International Journal of Computational Geometry & Applications

EFFICIENT DATA STRUCTURE FOR REPRESENTING AND SIMPLIFYING SIMPLICIAL COMPLEXES IN HIGH DIMENSIONS

Dominique Attali

Gipsa-lab
Grenoble, France
dominique.attali@gipsa-lab.grenoble-inp.fr

André Lieutier

Dassault Système
Aix-en-Provence, France
andre.lieutier@3ds.com

David Salinas

Gipsa-lab
Grenoble, France
david.salinas@gipsa-lab.grenoble-inp.fr

We study the simplification of simplicial complexes by repeated edge contractions. First, we extend to arbitrary simplicial complexes the statement that edges satisfying the *link condition* can be contracted while preserving the homotopy type. Our primary interest is to simplify *flag complexes* such as *Rips complexes* for which it was proved recently that they can provide topologically correct reconstructions of shapes. Flag complexes (sometimes called *clique complexes*) enjoy the nice property of being completely determined by the graph of their edges. But, as we simplify a flag complex by repeated edge contractions, the property that it is a flag complex is likely to be lost. Our second contribution is to propose a new representation for simplicial complexes particularly well adapted for complexes close to flag complexes. The idea is to encode a simplicial complex K by the graph G of its edges together with the inclusion-minimal simplices in the set difference $\mathrm{Flag}(G) \setminus K$. We call these minimal simplices *blockers*. We prove that the link condition translates nicely in terms of blockers and give formulae for updating our data structure after an edge contraction. Finally, we observe in some simple cases that few blockers appear during the simplification of Rips complexes, demonstrating the efficiency of our representation in this context.

Keywords: Simplicial complexes; data structure; flag complexes; clique complexes; Vietoris-Rips complexes; shape reconstruction; shape simplification; edge contraction; collapse; homotopy equivalence; high dimensions.

1. Introduction

As datasets are growing larger in size and in dimension, simplicial complexes built upon these data become gigantic, challenging our ability to extract useful and concise information. In particular, storing all simplices becomes prohibitive. A way to overcome this difficulty is reducing the size of the simplicial complex prior to analysis. During that process, it is desirable to preserve the homotopy type.

In this work, we focus on the simplification of a particular class of simplicial complexes, likely to be encountered in high dimensional data analysis and manifold learning. Specifically, we are interested in *flag complexes* also known as *clique complexes* that have the property of containing simplices wherever the adjacency of vertices permits one. Precisely, the flag complex of a graph G is the largest simplicial complex whose 1-skeleton is G. Obviously, flag complexes are completely determined by their 1-skeletons, which provide a concise form of storage. A standard way of building the 1-skeleton of a flag complex is to consider the *proximity graph* of a point cloud. The flag complex of such a graph is called a *Rips complex.* In the light of recent results,[1,3] Rips complexes seem to be good candidates for reproducing the homotopy type of the shape sampled by the point cloud. In this context, simplification can be used as a preprocessing phase for reducing, for instance, the cost of computing topological invariants such as Betti numbers.[9,16])

Following what has been done within the computer graphics and visualization communities, one can consider several elementary operations for simplifying a simplicial complex: vertex removal,[18] vertex clustering,[17] triangle contraction.[11] We primarily concentrate here on *edge contraction*, the operation that consists in merging two vertices. It was used in the pioneering work of Hoppe et al.[12] for generating progressive meshes and has been intensively studied ever since. Garland and Heckbert[10] proposed an elegant way of prioritizing edge contractions for surface simplification. Dey et al.[7] introduced a local condition called the *link condition* that characterizes edge contractions that permit a homeomorphic modification of 2- and 3-manifolds.

It would be tempting as we repeatedly apply edge contractions on a flag complex to keep its nature of flag complex, thus preserving its light form of storage along the simplification process. As already observed by Zomorodian in Ref. 22 and confirmed by our first experiments, this seems to be "almost" possible. Indeed, Zomorodian uses a representation by simplicial sets that allows him to collapse any edge while keeping the homotopy type unchanged. He observed that, along the simplification process, most cells remain regular simplices. We suggest here another strategy that preserves the representation by simplicial complexes along the simplification.

Our first contribution is the proof that the link condition introduced in Ref. 7 can also be used to guarantee homotopy-preserving edge contractions in arbitrary simplicial complexes. Our second contribution is a new data structure for encoding arbitrarily simplicial complexes. This data structure is particularly well-adapted for high-dimensional simplicial complexes which are "almost" flag complexes. Besides the 1-skeleton, we encode parsimoniously how the complex differs from the flag complex of its 1-skeleton. Precisely, we represent any simplicial complex K by its 1-skeleton G together with the set of inclusion-minimal simplices in the set difference $\mathrm{Flag}(G) \setminus K$. These minimal simplices are called *blockers*. The intuition is that simplicial complexes "close" to flag complexes will have a small amount of blockers.

We show that the link condition translates nicely in terms of blockers in our new data structure and give an explicit expression of the blockers created (and destroyed) during an edge contraction. We have implemented the data structure and the edge contraction operation. Our first experiments indicate that the simplification of Rips complexes in some simple cases and using a reasonable strategy for prioritizing edge contractions leads to the apparition of very few blockers. This seems to make the proposed representation efficient in practice.

When drawing simplicial complexes in the figures, we adopt the convention that besides drawing 1-skeletons, either we shade inclusion-maximal simplices or hatch blockers. When no triangles are shaded or hatched, the convention is that the blocker set is empty or equivalently that the simplicial complex is a flag complex.

2. Basic definitions

In this section, we recall standard definitions and notations that can be found in textbooks.[14] The cardinality of a set X will be denoted $\sharp X$. We will use the notation $A \subset B$ to indicate that A is a subset of B and $A \subsetneq B$ to indicate that A is a proper subset of B.

2.1. *Abstract simplicial complexes*

An *abstract simplex* is any finite non-empty set. The *dimension* of a simplex σ is one less than its cardinality, $\dim \sigma = \sharp\sigma - 1$. A k-simplex designates a simplex of dimension k. If $\tau \subset \sigma$ is a non-empty subset, we call τ a *face* of σ and σ a *coface* of τ. If in addition $\tau \subsetneq \sigma$, we say that τ is a *proper face* and σ is a *proper coface*. An *abstract simplicial complex* is a collection of simplices, K, that contains, with every simplex, the faces of that simplex. The dimension of K is the largest dimension of one of its simplices. The *closure* of a set of simplices Σ, denoted $\mathrm{Cl}(\Sigma)$, is the smallest simplicial complex containing Σ. The *closure* of a simplex σ is $\mathrm{Cl}(\sigma) = \bigcup_{\emptyset \neq \tau \subset \sigma} \{\tau\}$. The vertex set of the abstract simplicial complex K is the union of its elements, $\mathrm{Vert}(K) = \bigcup_{\sigma \in K} \sigma$. A *subcomplex* of K is a simplicial complex $L \subset K$. A particular subcomplex is the i-skeleton consisting of all simplices of dimension i or less, which we denote by $K^{(i)}$. The 0-skeleton is the set of inclusion-minimal simplices. Besides classical definitions, the following concept will be useful:

Definition 1. Let K be a simplicial complex whose dimension is k. The *expansion* of K, denoted $\mathrm{Expand}(K)$, is the largest simplicial complex having K as a k-skeleton. In particular, the expansion of a graph G is the *flag complex* of G and is denoted $\mathrm{Flag}(G)$.

