# Manipulating Topological Decompositions of Non-Manifold Shapes

David Canino

Independent Researcher in Computer Science
PhD. in Computer Science

canino.david@gmail.com

October 3, 2016

# The Problem (Automatic Editing of a Decomposition) #1

## The Starting Point

A (cell) complex $\Gamma$ discretizes a *digital shape*, and may be:

- decomposed into the relevant components of any *decomposition* $D(\Gamma)$, always computable (the *Batch* Approach);
- updated by an *editing operator* $u = (u^-, u^+)$. The resulting complex is $\Gamma_u = \{\Gamma | u^-\} \cup \{u^+\}$.

## Objective of This Paper (the *Interactive* Approach)

Updating automatically $D(\Gamma)$ when applying an *editing operator* $u$ on $\Gamma$.

## The Naive Approach

$$\begin{array}{ccc} \Gamma & \xrightarrow{\quad u \quad} & \Gamma_u \\ D \downarrow & & \downarrow D \\ D(\Gamma) & \xrightarrow{\quad ??? \quad} & D(\Gamma_u) \end{array}$$

This approach is always *computable*, and works in all cases (general solution)

*BUT* the relation between $D(\Gamma)$ and $D(\Gamma_u)$ is not known and exploited.

# The Problem (Automatic Editing of a Decomposition) #2

In any case, the *naive approach* may be *INEFFICIENT*:

- an update $u$ modifies only locally $\Gamma$ (a local *Region-Of-Influence*, *ROI*);
- the components in $D(\Gamma)$, related to the unchanged portions of $\Gamma$ (*NOT AFFECTED* by $u$), may be reused directly in $D(\Gamma_u)$;
- only the components of $D(\Gamma)$, related to the portions of $\Gamma$, modified by $u$ (*AFFECTED* by $u$), are recomputed, modified, and added to $D(\Gamma_u)$.

### Consequences

- no need to recompute $D$ from scratch after every $u$ (*expensive*);
- this task is performed even if at *interactive rate*, when reusing the components from $D(\Gamma)$.

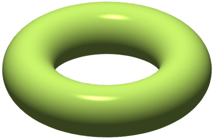
### However ...

- a general solution *does not exist*;
- a solution depends on the *update* $u$ and the *decomposition* $D$.

# The Structural Models for the Non-Manifold Shapes #1

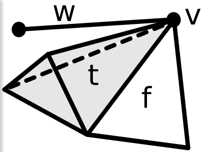
### Manifold Condition at a Point *p*

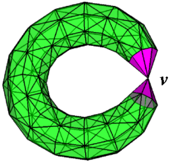
Its neighborhood is locally *homeomorphic* to the *ball*, centered at *p*.



### The Non-Manifold Shapes

- some *non-manifold singularities*, where the manifold condition is violated;
- several subcomponents of *different dimension*, often *(almost) manifold*.




### Classic Approach (*cell complexes* in $\mathbb{E}^d$)

The *topological data structures* (mangroves):

- cells (vertices, edges, 2-cells, ...);
- the *topological relations* for each cell.

Too many contributions:

- De Floriani and Hui, 2005
- Botsch et. al., 2010
- Canino, 2012

# The Structural Models for the Non-Manifold Shapes #2

### Drawbacks (wrt non-manifolds)

- the *non-manifold singularities* are not exposed directly (not always recognizable, *Nabutovski, 1996*);
- the *subcomponents* are not exposed.

ONLY the *local connectivity* for a cell in a (non-manifold) shape

### The Structural Model

- the *subcomponents* are exposed explicitly;
- the connections along the *non-manifold singularities.*

The *global structure* of a non-manifold shape

### Mesh Repairing (Result=manifold)

- Falcidieno and Ratto, 1992
- Gueziec and Cardoze, 1998
- Rossignac et all., 1999
- Attene et all., 2009 / 2013

### (Combinatorial) Stratifications

- Whitney, 1965
- Lopes et all., 1999
- De Floriani et all., 2003
- Pesco et all., 2004

# The Manifod-Connected (MC-) decomposition #1

The *Manifold-Connected (MC)-* decomposition (*De Floriani and Hui, 2007*) is extended *here* to any *cell d-complex* $\Gamma$ (initially for simplicial complexes).

