SYNASC 2014 - September 22-25, 2014

5th International Workshop on Computational Topology in Image Context

EFFICIENT COMPUTATION OF SIMPLICIAL HOMOLOGY THROUGH ACYCLIC MATCHING

Ulderico Fugacci, Federico Iuricich, Leila De Floriani

University of Genova, Italy

University of Maryland, MD, USA

MOTIVATION

Apply *topological methods* to the *description* and the *analysis of shapes*

We are interested in:

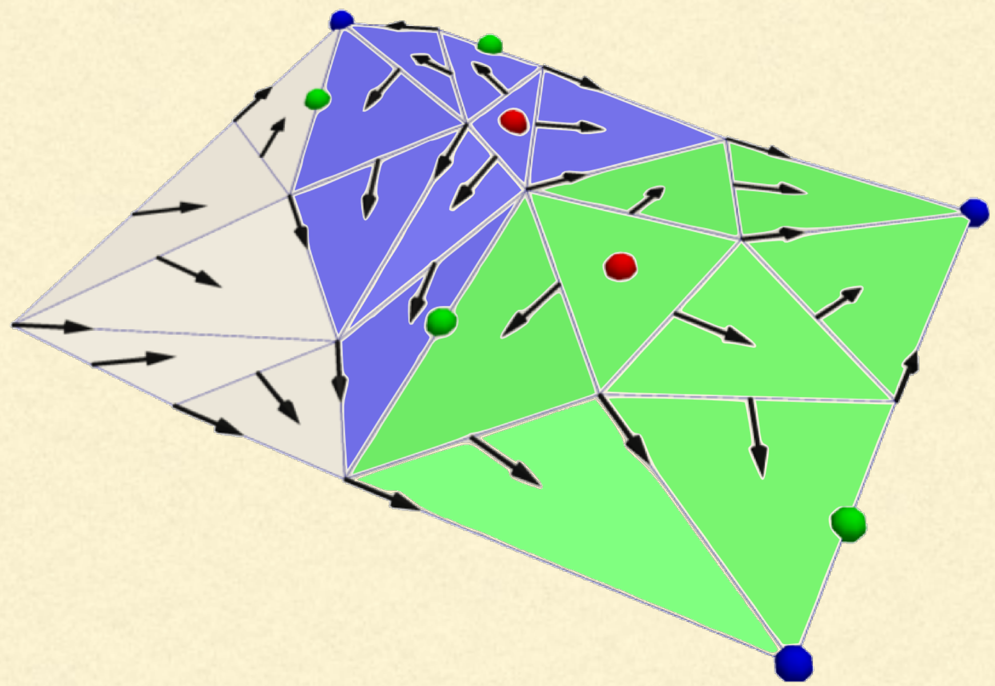
- *Large-size* data
- *High-dimensional* data

Main tool: Simplicial Homology

OUTLINE

- **Background notions**
  - Discrete Morse theory
  - Reductions and coreductions
- **Discrete Morse theory through reductions and coreductions**
  - Reduction-based and coreduction-based approach
  - Equivalence of the two approaches
  - Interleaving reductions and coreductions
- **Our algorithm**
  - Efficient encoding for the simplicial complex
  - Efficient encoding for the gradient vector field
- **Conclusions**

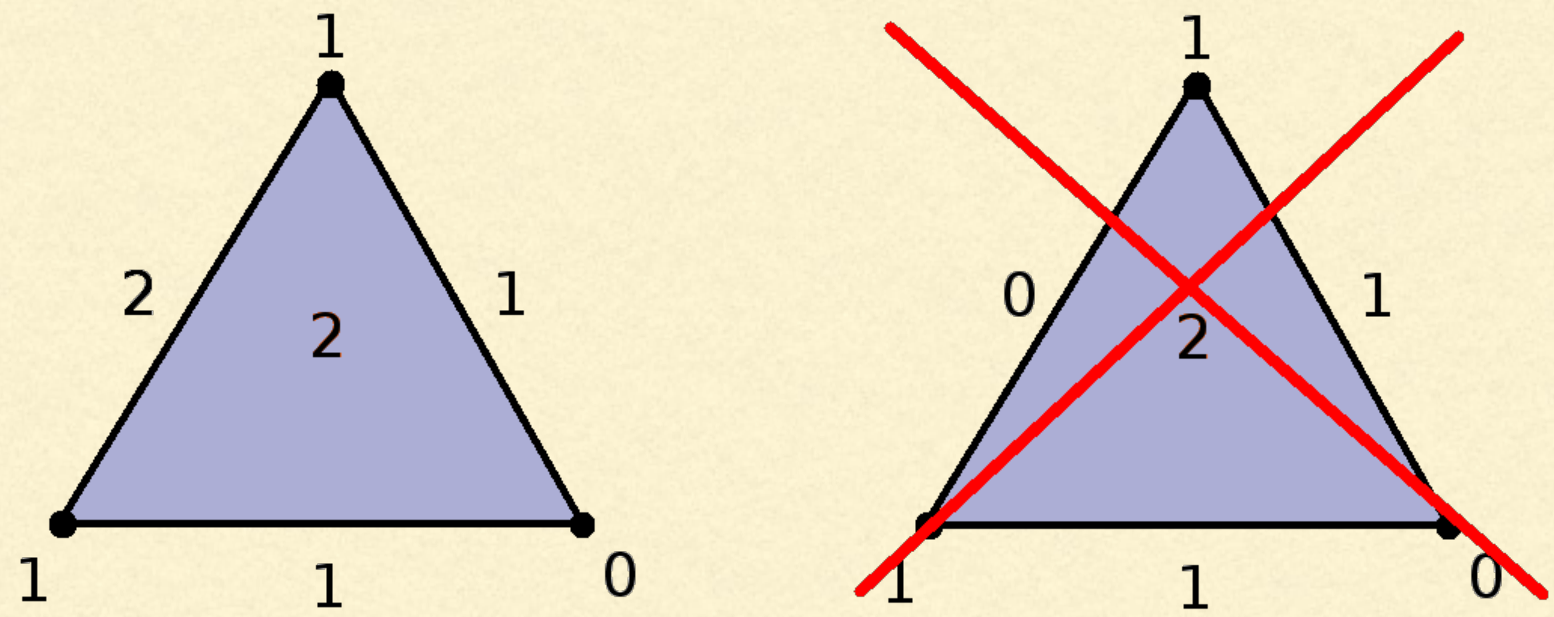
DISCRETE MORSE THEORY
[FORMAN 1998]



- *Combinatorial counterpart* of Morse theory [Milnor 1963]
- Introduced for *CW complexes*
- Gives a compact *homology-equivalent model* for a shape
- Provides *topological invariants* from a gradient vector field

DISCRETE MORSE FUNCTION

Let Σ simplicial complex

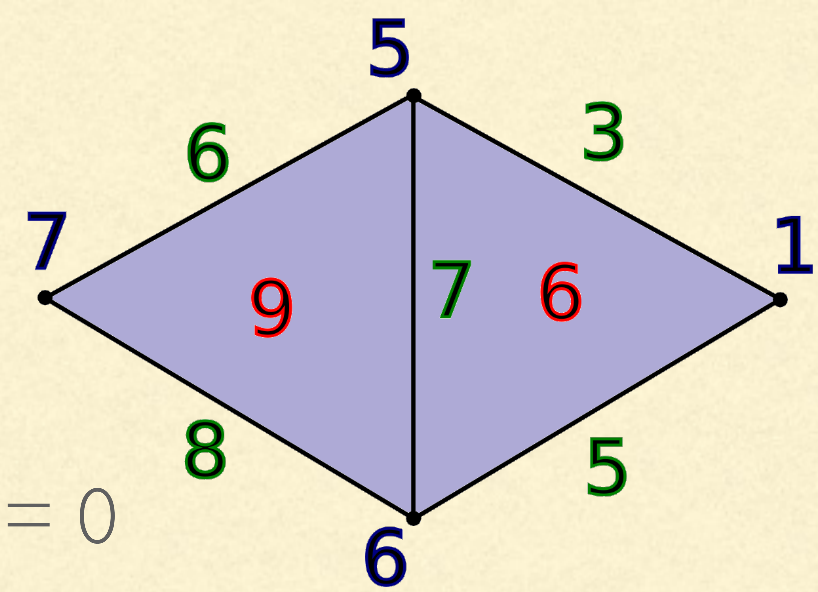


f : Σ → ℝ is called ***discrete Morse function*** if, for every simplex σ,

$$\# \{ \rho > \sigma \mid f(\rho) \leq f(\sigma) \} \leq 1$$

$$\# \{ \tau < \sigma \mid f(\tau) \geq f(\sigma) \} \leq 1$$

DISCRETE MORSE COMPLEX



A k-simplex σ is ***critical*** with index k if

$\# \{ \rho > \sigma \mid f(\rho) \leq f(\sigma) \} = \# \{ \tau < \sigma \mid f(\tau) \geq f(\sigma) \} = 0$

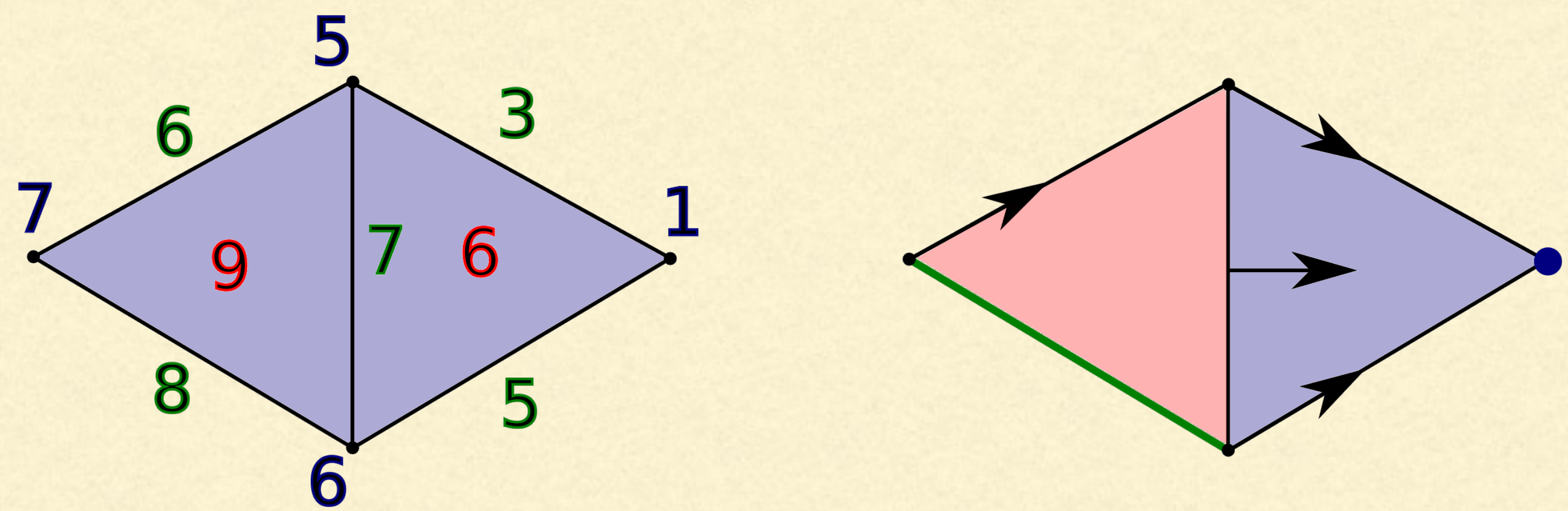
Critical simplices generate a chain complex $\mathscr{M}_*$ called

discrete Morse complex

Proposition. $H_k(\mathscr{M}_*) \cong H_k(\Sigma)$

DISCRETE MORSE FUNCTION AND GRADIENT VECTOR FIELD

A ***discrete vector field V*** on Σ is a collection of pairs of simplices $(\tau, \sigma) \in \Sigma \times \Sigma$ such that $\tau < \sigma$ and each simplex of Σ is in at most one pair of V



