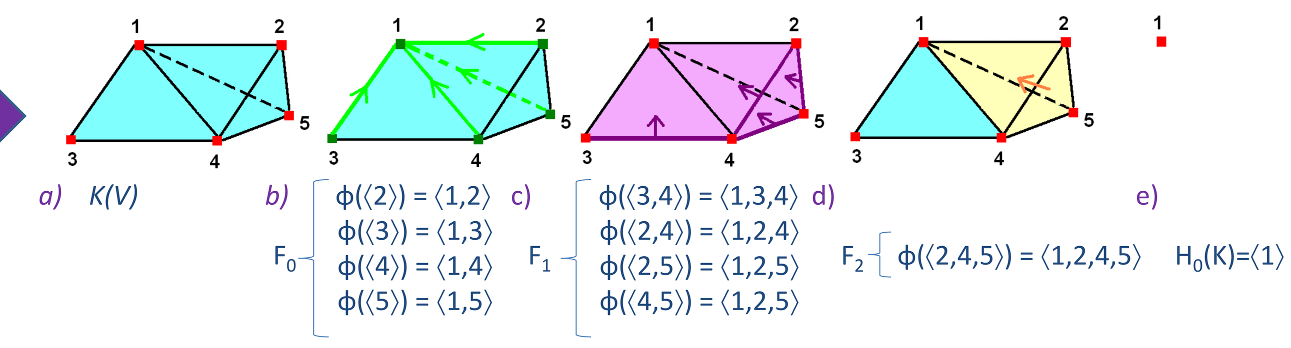
# Homological tree-based strategies for image analysis

**Pedro Real[1], Helena Molina-Abril[1,2], Walter G. Kropatsch[2]**
[1]Dpto. Matemática Aplicada I,Universidad de Sevilla,
real@us.es,habril@us.es
[2]PRIP, Vienna University of Technology, Austria
habril@prip.tuwien.ac.at

## Abstract

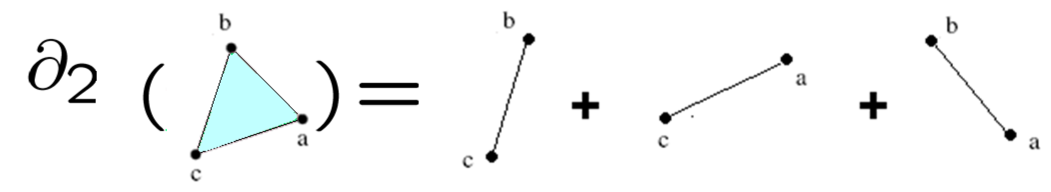
- $V$ → digital volume
- $K(V)$ → **cell complex** homologically equivalent to $V$
- **ϕ** → algebraic map over K(V) → Obtain homological characteristics of V (Betti numbers, homology generators, relations between them, etc.)

Describe **ϕ** using a mixed **three-level forest F**




## Homological information on cell complexes (ground ring $F_2 = \{0, 1\}$)

- (K, ∂) ➡ 3-d cell complex.
- **q-chain** ➡ $\sum \alpha_i \sigma_i$, $\sigma_i \in K(q)$, $\alpha_i \in F_2$. $\cdots \to C_{k+1} \xrightarrow{\partial_{k+1}} C_k \xrightarrow{\partial_k} C_{k-1} \to \cdots$
- q th chain complex ($C_q(K)$) ➡ Abelian group
- The boundary map $\partial_q(a)$ ➡ the collection of (q -1)-faces of a.
- **$\partial_q \partial_{q-1} = 0.$**
- **q-cycle** ➡ A chain a ∈ $C_q(K)$\ $\partial(a)_q=0$
- **q-boundary** ➡ A chain a ∈ $C_q(K)$\ $a=\partial_{q+1}(b)$ , b ∈ $C_{q+1}(K)$.
- **q th homology group** ➡ quotient group of q-cycles and q-boundaries.



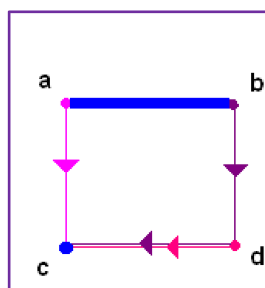

- $\partial_2(\langle a,b,c\rangle) = \langle a,b\rangle + \langle c,d\rangle + \langle b,a\rangle$
- $\partial_1(\langle a,b\rangle + \langle c,d\rangle + \langle b,a\rangle) = a+b+c+d+b+a = 0$

⬇

⟨a,b⟩ + ⟨c,d⟩ + ⟨b,a⟩ is a q-cycle and a q-boundary

## Homology gradient vector field (HGVF)

- Solve the Homology Problem (∂ =0 up to boundary) finding a map **ϕ**: $C_*(K) \to C_{*+1}(K)$
- **ϕ** defines an algebraic deformation process (chain homotopy). ➡ **ϕϕ = 0, ϕ∂ϕ = ϕ, ∂ϕ∂ = ∂.**
- **Homology groups** can be deduced in a straightforward manner from **ϕ.**
- The image of HGVF can be seen as a **mixed hierarchical forest**




- ϕ (⟨a⟩) = ⟨a,c⟩
- ϕ (⟨b⟩) = ⟨b,d⟩ + ⟨c,d⟩
- ϕ (⟨d⟩) = ⟨c,d⟩
- $H_0$= ⟨c⟩, $H_1$= ⟨a,b⟩

