# *Multi-Resolution Cell Complexes Based on Homology-Preserving Euler Operators*

**Lidija Čomić**

**University of Novi Sad, Serbia**

**Leila De Floriani, Federico Iuricich**

**University of Genova, Italy**

# Cell Complexes

* represent compactly geometry and topology of shapes
* form a basis modeling tool in many application domains
* many proposed data structures for representing cell complexes
* many proposed update operators
  - homology-preserving operators
  - homology-modifying operators

# Cell Complexes

Euler-Poincaré formula for cell complexes

$$n_0 - n_1 + .. + (-1)^d n_d = \beta_0 - \beta_1 + .. + (-1)^d \beta_d.$$

* $n_i$ is the number of $i$-cells in $\Gamma$
* $\beta_i$ is the $i$th Betti number of $\Gamma$
* $\chi(\Gamma) = n_0 - n_1 + .. + (-1)^d n_d$ is the *Euler-Poincaré characteristic* of $\Gamma$.

# Topological Operators on a Cell Complex

Homology-preserving operators preserve $\beta_i$ and $\chi(\Gamma)$.

$KiC(i+1)C(p,q)$ (*Kill i-Cell and (i+1)-Cell*)

- delete $i$-cell $p$ and $(i+1)$-cell $q$
- decrease $n_i$ and $n_{i+1}$ by one

$MiC(i+1)C(p,q)$ (*Make i-Cell and (i+1)-Cell*) is inverse to $KiC(i+1)C(p,q)$.

# Topological Operators on a Cell Complex

$KiC(i+1)C(p,q)$ is feasible on $\Gamma$ if:

* exactly two $i$-cells $p$ and $p'$ are on the boundary of $(i+1)$-cell $q$
* $i$-cell $p$ appears exactly once on the boundary of $(i+1)$-cell $q$

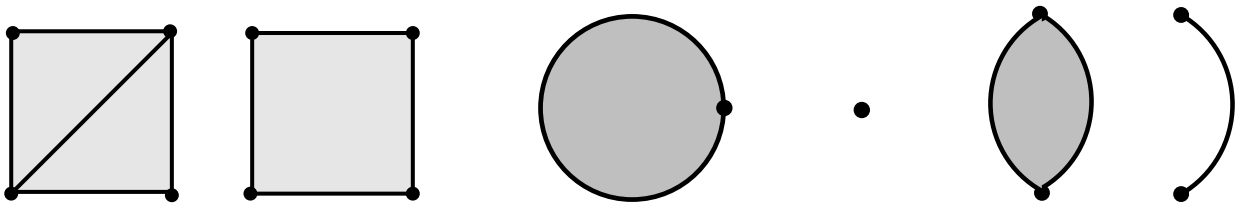
$KiC(i+1)C(p,q)$ creates a simplified complex $\Gamma'$:

* $i$-cell $p$ and $(i+1)$-cell $q$ are deleted
* each $(i+1)$-cell $r$ in the co-boundary of $p$ is merged with a copy of $q$ for each time $p$ appears on the boundary of $r$

There is a dual operator with reversed roles of $i$-cell and $(i+1)$-cell.

