combinatorial and algebraic aspects of the topology of the image.

This paper is structured as follows. First, some aspects related with cubical homology and DMT are reviewed. Then, basic concepts and techniques from Membrane computing are given. Next, a theoretical framework for designing algorithms in this "hibrid" context and some examples of its use are presented. Finally, some conclusions are discussed,where the most important fact is the theoretical viability of framework shown before, that allows us to implement algorithms computing the homology groups of 3D images using parallel computing tools.

2 Discrete Morse theory

In this section we introduce the topological background required by our framework. First of all, we review the required combinatorial structure for the topological spaces that are used. We mainly follow the process introduced by Kaczyński et al. [16] with some minor changes. Specifically, Kaczyński et al. uses closed cubes as main combinatorial objects while we use open cubes.

A *cubical cell* σ is a finite product of intervals:

$$\sigma = I_1 \times \cdots \times I_k \subset \mathbb{R}^k$$

where I_j is an interval with integer extremes $(m_j, m_j + 1)$ and length one or the singleton $\{m_j\}$, denoted as (m_j), for each $j \in \{1, \ldots, k\}$. The interval I_j is referred as the jth component of σ and denoted by $I_j(\sigma)$. The set of all cubical cells in $\mathbb{R}^k$ is denoted by $\mathcal{K}^k$. The set of all cubical cells is

$$\mathcal{K} = \bigcup_{k=1}^{\infty} \mathcal{K}^k \tag{1}$$

We usually require the cubical cells to be bounded. Hence, we define

$$\mathcal{K}^k_{*,n} = \{\sigma \in \mathcal{K}^k : 0 \leq \inf I_p(\sigma), \sup I_p(\sigma) < n, 1 \leq p \leq k\} \tag{2}$$

where k and n are nonnegative integer numbers.

Given a cubical cell σ in $\mathbb{R}^k$, its *embedding number* k is denoted by $\operatorname{emb}\sigma$. The dimension of σ is defined as the number of unitary intervals in its expression as product of intervals and is denoted by $\dim\sigma$. The set of all elementary cubes with dimension p is denoted by $\mathcal{K}_p$. The set of all elementary cubes in $\mathbb{R}^k$ with dimension p is denoted by $\mathcal{K}^k_p$. Whenever the dimension of a cubical cell require to be made explicit, it is denoted as a superscript between parenthesis. Therefore, $\dim \sigma^{(p)} = p$. We also explicitly indicate the dimension of a cell $\sigma^{(p)}$ writing that σ is a p-cell.

The closure of a cubical cell can be decomposed into the union of lower-dimensional cubical cells. If δ and σ are two cubical cells in $\mathbb{R}^k$ of any dimension and $\delta \subset \overline{\sigma}$, then δ is a *face* of σ and is denoted as $\delta \leq \sigma$. If δ is a face of σ and $\delta \neq \sigma$, then δ is a *proper face* of σ, denoted as $\delta < \sigma$. If δ is a face of σ and $\dim \delta = \dim \sigma - 1$ then δ is a *primary face* of σ, denoted by $\delta \in \partial\sigma$.

A *cubical complex* is a collection K of cubical cells with the same embedding number and such that, for every cubical cell $\sigma \in K$, all of its primary faces are in the complex. We denote the set of p-cells in K as K_p.

As usual, we define the chain complex $\mathcal{C}(K)$ as the graded $\mathbb{Z}$-module $\{C_p(K)\}_{p\in\mathbb{Z}}$ where $C_p(K)$ is the free abelian $\mathbb{Z}$-module generated by the cubical p-cells in K. The boundary map ∂ in the chain complex is defined in [16]. Concretely, for any cubical cell σ, $\partial\sigma$ is an alternating sum over the faces of σ and such that $\partial \circ \partial = 0$. As usual, let $Z_p = \ker \partial_p$ the subgroup of *cycles* and $B_p = \operatorname{Im} \partial_{p+1}$ the subgroup of *boundaries*. The pth cubical homology group of K is

$$H_p(K) = \frac{Z_p K}{B_p(K)}$$

The cubical homology of K is the collection $H_*(K) = \{H_p(K)\}_{p\in\mathbb{Z}}$. Recall that, for $p < 0$ or $p > \dim K$, is $H_p(K) = 0$.

2.1 Discrete Morse theory

So far, we have recalled all the required notions from cubical complexes. Now, we introduce the needed elements from DMT.

A real-valued smooth map defined over a compact d-manifold is a *Morse function* if all its critical points have non-singular Hessian matrix and no two critical points have the same function value [15]. Morse functions allow to endow the manifold with a cellular structure.

A CW complex is, roughly speaking, a topological space endowed with a decomposition into smaller pieces called cells, being homeomorphic to Euclidean spheres. A formal definition can be found in [13].

Discrete Morse theory introduced by Forman [13] adapts Morse Theory to CW complexes instead of smooth manifolds. Although DMT has been developed for finite regular CW complexes, we restrict our research to finite cubical complexes, which are a particular type of CW-complexes.

Let K be a cubical complex and $f : K \to \mathbb{R}$ a function that assigns scalar values to every cell of K. f is a *discrete Morse function* if, for every cubical cell $\sigma^{(p)} \in K$ the following conditions hold:

- $\#\{\tau^{(p+1)} \in K | \tau > \sigma \wedge f(\tau) \leq f(\sigma)\} \leq 1$
- $\#\{\mu^{(p-1)} \in K | \mu < \sigma \wedge f(\tau) \geq f(\sigma)\} \leq 1$

A discrete Morse function f can be thought as a discrete function that is increasing with respect to cell dimension except for, at most, two cells for each cell. Concretely, for each cell σ there is at most one face μ ($\mu \in \partial\sigma$) with $f(\mu) \geq f(\sigma)$ and there is at most one facet τ ($\sigma \in \partial\tau$) 0 with $f(\tau) \leq f(\sigma)$.

A cubical cell $\sigma^{(p)}$ is *critical* if one of the following conditions hold:

- $\#\{\tau^{(p+1)} \in K | \tau > \sigma \wedge f(\tau) \leq f(\sigma)\} = 0$
- $\#\{\mu^{(p-1)} \in K | \mu < \sigma \wedge f(\tau) \geq f(\sigma)\} = 0$

Given a cubical cell σ its *barycenter* is the point

$$b(\sigma) = \left(\frac{\inf I_1(\sigma) + \sup I_1(\sigma)}{2}, \ldots, \frac{\inf I_k(\sigma) + \sup I_k(\sigma)}{2}\right).$$

As the closure of a cubical cell is convex, its barycenter is an interior point.

We define a *discrete vector* as a pair of cells $\{\sigma^{(p)} < \tau^{(p+1)}\}$. Vectors are represented as arrows from the barycenter of the lower dimension cell to barycenter of the higher dimension cell. A *discrete vector field* V on K is a collection of vectors in K such that each cubical cell in the vector belongs to, at most, one pair of V.

Formally, a discrete vector field is a map $V : K \to K \cup \{0\}$ such that:

1. for each $\sigma \in K$, if $V(\sigma) \neq 0$ then $\dim V(\sigma) = \dim \sigma + 1$
2. for each $\sigma \in K_p$, either $V(\sigma) = 0$ or $\sigma \in \partial V(\sigma)$
3. if $\sigma \in \mathrm{Im}(V)$ then $V(\sigma) = 0$
4. for each $\sigma \in K_p$

$$\#\{\mu^{(p-1)} \in K | V(\mu) = \sigma\} \leq 1$$

Given a discrete vector field V on K, a V*-path* of dimension p, γ, is a sequence of cubical p-cells $\sigma_0, \sigma_1, \sigma_2, \ldots, \sigma_r$ such that

1. If $V(\sigma_i) = 0$, then $\sigma_{i+1} = \sigma_i$.
2. If $V(\sigma_i) \neq 0$, then $\sigma_{i+1} < V(\sigma_i)$ with $\sigma_{i+1} \neq \sigma_i$.

The set of V-paths is denoted as $\Gamma(V)$.

A V-path $\gamma = \sigma_0, \sigma_1, \ldots, \sigma_r$ is called *closed* is $\sigma_r = \sigma_0$ and is called *non-stationary* if $\sigma_1 \neq \sigma_0$.

Given two V-paths $\gamma_1 = \sigma_0, \ldots, \sigma_r$ and $\gamma_2 = \sigma_r, \ldots, \sigma_{r+s}$, the *concatenation* $\gamma_2 \circ \gamma_1$ is defined by

$$\gamma_2 \circ \gamma_1 = \sigma_0, \ldots, \sigma_r, \sigma_{r+1}, \ldots, \sigma_{r+s} \tag{3}$$

Two V-paths γ_1 and γ_2 are said to be *connectible* if there is a V-path γ_3 such that $\gamma_3 = \gamma_2 \circ \gamma_1$

A V-path γ_1 is said to be *extendable* by another V-path γ_2 whenever there is a V-path γ' such that $\gamma_2 = \gamma' \circ \gamma_1$, and is denoted by $\gamma_1 \trianglelefteq \gamma_2$. It can be easily verified that $(\Gamma(V), \trianglelefteq)$ is a partially ordered set. Whenever a V-path γ_1 is extendable by a V-path γ_2 so that $\gamma_2 = \gamma' \circ \gamma_1$ as well as γ' is non-stationary, it is denoted as $\gamma_1 \triangleleft \gamma_2$.