Throughout the paper, we will restrict ourselves to finite simplicial complexes. Note that the expansion of the 0-skeleton $K^{(0)}$ is the power set of the vertex set $\mathrm{Vert}(K)$ minus the empty set, $\mathrm{Expand}\left(K^{(0)}\right) = 2^{\mathrm{Vert}(K)} \setminus \{\emptyset\}$ and consists of all simplices $\sigma \subset \mathrm{Vert}(K)$ spanned by vertices in K. In particular, it has a unique inclusion-maximal simplex which is $\mathrm{Vert}(K)$. In Section 4, we will be interested by expansions of i-skeletons, $\mathrm{Expand}\left(K^{(i)}\right)$, which form for increasing values of i an inclusion-decreasing sequence of simplicial complexes all containing K; see Figure 3 for a schematic drawing of K and the expansion of its 0- and 1-skeletons.

2.2. *Intersection and union*

Two abstract simplices τ and σ are disjoint if they have no vertices in common or equivalently if $\tau \cap \sigma = \emptyset$. It will be convenient to denote the union of two simplices σ and τ simply $\sigma\tau$ instead of $\sigma \cup \tau$. In the same spirit, we shall use indifferently one of the two notations $\{v_0, v_1, \ldots, v_k\}$ or $v_0 v_1 \ldots v_k$ to designate the k-simplex spanned by vertices v_0, v_1, $\ldots$, v_k. In particular, we shall make no distinction between the vertex v of K and the 0-simplex $\{v\} \in K$. We shall also use the notation ab instead of $\{a, b\}$ to designate the edge connecting vertex a and vertex b.

2.3. *Underlying space*

Let $\pi : \mathrm{Vert}(K) \to \mathbb{R}^n$ be an injective map that sends the n vertices of K to n affinely independent points of $\mathbb{R}^n$, such as for instance the n vectors of the standard basis of $\mathbb{R}^n$. The *underlying space* of K is the point set $|K| = \bigcup_{\sigma \in K} \mathrm{Hull}\, \pi(\sigma)$ and is defined up to a homeomorphism. We shall say that a transformation f between two simplicial complexes K and K' preserves the topological type if the underlying spaces of K and K' are homeomorphic, $|K| \approx |K'|$ and we say that f preserves the homotopy type if the underlying spaces are homotopy equivalent, $|K| \simeq |K'|$.

3. Homotopy-preserving edge contraction

In this section, we give a local condition on the link of an edge ab in a simplicial complex K under which the contraction of the edge ab preserves the homotopy type of K. This condition, called the *link condition*, was introduced in Ref. 7 to characterize edge contractions that permit a homeomorphic modification when the simplicial complex K is the triangulation of a 2-manifold or a 3-manifold. Unlike previous works,[7,19] we make no assumptions on the simplicial complex K. In particular, we do not require that K triangulates a manifold.

3.1. *Edge contraction*

Recall that $\mathrm{Vert}(K)$ designates the set of vertices of K and consider $a, b \in \mathrm{Vert}(K)$ and $c \notin \mathrm{Vert}(K)$. To describe the edge contraction $ab \mapsto c$, we define a vertex map f that takes vertices a and b to c and takes all other vertices to themselves:

$$f(v) = \begin{cases} c & \text{if } v \in \{a, b\}, \\ v & \text{if } v \notin \{a, b\}. \end{cases} \tag{1}$$

We then extend f to all simplices $\sigma = \{v_0, \ldots, v_k\}$ of K, setting $f(\sigma) = \{f(v_0), \ldots, f(v_k)\}$. The edge contraction $ab \mapsto c$ is the operation that changes

K to $K' = \{f(\sigma) \mid \sigma \in K\}$. By construction f is surjective and K' is a simplicial complex. Note that the edge contraction $ab \mapsto c$ is well defined even when ab does not belong to K.

3.2. *Link condition*

Let σ be a simplex of the simplicial complex K. The *link* of σ in K is the simplicial complex

$$\mathrm{Lk}_K(\sigma) = \{\tau \in K \mid \tau \cup \sigma \in K, \tau \cap \sigma = \emptyset\}.$$

When K is clear from the context, we will drop it and denote the link of σ in K simply by $\mathrm{Lk}(\sigma)$. In particular, whenever we contract an edge $ab \in K$ to a new vertex $c \in K'$, it is unambiguous to write $\mathrm{Lk}(a)$, $\mathrm{Lk}(b)$, $\mathrm{Lk}(ab)$ for the links of a, b and ab in K and $\mathrm{Lk}(c)$ for the link of c in K'. We are now ready to state our main result (see Figure 1).

Theorem 1 (Link condition theorem). *Let K be a simplicial complex. The contraction of the edge $ab \in K$ preserves the homotopy type whenever* $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$.

An example of edge ab satisfying the link condition $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$ is given in Figure 1, where simplicial complexes are depicted with the convention adopted at the end of the introduction. Note that the converse of Theorem 1 is in general not true (see Figure 2). The end of the section is devoted to the proof of Theorem 1. First, we review the Nerve Theorem on which rests our proof. We then give the proof before finishing the section with two technical lemmas about links.

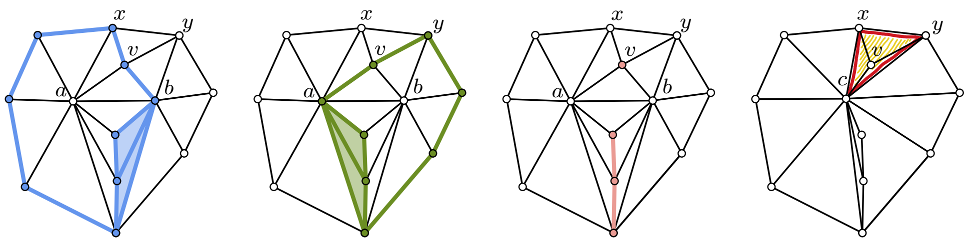

Fig. 1. Form left to right: link of a, link of b, link of ab and simplicial complex after contraction of the edge ab. One can check that ab satisfies the link condition. Equivalently, no blocker contains ab (see Section 4.3). As a consequence, the contraction of ab preserves the homotopy type. Note that the edge contraction $ab \mapsto c$ creates the blocker cxy and $\alpha = x$ and $\beta = y$ satisfy (i) and (ii) in Lemma 4. The contraction of any of the edges cv, xv and yv removes blocker cxy.