## Algorithm

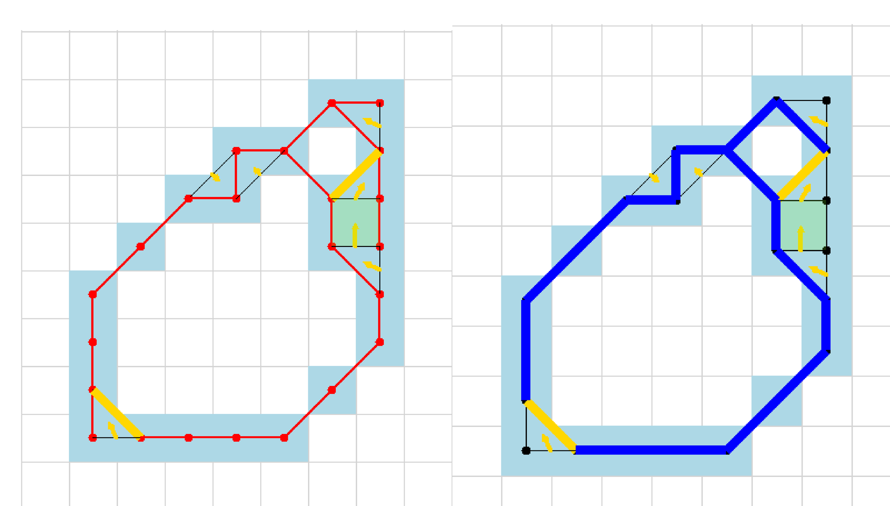
- Algorithm for computing **homological information** of K(V )

*Let* $(K,\partial)$ *be a finite cell complex with the filter* $K := \langle c_0, \ldots, c_m \rangle$
$\phi_0(c_0) := 0$
*For* $i := 0$ *to* $m$ *do*
  *define* $\bar{c}_i := c_i + \phi_{i-1}\partial_i(c_i)$
  $\forall e_{s_j} \in K_{i-1}$ *such that* $\partial_i(\bar{c}_i) = \sum_{j=1}^{r} \lambda_j e_{s_j}$
  *define* $\bar{e}_{s_j} := (id_{c(K_{i-1})} - \phi_{i-1}\partial_{i-1} - \partial_{i-1}\phi_{i-1})(e_{s_j}) \forall j = 1,\ldots,r$
  $\phi_i(c_i) := 0$
  *If* $\phi(c_i) := 0$ *then*
    *For* $j := 0$ *to* $i-1$ *do*
      $\phi_i(c_j) := \phi_{i-1}(c_j)$
  *Otherwise* *choose an element* $\bar{e}_{s_k} \neq 0$ *and define* $\tilde{\phi}(\bar{e}_{s_k}) := (\bar{c}_i)$ *and* $\tilde{\phi} := 0$ *otherwise*
  *For* $j := 0$ *to* $i-1$ *do*
    $\phi_i(c_j) := (\phi_{i-1} + \tilde{\phi}(id_{C(K_i)} + \phi_{i-1}\partial_i + \partial_{i-1}\phi_{i-1}))(c_j)$
  *OUTPUT: a homology gradient vector field* $\phi_m$ *for* $K$

Choose one filter

Chose one element

Determined in order to obtain as an output a homological forest



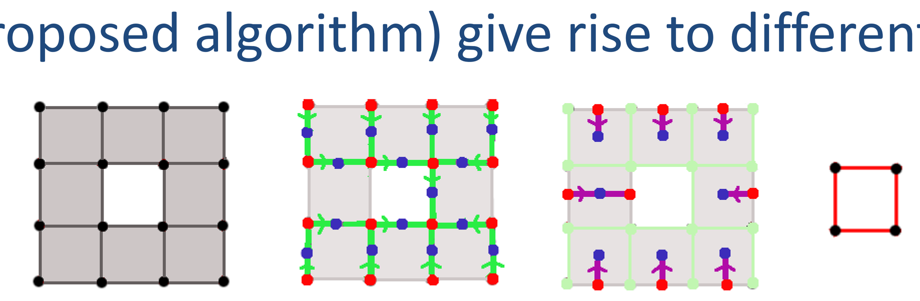
- The homology of K is given by **$\mathrm{Im}(1_{K_m} + \phi_m\partial_m + \partial_m \phi_m)$**

## Homological strategies

- Different strategies for computing **ϕ** (by changing only 2 parameters in the proposed algorithm) give rise to different maps.

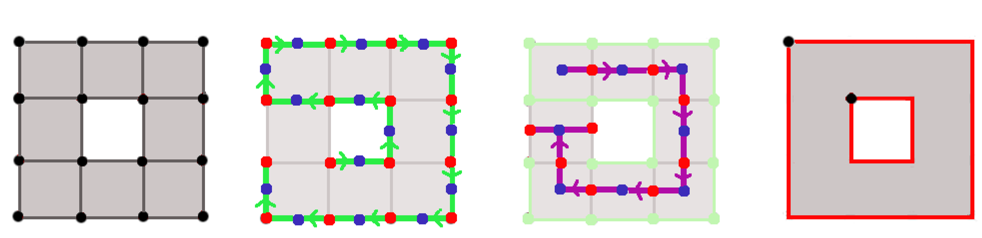
**Topological skeletons:**

- Filter → Taking first border cells.
- $e_{s_k}$ → is a free face of $\bar{c}_i$ (non-shared with other cells).



**Segmentation:**

- Filter → Taking first inner cells.
- $e_{s_k}$ → is a shared face of the inner cell $\bar{c}_i$



**Topological Image Pyramids:**

- Operations used to construct an irregular graph pyramid (contraction and removal) are HGVF → an irregular graph pyramid can be directly built using the proposed Algorithm
  - Given two cells $c_i$ and $c_j$ sharing a face $u_s$ and defining $\tilde{\phi}(u_s) = c_i$
    ↳ the cell $u_s$ will be removed and $c_i$ and $c_j$ will be merged
- The advantage of using this algorithm for building a pyramid is the complete topological control during the whole process, and the possibility of directly compute topological invariants at each level of the pyramid.