Given a V-path $\gamma = \sigma_0, \sigma_1, \ldots, \sigma_r$, the *multiplicity* of γ is defined as

$$m(\gamma) = \prod_{\substack{i=0 \\ V(\sigma_i)\neq 0}}^{r-1} \langle \sigma_i, \partial V(\sigma_i)\rangle \langle \sigma_{i+1}, \partial V(\sigma_i)\rangle \tag{4}$$

Given two connectible paths γ_1 and γ_2, then

$$m(\gamma_2 \circ \gamma_1) = m(\gamma_2) \cdot m(\gamma_2) \tag{5}$$

A V-path γ is said to be *maximal* if there is no V-path γ' such that $\gamma \lhd \gamma'$. A maximal path is called *critical* if its last cell is critical. Notice that, if $\gamma = \sigma_0, \ldots, \sigma_r$ is a maximal critical path, then there is a critical cell μ such that $\langle \sigma_0, \partial\mu\rangle \neq 0$.

Given a discrete Morse function, a special discrete vector field called *discrete gradient vector field* is defined so that $f(V(\sigma)) \leq f(\sigma)$. As shown in [13], a cubical complex can be transformed into another homotopically equivalent following a series of simplicial collapses,[1] where each of them collapses both cubical cells in each vector in the corresponding discrete gradient vector field.

We recall here one of the main results of DMT.

Theorem 1 ([13], 9.3) *A discrete vector field V is the gradient vector field of some discrete Morse function if and only if there are no non-stationary closed V-paths.*

A vector field V is extended to a graded group homomorphism $V : C_p(K) \to C_{p+1}(K)$ such that

$$V(\sigma^{(p)}) = \begin{cases} \pm\tau^{(p+1)} & \text{if } \{\sigma < \tau\} \text{ is a vector in } V \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where the sign is chosen so that $\langle \sigma, \partial V(\sigma)\rangle = -1$.

The *reduced (discrete time) flow map*, denoted as $\tilde{\phi}$ is defined by

$$\tilde{\phi} = \mathrm{id} + \partial \circ V \tag{7}$$

The reduced flow map associates a $(p+1)$-chain $\tilde{\phi}(\sigma)$ to any p-cell σ, such that $\partial\, V(\sigma) = \tilde{\phi}(\sigma) - \sigma$. Hence, $\langle \sigma, \tilde{\phi}(\sigma)\rangle = 0$. The map $\tilde{\phi}$ establishes a way of *following* the flow determined by the vector field. Concretely, if there is a V-path of length r from a p-cell σ to another p-cell σ', then $\langle \tilde{\phi}^r(\sigma), \sigma'\rangle \neq 0$. This fact can be proved by induction in the length of the V-path.

Definition 1 Let K be a cubical complex and f be a Morse discrete function on K. Let $C_p(K)$ denote the p-chains on K and $\mathcal{M}_p \subseteq C_p(K)$ the span of the critical p-cells. The graded group $\mathcal{M} = \{\mathcal{M}_p\}_{p\in\mathbb{Z}}$ is called the space of Morse chains.

[1] In [13] Forman works with simplicial complexes, however the mathematical scaffolding provided by DMT can be settled with no change to finite cubical complexes.

In [13] Forman makes use of the set of V-paths to build the boundary map of the Morse complex associated to a given complex and an acyclic vector field. It is shown in the following result.

Theorem 2 ([13], 7.1) *There are boundary maps $\tilde{\partial}_p : \mathcal{M}_p \to \mathcal{M}_{p-1}$, for each $p \in \mathbb{Z}$, so that*

$$\tilde{\partial}_{p-1} \circ \tilde{\partial}_p = 0$$

and such that the differential complex

$$0 \xrightarrow{\tilde{\partial}_{k+1}} \mathcal{M}_k \xrightarrow{\tilde{\partial}_k} \mathcal{M}_{k-1} \xrightarrow{\tilde{\partial}_{k-1}} \cdots \xrightarrow{\tilde{\partial}_2} \mathcal{M}_1 \xrightarrow{\tilde{\partial}_1} \mathcal{M}_0 \xrightarrow{\tilde{\partial}_0} 0$$

calculates de homology of K. That is, if we define

$$H_p(\mathcal{M}, \tilde{\partial}) = \frac{\ker \tilde{\partial}_p}{\operatorname{Im} \tilde{\partial}_{p+1}}$$

then, for each p

$$H_p(\mathcal{M}, \tilde{\partial}) \equiv H_p(K).$$

In [13] an explicit expression for the boundary operator in the Morse complex $(\mathcal{M}, \tilde{\partial})$ is also given in terms of V-paths.

Theorem 3 ([13], 7.3) *For any critical cubical $(p+1)$-cell τ, let $\tilde{\partial}\tau$ be defined as*

$$\tilde{\partial}\tau = \sum_{\sigma \in \mathcal{M}_p} c_{\sigma,\tau}\sigma$$

where

$$c_{\sigma,\tau} = \sum_{\gamma \in \Gamma(\tau,\sigma)} m(\gamma)$$

and $\Gamma(\tau, \sigma)$ is the set of maximal critical V-paths (for a given gradient vector field V) which go from a maximal face of τ to σ.

The complex $(\mathcal{M}, \tilde{\partial})$ and K have isomorphic homology groups.

2.2 Examples

In Fig. 1 a 2D cubical complex K is shown (Fig. 1a). In this example, cubical cells are referred as ordered tuples of vertices. A vector field V is shown in Fig. 1b. As the previous vector field is acyclic, it is in fact the gradient vector field for some discrete Morse function f, whose critical cells are showed in Fig. 1c.

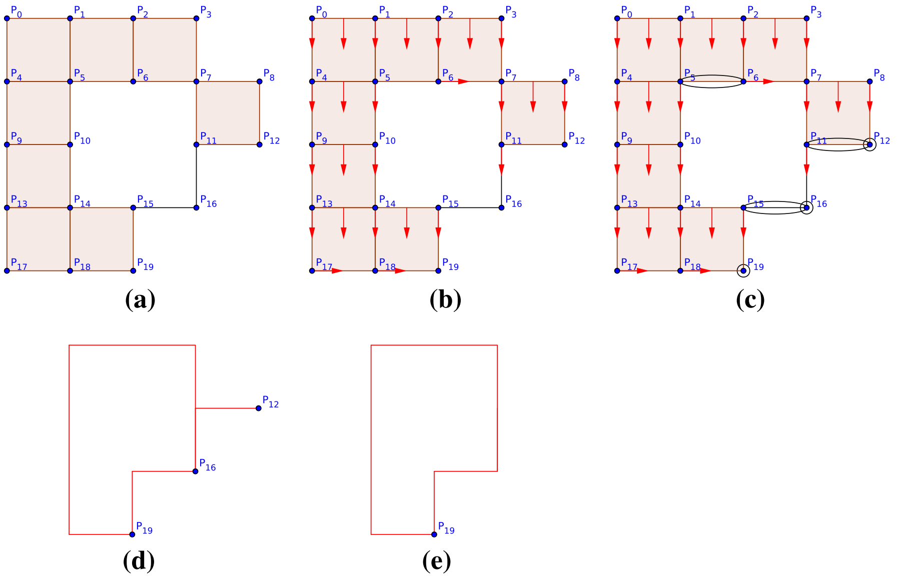

Fig. 1 **a** Example of cubical complex, **b** gradient vector field, **c** critical cells where two kinds of critical cells are shown: ordinary critical cells representing a hole at any dimension and other critical cells showing a "step" configuration which will be canceled later (see [13], section 11, for details), **d** critical complex and **e** final homological complex with null differential

The reduced flow map $\tilde{\phi}$ maps a cell $\sigma^{(p)}$ into a p-chain c, such that all the cells in c are those in $\partial V(\sigma)$ except σ. For example, $\tilde{\phi}$ applied to the 1-cell spanned by P_4 and P_5, denoted by $< P_4, P_5 >$ (see Fig. 1a) returns

$$\tilde{\phi}(\langle P_4, P_5\rangle) = \langle P_4, P_9\rangle + \langle P_9, P_{10}\rangle - \langle P_5, P_{10}\rangle$$

Informally speaking, the application of $\tilde{\phi}$ means the annihilation of some cells involved in the vector field, as well as the "extension" of critical cells in the sense of creating critical "super-cells" in terms of chains. For example, the application of $\tilde{\phi}$ to $\langle P_4, P_5\rangle$ not only removes $\langle P_4, P_5\rangle$ and $\langle P_4, P_9, P_{10}, P_5\rangle$, but it extends the cell $\langle P_0, P_4, P_5, P_1\rangle$ to the chain $\langle P_0, P_4, P_5, P_1\rangle + \langle P_4, P_9, P_{10}, P_5\rangle$.

This previous example provides us the key idea in the development of our framework. The homology of a 3D cubical complex can be calculated in only two stages. First the complex is reduced by the construction of a gradient vector field and the calculation of the associated Morse complex (see Fig. 1d). Second, the process is repeated in the Morse complex and, finally, the result is a chain complex with null differential whose pth homology group is easily calculable and coincides with the pth homology group of the cubical complex (see Fig. 1e).

2.3 Encoding images as cubical complexes and cubes as tuples

In this section, we work with a cubical cell version of a digital image [16]. A common problem appearing in the development and implementation of algorithms with cubical complexes is the requirement of a huge amount of memory resources. Below, an efficient encoding of cubical cells is described.