Recall that the *nerve* of a finite collection of sets, $\mathcal{F}$ is the simplicial complex that consists of all non-empty subcollections whose sets have a non-empty common intersection,

$$\mathrm{Nrv}\, \mathcal{F} = \{\mathcal{G} \subset \mathcal{F} \mid \bigcap \mathcal{G} \neq \emptyset\}$$

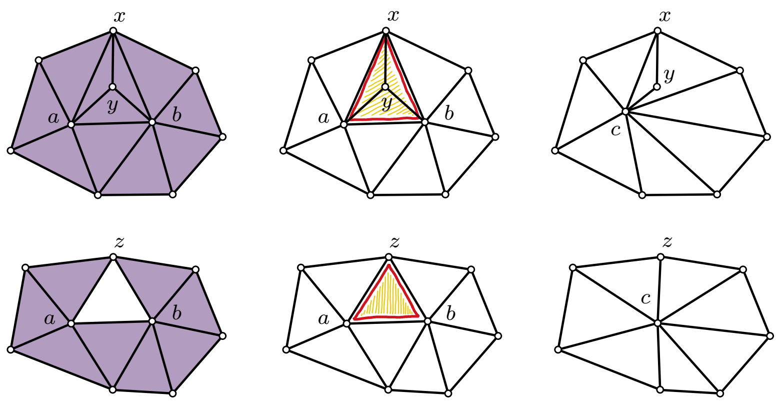

Fig. 2. Left and middle: two 2-dimensional simplicial complexes with a blocker through ab. Right: The edge contraction $ab \mapsto c$ may (top) or may not (bottom) preserves the homotopy type.

The Nerve Theorem has several versions,[5] one of the earliest being due to Leray.[13] For the proof of Theorem 1, we shall use the following form:

Theorem 2 (Nerve Theorem). *Consider a triangulable space X which is the union of a finite collection of closed sets $\mathcal{F}$, i.e. $X = \bigcup \mathcal{F}$. If for every subcollection $\mathcal{G} \in \mathrm{Nrv}\,\mathcal{F}$, the intersection $\bigcap \mathcal{G}$ is contractible, then the underlying space of $\mathrm{Nrv}\,\mathcal{F}$ is homotopy equivalent to X.*

Proof of Theorem 1. Suppose $ab \in K$ satisfies the link condition $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$ and let K' be the complex obtained after the edge contraction $ab \mapsto c$. The proof considers two coverings, one for $|K|$ and one for $|K'|$, whose nerves N and N' are proved to be isomorphic and for which we establish that $|K| \simeq |N|$ and $|N'| \simeq |K'|$. By abuse of language, we will write $|\sigma|$ for the underlying space of the closure of σ.

Consider first the collection of sets $\{|\sigma'|, \sigma' \in K'\}$ which covers $|K'|$ and let N' denote its nerve. Clearly, for any non-empty subcollection $\Sigma' \subset K'$, the intersection $\bigcap_{\sigma' \in \Sigma'} \sigma'$ is either empty or a simplex of K' and therefore the common intersection $\bigcap_{\sigma' \in \Sigma'} |\sigma'|$ of sets in the subcollection $\{|\sigma'|, \sigma' \in \Sigma'\}$ is either empty or contractible. The Nerve Theorem then implies that $|K'| \simeq |N'|$. Let $\bar{f} : |K| \to |K'|$ be the simplicial map induced by the vertex map $f : \mathrm{Vert}(K) \to \mathrm{Vert}(K')$ defined in Eq. (1). Consider the collection of sets $\{\bar{f}^{-1}[|\sigma'|], \sigma' \in K'\}$ obtained by taking the preimages of sets in the first collection. This collection covers $|K|$ and we denote its nerve by N. The two nerves N and N' are isomorphic because the surjectivity of $\bar{f}$ implies that for all $\Sigma' \subset K'$, we have the equivalence:

$$\bigcap_{\sigma' \in \Sigma'} \bar{f}^{-1}[|\sigma'|] = \bar{f}^{-1}\left[\bigcap_{\sigma' \in \Sigma'} |\sigma'|\right] \neq \emptyset \iff \bigcap_{\sigma' \in \Sigma'} |\sigma'| \neq \emptyset.$$

Furthermore, if the intersection on the right-hand side $\bigcap_{\sigma' \in \Sigma'} |\sigma'|$ is non-empty, then there exists a simplex $\tau' \in K'$ such that $\tau' = \bigcap_{\sigma' \in \Sigma'} \sigma'$ and by Lemma 2 which is proved below, the intersection on the left-hand side $\bigcap_{\sigma' \in \Sigma'} \bar{f}^{-1}[|\sigma'|] = \bar{f}^{-1}[|\tau'|] = |f^{-1}[\mathrm{Cl}(\tau')]|$ is contractible. To summarize, we established that $|K| \simeq |N| \approx |N'| \simeq |K'|$, showing that $|K|$ and $|K'|$ have same homotopy type. $\square$

The proof of Theorem 1 requires to establish that $|f^{-1}[\mathrm{Cl}(\tau)]|$ is contractible for all simplices τ in $K' = f(K)$ under the condition that $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$. We will prove a stronger result, namely that $|f^{-1}[\mathrm{Cl}(\tau)]|$ is collapsible. Recall that the *star* of a simplex σ in K, denoted $\mathrm{St}_K(\sigma)$, is the collection of simplices of K having σ as a face. Provided that there is a unique inclusion-maximal simplex $\eta \neq \sigma$ in the star of σ, it is well-known that $|K|$ deformation retracts to $|K \setminus \mathrm{St}_K(\sigma)|$ and the operation that removes $\mathrm{St}_K(\sigma)$ is then called a *collapse*.[8] A simplicial complex is said to be *collapsible* if it can be reduced to a single vertex by a finite sequence of collapses. In particular, the underlying space of a collapsible complex is contractible. We start with a technical lemma.

Lemma 1. *Let $\sigma \subset \mathrm{Vert}(K) \setminus \{a, b\}$ be a simplex spanned by vertices of K disjoint from a and b. The simplex $c\sigma$ belongs to K' if and only if either $a\sigma$ or $b\sigma$ belongs to K. Equivalently, $\sigma \in \mathrm{Lk}(c)$ if and only if $\sigma \in \mathrm{Lk}(a) \cup \mathrm{Lk}(b)$.*

Proof. Using $f^{-1}[\{c\sigma\}] = \{a\sigma, b\sigma, ab\sigma\} \cap K$ and the surjectivity of f, we get that $c\sigma \in K' \iff f^{-1}[\{c\sigma\}] \neq \emptyset \iff \{a\sigma, b\sigma, ab\sigma\} \cap K \neq \emptyset \iff \{a\sigma, b\sigma\} \cap K \neq \emptyset \iff a\sigma \in K$ or $b\sigma \in K$. $\square$

Lemma 2. *Suppose $ab \in K$ satisfies $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$ and let K' be the simplicial complex obtained after the edge contraction $ab \mapsto c$. The preimage $f^{-1}[\mathrm{Cl}(\sigma)]$ of the closure of any simplex $\sigma \in K'$ is non-empty and collapsible.*

Proof. For all $\sigma \in K'$, we give an expression of the preimage $f^{-1}[\mathrm{Cl}(\sigma)]$ which entails its collapsibility. Recalling that the closure of a simplex is $\mathrm{Cl}(\sigma) = \bigcup_{\emptyset \neq \tau \subset \sigma} \{\tau\}$ and noting that the preimage of a union is the union of the preimages, we consider three cases:

Case 1: $f^{-1}[\mathrm{Cl}(c)] = f^{-1}[\{c\}] = \{a, b, ab\}$ is collapsible.