# Topological Operators on a Cell Complex

$K1C2C$ operator on a 2-complex



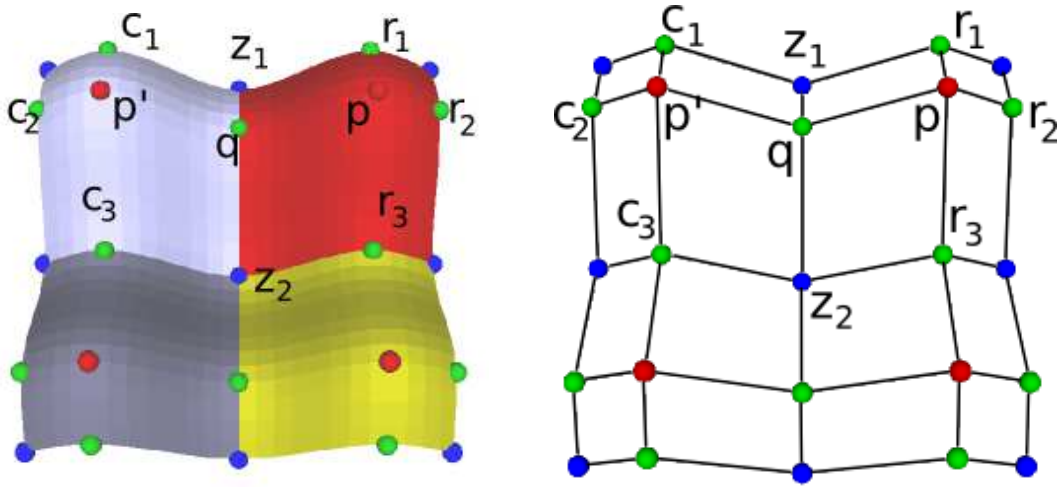
# Representing a Cell Complex

The topology of a cell complex $\Gamma$ is represented in the form of the *Incidence Graph (IG)* $G = (N, A, \psi)$, where

1. $N = N_0 \cup N_1 \cup ... \cup N_n$,
2. $(p, q) \in A$ if $i$-cell $p$ is on the boundary of $(i + 1)$-cell $q$ in $\Gamma$,
3. $\psi(p, q) = k$ if $i$-cell $p$ appears $k$ times in the boundary of $(i + 1)$-cell $q$.

# Representing a Cell Complex

A cell complex and the corresponding $IG$.



# Topological Operators on the $IG$

$KiC(i+1)C(p,q)$ is feasible on graph $G=(N,A,\psi)$ if:

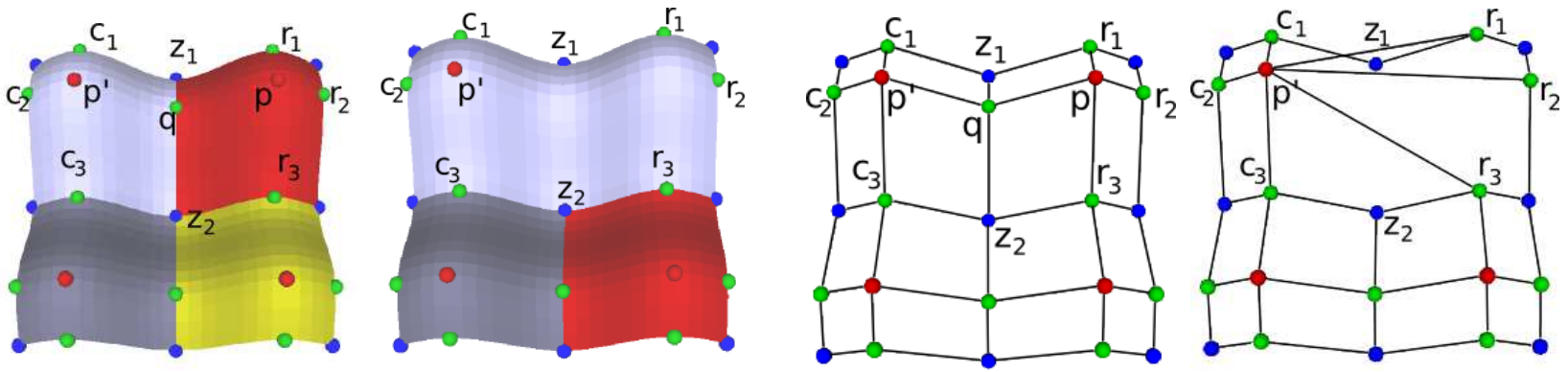
1. $(i+1)$-node $q$ is connected to exactly two $i$-nodes $p$ and $p'$,
2. there is exactly one arc in $A$ connecting $(i+1)$-node $q$ and $i$-node $p$ $\quad(\psi(p,q)=1)$.

In the simplified graph $G'=(N',A',\psi')$:

1. $i$-node $p$, $(i+1)$-node $q$ and all the incident arcs are deleted,
2. an arc $(p',r)$ is created for each arc $(p,r)\in A$, $r$ is an $(i+1)$-node $\quad(\varphi'(p',r)=\varphi(p',r)+\varphi(p',q)\cdot\varphi(p,r))$.