Given two non-negative integers k and $n \geq 2$, the following auxiliary sets are defined

$$T_n^k := \{0, 1, 2, \ldots, n-1\}^k, \tag{8}$$

$$\overline{T}_n^k := T_{2n-1}^k, \tag{9}$$

$$I_{n,k} := T_{n^k} = \{0, 1, 2, \ldots, n^k - 1\}, \tag{10}$$

$$\overline{I}_{n,k} := T_{(2n-1)^k} = \{0, 1, 2, \ldots, (2n-1)^k - 1\}. \tag{11}$$

The set of points[2] for the source images is the set $T_n^k \subset \mathbb{Z}^k$ equipped with a *cubic* neighborhood function, described as follows: two points $\mathbf{i} = (i_1, \ldots, i_k)$ and $\mathbf{j} = (j_1, \ldots, j_k)$ are said $(2k)$-*adjacent* if $\sum_{p=1}^{k} |i_p - j_p| = 1$. The neighborhood function is given by

$$N(i_1, \ldots, i_k) = \left\{ (j_1, \ldots, j_k) \in T_n^k : \sum_{p=1}^{k} |i_p - j_p| = 1 \right\}$$

This function restricted to $k = 2$ defines 4-adjacency and to $k = 3$ defines 6-adjacency.

Let $\mathcal{I} : T_n^k \to \{0, 1\}$ be a k-D binary image of size n^k, where the set of points in the object (or black points) is given by $\mathcal{I}^{-1}(1)$. Let $K = K(\mathcal{I})$ be the cubical cell complex associated to $\mathcal{I}$. In K, the 0-cells are points in the object, the 1-cells represent pairs of $(2k)$-adjacent points, the 2-cells unit squares where its edges are pairs of $(2k)$-adjacent points, and so on. In general, each p-cell is a p-dimensional unit hypercube whose edges are determined by pairs of $(2k)$-adjacent points.

A cubical cell is encoded as a tuple through the function $\mathbb{T} : \mathcal{K}^k \to \mathbb{Z}^k$ defined as follows

$$\mathbb{T}(\sigma) = (\sup I_1(\sigma) + \inf I_1(\sigma), \ldots, \sup I_k(\sigma) + \inf I_k(\sigma)) \tag{12}$$

The main property of $\mathbb{T}$ is given in the following lemma.

Lemma 1 *The function $\mathbb{T}$ is a bijection and $\mathbb{T}(\mathcal{K}_{*,n}^k) = \overline{T}_n^k$.*

Proof First at all, let us prove that $\mathbb{T}$ is injective. Let σ and μ be two distinct cells. Then, let $j_0 \in \{1, 2, \ldots k\}$ be the first index such that $I_{j_0}(\sigma) \neq I_{j_0}(\mu)$. The following cases may come up:

- $I_{j_0}(\sigma) = (a, a+1)$ and $I_{j_0}(\mu) = (b, b+1)$ with $a \neq b$. Then

$$\inf I_{j_0}(\sigma) = a \neq b = \inf I_{j_0}(\mu)$$

[2] The reader is supposed to be familiar with concepts of Image Algebra. For a detailed text, see [31].

- $I_{j_0}(\sigma) = (a, a+1)$ and $I_{j_0}(\mu) = (b)$. Then, either $a \neq b$ or $a+1 \neq b$. In the first case,

$$\inf I_{j_0}(\sigma) = a \neq b = \inf I_{j_0}(\mu)$$

  and in the second case

$$\sup I_{j_0}(\sigma) = a+1 \neq b = \sup I_{j_0}(\mu)$$

In all the cases showed above $\inf I_{j_0}(\sigma) + \sup I_{j_0}(\sigma) \neq \inf I_{j_0}(\mu) + \sup I_{j_0}(\mu)$ and, hence, $\mathbb{T}(\sigma) \neq \mathbb{T}(\mu)$.

To prove that $\mathbb{T}$ is surjective, it is enough to find a cell $\sigma = I_1 \times \cdots \times I_k$ such that, given a tuple $(c_1, \ldots, c_k) \in \mathbb{Z}^k$, $\mathbb{T}(\sigma) = (c_1, \ldots, c_k)$. It is easily verified that such cell σ si given by setting each interval I_p as follows

$$I_p = \begin{cases} \left(\frac{c_p-1}{2}, \frac{c_p+1}{2}\right) & \text{if } c_p \equiv 1 \pmod{2} \\ \left(\frac{c_p}{2}\right) & \text{if } c_p \equiv 0 \pmod{2} \end{cases} \tag{13}$$

for $1 \leq p \leq k$.

Finally notice that, if $I = (m, m+1)$ with $0 \leq m, m+1 \leq n-1$, then $\sup I + \inf I = 2m+1 \leq 2n-2$. In consequence, $\sup I + \inf I \in T_{2n-1}$, proving the last statement of the Lemma. □

As an example, the cell $\sigma = (0, 1) \times (3, 4) \times (2)$ is encoded as $\mathbb{T}(\sigma) = (1, 7, 4)$.

A cubical cell can also be encoded as an integer. Concretely, a cubical cell[3] is encoded as a number $\mathbb{I}_{n,k}(\sigma)$ in $\overline{I}_{n,k}$ as follows:

$$\mathbb{I}_{n,k}(\sigma) = \mathbb{I}_{n,k}(c_1, \ldots, c_k) = \sum_{p=1}^{k} c_p \cdot (2n-1)^{k-p} \tag{14}$$

Lemma 2 *The function $\mathbb{I}_{n,k} : \overline{T}_n^k \to \overline{I}_{n,k}$ defined above is a bijection.*

Proof To prove that the function $\mathbb{I}_{n,k}$ is surjective is enough to find a cell σ_l for any $l \in \overline{I}_{n,k}$ such that $\mathbb{I}_{n,k}(\sigma_l) = l$. The cell $\sigma_l = (c_1, \ldots, c_k)$ such that

$$c_p = \left\lfloor \frac{l}{(2n-1)^{(k-p)}} \right\rfloor \pmod{(2n-1)} \text{ for } 1 \leq p \leq k \tag{15}$$

satisfies that $\mathbb{I}_{n,k}(\sigma_l) = l$.

[3] Lemma 1 grants the identification of cubical cells in a cubical complex in $\mathbb{R}^k$ with k-tuples.

To prove that $\mathbb{I}_{n,k}$ is injective, let $\sigma, \sigma' \in \overline{T}_n^k$ be two cells such that $\mathbb{I}_{n,k}(\sigma) = \mathbb{I}_{n,k}(\sigma')$. Then, if $\sigma = (c_1, \ldots, c_k)$ and $\sigma' = (c'_1, \ldots, c'_k)$,

$$0 = \mathbb{I}_{n,k}(\sigma') - \mathbb{I}_{n,k}(\sigma) = \sum_{p=1}^{k} (c'_p - c_p)(2n-1)^{(k-p)}$$

hence, taking iteratively remainders modulo $(2n-1)$, it turns out that $c'_p \equiv c_p \pmod{2n-1}$ and, as $0 \leq c'_p, c_p < 2n-1$, it is proved that $c'_p = c_p$. □

3 Membrane computing formal framework

The chosen P system model for our framework is based in the works of Păun and Pérez-Jiménez in [26]. Here, a finite set of tissue like P systems working in a cooperative way is used, where each of them solves a part of the problem and, in some conditions, sends each output to other P system to be processed by it.

First of all, tissue like P systems are introduced. As it is well-known, the biological inspirations of this model are intercellular communication and cooperation between neurons [18,19]. The communication among cells is based on symport/antiport rules.[4] Tissue-like P systems have been widely used to solve computational problems in other areas (see e.g. [10,11]), but recently, they have been also used in the study of digital images (e.g., [3–5,12,23,24]).

In this paper, a variant of tissue-like P systems is used where the application of the rules are regulated by *promoters* and *inhibitors*. These promoters have a clear biological inspiration. The rule is applied if the reactants are present, but it is also necessary the presence of all the promoters and none of the inhibitors in the corresponding cell. The promoters are not *consumed* nor *produced* by the application of the rule, but if they are not in the cell, the rule cannot be applied. In one step, each reactant in a membrane can be used only for one rule, but if several rules need the presence of the same promoter, then the presence of one unique copy of the promoter suffices for the application of all the rules.

In the general case, if there are several possibilities, the rule is non-deterministically chosen, but sometimes a priority relation between rules is considered, so the concept of *priority* in our P systems is required.

Next, the formal definition of these P systems is recalled.

Definition 2 A *tissue P system with promoters, inhibitors, priorities and input* of degree $q \geq 1$ is a tuple of the form

$$\Pi = (\Gamma, \Sigma, \mathcal{E}, w_1, \ldots, w_q, \mathcal{R}, Pri, i_{in}, i_{out})$$

where q is the number of cells (or membranes) of the P system and

1. Γ is a finite alphabet, whose symbols are called objects. These objects can be placed in the cells or in the surrounding space (called the *environment*).

[4] Introduced in [25].

