The Gudhi library: Simplification of Simplicial Complexes

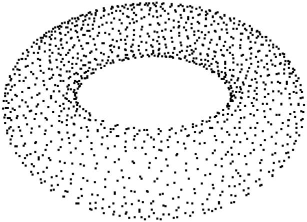
Gudhi workshop

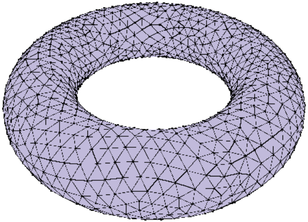
David Salinas

November 4, 2014

Introduction

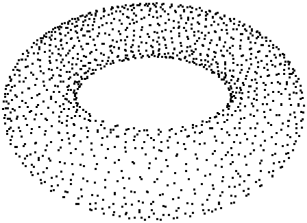
- Goal : represent and simplify huge simplicial complexes
- Problem : in some cases, the number of simplices may be too large for full representation

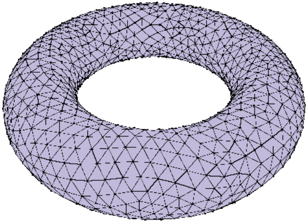




Introduction

- Goal : represent and simplify huge simplicial complexes
- Problem : in some cases, the number of simplices may be too large for full representation
- *Implicit* representation of simplices





Introduction

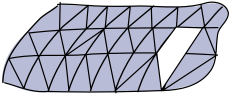
Plan

- Skeleton-blocker data-structure
  [SoCG11, IJCGTA12 Attali Lieutier Salinas]
- Experiments
  - Memory size
  - Simplification time

Introduction

Storing implicitly the set of simplices

- Compact representation for Flag complexes (can be represented only with a graph)
- What about other complexes?

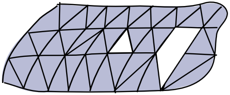


Flag-complex = every clique is a simplex

Introduction

Storing implicitly the set of simplices

- Compact representation for Flag complexes (can be represented only with a graph)
- What about other complexes?

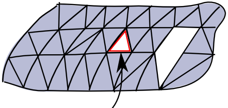


Flag-complex nearly everywhere

Introduction

Storing implicitly the set of simplices

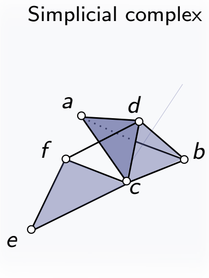
- Compact representation for Flag complexes (can be represented only with a graph)
- What about other complexes?



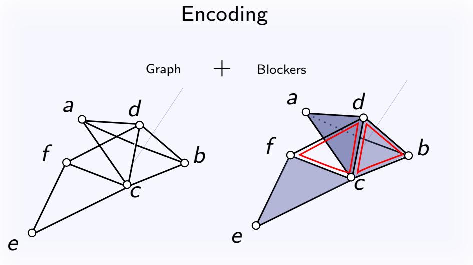
Flag-complex nearly everywhere
but here

Skeleton-blockers data-structure

Definition of the data-structure



Simplicial complex

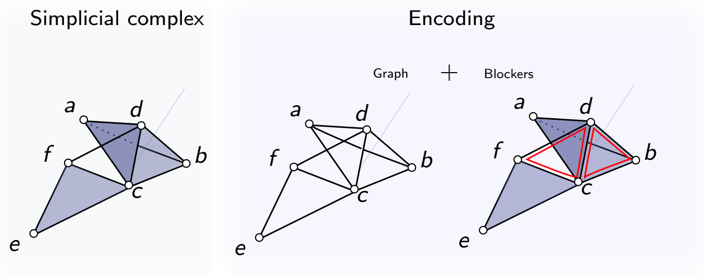


Encoding

Graph + Blockers

Skeleton-blockers data-structure

Definition of the data-structure



Simplicial complex

Encoding

Graph + Blockers

- $Sk^{(1)}(K) = \{ab, ac, ad, bc, bd, dc, df, ec, fc, fe\}$
- $Blockers(K)$=$\{bcd, cdf\}$

Skeleton-blockers data-structure [ALS 11]

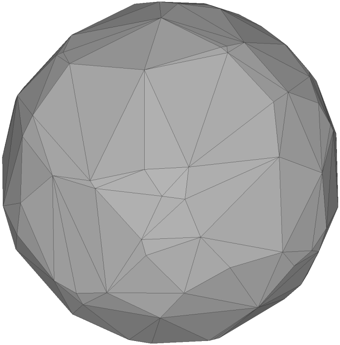
- $Blockers(K) = \{\sigma \subset P \mid \sigma \notin K \text{ and } \forall \tau \subsetneq \sigma, \tau \in K\}$
- The pair $[Sk^{(1)}(K), Blockers(K)]$ is sufficient to encode entirely K