A discrete Morse function $f : \Sigma \to \mathbb{R}$ induces a discrete vector field on Σ

$$V = \{ (\tau, \sigma)\ \Sigma \times \Sigma \mid \tau < \sigma \text{ and } f(\tau) \geq f(\sigma) \}$$

called the ***gradient vector field*** of f

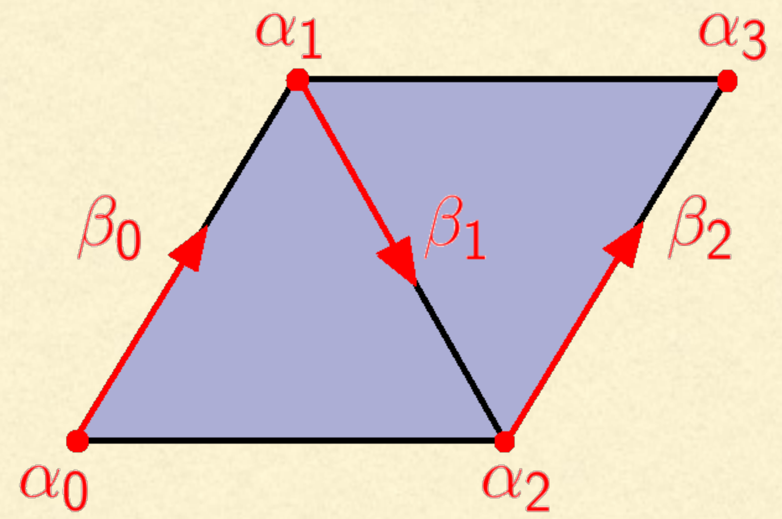
DISCRETE MORSE FUNCTION AND GRADIENT VECTOR FIELD

Given a discrete vector field V, a ***gradient path*** is a sequence of simplices of Σ

$$\alpha_0, \beta_0, \alpha_1, \beta_1, \alpha_2, \dots, \alpha_{r-1}, \beta_{r-1}, \alpha_r$$

where $(\alpha_i, \beta_i) \in V$, $\alpha_{i+1} < \beta_i$ and $\alpha_i \neq \alpha_{i+1}$

A gradient path is a ***non-trivial closed path*** if $r \geq 0$ and $\alpha_0 = \alpha_r$



Theorem. A discrete vector field V is the gradient vector field of a discrete Morse function if and only if there are no non-trivial closed paths

REDUCTIONS AND COREDUCTIONS
[MROZEK ET AL. 2009]

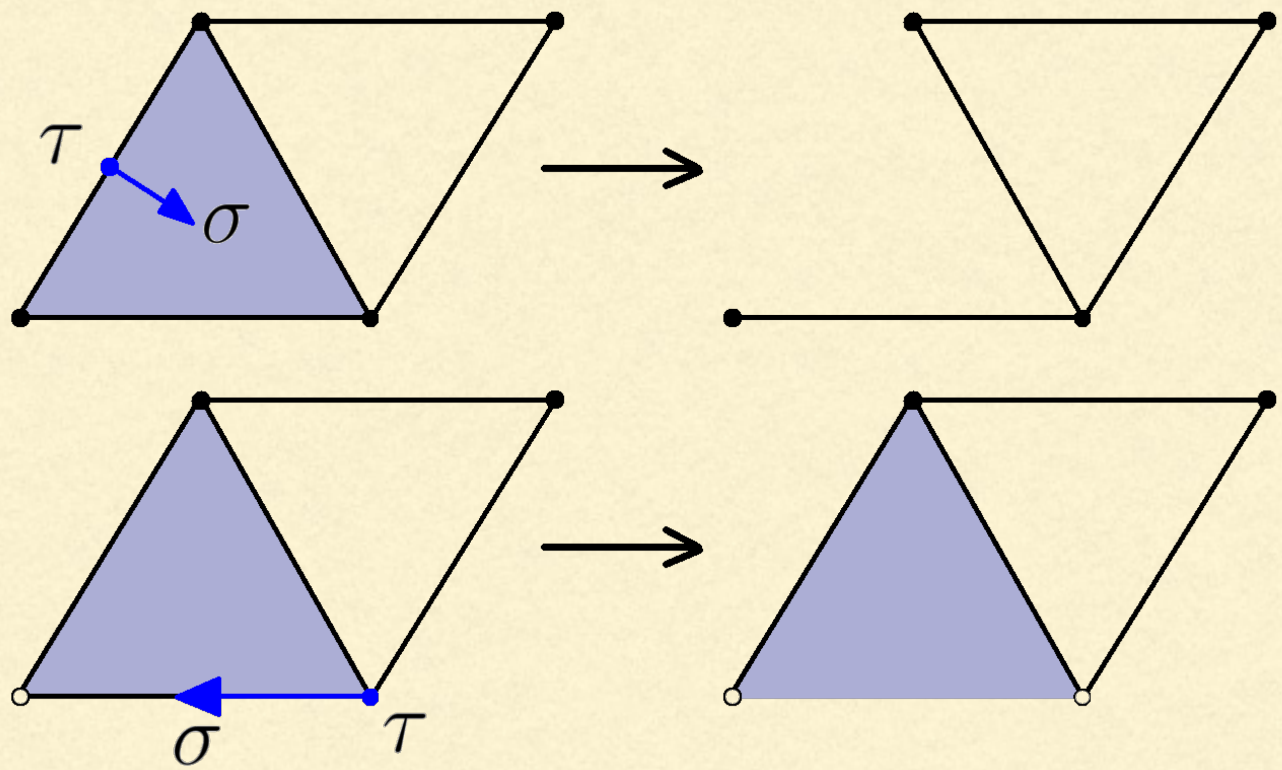
Let (τ, σ) be a pair of Σ such that $< \partial\sigma, \tau >= \pm 1$

(τ, σ) is a ***reduction pair*** if

$\mathrm{cbd}_\Sigma \tau = \{\sigma\}$

(τ, σ) is a ***coreduction pair*** if

$\mathrm{bd}_\Sigma\, \sigma = \{\tau\}$



Proposition. The removal of a reduction or of a coreduction pair is a homology-preserving operator

GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

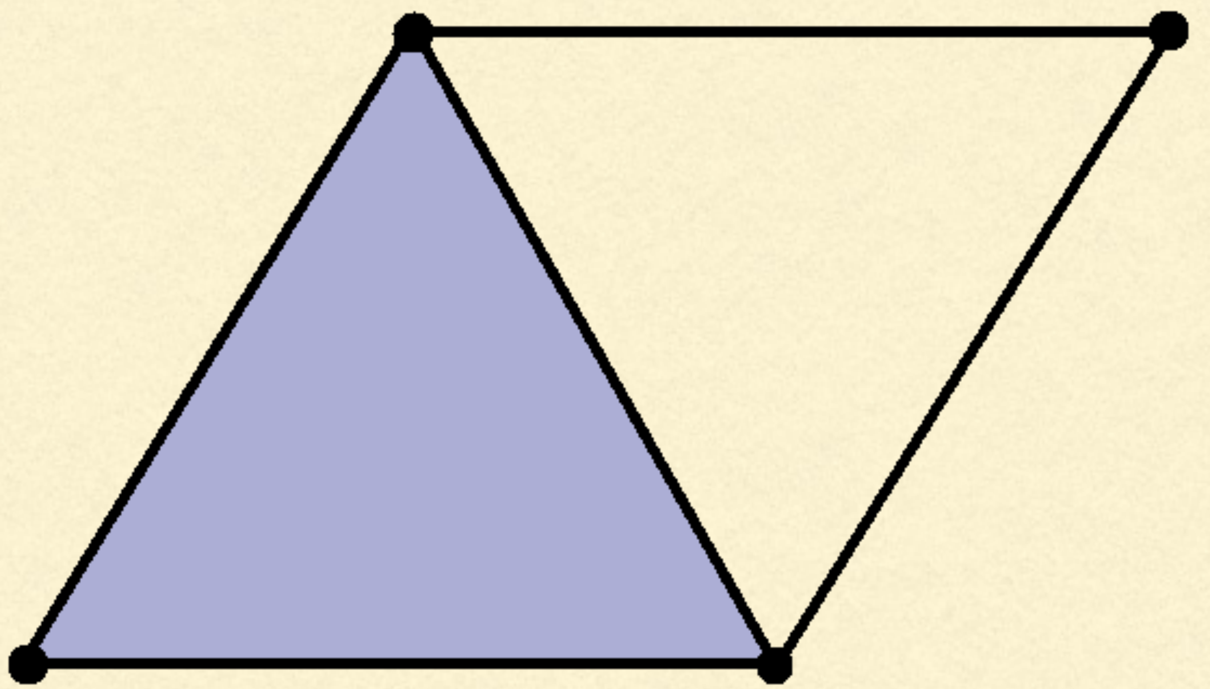
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

  while Σ' admits a *reduction pair* (τ, σ) do
    V ← V ∪ { (τ, σ) }
    Σ' ← Σ' \ { τ, σ }
  end while

  Let η be a *top simplex* in Σ'
    A ← A ∪ { η }
    Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