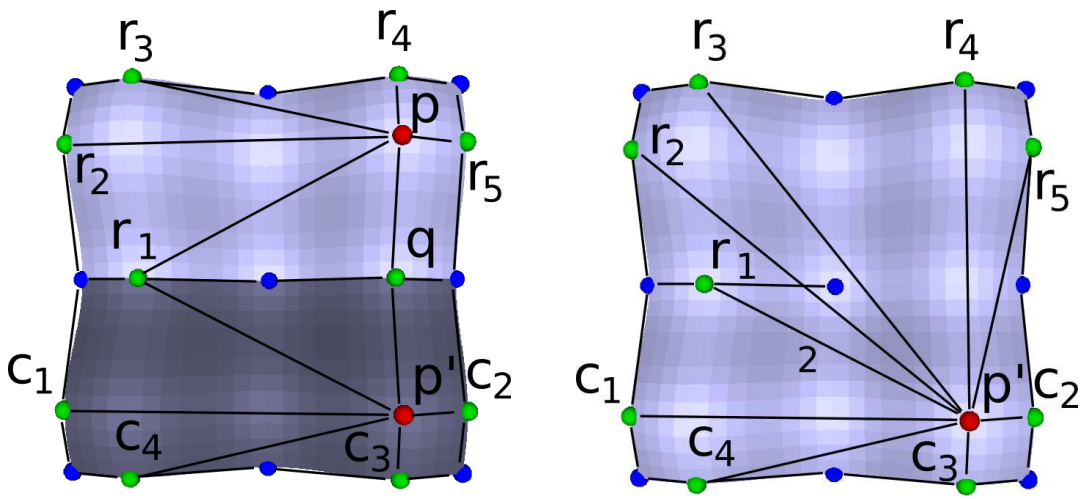
# Topological Operators on the $IG$

$K1C2C$ on a 2D cell complex and on the corresponding $IG$.



# Topological Operators

$K1C2C$ on a 2D cell complex and on the corresponding $IG$.




After simplification, 1-cell $r_1$ appears two times on the boundary of 2-cell $p'$ $\quad (\psi(p', r_1) = 2)$.

# Topological Operators on the $IG$

$MiC(i+1)C(p,q)$ is feasible on a graph $G=(N,A,\psi)$ if

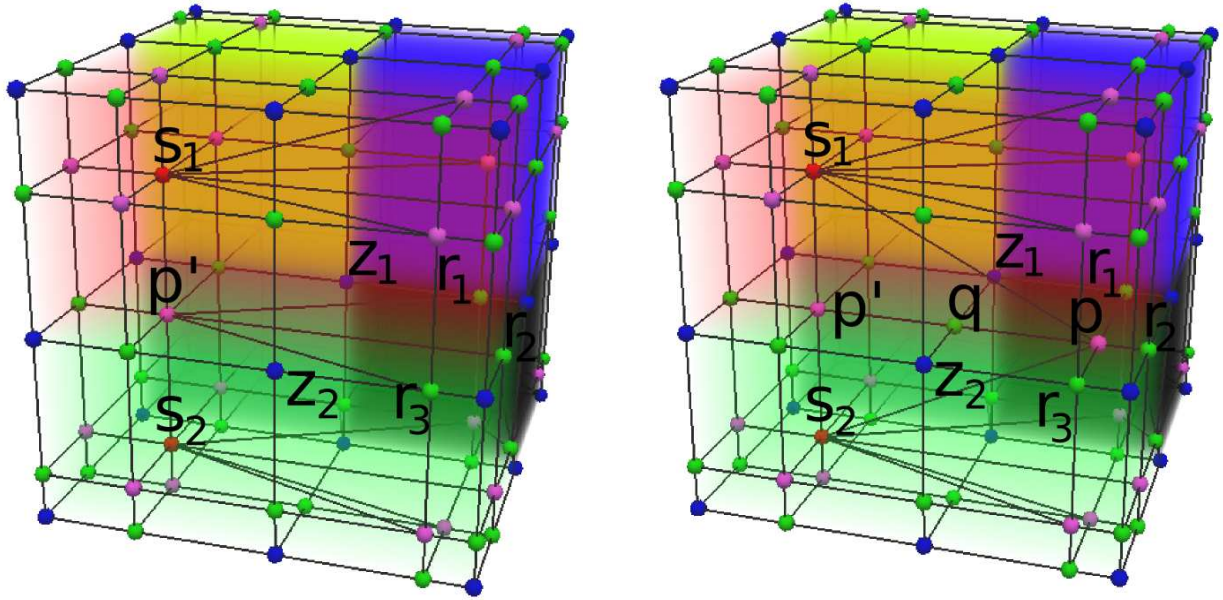
1. all the nodes that will be connected to either $p$ or $q$ are in $N$
2. all the arcs $(p',r)$ are in $A$
   ($(i+1)$-node $r$ will be connected to $p$).