**Top $k$-cell $\gamma$ (with $0 \leq k \leq d$)**

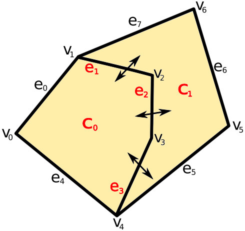
Does not bound another cell in $\Gamma$.

**MC-adjacent top $k$-cells $\gamma'$ and $\gamma''$**

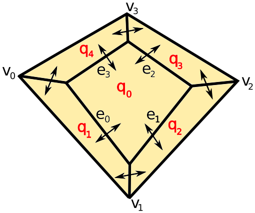
They are the unique top cells in the star of a common $(k-1)$-face $\tau$.

**The MC-path from $\gamma$ to $\gamma'$**

- a sequence of top $k$-cells, such that a pair of *consecutive* cells is *MC-adjacent*;
- all top $k$-cells in the MC-path are *MC-equivalent*.




2-cells $c_0$ and $c_1$ are MC-adjacent




MC-path (always computable):

$q_0 \rightarrow q_1 \rightarrow q_2 \rightarrow q_3 \rightarrow q_4$

# The Manifod-Connected (MC-) decomposition #2

**The *MC-connectivity* relation $\sim_{MC}$ among top $k$-cells (equivalence relation)**

Here, $\gamma \sim_{MC} \gamma'$, iff $\gamma$ and $\gamma'$ are MC-equivalent (*always computable).*

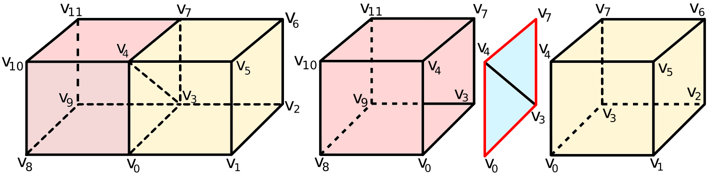
It is the *transitive closure* of the *MC-adjacency*.

**MC-component $[\gamma]$ (wrt $\sim_{MC}$)**

Maximal collection of all top cells, MC-equivalent to $\gamma$ (*equivalence class*).

**MC-decomposition $\mathcal{MC}_\Gamma$ (unique)**

Quotient space $\Gamma / \sim_{MC}$.




A more formal description (and more details) in the paper

For more details, see *Canino, 2012* - *Canino, De Floriani, 2013*

# How and When an Update affects the MC-components

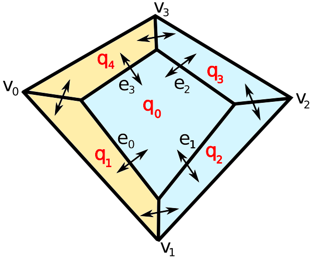
### The Key Target in the Running Pipeline

- understanding how *modifying* any equivalence class $[\gamma]$ (wrt $\sim_{MC}$) when applying an *update* $u = (u^-, u^+)$;
- $[\gamma]$ is *affected* by $u$, if it intersects the *generalized neighborhood* $\sigma^h(u^-)$ for any order $h$ (minimum order $\bar{h}$ such that this happens).

### The Generalized Neighborhood $\sigma^h(u^-)$

- $\sigma^0(u^-) \equiv$ all top cells in the star of vertices in $u^-$;
- $\sigma^h(u^-) \equiv \sigma^0(\sigma^{h-1}(u^-))$.

By construction, $\sigma^\infty(u^-) \equiv \Gamma$




$\sigma^0(v_0) = \{q_1, q_4\}$ (yellow)
$\sigma^1(v_0) = \sigma^0(v_0) \cup \{q_0, q_2, q_3\}$ (light blue)

# How Updating the MC-decomposition

- Modifying the MC-decomposition $\mathcal{MC}_\Gamma$ of a 2-complex $\Gamma$ with:
  - $V$ vertices (i.e, 0-cells), $E$ edges (i.e., 1-cells), $F$ polygons (i.e., 2-cells);
  - $R$ connected regions and $L$ hole loops (1-cycles).
- The *Euler operators* in *Lee and Lee, 2001*, satisfying the Euler formula:

$$V - E + F = R - L$$

- This forms a *basis* for all updates on cell 2-complexes + MC-decompositions.
- Exploiting the Compact MC-graph (see *Canino and De Floriani, 2013*), where every operation is *efficient*.
- Defined on any *mangrove* (see *Canino, 2012*) [ nodes clustering ]

**Key Operation (Theoretical Validity)**

- splitting and merging together the MC-paths of interest;
- $\sim_{MC}$ is an equivalence relation, thus is *transitive*.