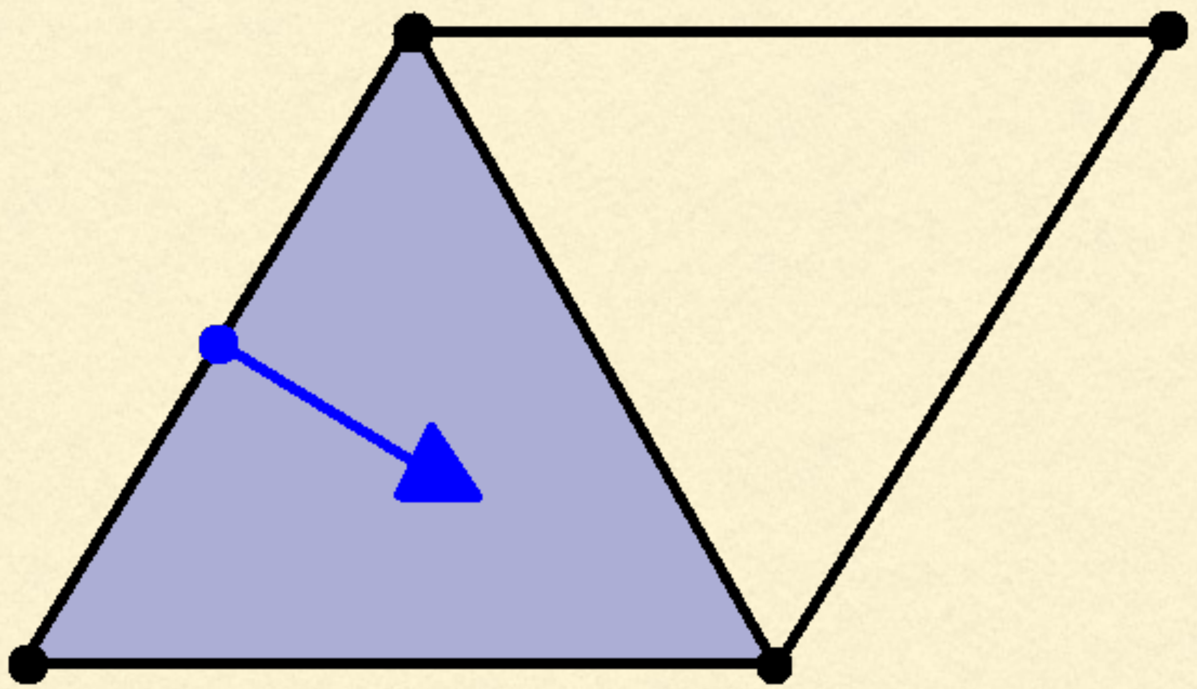
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

  while Σ' admits a *reduction pair* (τ, σ) do
    V ← V ∪ { (τ, σ) }
    Σ' ← Σ' \ { τ, σ }
  end while

  Let η be a *top simplex* in Σ'
    A ← A ∪ { η }
    Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

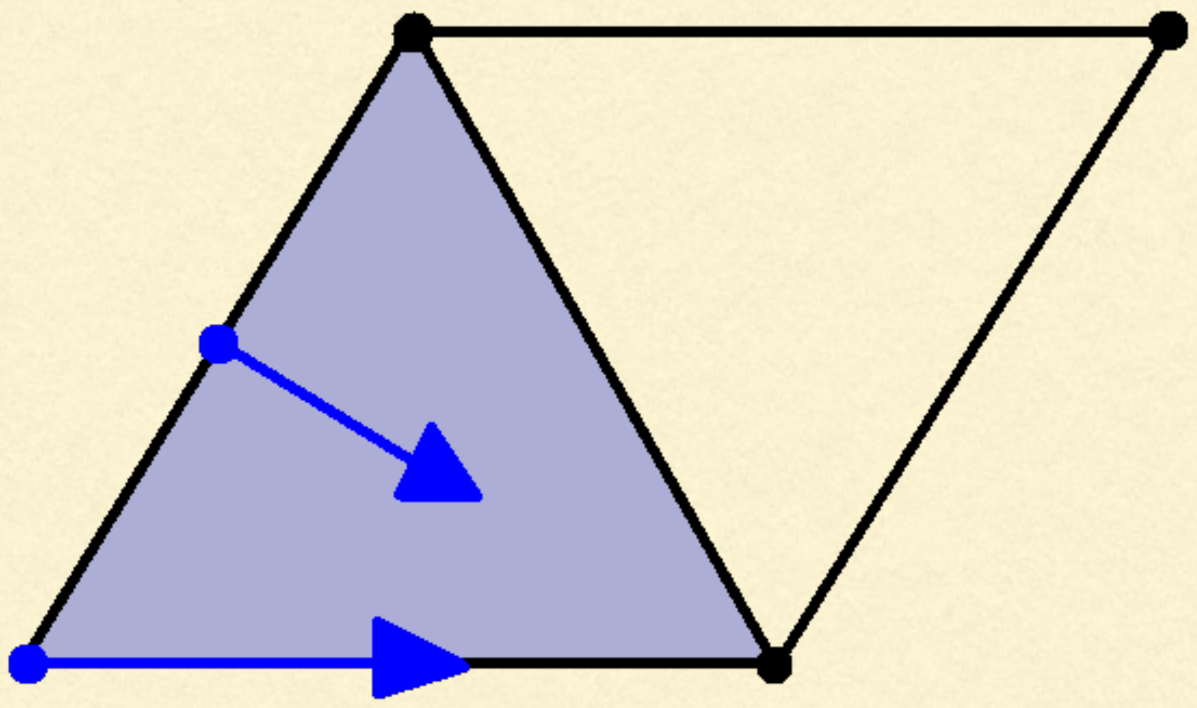
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

  while Σ' admits a *reduction pair* (τ, σ) do
    V ← V ∪ { (τ, σ) }
    Σ' ← Σ' \ { τ, σ }
  end while

  Let η be a *top simplex* in Σ'
    A ← A ∪ { η }
    Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

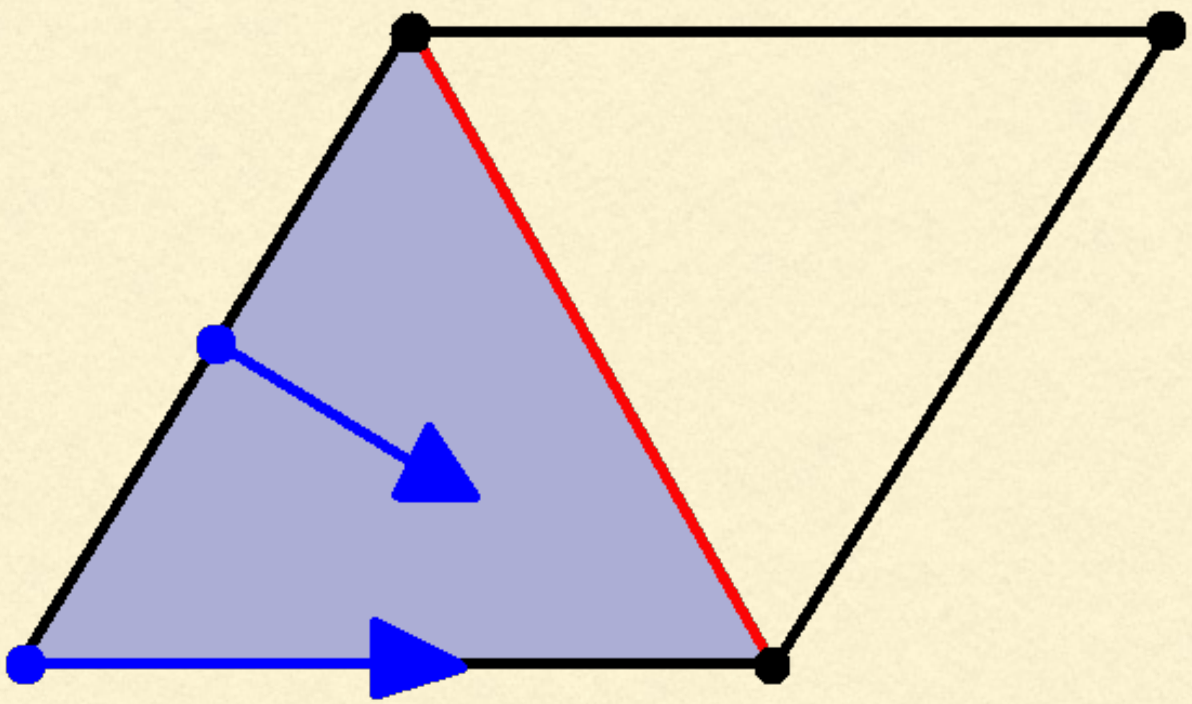
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *reduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *top simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

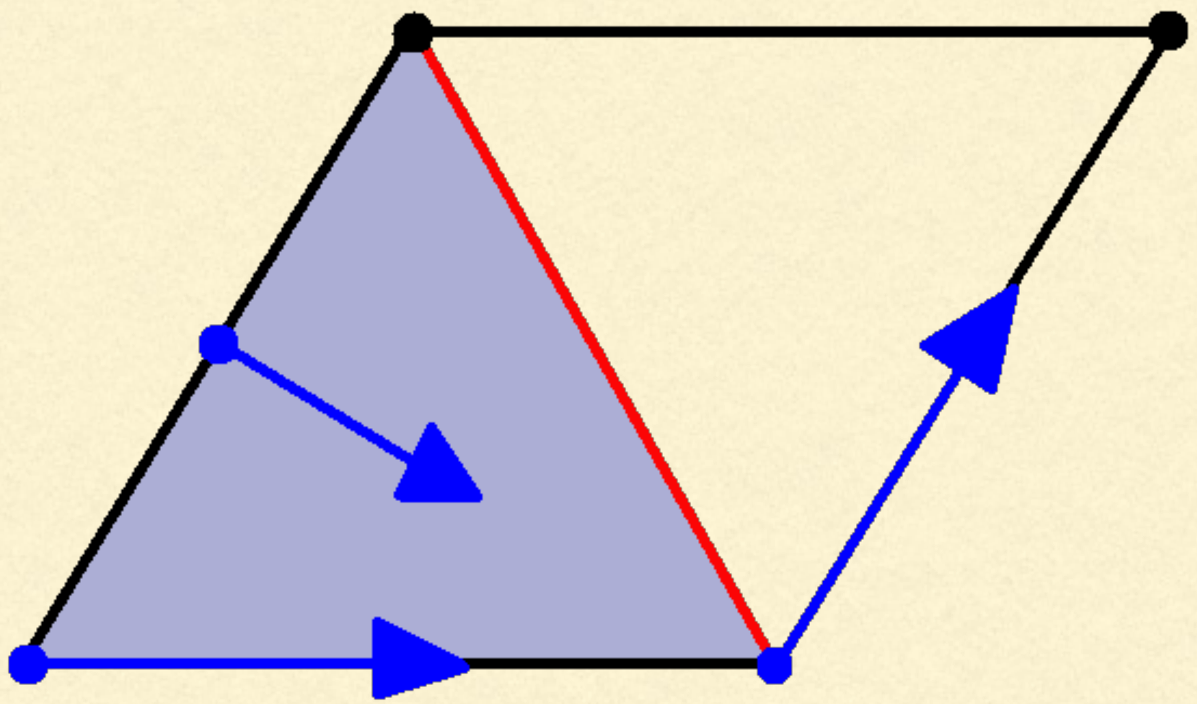
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *reduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *top simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