In the refined graph $G'=(N',A',\psi')$:

1. $i$-node $p$, $(i+1)$-node $q$ and all the incident arcs are created
2. $\varphi'(p',r)=\varphi(p',r)-\varphi'(p',q)\cdot\varphi'(p,r)$.

# Topological Operators

$M1C2C$ on a 3D cell complex and on the corresponding $IG$.



# Multi-Resolution Model

- $\mathcal{S}$ is a set of $KiC(i+1)C$ operators, applied iteratively to the $IG$ at full resolution.
- $G_B$ is the $IG$ at coarsest resolution.
- $\mathcal{M}$ is the set of operators inverse to the ones in $\mathcal{S}$.
- $\mathcal{R}$ is the dependency relation between refinements in $\mathcal{M}$.

Multi-Resolution Cell Complex $MCC = (G_B, \mathcal{M}, \mathcal{R})$

# Multi-Resolution Model

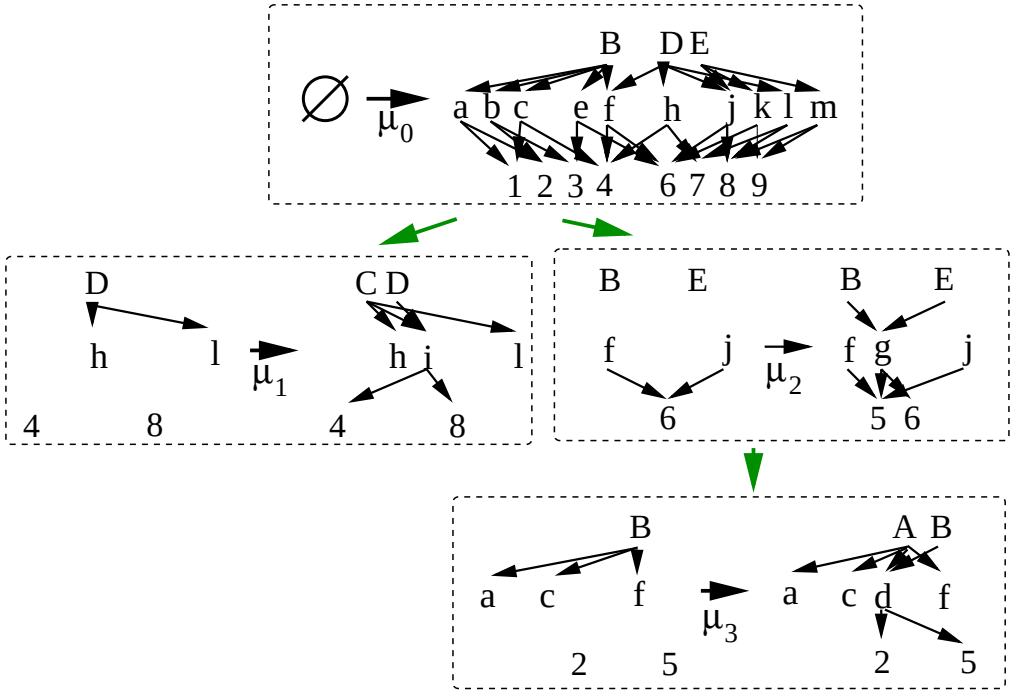
- Refinement $\mu = MiC(i+1)C(p,q)$ directly depends on refinement $\mu^*$ if and only if $\mu^*$ creates at least one node that is connected to either $p$ or $q$ by $\mu$.
- Dependency relation is a partial order.
- Independent refinements are interchangeable.
- A closed set $\mathcal{U} = \{\mu_0, \mu_1, \mu_2, ..., \mu_m\}$ of refinements can be applied on $G_B$ in any order that extends the partial order, producing the same $IG$ at an intermediate resolution.