# The MEV and the KEV Updates

### The Make-Edge-Vertex (MEV) Update

- adds a new vertex $v'$ to $\Gamma$;
- adds a new top edge $e = (v, v')$ between $v'$ and an existing vertex $v$.

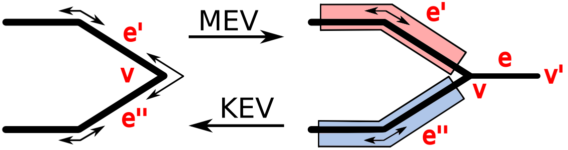
V=V+1, E=E+1

MC-components intersect $\sigma^0(v)$

### Key Idea

- candidate $[e]$ is merged with an existing $[e']$ in $\mathcal{MC}_\Gamma$, iff a MC-adjacency occurs at $v$;
- otherwise, $[e]$ is added, and $[e']$ may be (even) split.

Time complexity: $O(1)$ [best]
#top edges in $[e']$ (worst)




### The *Kill-Edge-Vertex (KEV)* Update

The *reverse* wrt the MEV update.

More details in the paper.

# The MEL and the KEL Updates

### The Make-Edge-Loop (MEL) Update

Completes a hole loop with a new top edge $e = (v, v')$, connecting vertices $v$ and $v'$.

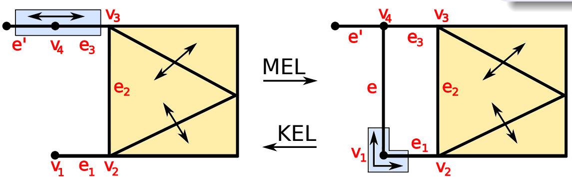
$$E{=}E{+}1, L{=}L{+}1$$

The hole loops are not relevant for $\sim_{MC}$.

MC-components intersect $\sigma^0(v) \cup \sigma^0(v')$

### Key Idea

- candidate $[e]$ is merged with the existing MC-components in $\mathcal{MC}_\Gamma$, iff a MC-adjacency occurs at $v$ or/and $v'$ (also both, up to 2 fusions);
- otherwise, $[e]$ is added, and the existing MC-components may be (even) split.




### The *Kill-Edge-Loop (KEL)* Update

The *reverse* wrt the MEL update.

Time complexity: #top edges in the star of $v$ and $v'$.

# The MEJR, KESR, MVR and KVR Updates

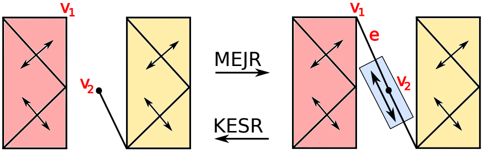
**The Make-Edge-Join-Region (MEJR) Update**

Creates a new top edge $e = (v_1, v_2)$ between two existing vertices $v_1$ and $v_2$ in two distinct regions.

**The Kill-Edge-Split-Region (KESR) Update**

Removes a top edge $e = (v_1, v_2)$, disconnecting two regions in $\Gamma$ (connected only through $e$).

E=E+1, R=R-1




E=E-1, R=R+1

- Mutually reverse
- Similar to the MEL and KEL updates (without loops)

**The Make-Vertex-Region (MVR) Update**

Adds a new top vertex $v$, i.e., a new $[v]$ to $\mathcal{MC}_\Gamma$ ($V = V + 1$, $R = R + 1$).

**The Kill-Vertex-Region (KVR) Update**

Removes a top vertex $v$, i.e., an existing $[v]$ from $\mathcal{MC}_\Gamma$ ($V = V - 1$, $R = R - 1$).

# The MFKL and the KFML Updates

### The Make-Face-Kill-Loop (MFKL) Update

Fills the void, bounded by a hole loop $(e_i)_{i=1}^n$, with a new 2-cell $\gamma$

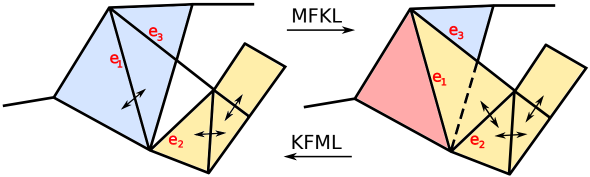
F=F+1,L=L-1

MC-components in $\bigcup_i \sigma^0(e_i)$

Time complexity: #2-cells in the star of all edges $e_i$ in the hole loop

### Key Idea

- candidate $[\gamma]$ is merged with the existing MC-components in $\mathcal{MC}_\Gamma$, iff a MC-adjacency occurs at $e_i$
- otherwise, $[\gamma]$ is added, and the existing MC-components may be (even if) split




### The *Kill-Face-Make-Loop (KFML)* Update

The *reverse* update wrt the MFKL update.