2. $\Sigma \subseteq \Gamma$ is the input alphabet. The *input* of the computation performed by the P system is encoded by using this alphabet.
3. $\mathcal{E} \subseteq \Gamma$ is a finite alphabet representing the set of the objects in the environment. Following a biological inspiration, the objects in the environment are available in an arbitrary large amount of copies;
4. $w_1, \ldots, w_q$ are strings over Γ representing the multisets of objects placed inside the cells at the beginning of the computation;
5. $\mathcal{R}$ is a finite set of rules of the following form:

$$(pro \, \neg inh \,|\, i, u/v, j), \quad \text{for } 0 \leq i \neq j \leq q, \;\; pro, inh, u, v \in \Gamma^*$$

6. Pri is a finite set of relations $R_i > R_j$, where R_i and R_j are rules from $\mathcal{R}$. It means that if R_i and R_j can be applied, then the application of R_i has *priority* on the application of R_j.
7. $i_{in} \in \{1, 2, \ldots, q\}$ denotes the input cell, i.e., the cell where the *input* of the computation will be placed.
8. $i_{out} \in \{1, 2, \ldots, q\}$ denotes the output cell, i.e., the cell where the *output* of the computation will be placed.

Informally, a tissue-like P system with promoters, inhibitors and priorities of degree $q \geq 1$ can be seen as a set of q cells labeled by $1, 2, \ldots, q$. The cells are the nodes of a virtual graph, where the edges connecting the cells are determined by the communication rules of the P system, i.e., as usual in tissue-like P systems, the edges linking cells are not provided explicitly: if a rule $(pro \, \neg inh \,|\, i, u/v, j)$ is given, then cells i and j are considered linked. The application of a rule $(pro \, \neg inh \,|\, i, u/v, j)$ consists of trading the multiset u (initially in the cell i) against the multiset v (initially in j). After the application of the rule, the multiset u disappears from the cell i and it appears in the cell j. Analogously, the multiset v disappears from the cell j and it appears in the cell i. The trade can also be between one cell and the environment, labeled by 0. The rule is applied if in the cell with label i the objects of pro are present in the cell i (*promoters*), while any of the objects in inh do not appear in the cell (*inhibitors*). The promoters or the inhibitors are not modified by the application of the rule. If the promoters and inhibitors are empty, we write $(i, u/v, j)$ instead of $(\emptyset \, \neg\emptyset |\, i, u/v, j)$. Finally, we write $(pro \,|i, u/v, j)$ or $(\neg inh \,|i, u/v, j)$ when only promoters or inhibitors appear, respectively.

As usual, some objects not belonging to $\mathcal{E}$ can arrive to the environment during a computation. So, in a configuration (not initial) it could be found two types of objects in the environment: firstly, those which belong to the environment and appear in an arbitrary large number of copies. Secondly, those which do not belong to the environment but have been sent to the environment by the application of a rule.

Rules are used as usual in the framework of membrane computing, that is, in a maximally parallel way (a universal clock is considered). A *configuration* is an instantaneous description of the P system and it is represented as a tuple $(w_0, w_1, \ldots, w_q)$, where w_0 is the multiset of objects from $\Gamma - \mathcal{E}$ placed in the environment (initially, $w_0 = \emptyset$). Given a configuration, we can perform a computation step and obtain a new configuration by applying the rules in a parallel manner as it is shown above. A configuration is *halting* when no rules can be applied to it. A *computation* is a sequence of computation steps such that either it is infinite or it is finite and the last step yields a halting configuration (i.e., no rules can be applied to it). Then, a computation halts when the P system reaches a halting configuration. The output of a computation is collected from its halting configuration by reading the objects contained in the output cell.

Definition 3 A dP scheme (of degree $s \geq 1$) is a construct

$$\Delta = (\Gamma, \Sigma, \Pi_1, \ldots, \Pi_s, R, i_{in}, i_{out})$$

where:

1. Γ is an alphabet of (communicated) objects.
2. $\Sigma \subset \Gamma$ is the input alphabet, used as input for the module $\Pi_{i_{in}}$.
3. $\Pi_1, \ldots, \Pi_s$ are tissue like P systems with promoters, inhibitors and priorities whose alphabets contain Γ. Each Π_i is called a component or module of Δ.
4. R is a set of distribution rules of the form $(pro\neg inh|i, u \rightarrow v, j)$ where pro, inh, u and v are multisets on Γ, $0 \leq i \neq j \leq s$ and $i \neq 0$. The objects in pro are called promoters of the rule and the objects in inh are called inhibitors, as usual in tissue like P systems with promoters and inhibitors.
5. i_{in} and i_{out} represents the modules used as input and output, respectively.

Informally, the evolution of a dP scheme can be summarized as follows. A universal common clock is used for all the modules in the scheme. Initially all the modules start evolving together, each of them with their respective initial configuration except the input module, which also uses the input alphabet Σ as input. On each computation step, every time a module reaches its halting configuration, a maximal multiset of distribution rules is selected. Then, we apply these rules so some objects are sent from the output membrane of some modules to the input membrane of another. Then, the evolution starts again until all the modules have reached their halting configuration and no distribution rule can be selected. We use an special destination module, indexed by 0. Every time a rule $(pro\neg inh|i, u \rightarrow v, 0)$ is executed, the multiset u is removed from the output membrane of the ith module. This is used as some kind of garbage collection.

4 Membrane computing framework for cell complexes algorithms

In previous section, the formal requirements for a Membrane Computing algorithm are presented, i.e. all the elements required to formally describe a P system. Hence, a language must be defined in order to characterize all the elements present in each membrane. The main goal in this paper is the introduction of a membrane computing framework flexible enough to be useful in many others applications than those presented below. Hence, we define now those essential notions in order to represent the objects and relations already outlined in Sect. 2.

Let $K \subset \overline{T}_n^k$ be a cubical complex. Let ∂ represents the boundary map in the associated chain complex. The cubical cells in K are represented as σ_i, for some $i \in \overline{I}_{n,k}$. The cubical geometry of K is achieved using objects $U^{\pm}_{ij,d}$ for $i, j \in \overline{I}_{n,k}$

and $1 \leq d \leq k$. The presence of such an object is interpreted as the cell σ_i is in the boundary of σ_j and the vector joining their barycenters (from lower dimension to higher dimension cell) is $\pm\frac{1}{2}\mathbf{u}_d$, where $\mathbf{u}_d$ is the d-th vector in the canonical base in $\mathbb{R}^k$.

As the cubical complex K is usually built from a binary k-D image[5] $\mathcal{I} : T_n^k \to \{0, 1\}$, black pixels in $\mathcal{I}^{-1}(1)$ are denoted as $B_\mathbf{i}$ for $\mathbf{i} \in T_n^k$.

The dimension of a cubical p-cell σ_i is represented by the presence of the object $\mathbf{d}_{ip}$, for $i \in \overline{I}_{n,k}$ and $0 \leq p \leq k$.

The boundary map is symbolized by objects $\delta_{ij}^{\pm}$ whenever $\langle \sigma_i, \partial\sigma_j \rangle = \pm 1$, respectively. Recall that cubical complexes are regular, so it is ensured that $\langle \sigma, \partial\mu \rangle \in \{0, \pm 1\}$ for any pair of cells $\sigma^{(p)}$ and $\mu^{(p+1)}$. Sometimes the knowledge of how many cells share a given cell in their border is required. The presence of objects $\mathbf{D}_{if}$ means that the set $\{\sigma_j \in K : \langle \sigma_i, \partial\sigma_j \rangle \neq 0\}$ has f elements.

Gradient vector fields are represented by the presence of objects $V_{ij}^{\pm}$ where the sign is given by the sign of $-\langle \sigma_i, \partial\sigma_j \rangle$ with $i, j \in \overline{I}_{n,k}$. Objects V_i depict that the cubical cell σ_i is used as part of a vector. If a cubical cell is critical for a given gradient vector field V, it is denoted by the presence of some object $\mathbf{C}_i$ for $0 \leq i \leq N$.

In order to stand for the operator $\tilde{\phi}$ defined above, objects $\phi_{ij}^{\pm}$, for $0 \leq i, j \leq N$, are used.

Hence, the common starting language for all the algorithms designed in the proposed framework is defined below:

$$\begin{aligned}
\Gamma_0 = \{B_\mathbf{i} : \mathbf{i} \in T_n^k\} &\cup \left\{\sigma_i : i \in \overline{I}_{n,k}\right\} \cup \{\mathbf{C}_i : i \in \overline{I}_{n,k}\} \\
&\cup \left\{\mathbf{d}_{ip} : i \in \overline{I}_{n,k} \wedge 0 \leq p \leq k\right\} \cup \left\{\mathbf{D}_{if} : i \in \overline{I}_{n,k} \wedge 0 \leq f \leq 2^k\right\} \\
&\cup \left\{\delta_{ij}^{\pm} : i, j \in \overline{I}_{n,k}\right\} \cup \left\{V_{ij}^{\pm} : i, j \in \overline{I}_{n,k}\right\} \cup \left\{V_i : i \in \overline{I}_{n,k}\right\} \cup \{V\} \\
&\cup \left\{U_{ij,p}^{\pm} : i, j \in \overline{I}_{n,k} \wedge 0 \leq p \leq k\right\} \cup \left\{\phi_{ij}^{\pm} : i, j \in \overline{I}_{n,k}\right\}
\end{aligned}$$

The framework presented here does not only consist of a language, but also some dP modules and distribution rules are defined. Firstly, the dP modules have to be such that allows performing only simple tasks in each module. Hence, the implementation of the membrane computing algorithm in current parallel devices is simplified. Secondly, the development of algorithms in computational algebraic topology usually reduces to simplify some cell complexes by using DMT. We specify here some modules related to the process of removal of cells which are the head and tails of discrete gradient vectors.

Previously mentioned common tasks are introduced as modules of a dP scheme. Namely, two main tasks are to be defined:

1. Build a cubical complex from a binary k-D image.
2. Specify the Morse complex from a given cubical complex and a gradient vector field.