# Multi-Resolution Model

* $MCC$ is encoded in a DAG
  - the nodes encode refinements in $\mathcal{M}$
  - the arcs encode the direct dependency relation
  - the root $\mu_0$ of the DAG is a dummy refinement that creates the base graph $G_B$

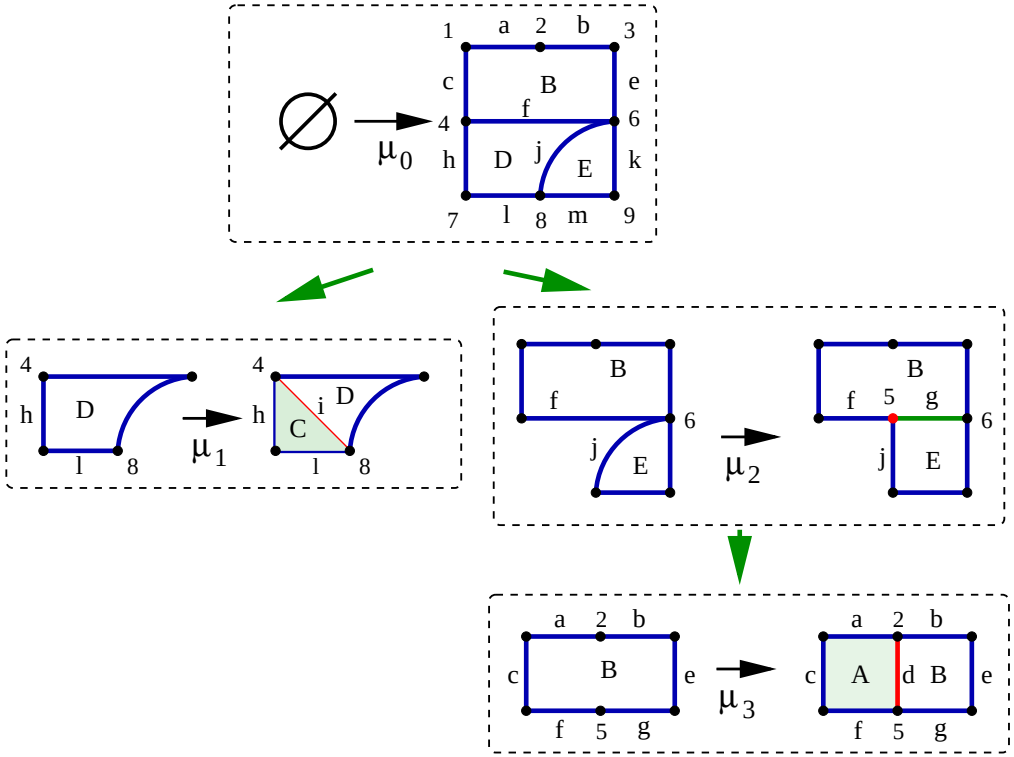
# Multi-Resolution Model

Example of an $MCC$.



# Multi-Resolution Model

Refinements expressed in terms of cell complexes.



# Multi-Resolution Model

An $MCC$ encodes a large number of representations at intermediate resolution.