# Other Updates

This forms a basis for manipulating $\mathcal{MC}_\Gamma$ with any update.

**MC-equivalent Meshings**

Replacing a MC-path with another MC-path (with the same domain).

$\mathcal{MC}_\Gamma$ remains unchanged.

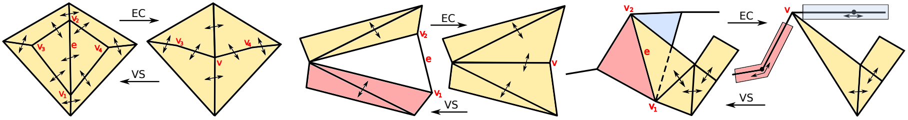
**Collapsing a $p$-cell $\gamma$**

Removing $\gamma$ and one of its border $(p-1)$-faces.

- Template-based and Stellar Updates
- Delaunay/Voronoi Mesh Generation
- Automatic Retopology
- Merging/Splitting MC-adjacent cells

- 1-cell $\gamma$: KEV + KVR
- 2-cell $\gamma$: KFML + KEL

Collapsing an edge is an *open problem* - updates are propagated to the entire Γ.




$\sim$ a template-based operator    EC on a *top* edge    EC on a *not top* edge

# Experimental Results

Applying $m$ updates on $\Gamma$ and $\mathcal{MC}_\Gamma$.

Average Running times (ms) on the non-manifold 2D shapes from `http://ggg.disi.unige.it`.

- $\mathcal{T}_m^N$: the *naive* approach
- $\mathcal{T}_m^B$: the *batch* approach
- $\mathcal{T}_m^I$: the *interactive* approach

| $m$ | $\mathcal{T}_m^N$ | $\mathcal{T}_m^B$ | $\mathcal{T}_m^I$ | $\mathcal{T}_m^B$ / $\mathcal{T}_m^I$ |
|---|---|---|---|---|
| $10K$ | $289.4K$ | 36 | 5.5 | 6.5 |
| $40K$ | – | 85 | 22 | 3.9 |
| $100K$ | – | 196 | 58 | 3.3 |
| $500K$ | – | $1.1K$ | 372 | 3.9 |
| $1M$ | – | $2.4K$ | $746K$ | 3 |
| $3M$ | – | $10.8K$ | $2.1K$ | 5.1 |
| $6M$ | – | $23.4K$ | $4.3K$ | 5.4 |

- $\mathcal{T}_m^N > 10$ minutes after only $40K$ updates;
- $\mathcal{T}_m^B \approx 4 \times \mathcal{T}_m^I$ (on average).

**Consequences**

- $\mathcal{T}_m^N$ is too high;
- $\mathcal{MC}_\Gamma$ could be built *interactively*.

Our implementations exploit the *Mangrove TDS Library* - `http://mangrovetds.sourceforge.net`

# The Future Work

These idea are the basis for a very large number of applications:

- the *random* and the *interactive manipulation* for a non-manifold shape and its MC-decomposition (*automatically*)
- improving the internal *meshing quality* of the MC-components (e.g., optimal for the 3D printing, but not only)
- improving the efficiency for computing the simplicial homology (*Constructive Homology Theory* by F. Sergeraert, see *Boltcheva, Canino, et. al., 2011*)
- extension to the *higher dimensional* shapes
- defining a *multiresolution structural model* for the non-manifold shapes

**Main Consequence**

- The internal meshing of the MC-components is *not mandatory*
- Generation at run-time, like the *Catalogs*, by *Castelli Aleardi et. al., 2011*

## That's All (for Now). But I hope not ...

The source code of the implementations will be available as GPL v3 from:

- `http://mangrovetds.sourceforge.net`
- `http://mangrovetds.github.io`

distributed as an *Extra Program* of the *Mangrove TDS Library* (new version 3.0).

# THANK YOU for YOUR ATTENTION!