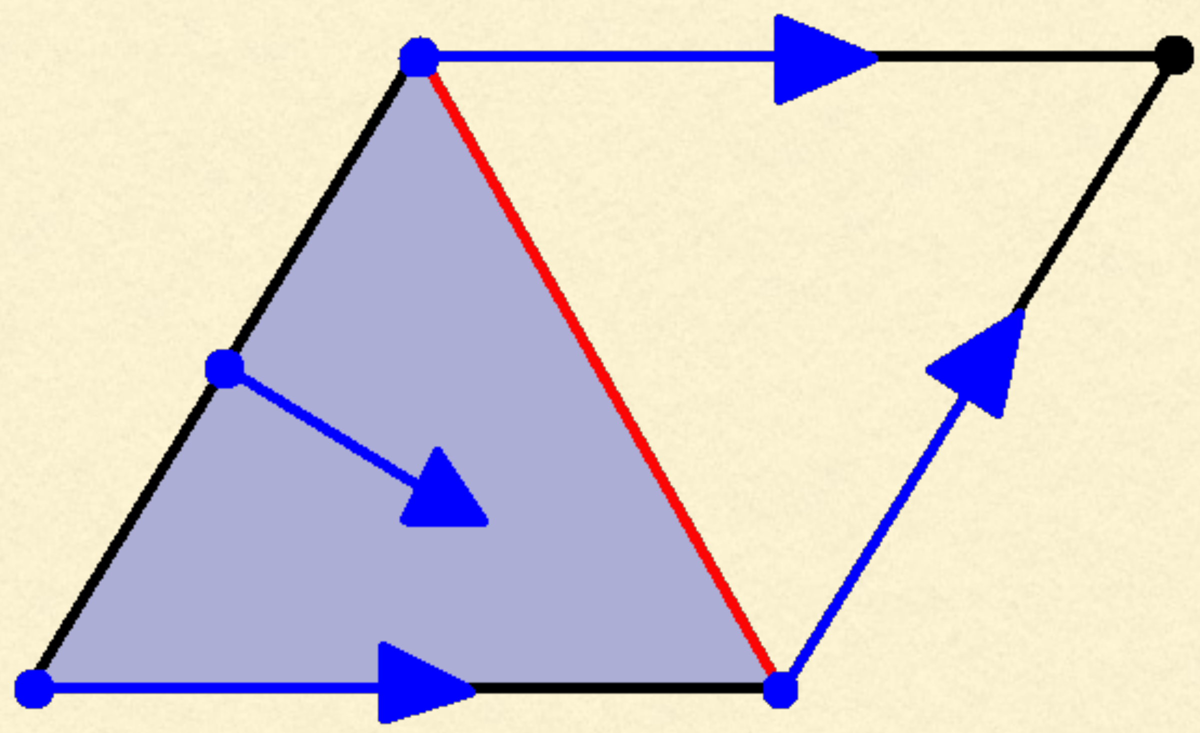
Set $\Sigma' \leftarrow \Sigma$, $V \leftarrow \varnothing$, $A \leftarrow \varnothing$

while $\Sigma' \neq \varnothing$ do

  while Σ' admits a *reduction pair* (τ, σ) do
    $V \leftarrow V \cup \{ (\tau, \sigma) \}$
    $\Sigma' \leftarrow \Sigma' \setminus \{ \tau, \sigma \}$
  end while

  Let η be a *top simplex* in Σ'
    $A \leftarrow A \cup \{ \eta \}$
    $\Sigma' \leftarrow \Sigma' \setminus \{ \eta \}$

end while



GRADIENT VECTOR FIELD BY REDUCTIONS
[BENEDETTI ET AL. 2009]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

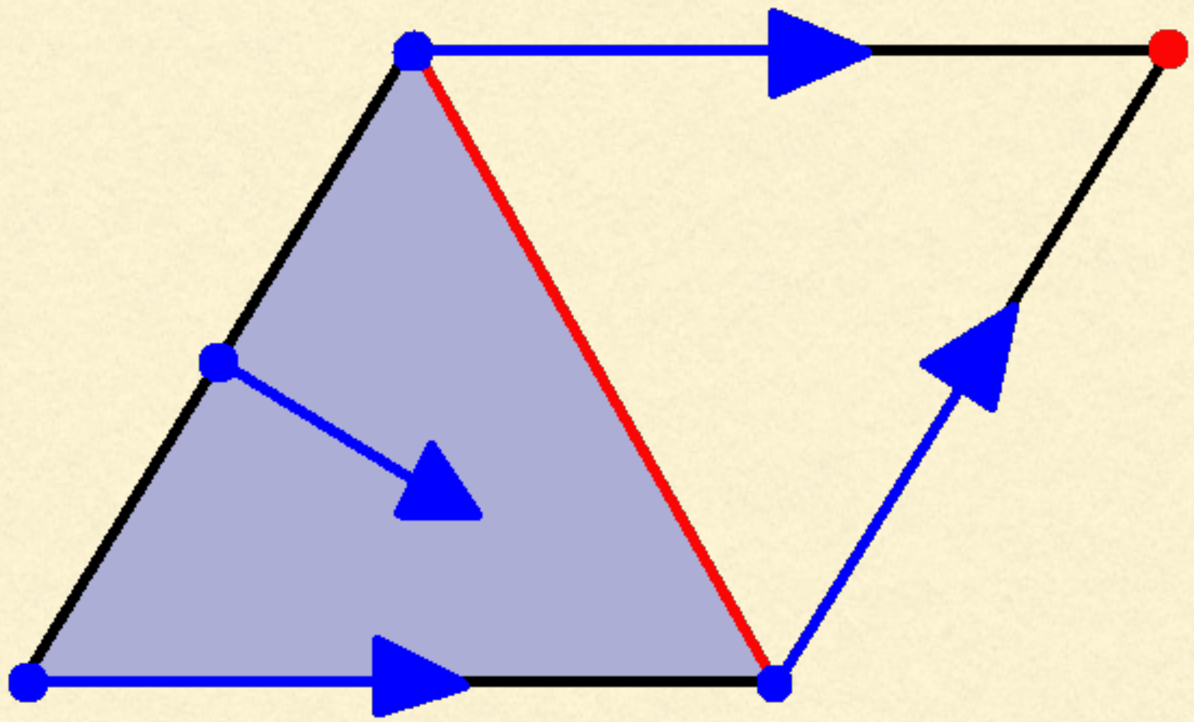
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *reduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *top simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

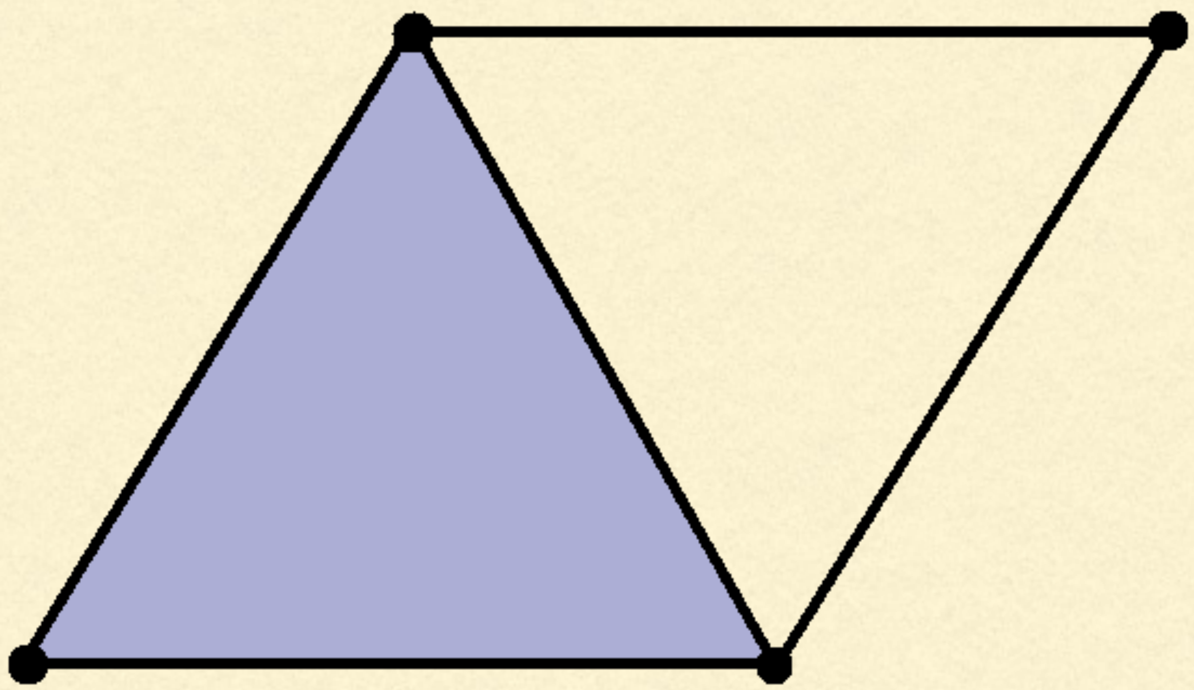
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *coreduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *free simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

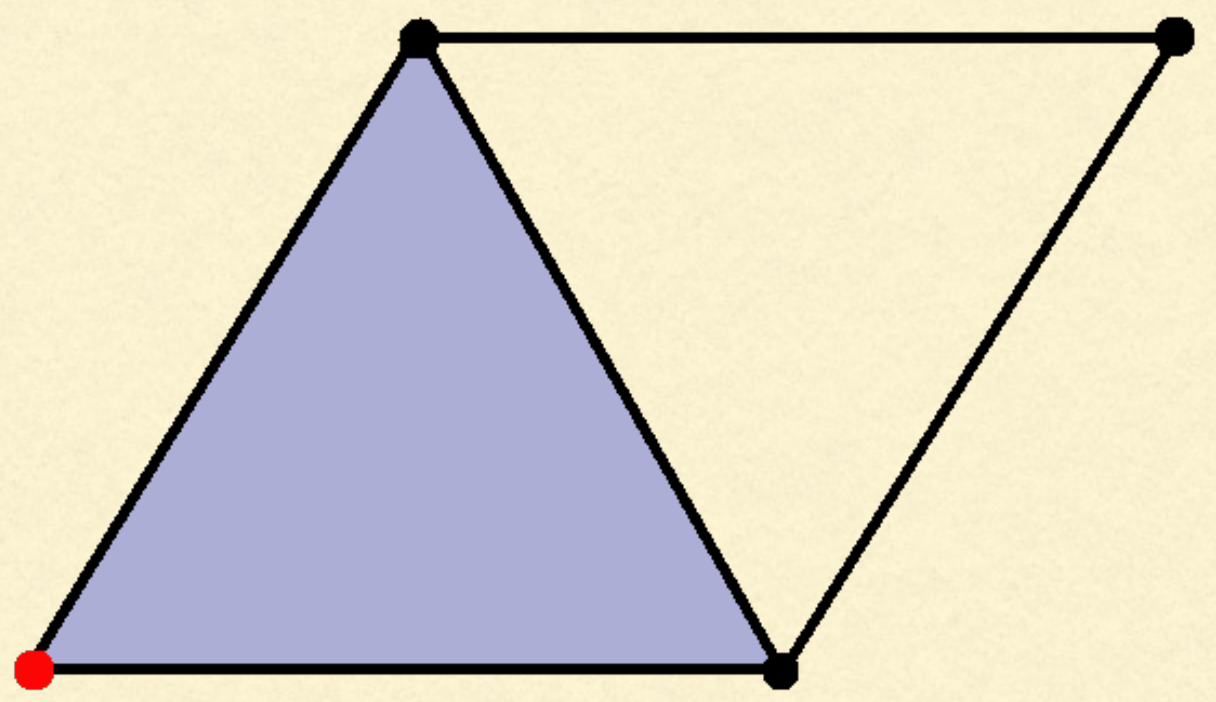
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *coreduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *free simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