Data structure size

- How many blockers in a simplicial complex?
- None for flag complexes such as the Rips complex
- How many in Delaunay, Tangential Delaunay, Cech complexes?
- How many in random 3-spheres?
  (3-dimensional manifold embedded in $\mathbb{R}^4$)

Data structure size

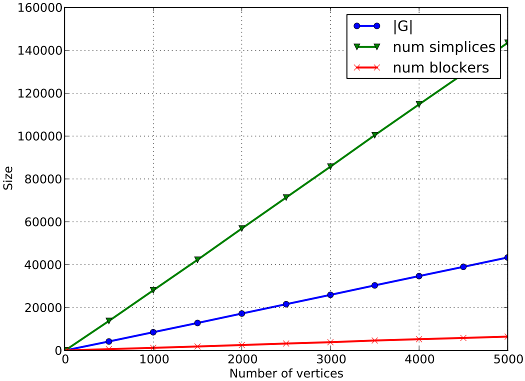
- How many blockers in a simplicial complex?
- None for flag complexes such as the Rips complex
- How many in Delaunay, Tangential Delaunay, Cech complexes?
- How many in random 3-spheres?
  (3-dimensional manifold embedded in $\mathbb{R}^4$)



Random 2-sphere with 200 points

Data structure size

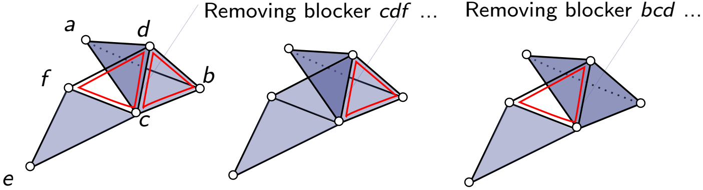
- How many blockers in a simplicial complex?
- None for flag complexes such as the Rips complex
- How many in Delaunay, Tangential Delaunay, Cech complexes?
- How many in random 3-spheres?
  (3-dimensional manifold embedded in $\mathbb{R}^4$)



Data structure size

Popable blockers

- Removing a blocker in the data-structure may change the homotopy type



Removing blocker *cdf* ...

Removing blocker *bcd* ...

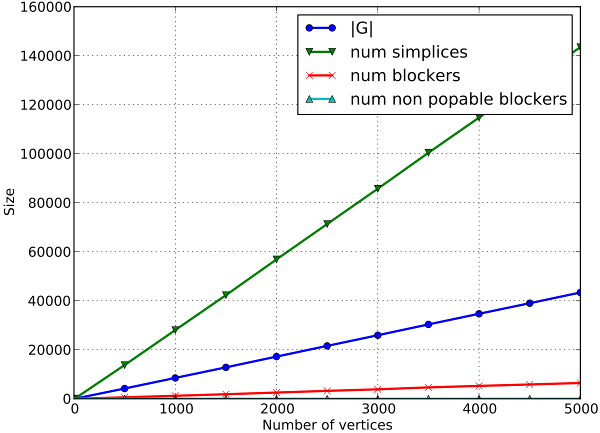
changes homotopy type

preserves homotopy type

- Removing a popable blocker does not change the homotopy type

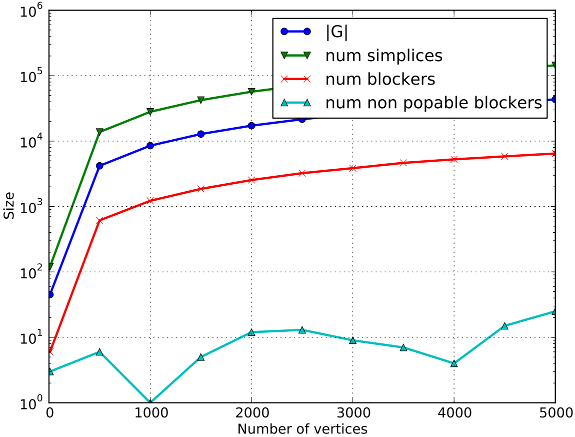
Data structure size

- How many blockers in a simplicial complex?
- How many non popable blockers in random 3-spheres? (3-dimensional manifold embedded in $\mathbb{R}^4$)



Data structure size

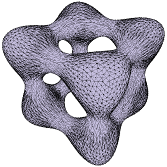
- How many blockers in a simplicial complex?
- How many non popable blockers in random 3-spheres?
  (3-dimensional manifold embedded in $\mathbb{R}^4$)



Edge contractions

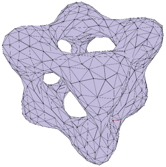
Definition

To simplify a complex, perform iterative topology-preserving edge contraction :

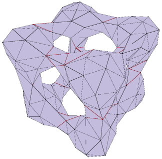


Rips complex

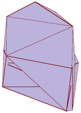
($\approx 70.10^6$ simplices)



6000 contractions



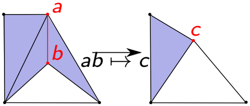
6700 contractions



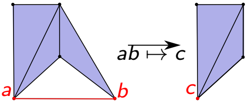
6787 contractions

(≈ 100 simplices)