[5] In case of 2D images, $B_\mathbf{i} = B_{i_1 i_2}$.

For any $i \in \overline{I}_{n,k}$, the cubical cell $\sigma(i) \in \mathcal{K}^k_{*,n}$ is defined as

$$\sigma(i) = \mathbb{T}^{-1}(\mathbb{I}^{-1}_{n,k}(i))$$

To determine the vertices of a cubical cell σ the function gen : $\mathcal{K}^k_n \to 2^{\mathbb{Z}^k}$ defined as

$$\text{gen}(I_1 \times \cdots \times I_k) = \{\inf I_1, \sup I_1\} \times \cdots \times \{\inf I_k, \sup I_k\} \tag{16}$$

is used. For example,

$$\begin{aligned} \text{gen}\,((1,2) \times (2) \times (0,1)) &= \{1,2\} \times \{2\} \times \{0,1\} \\ &= \{(1,2,0), (1,2,1), (2,2,0), (2,2,1)\} \end{aligned}$$

are the vertices of a square in $\mathbb{R}^3$.

Let σ and μ be cubical cells such that $\sigma \in \partial\mu$. Let $b(\sigma)$ and $b(\mu)$ represent the barycenter of cells σ and μ, respectively. Finally, denote the vector $\mathbf{b(\sigma)b(\mu)}$ as $\mathbf{v}_{\sigma\mu}$. For example, let $\sigma = (0,1) \times (2) \times (2,3)$ and $\mu = (0,1) \times (2,3) \times (2,3)$. Then, $\mathbf{v}_{\sigma\mu} = \left(0, \frac{1}{2}, 0\right)$ as the barycenters are, respectively, $b(\sigma) = \left(\frac{1}{2}, 2, \frac{5}{2}\right)$ and $b(\mu) = \left(\frac{1}{2}, \frac{5}{2}, \frac{5}{2}\right)$.

Notice that the barycenter of a cubical cell is defined as

$$b(I_1 \times \cdots \times I_k) = \left(\frac{\inf I_1 + \sup I_1}{2}, \ldots, \frac{\inf I_k + \sup I_k}{2}\right) \tag{17}$$

Definition 4 Let $\mathcal{I} : T^k_n \to \{0,1\}$ be a binary k-D image. The tissue like P system with promoters, inhibitors and priorities Π_{Cub} is defined as

$$\Pi_{\text{Cub}} = \{\Gamma, \Sigma, \mathcal{E}, w_1, \mathcal{R}, Pri, i_{\text{in}}, i_{\text{out}}\}$$

where

- $\Gamma = \Gamma_0$
- $\Sigma = \{B_{\mathbf{i}} : \mathbf{i} \in \mathcal{I}^{-1}(1)\}$
- $\mathcal{E} = \Gamma_0$
- $w_1 = \emptyset$
- $\mathcal{R}$ is the following set of rules:
  - $R_1 \equiv (\{B_{\mathbf{i}} : \mathbf{i} \in \text{gen}(\sigma(l))\} \neg\{\sigma_l\}|1, \lambda/\sigma_l\mathbf{C}_l, 0)$ for $l \in \overline{I}_{n,k}$
  - $R_2 \equiv \left(\{\sigma_i, \sigma_j\} \neg\{\delta^{\pm}_{ij}\}|1, \lambda/\delta^{\pm}_{ij}, 0\right)$ for $i, j \in \overline{I}_{n,k}$ and $\langle\sigma(i), \partial\sigma(j)\rangle = \pm 1$
  - $R_3 \equiv \left(\{\sigma_i, \sigma_j\} \neg\{U^{+}_{ij,d}\}|1, \lambda/U^{+}_{ij,d}, 0\right)$ for $i, j \in \overline{I}_{n,k}$, $1 \leq d \leq k$, $\langle\sigma(i), \partial\sigma(j)\rangle \neq 0$ and $\mathbf{v}_{\sigma(i)\sigma(j)} \cdot \mathbf{u}_d = \frac{1}{2}$

  Note that $\mathbf{u}_d$ is the d-th vector of the canonical base in $\mathbb{R}^k$.
- $Pri = \emptyset$
- $i_{\text{in}} = i_{\text{out}} = 1$

Theorem 4 *Given a k-D binary image $\mathcal{I}$, the corresponding P system Π_{Cub} builds the cubical complex for $\mathcal{I}$ in two computation steps.*

Proof (*Remark*) Let $\mathcal{I} : T_n^k \to \{0, 1\}$ be a binary k-D image. Let K be the cubical complex built from $\mathcal{I}$. In the initial configuration, the first membrane consists on objects $B_{\mathbf{i}}$ for $\mathbf{i} \in \mathcal{I}^{-1}(1)$. In this situation, only some rules in R_1 can be selected, because the other rules require the presence of objects σ_i. The rules in R_1 that can be selected are those for $l \in \overline{I}_{n,k}$ such that all the points in $\text{gen}(\sigma(l))$ are in $\mathcal{I}^{-1}(1)$. Therefore, the first computation step builds the objects σ_l that represents cubical cells in the complex and, also, introduce the mark for critical cells ($\mathbf{C}_l$).

In the second configuration, no rule in R_1 can be selected since the proper objects representing each cubical cell act as inhibitors for this rules. However, the rules in R_2 and R_3 are susceptible to be selected. In fact, the selected rules in R_2 introduce into the first membrane the objects that symbolize the boundary map ($\delta_{ij}^{\pm}$). The selected rules in R_3 introduce the objects that mark the (positive) direction of the boundary barycentric vector $\mathbf{v}_{\sigma\mu}$ for every pair of cells $\sigma, \mu \in K$ such that $\sigma \in \partial\mu$ and $\mathbf{v}_{\sigma\mu} \cdot \mathbf{u}_d = \frac{1}{2}$.

The application of rules in R_2 and R_3 prevent the selection of any other rule in R_2 or R_3 any time, because the inserted objects act as inhibitors. Hence, the next computation step (the third) makes the system reach the halting condition, inasmuch as no rule can be selected. □

Remark 1 If σ is in the boundary of μ, then both cells share all its intervals except one, degenerated in σ and non-degenerated in μ. We suppose here that this interval is the d-th. Then, all the coordinates of the vector $\mathbf{v}_{\sigma\mu}$ are 0, except the d-th one, whose value is $\frac{1}{2}$ and, hence, $\mathbf{v}_{\sigma\mu} = \frac{1}{2}\mathbf{u}_d$.

Definition 5 Let K be a cubical complex and ∂ the associated boundary map. Let V be a gradient vector field in K. The P system Π_{Flow} is defined as

$$\Pi_{\text{Flow}} = (\Gamma, \Sigma, \mathcal{E}, w_1, \mathcal{R}_{\text{Flow}}, Pri, i_{\text{in}}, i_{\text{out}})$$

where

- The alphabet of objects is given by $\Gamma = \Gamma_0 \cup \{E_{ij} : i, j \in \overline{I}_{n,k}, i \neq j\}$.
- The input alphabet is given by

$$\begin{aligned}\Sigma = \{\sigma_i : i \in \mathbb{I}_{n,k}(\mathbb{T}(K))\} &\cup \{\delta_{ij}^{\pm} : i, j \in \mathbb{I}_{n,k}(\mathbb{T}(K)) \wedge \langle \sigma(i), \partial\sigma(j) \rangle = \pm 1\} \\ &\cup \{V_{ij}^{\pm}, V_i, V_j : i, j \in \mathbb{I}_{n,k}(\mathbb{T}(K)) \wedge V(\sigma(i)) = \pm\sigma(j)\} \\ &\cup \{\mathbf{C}_i : i \in \mathbb{I}_{n,k}(\mathbb{T}(K)) \wedge V(\sigma(i)) = 0 \wedge V^{-1}(\sigma(i)) = \emptyset\} \\ &\cup \{E_{ij} : i, j \in \mathbb{I}_{n,k}(\mathbb{T}(K)) \wedge \langle \sigma(j), \partial V(\sigma(i)) \rangle \neq 0 \wedge i \neq j\}\end{aligned}$$

- The environment alphabet is given by $\mathcal{E} = \Gamma_0$.
- The set of rules $\mathcal{R}$ is given by
  - $R_1 \equiv \left(\{\sigma_i \sigma_j \sigma_l \delta_{il}^{s_1} \delta_{jl}^{s_2} V_{il}^{-s_1}\} \neg\{\phi_i \phi_j\} | 1, \lambda / \phi_{ij}^{-s_1 s_2} \phi_i \phi_j, 0\right)$ for $i, j, l \in \overline{I}_{n,k}$, $i \neq j \neq l \neq i$ and $s_1, s_2 \in \{+, -\}$
  - $R_2 \equiv \left(\{\sigma_i \mathbf{C}_i\} \neg\{\phi_{ii}^{+}\} | 1, \lambda / \phi_{ii}^{+}, 0\right)$ for $i \in \overline{I}_{n,k}$

  - $R_3 \equiv \left(\{\phi^{s_1}_{i_1 i_2} V^{s_2}_{i_2 j_1} \phi^{s_3}_{j_1 j_2}\}|1,\, E_{i_2 j_1}/\phi_{ij}^{-s_1 s_2 s_3}, 0\right)$ for $i, j, l \in \overline{I}_{n,k}$, $s_1, s_2, s_3 \in \{+, -\}$ and $i \neq j \neq l \neq i$.
  - $R_4 \equiv \left(\{\delta^{s_1}_{li} \mathbf{C}_i \mathbf{C}_j\} \neg \{\tilde{\delta}^{s_1 s_2}_{ji}\}|1,\, \phi^{s_2}_{lj}/\tilde{\delta}^{s_1 s_2}_{ji} \tilde{\mathbf{C}}_i \tilde{\mathbf{C}}_j \tilde{\sigma}_i \tilde{\sigma}_j, 0\right)$ for $i, j, l \in \overline{I}_{n,k}$ with $i \neq j \neq l \neq i$.
  - $R_5 \equiv \left(1, \tilde{\delta}^{+}_{ij} \tilde{\delta}^{-}_{ij}/\lambda, 0\right)$ for $i, j \in \overline{I}_{n,k}, i \neq j$.
- $Pri = \emptyset$
- $i_{\text{in}} = i_{\text{out}} = 1$.