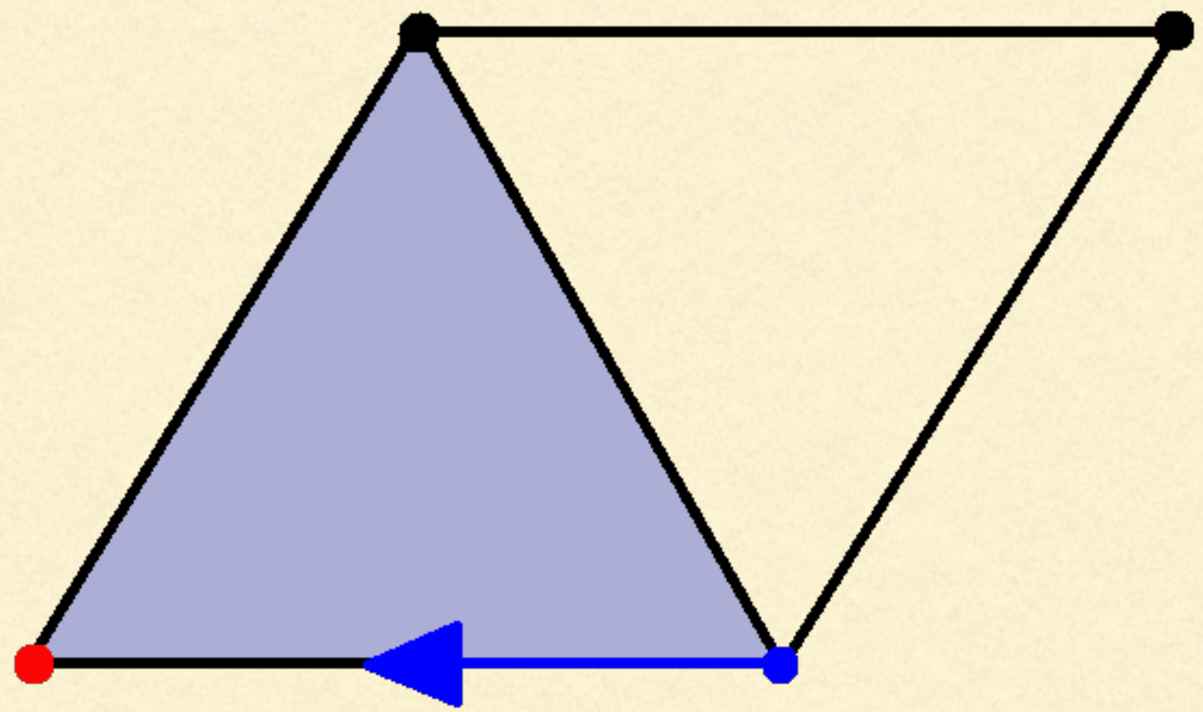
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *coreduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *free simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

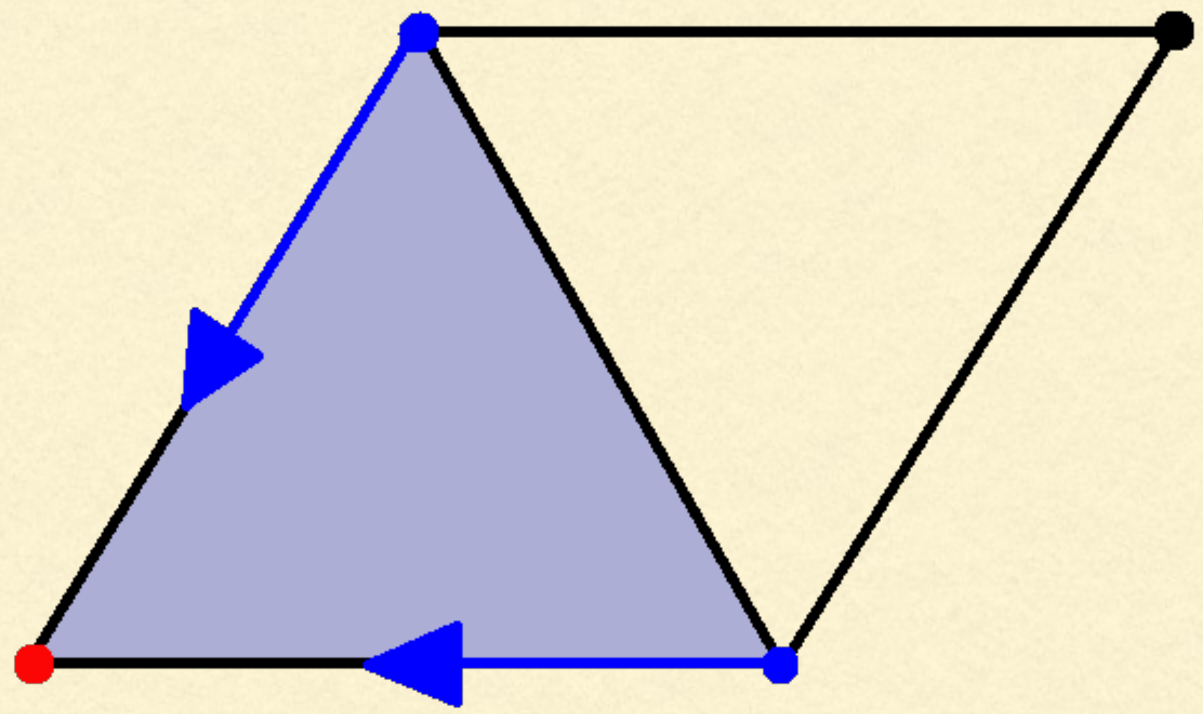
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

  while Σ' admits a *coreduction pair* (τ, σ) do
    V ← V ∪ { (τ, σ) }
    Σ' ← Σ' \ { τ, σ }
  end while

  Let η be a *free simplex* in Σ'
    A ← A ∪ { η }
    Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

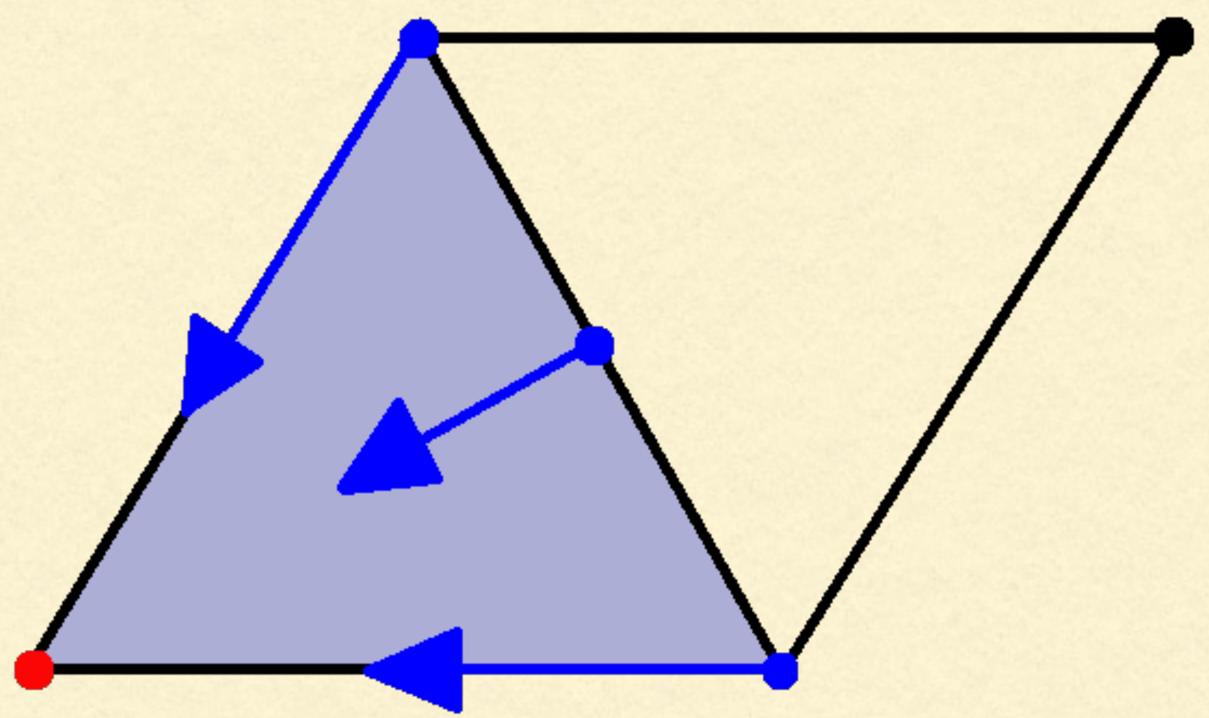
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

  while Σ' admits a *coreduction pair* (τ, σ) do
    V ← V ∪ { (τ, σ) }
    Σ' ← Σ' \ { τ, σ }
  end while

  Let η be a *free simplex* in Σ'
    A ← A ∪ { η }
    Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