Selective refinement query:

* Define a Boolean criterion $\tau$ on the nodes of the $MCC$
* Depth-first algorithm
  - start from $G_B$
  - apply recursively refinements $\mu$ required to satisfy the criterion
  - apply all ancestor refinements before $\mu$

# Experimental Results

Simplification approaches used to build the $DAG$

* step-by-step
* batch

# Experimental Results

Step-by-step simplification:

- put all feasible simplifications in a priority queue
- perform the first simplification from the queue
- update the queue

# Experimental Results

Batch simplification:

* put all feasible simplifications in a priority queue
* perform a set of independent simplifications from the queue
* update the queue

# Experimental Results

| | Data set | N. Cells | N. Simpl. | Time Simpl. | Time MCC | Storage MCC | Time Ref. | Full Compl. | Base Compl. |
|---|---|---|---|---|---|---|---|---|---|
| | | Step-by-step simplification | | | | | | | |
| 2D | *Eros* | 2859566 | 1429781 | 74.4 | 5.3 | 254.9 | 18.1 | 349.0 | 0.0002 |
| | *Hand* | 1287532 | 643694 | 35.4 | 2.3 | 117.2 | 7.58 | 157.1 | 0.01 |
| | *VaseLion* | 1200002 | 599999 | 26.7 | 2.1 | 105.8 | 6.8 | 146.4 | 0.00028 |
| | | Batch simplification | | | | | | | |
| | *Eros* | 2859566 | 1429781 | 218.8 | 6.4 | 241.0 | 18.7 | 349 | 0.0002 |
| | *Hand* | 1287532 | 643741 | 99 | 2.6 | 120.7 | 7.6 | 157.1 | 0.004 |
| | *VaseLion* | 1200002 | 599999 | 90.7 | 2.3 | 110.5 | 7.7 | 146.4 | 0.00028 |

# Experimental Results

3D

| Data set | N. Cells | N. Simpl. | Time Simpl. | Time MCC | Storage MCC | Time Ref. | Full Compl. | Base Compl. |
|---|---|---|---|---|---|---|---|---|
| | | | | Step-by-step simplification | | | | |
| *VisMale* | 297901 | 147594 | 45.1 | 0.6 | 40.4 | 5.1 | 48 | 0.46 |
| *Bonsai* | 1008357 | 498790 | 380.6 | 2.7 | 146.9 | 27.2 | 162.5 | 1.8 |
| *Hydrogen* | 2523927 | 1248743 | 8643.8 | 7.8 | 395.7 | 419.5 | 407.4 | 4.4 |
| | | | | Batch simplification | | | | |
| *VisMale* | 297901 | 148116 | 69.2 | 0.7 | 37.6 | 2.5 | 48 | 0.28 |
| *Bonsai* | 1008357 | 501524 | 305.8 | 2.69 | 126.4 | 10.4 | 162.5 | 0.89 |
| *Hydrogen* | 2523927 | 1253913 | 1412.9 | 7.4 | 321.3 | 33.9 | 407.4 | 2.7 |

# Experimental Results

| 2D | | | | 3D | | | |
|---|---|---|---|---|---|---|---|
| Data set | Perc. | Refinement Time (sec) | | Data set | Perc. | Refinement Time (sec) | |
| | | step-by-step | batch | | | step-by-step | batch |
| *Eros* | 50% | 0.80 | 0.92 | *VisMale* | 50% | 3.45 | 0.12 |
| | 80% | 1.42 | 1.01 | | 80% | 3.77 | 0.15 |
| | 100% | 2.63 | 2.60 | | 100% | 4.01 | 0.53 |
| *Hand* | 50% | 0.31 | 0.57 | *Bonsai* | 50% | 15.3 | 0.65 |
| | 80% | 0.45 | 0.65 | | 80% | 17.4 | 0.69 |
| | 100% | 1.20 | 1.19 | | 100% | 19.1 | 1.88 |
| *VaseLion* | 50% | 0.73 | 0.69 | *Hydrogen* | 50% | 106.3 | 8.1 |
| | 80% | 1.01 | 0.99 | | 80% | 127.7 | 8.7 |
| | 100% | 1.10 | 1.06 | | 100% | 172.1 | 11.3 |

# Experimental Results

The representations obtained from the $MCC$ after 10K, 50K and 200K refinements, the complex at full resolution of the *VaseLion* data set and the representation obtained with a query at variable resolution.

# Future Work

Computation of homology:

- Simplify the complex using homology-preserving operators
- Compute homology generators on the simplified complex
- Propagate the generators to the full-resolution complex using the $MCC$