Theorem 5 *Let $K \subset \mathcal{K}^k_{*,n}$ be a cubical complex and V a gradient vector field in K. The module Π_{Flow} computes the Morse complex for K associated to V in, at most, $4 + \lfloor \log_2 \nu \rfloor$ computation steps, where $\nu \leq n^k$ is the length of the greatest V-path on K.*

Proof (*Remark*) The initial configuration for Π_{Flow} consists in all the elements that codify the cubical complex K (objects σ_i for cubical cells and $\mathbf{C}_i$ marking critical cells), its boundary map (objects $\delta^{\pm}_{ij}$) and the gradient vector field (objects $V^{\pm}_{ij}$). There also are objects E_{ij} depicting that σ_i and σ_j are distinct maximal faces in $V(\sigma_i)$. In this situation, only rules in R_1 and R_2 can be selected, as the other rules require the existence of objects $\phi^{\pm}_{ij}$ introduced in the first membrane by these rules. The application of rules R_1 creates one object $\phi^{\pm}_{ij}$ for each pair of cubical cells $\sigma_i, \sigma_j \in K$ such that $\langle \tilde{\phi}(\sigma_i), \sigma_j \rangle = \pm 1$. Recall that $\tilde{\phi} : \mathcal{C} \to \mathcal{C}$ given by $\tilde{\phi} = \text{id} + \partial \circ V$ is the reduced (discrete time) flow of the gradient vector field [13].

The application of rules R_2 previously selected introduce objects ϕ^{+}_{ii} for each critical cubical cell σ_i, representing that, for each critical cell μ is $\tilde{\phi}(\mu) = \mu$. In the current situation, the second configuration, only rules in R_3, R_4 or R_5 are available to be selected.

Recall that objects $\phi^{\pm}_{ij}$ encodes the presence of a V-path from σ_i to σ_j with multiplicity equals to ± 1. Rules in R_3 concatenate two V-paths γ and γ' if there is a cell σ such that the ending cell of γ', named $\sigma(i)$, the starting cell of γ lay on its boundary and $V(\sigma(i)) = \sigma$. Hence, rules in R_3 are applied until a maximal path is constructed. This process needs, at most, $\log_2 \nu$ steps, where ν is the length of the longest maximal V-path.

Rules in R_4 get some objects from the environment, encoding the cells of the Morse complex. Indeed, for each critical path from σ_i to σ_j with multiplicity ± 1, there is an object $\phi^{\pm}_{ij}$, hence for each critical cubical cell σ_l with $\langle \sigma_i, \partial \sigma_l \rangle \neq 0$ there will be as many objects $\phi^{\pm}_{ij}$ as the number of critical V-paths from σ_i to σ_j with multiplicity ± 1, respectively.

Finally, rules in R_5 cancel pairs of objects $\tilde{\delta}^{+}_{ij}$, $\tilde{\delta}^{-}_{ij}$. The halting configuration of Π_{Flow} has as many objects $\tilde{\delta}^{\pm}_{ij}$ as the incidence of σ_i in the boundary of σ_j.

Notice that the entire process needs, at most, $4 + \lfloor \log_2 \nu \rfloor$ steps, where ν is the length of the longest critical V-path. □

In the following section, we show an example of the use of our framework by designing a dP scheme for solving one Computational Algebraic Topology related problem.

5 dP systems for the calculation of the homology groups for a 3D binary image

We deal here with a membrane algorithm for computing an optimal gradient vector field for a 3D digital object $\mathcal{I}$. Working with a cubical "cellularization" $K(\mathcal{I})$ of $\mathcal{I}$, we show that the critical cells of dimension p ($p = 0, 1, 2, 3$) of this particular gradient vector field coincide with the pth Betti number of $K(\mathcal{I})$.

Discrete Morse theory establishes a tool to simplify a cell complex[6] while the homology groups are kept. This can be used to calculate the homology groups of appropriate cell complexes. Concretely, the technique explained below require the complex to be torsion free.

An acyclic vector field over a cubical complex K is built, removing as many cubical cells as possible. In case that the resulting Morse complex has non-null differential, another acyclic vector field is constructed to accomplish the so called *critical cell cancellation* (see section 11 in [13]). In this way, a sequence of Morse reductions is built until a complex with null-differential is "reached". To ensure the finiteness of this process, the cubical complex has to be torsion free.

Figure 1 presents an example of the calculation of homology groups for a 2D cubical complex. In this example the strategy presented above is shown. First, an acyclic vector field is built such that almost every cubical p-cell σ is paired to one of its co-faces μ (cubical $(p+1)$-cell with $\sigma \in \partial\mu$). The algorithm used to determine the acyclic vector field returns two kind of critical cubical cells: on one hand, cubical p-cells that represent a p-dimensional hole and, on the other hand, 1-cells that represents an "step" configuration (as $\langle P_{11}, P_{12}\rangle$ or $\langle P_{15}, P_{16}\rangle$ in Fig. 1). Therefore, another acyclic vector field is required to cancel these spurious critical cells. Notice that the first Morse complex is not a cubical complex but a CW complex. However, it is usually much more simpler than the original cubical complex.

For binary 3D images, it suffices to build up to two acyclic vector fields to calculate an homotopically equivalent cell complex with null differential, which makes homology group calculation trivial. In the paragraphs below, two P modules are defined to perform this task: Π_{Vector} and Π_{Vector_2}.

Definition 6 Let $\mathcal{I} : T_n^3 \rightarrow \{0, 1\}$ be a k-D binary image. The module Π_{Vector} is defined as

$$\Pi_{\text{Vector}} = \{\Gamma, \Sigma, \mathcal{E}, w_1, \mathcal{R}, Pri, i_{\text{in}}, i_{\text{out}}\}$$

where

- $\Gamma = \Gamma_0$
- $\Sigma = \{\sigma_i : i \in \mathbb{I}_{n,k}(\mathbb{T}(K(\mathcal{I})))\}$
- $\mathcal{E} = \Gamma_0$
- $w_1 = \emptyset$
- $\mathcal{R}$ is the following set of rules:
  - $R_1 \equiv \left(\neg\{V_i V_j\}|1, \sigma_i\sigma_j\delta_{ij}^{\pm}\mathbf{d}_{ip}U_{ij,d}^{+}C_iC_j/\sigma_i\sigma_j\delta_{ij}^{\pm}U_{ij,d}^{+}V_{ij}^{\mp}V_iV_j, 0\right)$ for $i, j \in \overline{I}_{n,3}$, $i \neq j$, $0 \leq p \leq 3$ and $1 \leq d \leq 3$

[6] CW complex in [13] and cubical complex in this paper.

These rules take a critical cubical p-cell (σ_i), a critical cubical $(p+1)$-cell (σ_j) with $\sigma_i \in \partial\sigma_j$ and such that the barycentric vector $\mathbf{v}_{\sigma_i\sigma_j}$ is $\frac{1}{2}\mathbf{u}_d$. Then, objects $V_{ij}^{\pm}$ are brought from the environment (this represents that $V(\sigma_i) = \pm\sigma_j$) as long as objects V_i and V_j (used to prevent that a cubical cell is used in two different vectors in the vector field).

- The priorities are given by

$$\begin{aligned}Pri = \{&R_1[i,j,p,d] > R_1[i,j',p,d'] : i,j,j' \in \overline{I}_{n,3} \wedge 0 \le p \le 3 \wedge 1\\ &\le d < d' \le 3\} \cup \{R_1[i,j,p,d] > R_1[i',j,p,d'] : i,i',j \in \overline{I}_{n,3} \wedge 0\\ &\le p \le 3 \wedge 1 \le d < d' \le 3\} \cup \{R_1[i,j,p,d] > R_1[j,j',p+1,d'] :\\ &\ i,j,j' \in \overline{I}_{n,3} \wedge 0 \le p \le 2 \wedge 1 \le d \le 3\}\end{aligned}$$

  where $R_1[i,j,p,d]$ represents the corresponding instance of rule R_1 for the given values of the parameters.

  The priorities in the first set are used to assign to a cubical cell σ_i a cubical cell σ_j with the corresponding barycentric vector parallel to the vector $\mathbf{u}_d$ with the lowest d as possible.

  The priorities of the second kind are used to choose between two cubical cells σ_i and $\sigma_{i'}$ when another cubical cell σ_j can be assigned to be the image by V. The criteria is again to select the cubical cell with the lowest barycentric vector.