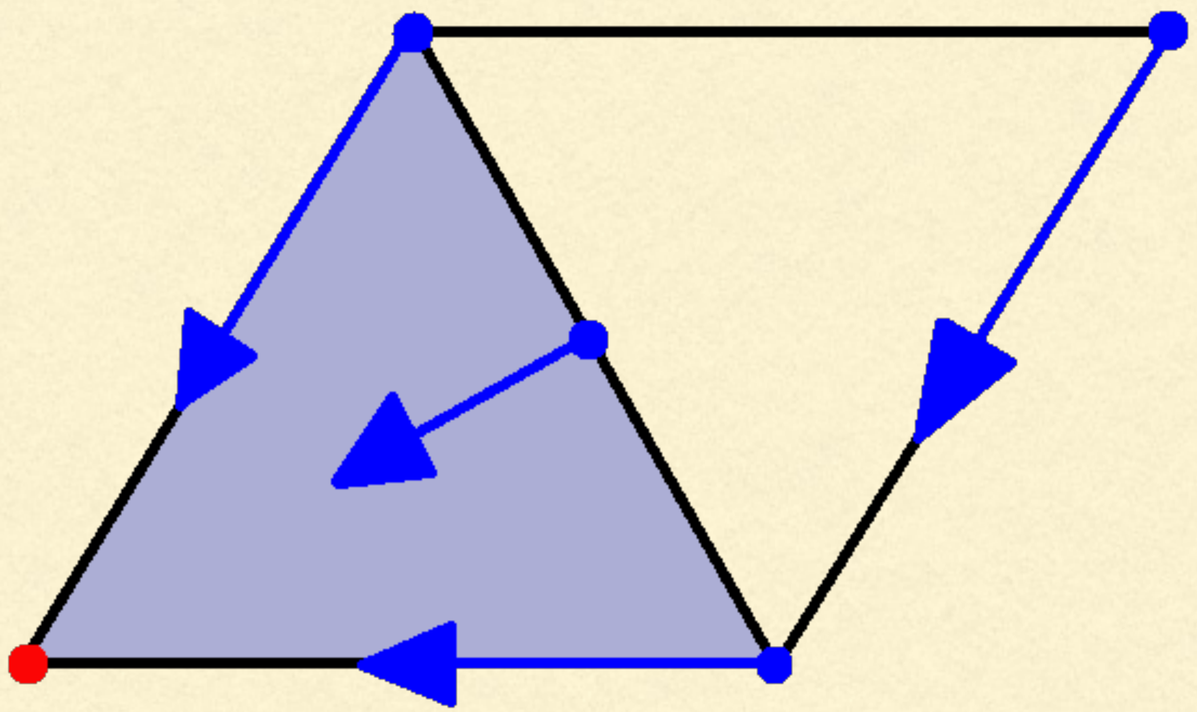
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

  while Σ' admits a *coreduction pair* (τ, σ) do
    V ← V ∪ { (τ, σ) }
    Σ' ← Σ' \ { τ, σ }
  end while

  Let η be a *free simplex* in Σ'
    A ← A ∪ { η }
    Σ' ← Σ' \ { η }

end while



GRADIENT VECTOR FIELD BY COREDUCTIONS [HARKER ET AL. 2010]

Input: Σ simplicial complex
Output: V gradient vector field, A set of critical simplices

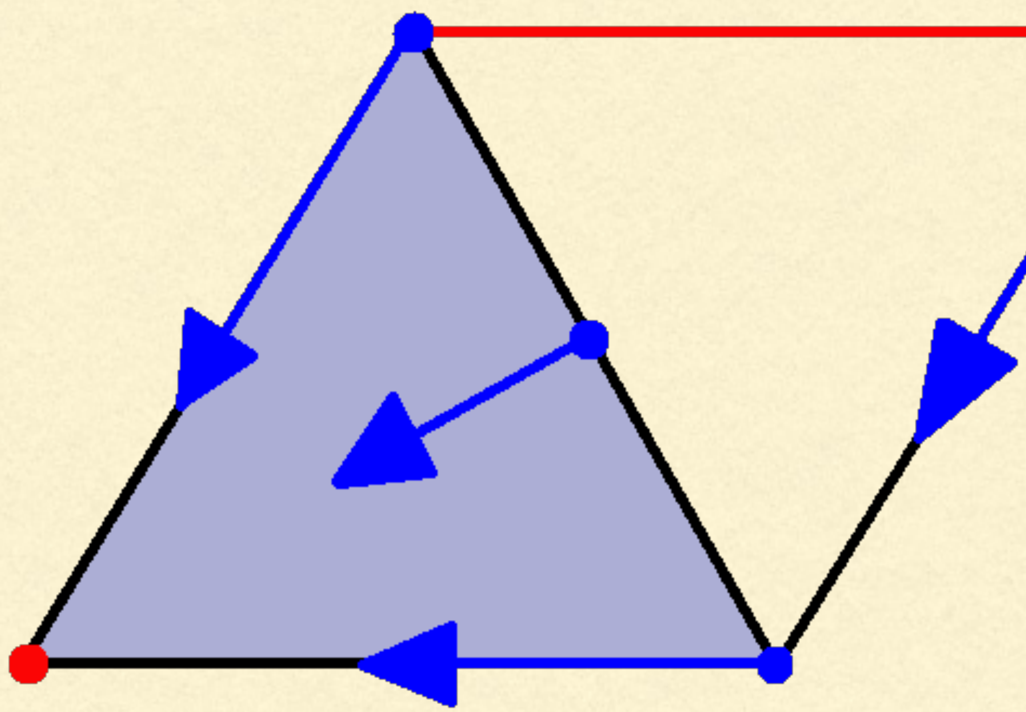
Set Σ' ← Σ, V ← ∅, A ← ∅

while Σ' ≠ ∅ do

while Σ' admits a *coreduction pair* (τ, σ) do
V ← V ∪ { (τ, σ) }
Σ' ← Σ' \ { τ, σ }
end while

Let η be a *free simplex* in Σ'
A ← A ∪ { η }
Σ' ← Σ' \ { η }

end while



REDUCTION-BASED AND COREDUCTION-BASED APPROACHES

Proposition.
Both the algorithms produce a gradient vector field on Σ

The same holds even if the following condition is not satisfied

(☆) A new critical simplex is created only if no more [co]reduction pair is available

EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Which approach is able to compute a gradient vector field with less critical simplices?

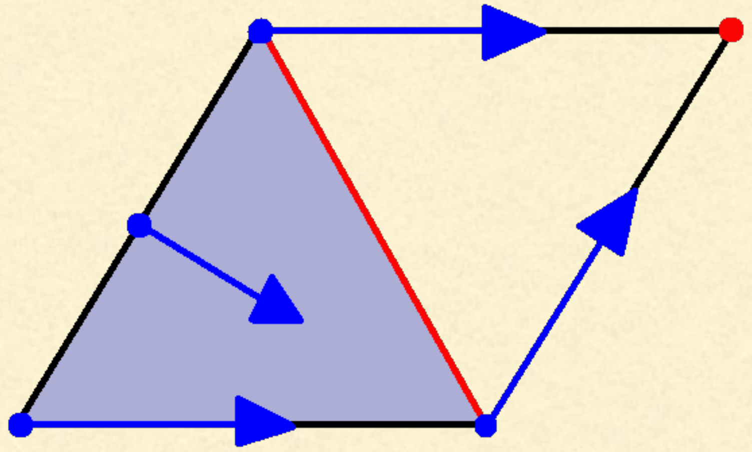
Reduction-based and coreduction-based approaches are ***equivalent***

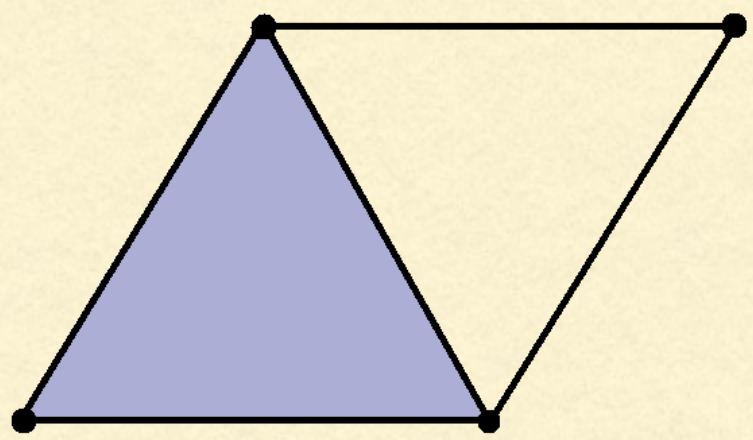
Proposition. Given a simplicial complex Σ and the gradient vector field V produced by a reduction-based algorithm, it is always possible to obtain the same gradient vector field with a coreduction-based algorithm
The reverse is also true

EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals

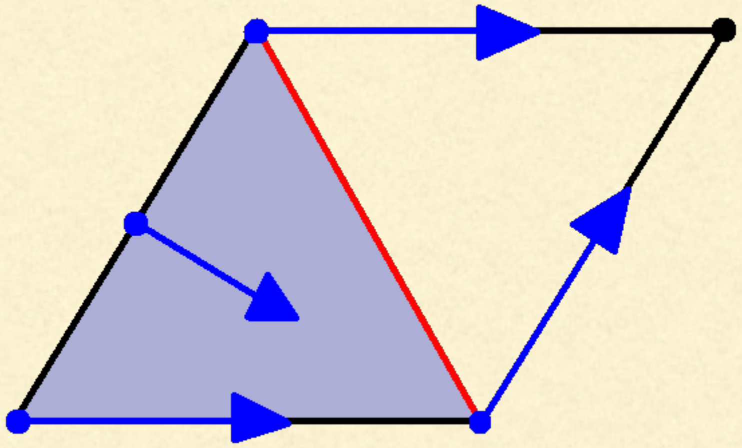


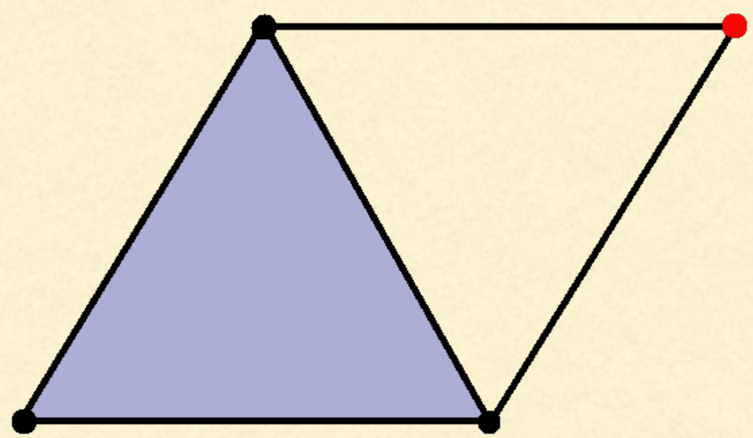


EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals

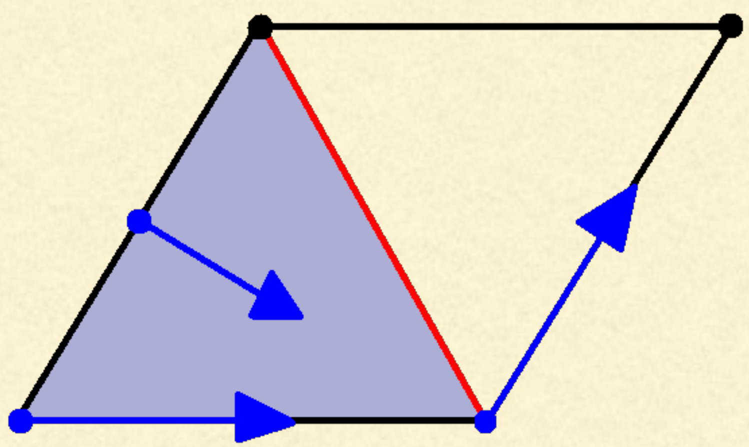


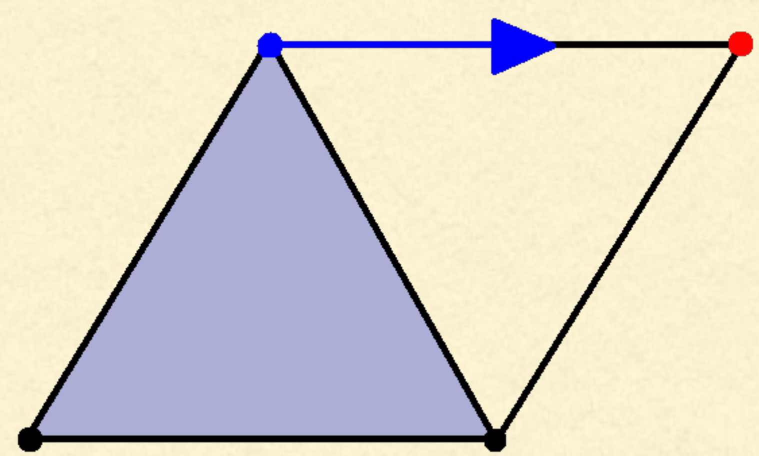


EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals

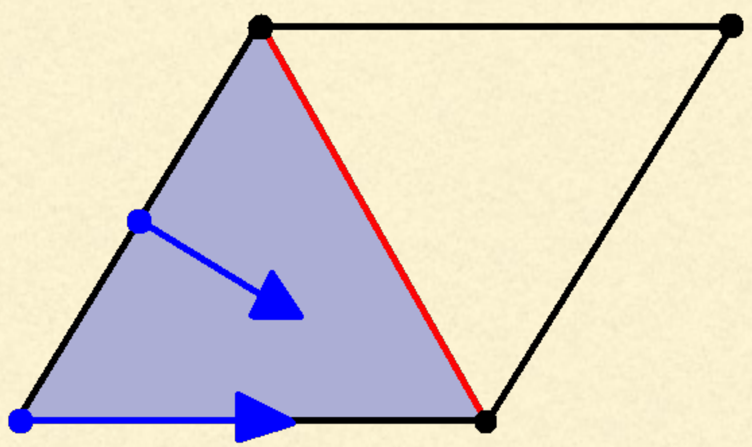


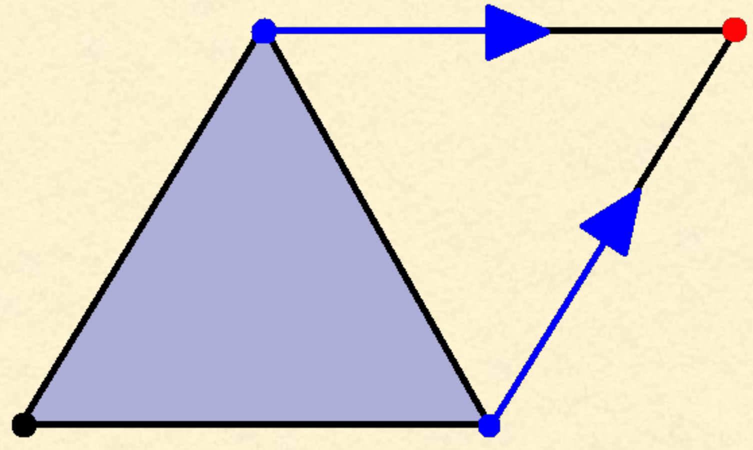


EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals

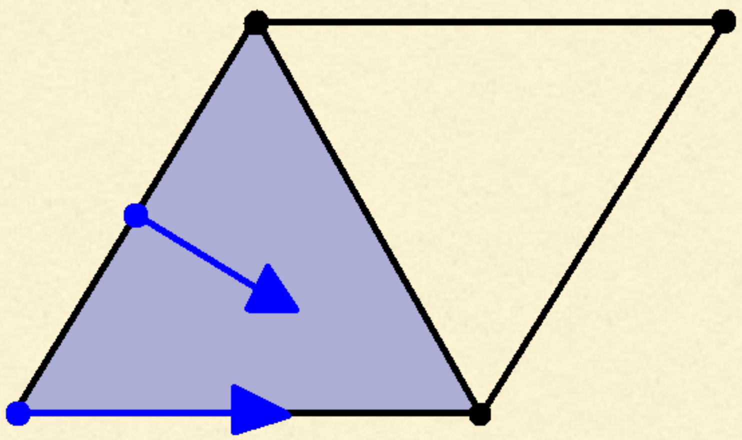


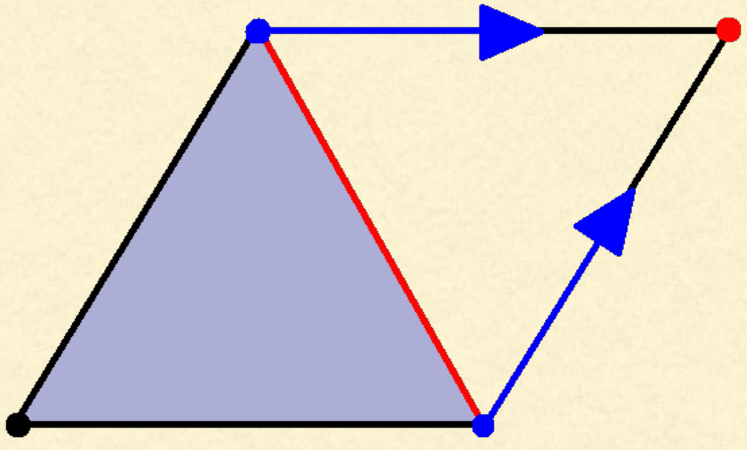


EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals

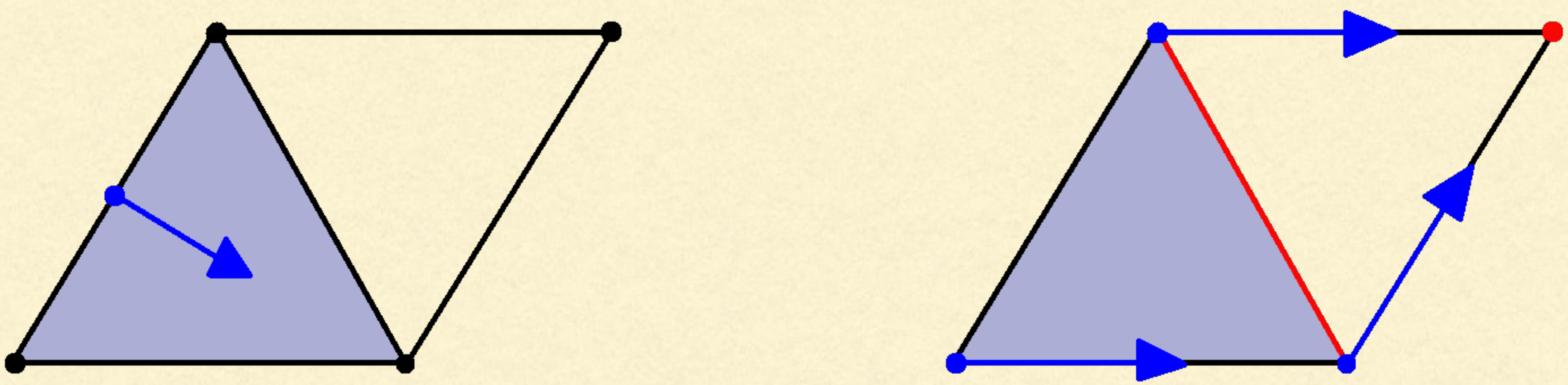




EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

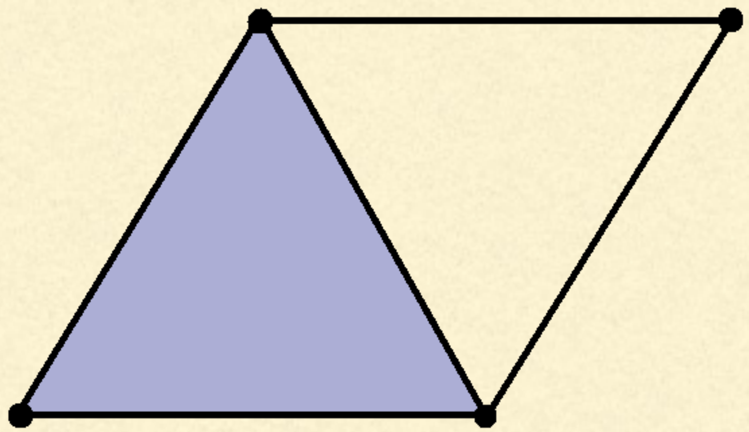
- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals

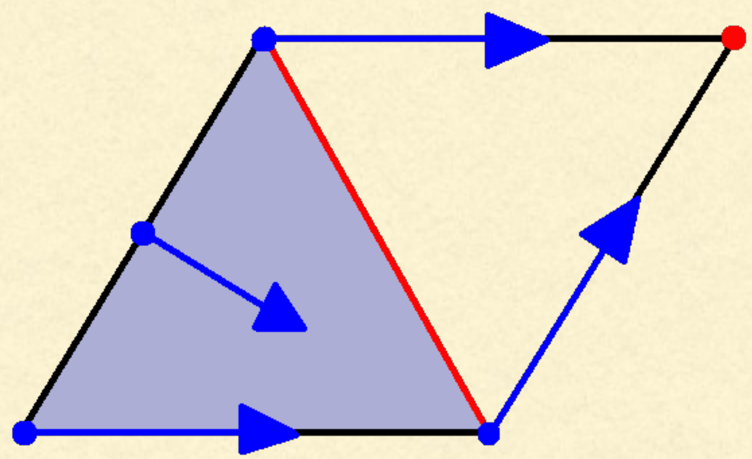


EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Proof's guidelines:

- Consider a simplicial complex Σ and run the reduction-based approach on it
- Take the sequence of reduction pairs and top simplex removals operated by the algorithm
- Reverse the order of the sequence: this new sequence represents for Σ a performable sequence of coreduction pairs and free simplex removals





EQUIVALENCE OF REDUCTION-BASED AND COREDUCTION-BASED APPROACH

Remark. It is not true, in general, that, given a gradient vector field V built by using a reduction-based algorithm satisfying condition (☆), V can be produced by a coreduction-based algorithm for which (☆) holds

INTERLEAVING REDUCTIONS AND COREDUCTIONS

Another class of approaches interleaving reductions and coreductions could be proposed