Case 2: If $\sigma \cap c = \emptyset$, then for all faces τ of σ, we also have $\tau \cap c = \emptyset$ and therefore $f^{-1}[\{\tau\}] = \{\tau\}$. It follows that

$$f^{-1}[\mathrm{Cl}(\sigma)] \quad = \quad \bigcup_{\emptyset \neq \tau \subset \sigma} f^{-1}[\{\tau\}] \quad = \quad \bigcup_{\emptyset \neq \tau \subset \sigma} \{\tau\} \quad = \quad \mathrm{Cl}(\sigma).$$

Case 3: If σ belongs to the link of c in K', then $\sigma \in \mathrm{Lk}(a) \cup \mathrm{Lk}(b)$ by Lemma 1. If in addition ab satisfies the link condition $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$, this implies that σ belongs either to $\mathrm{Lk}(ab)$ or to $\mathrm{Lk}(a) \setminus \mathrm{Lk}(ab)$ or to $\mathrm{Lk}(b) \setminus \mathrm{Lk}(ab)$. Observing that the same is true for all faces τ of σ, we deduce immediately that for all $\emptyset \neq \tau \subset \sigma$

$$f^{-1}[\{c\tau\}] = \begin{cases} \{a\tau, b\tau, ab\tau\} & \text{if } \tau \in \mathrm{Lk}(ab), \\ \{a\tau\} & \text{if } \tau \in \mathrm{Lk}(a) \setminus \mathrm{Lk}(ab), \\ \{b\tau\} & \text{if } \tau \in \mathrm{Lk}(b) \setminus \mathrm{Lk}(ab). \end{cases}$$

Since $\mathrm{Cl}(c\sigma) = \{c\} \cup \mathrm{Cl}(\sigma) \cup \bigcup_{\emptyset \neq \tau \subset \sigma} \{c\tau\}$, we obtain that

$$f^{-1}[\mathrm{Cl}(c\sigma)] = \{a, b, ab\} \ \cup \ \mathrm{Cl}(\sigma) \ \cup \ \bigcup_{\emptyset \neq \tau \subset \sigma} f^{-1}[\{c\tau\}].$$

Writing $\Sigma \cdot \Sigma' = \{\sigma\sigma' \mid \sigma \in \Sigma, \sigma' \in \Sigma\}$ and setting $L = \mathrm{Cl}(\sigma) \cap \mathrm{Lk}(ab)$ we get that

$$f^{-1}[\mathrm{Cl}(c\sigma)] = \begin{cases} \mathrm{Cl}(ab\sigma) & \text{if } \sigma \in \mathrm{Lk}(ab), \\ \mathrm{Cl}(a\sigma) \cup \{b, ab\} \cup \{b, ab\} \cdot L & \text{if } \sigma \in \mathrm{Lk}(a) \setminus \mathrm{Lk}(ab), \\ \mathrm{Cl}(b\sigma) \cup \{a, ab\} \cup \{a, ab\} \cdot L & \text{if } \sigma \in \mathrm{Lk}(b) \setminus \mathrm{Lk}(ab). \end{cases}$$

Hence, if $\sigma \in \mathrm{Lk}(ab)$, the preimage $f^{-1}[\mathrm{Cl}(c\sigma)]$ is clearly collapsible. If $\sigma \in \mathrm{Lk}(a) \setminus \mathrm{Lk}(ab)$, we can always find a set of simplices $\lambda_1, \ldots, \lambda_k$ whose closure is equal to the simplicial complex $L = \mathrm{Cl}(\sigma) \cap \mathrm{Lk}(ab)$ and such that, for all $1 \leq i, j \leq k$, the simplex λ_i is neither a face nor a coface of the simplex λ_j. In other words, the set of simplices $\lambda_1, \ldots, \lambda_k$ are the inclusion-maximal simplices of L. By construction, $ab\lambda_i$ is the only proper coface of $b\lambda_i$ in $f^{-1}[\mathrm{Cl}(c\sigma)]$. After a sequence of k elementary collapses consisting in removing pairs of simplices $(b\lambda_i, ab\lambda_i)$, we are left with the simplicial complex $\mathrm{Cl}(a\sigma) \cup \{b, ab\}$ which is collapsible. The case $\sigma \in \mathrm{Lk}(b) \setminus \mathrm{Lk}(ab)$ is done similarly. □

4. Encoding complexes with their skeletons and blocker sets

It is common to represent a simplicial complex K of small dimension by the subset $L \subset K$ of simplices that are inclusion-maximal, that is, the set of simplices of K which have no proper cofaces in K (see Figure 4, top left). The simplicial complex K can then be recovered from L by taking the closure, $K = \mathrm{Cl}(L)$. In this section, we introduce a new way of representing simplicial complexes (see Figures 4 to 5). Roughly, we store the 1-skeleton G of K together with a minimal set of simplices called *blockers* that indicates how much K differs from the flag complex of G. First, we describe our data structure for encoding simplicial complexes. Then, we explain how to check the link condition and how to maintain the data structure as we contract edges. Pseudo-codes and time complexities are given in Appendix A.

4.1. *Data structure*

Definition 2. Let $i \geq 0$. We say that a simplex $\sigma \subset \mathrm{Vert}(K)$ is an *order-i blocker* of K if it satisfies the following three conditions (1) $\dim \sigma > i$; (2) σ does not belong to K; (3) all proper faces of σ belong to K. The set of order-i blockers of K is denoted $\mathrm{Blockers}_i(K)$.

Equivalently, the order-i blockers of K are the inclusion-minimal simplices of $\mathrm{Expand}\left(K^{(i)}\right) \setminus K$ for all $i \geq 0$; see Figures 3 and 4. A key consequence is that the pair $(K^{(i)}, \mathrm{Blockers}_i(K))$ encodes entirely the simplicial complex K. Indeed, the simplicial complex whose i-skeleton is S and whose order-i blocker set is B can be retrieved from the pair (S, B) using the formula

$$K = \{\sigma \in \mathrm{Expand}\,(S) \mid \sigma \text{ has no face in } B\}. \tag{2}$$

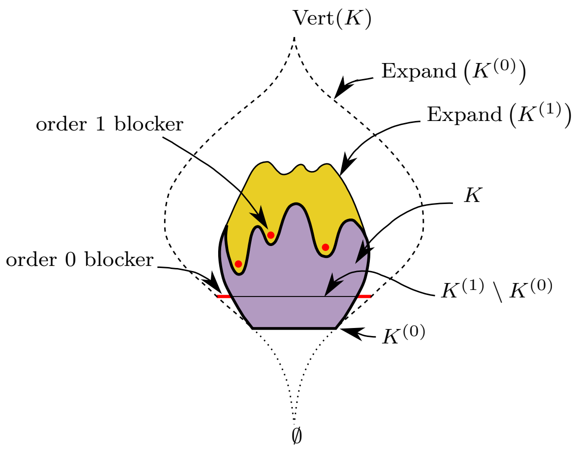

Fig. 3. Hasse diagram of K.