  The last priority type is used to choose whether a cubical cell is chosen as the source cell of a vector or the end cell of another. In this case it is used as the end cell.
- $i_{\text{in}} = i_{\text{out}} = 1$

Informally speaking, the vector field tries to "go" in the direction of $\mathbf{u}_1$. When it is not possible to go in that direction, it is used $\mathbf{u}_2$ and, if it cannot be performed, $\mathbf{u}_3$ is used. Hence, the flow of V can be thought as a waterfall going downside until it goes from left to right and, finally, rear to back.

Definition 7 Let $\mathcal{I} : T_n^3 \to \{0,1\}$ be a k-D binary image. The module Π_{Vector_2} is defined as

$$\Pi_{\text{Vector}_2} = \{\Gamma, \Sigma, \mathcal{E}, w_1, \mathcal{R}, Pri, i_{\text{in}}, i_{\text{out}}\}$$

where

- $\Gamma = \Gamma_0$
- $\Sigma = \{\sigma_i : i \in \mathbb{I}_{n,k}(\mathbb{T}(\mathcal{M}))\}$ where $\mathcal{M}$ is the Morse complex built from the acyclic vector field created by Π_{Vector}.
- $\mathcal{E} = \Gamma_0$
- $w_1 = \emptyset$
- $\mathcal{R}$ is the following set of rules:
  - $R_1 \equiv \left(\neg\{V_iV_j\}|1, \sigma_i\sigma_j\delta_{ij}^{\pm}C_iC_j/\sigma_i\sigma_j\delta_{ij}^{\pm}V_{ij}^{\mp}V_iV_jV, 0\right)$ for $i,j \in \overline{I}_{n,3}, i \neq j$. These rules take two cells that can be used to build a vector and create that vector. Notice that, once a cell is used in a vector, it will not be used in another.

- $Pri = \emptyset$.
- $i_{\text{in}} = i_{\text{out}} = 1$

This P module cancels pairs of critical cells of Π_{Vector} due to the previously mentioned "step" configuration.

The dP scheme

$$\Delta_{\text{Bin3DHom}} = (\Gamma_0, \Sigma, \Pi_{\text{Cub}}, \Pi_{\text{Vector}}, \Pi_{\text{Flow}}, \Pi_{\text{Vector}_2}, \mathcal{R}, i_{\text{in}}, i_{\text{out}})$$

with four modules defined below, computes the homology groups for the cubical complex generated from $\mathcal{I}$.

- $\Sigma = \{B_{\mathbf{i}} : \mathbf{i} \in \mathcal{I}^{-1}(1)\}$
- The modules are given by
  - Π_{Cub} This module builds a cubical complex from the 3D binary image $\mathcal{I}$.
  - Π_{Vector} This module creates a maximal acyclic vector field.
  - Π_{Flow} This module determines the Morse complex associated to the vector field created above.
  - Π_{Vector_2} This module creates a maximal acyclic vector field in the Morse complex.
- The distribution rules are given by
  - $\mathbf{R}_1 \equiv (\text{Cub}, \sigma_i \mathbf{C}_i \to \sigma_i \mathbf{C}_i, \text{Vector})$ for $i \in \overline{I}_{n,3}$
  - $\mathbf{R}_2 \equiv \left(\text{Cub}, \delta_{ij}^{\pm} \to \delta_{ij}^{\pm}, \text{Vector}\right)$ for $i, j \in \overline{I}_{n,3}$
  - $\mathbf{R}_3 \equiv \left(\text{Cub}, U_{ij,d}^{+} \to U_{ij,d}^{+}, \text{Vector}\right)$ for $i, j \in \overline{I}_{n,3}$ and $1 \leq d \leq 3$
  - $\mathbf{R}_4 \equiv \left(\text{Cub}, \mathbf{d}_{ip} \to \mathbf{d}_{ip}, \text{Vector}\right)$ for $i \in \overline{I}_{n,3}$ and $0 \leq p \leq 3$
  - $\mathbf{R}_5 \equiv (\text{Vector}, \sigma_i \to \sigma_i, \text{Flow})$ for $i \in \overline{I}_{n,3}$
  - $\mathbf{R}_6 \equiv (\text{Vector}, \mathbf{C}_i \to \mathbf{C}_i, \text{Flow})$ for $i \in \overline{I}_{n,3}$
  - $\mathbf{R}_7 \equiv \left(\text{Vector}, \delta_{ij}^{\pm} \to \delta_{ij}^{\pm}, \text{Flow}\right)$ for $i, j \in \overline{I}_{n,3}, i \neq j$
  - $\mathbf{R}_8 \equiv \left(\text{Vector}, \mathbf{d}_{ip} \to \mathbf{d}_{ip}, \text{Flow}\right)$ for $i \in \overline{I}_{n,3}$ and $0 \leq p \leq 3$
  - $\mathbf{R}_9 \equiv \left(\text{Vector}, V_{ij}^{\pm} \to V_{ij}^{\pm}, \text{Flow}\right)$ for $i, j \in \overline{I}_{n,3}, i \neq j$
  - $\mathbf{R}_{10} \equiv \left(\text{Flow}, \tilde{\sigma}_i \tilde{\mathbf{C}}_i \to \sigma_i \mathbf{C}_i, \text{Vector}_2\right)$ for $i \in \overline{I}_{n,3}$
  - $\mathbf{R}_{11} \equiv \left(\text{Flow}, \tilde{\delta}_{ij}^{\pm} \to \delta_{ij}^{\pm}, \text{Vector}_2\right)$ for $i, j \in \overline{I}_{n,3}, i \neq j$
  - $\mathbf{R}_{12} \equiv (\{V\}|\text{Vector}_2, \sigma_i \to \sigma_i, \text{Flow})$ for $i \in \overline{I}_{n,3}$
  - $\mathbf{R}_{13} \equiv (\{V\}|\text{Vector}_2, \mathbf{C}_i \to \mathbf{C}_i, \text{Flow})$ for $i \in \overline{I}_{n,3}$
  - $\mathbf{R}_{14} \equiv \left(\{V\}|\text{Vector}_2, \delta_{ij}^{\pm} \to \delta_{ij}^{\pm}, \text{Flow}\right)$ for $i, j \in \overline{I}_{n,3}, i \neq j$
  - $\mathbf{R}_{15} \equiv \left(\{V\}|\text{Vector}_2, V_{ij}^{\pm} \to V_{ij}^{\pm}, \text{Flow}\right)$ for $i, j \in \overline{I}_{n,3}, i \neq j$
- $i_{\text{in}} = \text{Cub}$, $i_{\text{out}} = \text{Vector}_2$

The modules Π_{Vector} and Π_{Vector_2} are the only modules that do not rely on the framework. In other words, the user of the framework only has to deal, most of the times, with writing modules for calculating acyclic vector fields, as the tasks for building the cubical complex associated to the binary image and the building of the Morse complex associated to it, are left to the framework.

Theorem 6 *The dP scheme $\Delta_{Bin3DHom}$ defined above calculates the homology groups for a given binary 3D image $\mathcal{I} : T_n^3 \to \{0, 1\}$ in $O(log_2 n)$ computation steps.*

Proof (*Remark*) The result is proved by following the behavior of the dP scheme Δ_{Bin3DHom} for an input image $\mathcal{I}$. First of all, the input alphabet for Π_{Cub} is set up from $\mathcal{I}$ as Π_{Cub} is the input module. No distribution rules can be selected until one module reaches its halting stage. By Theorem 4, Π_{Cub} halts in two steps. Then, the only distribution rules that can be selected are $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{R}_3$ and $\mathbf{R}_4$. After it application, the module Π_{Vector} starts calculating a vector field.

Rules R_1 in Π_{Vector} build a vector from a cell σ_i to a co-face σ_j. The priorities ensures that all the p-cells are mapped by V, the vector field, to a $p + 1$-cell, prioritizing that the vector joining its barycenter has as lowest index as possible. Also, the rules ensure that the vector field is acyclic, as all the vectors are "parallel" to the vectors in the canonical base in $\mathbb{R}^3$ with positive direction. This work is done in only one computation step. This way of building a vector field creates two kind of critical cells. First of all, there are critical cells due to homology generators. Second, there are critical cells due to "steps" in the cubical complex. In Fig. 1a, the cells $\langle P_{11}, P_{16}\rangle$ and $\langle P_{11}, P_{12}\rangle$ are in such position. It looks somewhat an "step" in a stair in a bottom view. This configuration of critical cell is due to the priority relation among rules in Π_{Vector}. Hence, we have two kinds of critical cells, those representing an homology generator and those cells that must have to be canceled in later computation steps.

Once Π_{Vector} halts, the only distribution rules that can be selected are $\mathbf{R}_5$ to $\mathbf{R}_9$, sending objects to Π_{Flow}. Concretely, those rules send cells (σ_i), boundary map ($\delta_{ij}^{\pm}$), critical cells ($\mathbf{C}_i$) and vector field $V_{ij}^{\pm}$.

Module Π_{Flow} calculates the Morse complex associated to the previous vector field in, at most, $4+3\log_2(n-1)$. To prove this note that the longest V-path in a 3D cubical complex with, at most, $(n-1)^3$ 1-cells have length, at most, $(n-1)^3$ and, then, use Theorem 5.

Once the Morse complex is built, it is sent to the module Π_{Vector_2} by distribution rules $\mathbf{R}_{10}$ and $\mathbf{R}_{11}$. This module builds an acyclic vector field that "removes" all the cells that do not represent an homology generator. This computation is completed in only one step. If some vector field is built, which means that there are cells that can be canceled, at least one object V comes from the environment that ensures the selection and application of distribution rules $\mathbf{R}_{12}$ to $\mathbf{R}_{15}$, sending the Morse complex and the gradient vector field to the module Π_{Flow}, where another Morse complex is built and sent to Π_{Vector_2}.