- Contracting an edge = identify two vertices in the complex
- May change the homotopy type if link condition does not hold



$ab \mapsto c$ preserves the homotopy type

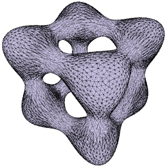


$ab \mapsto c$ changes the homotopy type

Edge contractions

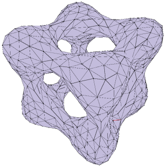
Definition

To simplify a complex, perform iterative topology-preserving edge contraction :

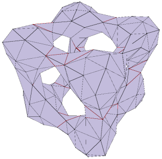


Rips complex

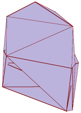
($\approx 70.10^6$ simplices)



6000 contractions



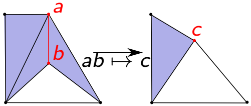
6700 contractions



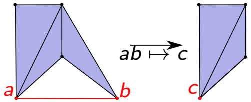
6787 contractions

(≈ 100 simplices)

- Contracting an edge = identify two vertices in the complex
- May change the homotopy type if link condition does not hold



$ab \mapsto c$ preserves the homotopy type



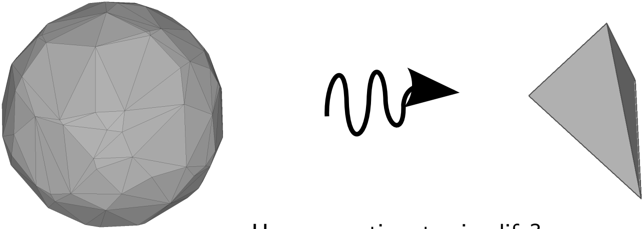
$ab \mapsto c$ changes the homotopy type

- Link condition iff no blocker through the edge

Edge contraction implementation efficiency

How much time to reduce a sphere?

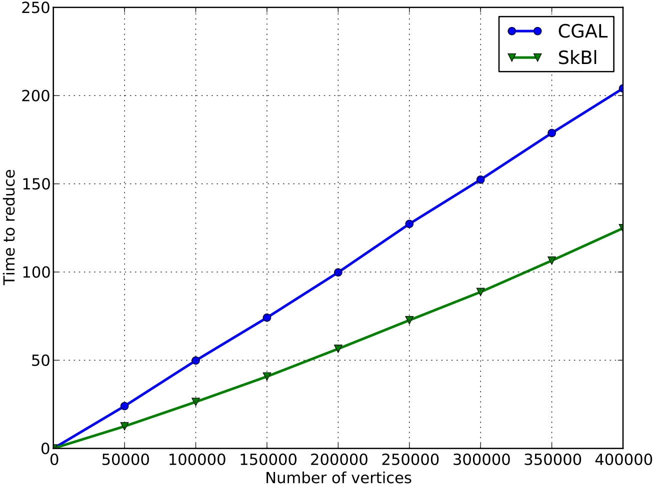
- Draw a random sphere in $\mathbb{R}^3$ with n points
- How much time to simplify to a tetrahedron with edge contraction?



How many time to simplify?

Edge contraction implementation efficiency

How much time to reduce a sphere?



- CGAL : Surface mesh simplification package (Polyhedron_3)
- ≈ 65% times faster

Skeleton-blockers data-structure

Implementation

- Template based, user can provide its vertices / edges classes
- Constructor from graph/blockers / list simplices / top faces
- STL iterators for vertices / edges / blockers / simplices
- Edge contraction, collapse, ...

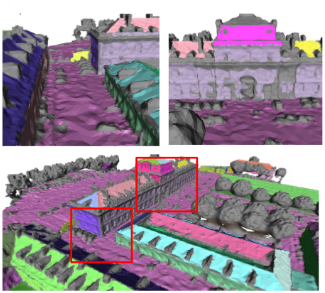
Simplex iteration

```
typedef ... Complex;

Complex complex;

// build complex

for(auto s : complex.simplex_range())
        cout << s <<endl;
```

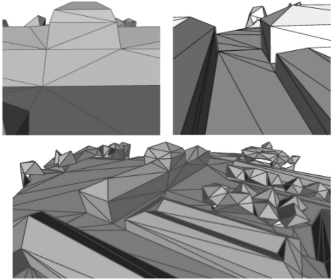
Edge contraction implementation

Genericity

- Policy-based design (adapted from CGAL Surface Mesh Simplification package)
- The user can provide it own cost, placement, validity and visitor policies
- Genericity : simplification of urban mesh (submitted to CGF)



Input mesh



Mesh simplified with SB package and specific policies

- Homology computation

Time for a demo

Conclusion

On a practical point of view :

- Very sparse representation
- Generic and efficient simplification

On a theoretical point of view :

- How many blockers and non popable blockers in a manifold?
- Persistence?