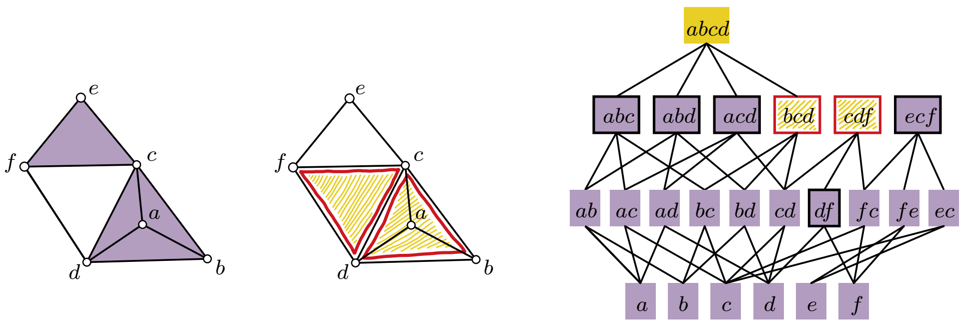

Fig. 4. Left: simplicial complex consisting of six vertices, ten edges and four non-overlapping shaded triangles. Middle: same simplicial complex represented by its 1-skeleton and order-1 blocker set $\{cdf, bcd\}$. Right: Hasse diagram of the expansion of the 1-skeleton. Nodes in dark gray are in the simplicial complex. Blockers and inclusion-maximal simplices are shown as framed nodes.

In this paper, we are primarily interested in simplicial complexes K "close" to flag complexes with a "small" 1-skeleton and therefore choose to represent them by the pair $(K^{(1)}, \mathrm{Blockers}_1(K))$. Indeed, if $K = \mathrm{Flag}(G)$ is a flag complex, then its blocker set is empty and K can be represented by the pair $(G, \emptyset)$. As we simplify the simplicial complex by edge contractions, we hope that the blocker set will remain small. This intuition is sustained by experiments we make in Section 5 in which K is the Rips complex of a point set that samples a shape. Hereafter, blockers will always refer to order-1 blockers.

Let $\mathcal{N}(v) = \mathcal{N}_K(v)$ be the set of vertices $w \neq v$ such that $vw \in K$ and write $\mathcal{B}(v) = \mathcal{B}_K(v)$ for the set of blockers that contain v. Clearly, encoding the pair $(K^{(1)}, \mathrm{Blockers}_1(K))$ boils down to encoding for each vertex v of K the pair $(\mathcal{N}(v), \mathcal{B}(v))$. Precisely, our data structure consists of a linear array V for the vertices and records for each vertex v the set of neighbors $\mathcal{N}(v)$ and a set of pointers to blockers in $\mathcal{B}(v)$ as illustrated in Figure 5. It follows that the size of our data structure is a constant times $\sum_{v \in \mathrm{Vert}(K)} (1 + \sharp\mathcal{N}(v) + 2\sharp\mathcal{B}(v))$. To see this, charge each vertex in a blocker to its corresponding vertex in V. During the operation, each vertex in V is charged at most $\sharp\mathcal{B}(v)$ times.

To conclude this section, we give a crude upper bound on the dimension and number of blockers in a simplicial complex with n vertices. Consider a blocker σ passing through a vertex v. Since $\sigma \subset \{v\} \cup \mathcal{N}(v)$, we get $\sharp\sigma \leq 1 + \sharp\mathcal{N}(v)$ and therefore $\dim \sigma \leq \max_{v \in \sigma} \sharp\mathcal{N}(v)$. It follows that $N_{\max} = \max_{v \in \mathrm{Vert}(K)} \sharp\mathcal{N}(v)$ is an upper bound on the dimension of the blockers and $O(2^{N_{\max}})$ is an upper bound on the number of blockers through v. The total number of blockers in our data structure is $O(n2^{N_{\max}})$.

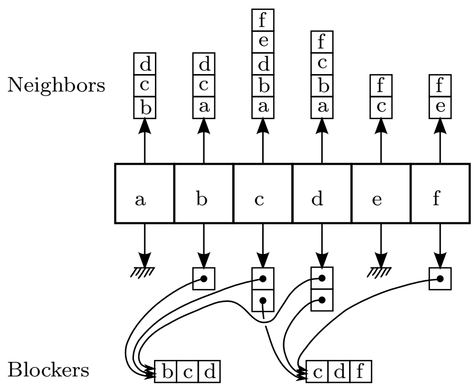

Fig. 5. Data structure representing the simplicial complex in Figure 4.

4.2. *Testing whether a simplex belongs to the complex*

Recall that a simplex σ belongs to K if and only if σ belongs to Expand $\left(K^{(i)}\right)$ and σ has no face in $\text{Blockers}_i(K)$ (see Eq. (2)). It will be convenient to use this equivalence with $i = 0$ for the proofs and $i = 1$ for computations. Precisely, in the proofs, we will use that for all $\sigma \subset \text{Vert}(K)$, we have the equivalence: $\sigma \in K \iff \sigma$ has no face in $\text{Blockers}_0(K)$. For the computations, we will set $i = 1$ and test whether $\sigma \subset \text{Vert}(K)$ belongs to K by checking whether its edges belong to the 1-skeleton and σ contains no order-1 blocker; see Algorithm 4 in Appendix A for the details.

4.3. *Checking the Link Condition*

The next lemma formulates the link condition in terms of blockers.

Lemma 3. *Let K be a simplicial complex. The edge $ab \in K$ satisfies the link condition* $\text{Lk}(ab) = \text{Lk}(a) \cap \text{Lk}(b)$ *if and only if no blocker of K contains ab.*

Proof. It is not difficult to see that for all simplicial complexes K and for all edges $ab \in K$, we have the inclusion $\text{Lk}(ab) \subset \text{Lk}(a) \cap \text{Lk}(b)$. Let us prove that $\text{Lk}(ab) = \text{Lk}(a) \cap \text{Lk}(b)$ implies that no blocker of K contains ab. Suppose for a contradiction that the simplex $ab\tau$ is a blocker of K for some simplex τ such that $ab \cap \tau = \emptyset$. By definition of a blocker, all proper faces of $ab\tau$ belong to K and in particular $a\tau \in K$ and $b\tau \in K$. On the other hand, $ab\tau$ does not belong to K. It follows that $\tau \in \text{Lk}(a)$, $\tau \in \text{Lk}(b)$ and $\tau \notin \text{Lk}(ab)$, implying that $\text{Lk}(ab) \neq \text{Lk}(a) \cap \text{Lk}(b)$. Conversely, suppose no blocker of K contains ab and let us prove that $\text{Lk}(a) \cap \text{Lk}(b) \subset \text{Lk}(ab)$. Consider a simplex $\sigma \in \text{Lk}(a) \cap \text{Lk}(b)$. By definition, $a\sigma \in K$, $b\sigma \in K$ and $ab \cap \sigma = \emptyset$. We claim that $ab\sigma$ belongs to K and therefore $\sigma \in \text{Lk}(ab)$. Let us prove the claim by contradiction. Suppose some of the faces of $ab\sigma$ do not belong to K and let $\tau_{\min}$ be an inclusion-minimal face among them. In other words, $\tau_{\min}$ is an order-0 blocker. Since $\tau_{\min} \subset ab\sigma$, $\tau_{\min} \notin K$, $a\sigma \in K$ and $b\sigma \in K$, we must have $ab \subset \tau_{\min}$ which contradicts the assumption that no (order-1) blocker of K contains ab. $\square$

Testing whether an edge $ab \in K$ satisfies the link condition can be done by traversing the blockers through a and testing for each blocker whether it contains b. See Algorithm 1 in Appendix A for the pseudo-code and a discussion of its time complexity.