Proposition. Given a simplicial complex Σ, any discrete vector field produced by an algorithm performing reduction and coreduction pairs, removals of top and of free simplices is a gradient vector field

Each interleaved approach has equivalent capabilities

Proposition. Given a simplicial complex Σ and the gradient vector field V produced by an interleaved algorithm, it is always possible to obtain the same gradient vector field with a reduction-based algorithm or, equivalently, with a coreduction-based algorithm

OUR ALGORITHM

We developed a new ***coreduction-based algorithm*** for ***simplicial homology computation*** based on a highly ***efficient data structure*** and on a ***compact encoding of the gradient vector field***

Main features:

- Compact encoding of the simplicial complex
- Computation of a gradient vector field through a coreduction-based approach
- Compact encoding of the gradient vector field on the data structure
- Extraction of boundary maps
- Retrieval of simplicial homology and homology generators

COMPACT ENCODING OF THE SIMPLICIAL COMPLEX

A simplicial complex Σ is encoded by ***Generalized Indexed data structure with Adjacencies (IA*)*** [Canino et al. 2011]

IA* encodes:

- top simplices and vertices of Σ
- partial (co)boundary and adjacency relations for each encoded simplex

Incidence Graph (IG) encodes:

- all the simplices of Σ
- (immediate) boundary and coboundary for each simplex

The IA* data structure is generally more compact than IG:

- about 46% in dimension 2
- about 70% in dimension 3
- up to 170 times for special cases in dimension 8

COMPACT ENCODING OF THE GRADIENT VECTOR FIELD

Encoding of the gradient vector field is associated ***only with the top simplices***

Each top k-simplex σ encodes a ***bitvector*** of length $\sum_{i=1}^{k} \binom{k+1}{i+1}(i+1)$ representing ***all the possible pairings*** on its boundary.

For efficiency, an additional bitvector, denoted as paired(σ), is encoded for each top simplex:

paired(σ) encodes, for each simplex τ in the boundary of σ, whether τ is paired

COMPACT ENCODING OF THE GRADIENT VECTOR FIELD

Let η, τ be two paired simplices both in σ of dimension j and j+1 respectively; the resulting pair will be encoded in the following position of the bitvector of σ

$$\textit{start} + (j+2)\ \textit{pos}\tau + \textit{pos}\eta$$

where

- start $= \sum_{i=j+2}^{k} \binom{k+1}{i+1}(i+1)$
- posτ is the position of τ on the boundary of σ
- posη is the position of η on the boundary of τ

EXPERIMENTAL EVALUATION

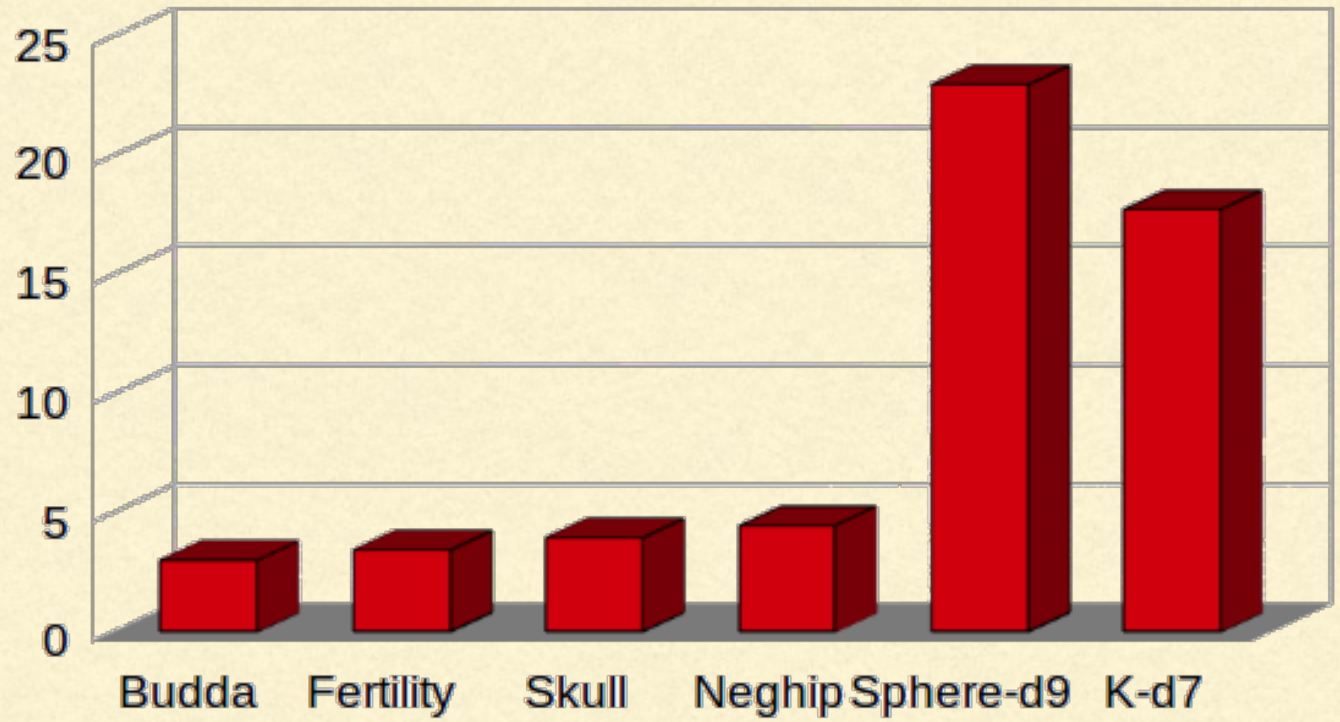
Dataset	d	n	n_0	n_{top}
Buddha	2	3.2M	0.54M	1.08M
Elephant	2	9.2M	1.5M	3.07M
Fertility	2	1.4M	0.24M	0.48M
Skull	3	0.75M	37K	0.15M
Neghip	3	2.1M	93K	0.48M
7Klein	7	0.1M	0.11K	0.6K
9Sphere	9	0.22M	2.0K	911

Column d indicates the complex dimension, n indicates the total number of simplexes, n_0 the number of vertices and n_{top} the number of top simplices

EXPERIMENTAL EVALUATION

We compared our performances with Perseus [Nanda 2012]

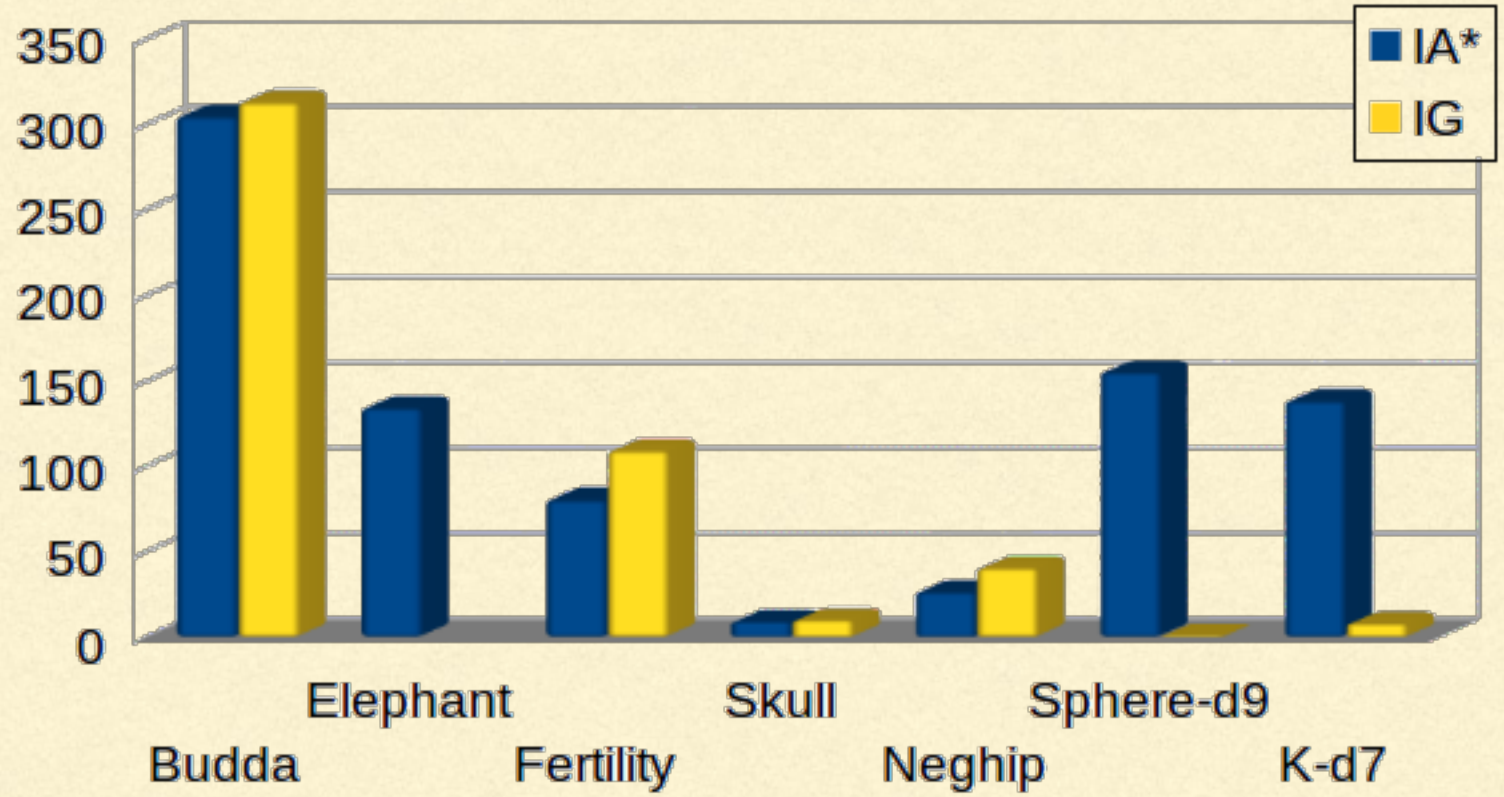
IA vs IG: storage cost*



Ratio (IG/IA*) of the storage costs of IG and IA* data structures endowed with gradient vector field encoding

EXPERIMENTAL EVALUATION

IA vs IG: timings*



Comparisons of timings for the homology computation algorithms based on the IA* and IG data structures

CONCLUSIONS

We have

- proven the ***equivalence of different methods*** based on homology-preserving operators to compute a gradient vector field
- developed a new ***algorithm based on coreductions***, on a ***space-efficient representation*** of the simplicial complex and on a ***compact encoding*** of the gradient vector field

We plan to develop

- an efficient encoding based for a simplicial complex in arbitrary dimension on the ***stellar tree*** data structure [Weiss et al. 2011]
- a new algorithm for ***persistent homology*** computation