Once the second Morse complex has came to Π_{Vector_2} the dP scheme $\Delta_{Bin3DHom}$ reaches the halting state, as no other distribution rule can be selected. This takes place as the only critical cells that are not homology generators are 1-cells that are in a "step" configuration, and those cells are removed by the second vector field. Formally, we have reduced the initial cubical complex to another complex with null differential, which means that all the cells are generators of the homology groups.

Recall that the sum of the computation steps required by each module is, in the worst case, at most $12 + 6log_2(n-1)$, which is $O(\log_2 n)$. □

The full calculation process of homology groups can be followed in Fig. 1. The module Π_{Cub} builds the cubical complex in Fig. 1a in two computation steps. Then, the module Π_{Vector} builds the acyclic vector field in Fig. 1b in one computation step. Following, the module Π_{Flow} constructs the (first) Morse complex in 7 steps (because the longest V-path in the complex has length 6). Figure 1c shows the two kinds of critical cells. On one hand, P_{19} is a critical 0-cell due to an homology generator. On the other hand, P_{12} and P_{16} are 0-cells due to "step" configuration. $\langle P_5, P_6\rangle$ is a critical 1-cell due to "hole" while $\langle P_{11}, P_{12}\rangle$ and $\langle P_{15}, P_{16}\rangle$ are critical 1-cells originated by a "step" configuration. Next, the module Π_{Vector_2} specify, in one computation step, another acyclic vector field in the complex in Fig. 1d which leads to the computation in 5 computation steps of the homological complex in Fig. 1e, that has null differential, so that all the cells are homology generators.

6 Conclusions and future work

In this paper a theoretical massively parallel framework to develop algorithms in homological analysis of 3D imagery has been presented. We focus our study in 3D digital images, due to the fact that "linear topology" or homology information can incorporate torsion algebraic data in higher dimension beyond classical DMT context. In fact, the computatibity problem of torsion homology subgroups for kD ($k \geq 4$) digital objects is a priority at long-term for this incipient and worthwhile research line.

At short term, we intend to implement this set of algorithms in massively parallel hardware as graphics cards supporting CUDA. We also work on applications of our parallel theoretical framework to other image processing tasks, as segmentation or skeletonization.

We are also interested in the representational power of our framework, as a parallel implementation of the works of Real and Molina-Abril [21,28].

References

1. Bauer, U., Lange, C., Wardetzky, M.: Optimal topological simplification of discrete functions on surfaces. Discrete Comput. Geom. **47**(2), 347–377 (2012)
2. Berciano, A., Molina-Abril, H., Real, P.: Searching high order invariants in computer imagery. Appl. Algebra Eng. Commun. Comput. **23**(1–2), 17–28 (2012)
3. Carnero, J., Díaz-Pernil, D., Gutiérrez-Naranjo, M.A.: Designing tissue-like P systems for image segmentation on parallel architectures. In: del Amor, M.A.M., Păun, Gh., de Mendoza, I.P.H., Romero-Campero, F.J., Cabrera, L.V. (eds.) Ninth Brainstorming Week on Membrane Computing, pp. 43–62. Fénix Editora, Sevilla, Spain (2011)
4. Christinal, H.A., Díaz-Pernil, D., Real, P.: Segmentation in 2D and 3D image using tissue-like P system. In: Bayro-Corrochano, E., Eklundh, J.O. (eds.) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, Lecture Notes in Computer Science, vol. 5856, pp. 169–176 (2009)
5. Christinal, H.A., Díaz-Pernil, D., Real, P.: Region-based segmentation of 2D and 3D images with tissue-like P systems. Pattern Recogn. Lett. **32**(16), 2206–2212 (2011)
6. Christinal, H.A., Díaz-Pernil, D., Real, P.: P systems and computational algebraic topology. Math. Comput. Model. **52**(11–12), 1982–1996 (2010)
7. Cazals, F., Chazal, F., Lewiner, T.: In: Proceedings of the Nineteenth Annual Symposium on Computational Geometry, pp. 351–360. ACM (2003)
8. Díaz-Pernil, D., Christinal, H.A., Gutiérrez-Naranjo, M.A., Real, P.: Using membrane computing for effective homology. Appl. Algebra Eng. Commun. Comput. **23**(5–6), 233–249 (2012)

9. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Molina-Abril, H., Real, P.: Designing a new software tool for digital imagery based on P systems. Nat. Comput. **11**(3), 381–386 (2012)
10. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Pérez-Jiménez, M.J., Riscos-Núñez, A.: A linear-time tissue P system based solution for the 3-coloring problem. Electr. Notes Theor. Comput. Sci. **171**(2), 81–93 (2007)
11. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Pérez-Jiménez, M.J., Riscos-Núñez, A.: Solving subset sum in linear time by using tissue P systems with cell division. In: Mira, J., Álvarez, J.R. (eds.) IWINAC (1). Lecture Notes in Computer Science, vol. 4527, pp. 170–179 (2007)
12. Díaz-Pernil, D., Gutiérrez-Naranjo, M.A., Real, P., Sánchez-Canales, V.: Computing homology groups in binary 2D imagery by tissue-like P systems. Roman. J. Inf. Sci. Technol. **13**(2), 141–152 (2010)
13. Forman, R.: Morse theory for cell complexes. Adv. Math. **134**, 90–145 (1998)
14. Gunther, D., Reininghaus, J., Hotz, I., Wagner, H.: Memory-efficient computation of persistent homology for 3D images using discrete morse theory. In: 2011 24th SIBGRAPI Conference on Graphics, Patterns and Images (Sibgrapi), pp. 25–32 (2011)
15. Gyulassy, A., Bremer, P.T., Hamann, B., Pascucci, V.: A practical approach to Morse–Smale complex computation: scalability and generality. IEEE Trans. Vis. Comput. Graph. **14**(6), 1619–1626 (2008)
16. Kaczyński, T., Mischaikow, K., Mrozek, M.: Computational homology. Applied Mathematical Sciences. Springer, Berlin (2004)
17. Lewiner, T., Lopes, H., Tavares, G.: Applications of Forman's discrete Morse theory to topology visualization and mesh compression. IEEE Trans. Vis. Comput. Graph. **10**(5), 499–508 (2004)
18. Martín-Vide, C., Pazos, J., Păun, Gh., Rodríguez-Patón, A.: A new class of symbolic abstract neural nets: tissue P systems. In: Ibarra, O.H., Zhang, L. (eds.) COCOON. Lecture Notes in Computer Science, vol. 2387, pp. 290–299 (2002)
19. Martín-Vide, C., Păun, Gh., Pazos, J., Rodríguez-Patón, A.: Tissue P systems. Theoret. Comput. Sci. **296**(2), 295–326 (2003)
20. Molina-Abril, H., Real, P.: Homological optimality in discrete Morse theory through chain homotopies. Pattern Recogn. Lett. **33**(11), 1501–1506 (2012)
21. Molina-Abril, H., Real, P.: Homological spanning forest framework for 2D image analysis. Ann. Math. Artif. Intell. **64**(4), 385–409 (2012)
22. Nithin, S., Vijay, N.: Parallel computation of 3D Morse–Smale complexes. Comput. Graph. Forum **31**(3), 965–974 (2012)
23. Peña-Cantillana, F., Díaz-Pernil, D., Berciano, A., Gutiérrez-Naranjo, M.A.: A parallel implementation of the thresholding problem by using tissue-like P systems. In: Real, P., Díaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W.G. (eds.) CAIP (2). Lecture Notes in Computer Science, vol. 6855, pp. 277–284 (2011)
24. Peña-Cantillana, F., Díaz-Pernil, D., Christinal, H.A., Gutiérrez-Naranjo, M.A.: Implementation on CUDA of the smoothing problem with tissue-like P systems. Int. J. Nat. Comput. Res. **2**(3), 25–34 (2011)
25. Păun, A., Păun, Gh.: The power of communication: P systems with symport/antiport. New Gener. Comput. **20**(3), 295–306 (2002)
26. Păun, G., Pérez-Jiménez, M.J.: Solving problems in a distributed way in membrane computing: dP systems. Int. J. Comput. Commun. Control **5**(2), 238–252 (2010)
27. Peterka, T., Ross, R., Gyulassy, A., Pascucci, V., Kendall, W., Shen, H.-W., Lee, T.-Y., Chaudhuri, A.: Scalable parallel building blocks for custom data analysis. 2011 IEEE Symposium on Large Data Analysis and Visualization (LDAV), pp. 105–112 (2011)
28. Real, P., Molina-Abril, H., Kropatsch, W.: Homological tree-based strategies for image analysis. Comput. Anal. Images Patterns (LNCS) **5702**, 326–333 (2009)
29. Reininghaus, J., Hotz, I.: Combinatorial 2d vector field topology extraction and simplification. In: Topological Methods in Data Analysis and Visualization, pp. 103–114. Springer, Berlin, Heidelberg (2011)
30. Reininghaus, J., Lowen, C., Hotz, I.: Fast combinatorial vector field topology. IEEE Trans. Vis. Comput. Graph. **17**(10), 1433–1443 (2011)
31. Ritter, G.X., Wilson, J.N., Davidson, J.L.: Image algebra: an overview. Comput. Vis. Graph. Image Process. **49**(3), 297–331 (1990)