4.4. *Updating the data structure after an edge contraction*

In this section, we describe how to update the data structure after an edge contraction. More precisely, we consider a simplicial complex K and let K' be the simplicial complex obtained after the edge contraction $ab \mapsto c$. Our goal is to compute the pair $(\mathcal{N}_{K'}(c), \mathcal{B}_{K'}(c))$. Clearly, $\mathcal{N}_{K'}(c) = (\mathcal{N}(a) \setminus \{b\}) \cup (\mathcal{N}(b) \setminus \{a\})$. The next lemma prepares the computation of $\mathcal{B}_{K'}(c)$ by characterizing blockers through c.

Lemma 4. *Let K' be the simplicial complex obtained after the edge contraction $ab \mapsto c$. Suppose $\sigma \subset \mathrm{Vert}(K) \setminus \{a, b\}$ is a simplex with $\dim \sigma \geq 1$. The simplex $c\sigma$ belongs to* $\mathrm{Blockers}_1(K')$ *if and only if the following two conditions are fulfilled:*

(i) $\sigma \in K$*; every proper face of* σ *belongs to* $\mathrm{Lk}(a) \cup \mathrm{Lk}(b)$*;*
(ii) $\sigma = \alpha\beta$ *with* $a\beta \in \mathrm{Blockers}_0(K)$ *and* $b\alpha \in \mathrm{Blockers}_0(K)$*.*

Proof. First, note that the proper faces of $c\sigma$ belong to K' if and only if (i) is satisfied. Indeed, using Lemma 1, (i) is equivalent to $\sigma \in K'$ and $c\tau \in K'$ for all proper faces $\tau \subsetneq \sigma$. Hence, $c\sigma \in \mathrm{Blockers}_1(K') \implies$ (i). Let us prove that $c\sigma \in \mathrm{Blockers}_1(K') \implies$ (ii). Since $c\sigma \notin K'$, neither $a\sigma$ nor $b\sigma$ belongs to K. It follows that $a\sigma$ has a face in $\mathrm{Blockers}_0(K)$ and since $\sigma \in K$, this face must contain a. Let us denote this face $a\beta$ with $\beta \subset \sigma$. Similarly, since $b\sigma \notin K$, there exists a face $\alpha \subset \sigma$ such that $b\alpha \in \mathrm{Blockers}_0(K)$. Let us prove that $\alpha\beta = \sigma$. Suppose for a contradiction that $\alpha\beta$ is a proper face of σ. (i) implies that $\alpha\beta \in \mathrm{Lk}(a) \cup \mathrm{Lk}(b)$. If $\alpha\beta$ belongs to $\mathrm{Lk}(a)$, then β being a face of $\alpha\beta$ must also belong to $\mathrm{Lk}(a)$ which contradicts $a\beta \in \mathrm{Blockers}_0(K)$. Similarly, if $\alpha\beta$ belongs to $\mathrm{Lk}(b)$, we also get a contradiction.

Conversely, let us prove that (i) and (ii) $\implies c\sigma \in \mathrm{Blockers}(K')$. We have seen that (i) implies that all proper faces of $c\sigma$ belong to K'. To prove that $c\sigma \notin K'$, we note that neither $a\sigma$ nor $b\sigma$ belongs to K. Indeed, $a\sigma = a\alpha\beta \notin K$ since its face $a\beta \in \mathrm{Blockers}_0(K)$ and $b\sigma = b\alpha\beta \notin K$ since its face $b\alpha \in \mathrm{Blockers}_0(K)$. □

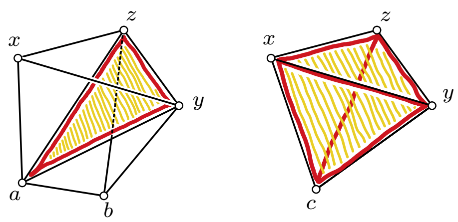

Fig. 6. Triangle ayz is a 2-blocker. We have that $\mathrm{Lk}(a) = \{x, y, z, xy, xz, b, by, bz\}$, $\mathrm{Lk}(b) = \{y, z, yz, a, ay, az\}$, $\mathrm{Lk}(ab) = \{y, z\}$. Note that $\sigma = \alpha\beta$ with $\alpha = x$ and $\beta = yz$ fulfills *(i)* and *(ii)* in Lemma 4. Therefore, the edge contraction $ab \mapsto c$ leads to the creation of the 3-blocker $cxyz$ and the destruction of the 2-blocker ayz.

A few remarks. Suppose $\sigma = \alpha\beta$ with $\dim \sigma \geq 1$ and $ab \cap \sigma = \emptyset$ satisfies (i) and (ii) in Lemma 4. Because order-0 blockers have dimension 1 or more, the two sets α and β are non-empty. Writing $d_i(v)$ for the largest dimension of order-i blockers through v, it follows directly from the lemma that the largest dimension of order-1 blockers through c satisfies $d_1(c) \leq d_0(a) + d_0(b)$ (see Figure 6 for an example in which equality is attained). Finally, we show that if α is a vertex, then $\alpha \in \mathcal{N}(a)$.

Suppose α is a vertex. Because $\dim\sigma \geq 1$, α is a proper face of σ. Thus, (i) implies $\alpha \in \mathrm{Lk}(a) \cup \mathrm{Lk}(b)$ and (ii) implies $\alpha \notin \mathrm{Lk}(b)$, yielding $\alpha \in \mathrm{Lk}(a)$.

We are now ready to derive an expression for the set of blockers through c. First, note that

$$\mathrm{Blockers}_0(K) \;=\; \mathrm{Blockers}_1(K) \cup \; \{xy \mid x \in K^{(0)}, y \in K^{(0)}, xy \notin K^{(1)}\}.$$

Hence, $b\alpha \in \mathrm{Blockers}_0(K)$ if and only if $b\alpha \in \mathrm{Blockers}_1(K)$ or $\alpha \in K^{(0)} \setminus \mathcal{N}(b)$. Thus, to exhaust simplices $\sigma = \alpha\beta$ with $\dim\sigma \geq 1$, $ab \cap \sigma = \emptyset$ that satisfy (i) and (ii), it suffices to take α in the set of simplices

$$Z_a(b) \;=\; \{\alpha \mid b\alpha \in \mathcal{B}(b),\, ab \cap \alpha = \emptyset\} \;\cup\; (\mathcal{N}(a) \setminus (\mathcal{N}(b) \cup \{b\}))\,.$$

Switching a and b, we define $Z_b(a)$ similarly and obtain

$$\mathcal{B}_{K'}(c) = \{\, c\alpha\beta \mid \alpha \in Z_a(b),\, \beta \in Z_b(a),\, \alpha\beta \in K,\, \forall\tau \subsetneq \alpha\beta,\, \tau \in \mathrm{Lk}(a) \cup \mathrm{Lk}(b)\,\}.$$

From this formula, we derive immediately an algorithm for computing $\mathcal{B}_{K'}(c)$ whose pseudo-code is given in Algorithm 5 of Appendix A. The only piece that we still need to explain is how to compute the link of a vertex. This is done in the next section, where more generally we compute the link of a simplex.

Let $N_\sigma = \max_{v\in\sigma} \sharp\mathcal{N}(v)$. Overall, updating the data structure after the edge contraction $ab \mapsto c$ has a cost which increases with the number and dimension of blockers in a neighborhood of a and b and can be done efficiently in $O(\, N_a N_b \log N_{\mathcal{N}(a)}\,)$ assuming there are no blockers in a neighborhood of a and b (see Appendix A for the details).

4.5. *Computing the link of a simplex*

Since the link of a simplex $\alpha \in K$ is a simplicial complex, we can also represent it by a pair consisting of its 1-skeleton and its order-1 blocker set. We give below formulas expressing each element in the pair. Let $\mathcal{N}(\alpha) = \bigcap_{u\in\alpha} \mathcal{N}(u)$.

Lemma 5. *For every simplex α in the simplicial complex K, we have:*

$$\mathrm{Lk}(\alpha)^{(1)} = \{\sigma \subset \mathcal{N}(\alpha) \mid \dim\sigma \leq 1,\, \alpha\sigma \in K\},$$
$$\mathrm{Blockers}_1(\mathrm{Lk}(\alpha)) = \{\sigma \subset \mathcal{N}(\alpha) \mid \dim\sigma \geq 2,\, \alpha\sigma \notin K,\;\, \forall\sigma' \subsetneq \sigma,\, \alpha\sigma' \in K\}.$$

Proof. By definition, σ is a vertex or an edge of the link of α if and only if $\dim\sigma \leq 1$, $\alpha \cap \sigma = \emptyset$ and $\alpha\sigma \in K$, yielding the first formula.

By definition, σ is a blocker of the link of α if and only if $\dim\sigma \geq 2$, $\sigma \notin \mathrm{Lk}(\alpha)$ and for all proper faces $\emptyset \neq \sigma' \subsetneq \sigma$, we have that $\sigma' \in \mathrm{Lk}(\alpha)$, yielding the second formula. □

Below, we give a characterization of the blockers in the link of a simplex from which we derive an algorithm for computing the link; See Algorithm 3 in Appendix A for the pseudo-code and a discussion of its time complexity.

Lemma 6. *Let $\alpha \in K$. The simplex σ is a blocker of the link of α if and only if $\sigma \subset \mathrm{Vert}(\mathrm{Lk}(\alpha))$, $\dim \sigma \geq 2$, there exists $\beta \in \mathrm{Blockers}_1(K)$ such that $\sigma = \beta \setminus \alpha$ and there exists no blocker of K of the form $\alpha'\sigma'$ with $\emptyset \neq \alpha' \subset \alpha$ and $\sigma' \subsetneq \sigma$.*

Proof. Suppose σ is a blocker in the link of α. By Lemma 5, we have that $\dim \sigma \geq 2$, $\sigma \subset \mathcal{N}(\alpha)$, $\alpha\sigma \notin K$ and $\forall \sigma' \subsetneq \sigma$, $\alpha\sigma' \in K$. In particular, it is easy to check that all vertices and all edges of σ belong to K and $\alpha \cap \sigma = \emptyset$. Therefore, the condition $\alpha\sigma \notin K$ implies that there exists a blocker β of K such that $\beta \subset \alpha\sigma$. The condition $\forall \sigma' \subsetneq \sigma$, $\alpha\sigma' \in K$ implies that the blocker β contains σ and therefore $\sigma = \beta \setminus \alpha$. Furthermore, no blocker of K has the form $\alpha'\sigma'$ with $\alpha' \subset \alpha$ and $\sigma' \subsetneq \sigma$.

Conversely, consider $\sigma = \beta \setminus \alpha \subset \mathrm{Vert}(\mathrm{Lk}(\alpha))$ such that $\dim \sigma \geq 2$, β is a blocker of K and there exists no blocker $\alpha'\sigma'$ of K with $\emptyset \neq \alpha' \subset \alpha$ and $\sigma' \subsetneq \sigma$. Since $\beta \subset \alpha\sigma$, this means that $\alpha\sigma$ is a coface of the blocker β and therefore does not belong to K. On the other hand, for all $\emptyset \neq \sigma' \subsetneq \sigma$, the simplex σ' is a proper face of the blocker β and therefore belongs to K. One can check that for all $\sigma' \subsetneq \sigma$, the simplex $\alpha\sigma'$ belongs to K since all its edges belong to K and none of its faces are blockers of K. By Lemma 5, $\sigma \in \mathrm{Blockers}_1(\mathrm{Lk}(\alpha))$. □

As an immediate corollary, we get that if K is a flag complex, so is the link of any of its vertices. Furthermore, for every vertex $x \in \mathrm{Lk}(v)$, we have $\sharp\mathcal{N}_{\mathrm{Lk}(v)}(x) \leq \sharp\mathcal{N}_K(x)$ and $\sharp\mathcal{B}_{\mathrm{Lk}(v)}(x) \leq \sharp\mathcal{B}_K(x)$. The pseudo-code for computing the link is given in Algorithm 3 of Appendix A.

4.6. *Poppable blockers*

The smaller the number of blockers the more efficient our data structure will be. Given a simplicial complex K represented by the pair (S, B), we are interested in the operation that removes a blocker σ from the blocker set B, while leaving intact the 1-skeleton S. The result is a complex K' represented by the pair $(S, B \setminus \{\sigma\})$. By definition of a blocker, all faces of σ but σ itself belong to K. Thus, removing σ from the blocker set of K has the effect of adding it to the complex, possibly with some cofaces. We get back K from K' by removing precisely the star of σ in K', that is $K = K' \setminus \mathrm{St}_{K'}(\sigma)$. As we have seen earlier, if there is a unique inclusion-maximal simplex $\eta \neq \sigma$ in the star of σ, then $|K'|$ deformation retracts to $|K|$ and the operation that removes σ from the blocker set of K is then called an *anti-collapse*. More generally, it has been established in Ref. 4 that if $\mathrm{Lk}_{K'}(\sigma)$ is a cone, then we can go from K' to K by a sequence of collapses. We recall that a simplicial complex L is said to be a *cone* if it contains a vertex o such that the following implication holds: $\sigma \in L \implies \sigma \cup \{o\} \in L$. The vertex o is called the *apex* of the cone. We will see shortly that checking whether a simplicial complex is a cone is not too complicated. This motivates the following definition:

Definition 3. Let K be a simplicial complex. The blocker σ of K is *poppable* if the link of σ in the simplicial complex K' encoded by the pair $(K^{(1)}, \mathrm{Blockers}_1(K) \setminus \{\sigma\})$

is a cone. The operation that goes from K to K' is called an *extended anti-collapse.*

Removing poppable blockers from the data structure does not change the homotopy type. Checking whether a blocker σ is poppable can be done efficiently by first computing the link L of σ in the complex $(K^{(1)}, \text{Blockers}_1(K) \setminus \{\sigma\})$ as explained in the previous section and then checking whether L is a cone. The following lemma leads to a simple algorithm for testing whether L is a cone; see Algorithm 2 in Appendix A for the pseudo-code.

Lemma 7. *A simplicial complex L is a cone with apex o if and only if $\mathcal{N}_L(o) = \text{Vert}(L) \setminus \{o\}$ and $\mathcal{B}_L(o) = \emptyset$.*

Proof. Suppose L is a cone with apex o. Clearly, every vertex $v \neq o$ of L is connected to o by an edge or equivalently $\mathcal{N}_L(o) = \text{Vert}(L) \setminus \{o\}$. Suppose for a contradiction that $o\tau$ is a blocker of L for some simplex τ such that $o \cap \tau = \emptyset$. By definition of a blocker, $o\tau \notin L$ and all proper faces of $o\tau$ belong to L. In particular $\tau \in L$ and by definition of a cone $o\tau \in L$, yielding a contradiction. Conversely, suppose $\mathcal{B}_L(o) = \emptyset$ and $\mathcal{N}_L(o) = \text{Vert}(L) \setminus \{o\}$. Let $\tau \in L$. It is easy to see that $o\tau \in L$ since all edges of $o\tau$ belong to L and $o\tau$ contains no blocker. □

5. Experiments

In this section we apply our representation to the simplification of a subfamily of flag complexes, namely Rips complexes and present the results of various computational experiments we performed.

5.1. *Rips complexes*

Given as input a point cloud P in a metric space and a real number $r \geq 0$, the *proximity graph* $G_r(P)$ is the graph whose vertices are the points P and whose edges connect all pairs of points within distance $2r$. By definition, the *Rips complex* is the flag complex of the proximity graph, $\mathcal{R}(P, r) = \text{Flag}(G_r(P))$. In our experiments, we consider finite point sets P that sample various d-dimensional manifolds X embedded in the D-dimensional Euclidean space. Typically, $d \in \{1, 2, 3\}$. In our experiments, $D \in \{3, 4, 9, 128^2\}$. Rips complexes are built using the extrinsic distance of the embedding space. In the remainder of this section, we establish an upper bound on the initial size of our data structure and compare it with a brute-force approach that stores all simplices in the Rips complex. For this, we suppose the sampling is neither too sparse nor too dense. Precisely, we suppose that every ball centered at X with radius ε contains at least one point of P and at most κ points of P for some positive number $\varepsilon \leq r$ and some constant $\kappa > 0$. In particular, $d_H(P, X) \leq \varepsilon$. We express our upper and lower bounds using respectively covering and packing numbers whose definitions can be found in Ref. 6 and are recalled below.

Write $B(x,r)$ for the closed ball with center x and radius r and let $U \subset \mathbb{R}^D$. An *ε-cover* of U is a set $Y \subset U$ with the property that the union of balls centered at Y with radius ε contains U. An *ε-packing* of U is a set $Y \subset U$ with the property that the balls centered at Y with radius ε are pairwise disjoint. The *covering number* $\mathcal{C}(x,r,\varepsilon)$ is the size of the smallest ε-cover of $X \cap B(x,r)$ and the *packing number* $\mathcal{P}(x,r,\varepsilon)$ is the size of the largest ε-packing of $X \cap B(x,r)$. We let $\mathcal{C}(r,\varepsilon) = \max_{x\in X} \mathcal{C}(x,r,\varepsilon)$ and $\mathcal{P}(r,\varepsilon) = \min_{x\in X} \mathcal{P}(x,r,\varepsilon)$. If X is a d-dimensional manifold with a positive reach $\rho > 0$ then the covering number and the packing number are both $\Theta(r^d/\varepsilon^d)$ as $\varepsilon \to 0$ and $r < \rho$. The constant hidden in Θ depends only upon the intrinsic dimension d and does not depend upon the ambient dimension D.

Let us bound the size of our data structure. The number of neighbors of v is upper bounded by

$$N_{\max} = \max_{v\in \mathrm{Vert}(\mathcal{R}(P,r))} \sharp\mathcal{N}(v).$$

Since the Rips complex contains no blocker, the size of our data structure initially is a constant times $\sum_{v\in\mathrm{Vert}(\mathcal{R}(P,r))}(1+\sharp\mathcal{N}(v)) \le n(1+N_{\max})$. Note that the neighbors of a vertex v in the Rips complex $\mathcal{R}(P,r)$ are the points of $P\setminus\{v\}$ in the ball $B(v,2r)$. This ball can be covered with $\mathcal{C}(v,2r,\varepsilon)$ balls of radius ε centered at $X \cap B(v,2r)$, each of them containing at most κ points of P. It follows that $\sharp\mathcal{N}(v) \le \kappa\mathcal{C}(v,2r,\varepsilon)$ and $N_{\max} \le \kappa\mathcal{C}(2r,\varepsilon)$. Hence, the size of our data structure is bounded by n times a quantity that depends only upon the ratio r/ε, the constant κ and the intrinsic dimension d and not upon the ambient dimension D.

Let us now compare our representation with the one that consists in storing all simplices. We first give an upper and lower bound on the number of k-simplices. Consider a k-simplex σ which has v as a vertex. If $\sigma \in \mathcal{R}(P,r)$ then its vertex set is contained in $B(v,2r)$. On the other hand, if the vertex set of σ is contained in $B(v,r)$ then $\sigma \in \mathcal{R}(P,r)$. Put another way, the set of k-simplices of $\mathcal{R}(P,r)$ with v as a vertex is a subset of the set of k-simplices obtained by picking v and k distinct neighbors of v in $\mathcal{R}(P,r)$. On the other hand, we get a k-simplex of $\mathcal{R}(P,r)$ by choosing v and k distinct neighbors of v in $\mathcal{R}(P,r/2)$. Let

$$N'_{\min} = \min_{v\in \mathrm{Vert}(\mathcal{R}(P,r/2))} \sharp\mathcal{N}(v).$$

We deduce the following upper and lower bounds on the number of k-simplices:

$$\frac{n}{k+1}\binom{N'_{\min}}{k} \quad\le\quad \sharp\{\sigma\in\mathcal{R}(P,r) \mid \dim\sigma = k\} \quad\le\quad \frac{n}{k+1}\binom{N_{\max}}{k}.$$

Summing over all dimensions, we get that $\frac{n}{N'_{\min}+1}2^{N'_{\min}} \le \sharp\mathcal{R}(P,r) \le n2^{N_{\max}}$. We saw above that $N_{\max} \le \kappa\mathcal{C}(2r,\varepsilon)$. Let us give a lower bound on $N'_{\min}$. Recall that the neighbors of a vertex v in the Rips complex $\mathcal{R}(P,r/2)$ are the points of $P\setminus\{v\}$ in the ball $B(v,r)$. This ball can be packed with $\mathcal{P}(v,r-\varepsilon,\varepsilon)$ pairwise disjoint balls of radius ε centered at $X \cap B(v,r-\varepsilon)$, each of them containing at least one point of P. It follows that $N'_{\min} \ge \mathcal{P}(r-\varepsilon,\varepsilon)-1$. To conclude, if N represents the typical number of neighbors in the proximity graph, a brute-force approach would have to enumerate something in the order of magnitude $\frac{n}{N+1}2^N$, which is prohibitive for realistic Rips complexes. This should be compared to the size of our data structure which initially is roughly equal to $O(nN)$. In Figure 7, we plotted the number of k-simplices in the Rips complex of a point set P that samples the boundary of a surface in $\mathbb{R}^3$. Already in this small example, the number of simplices is gigantic and so is the size of a brute-force representation.

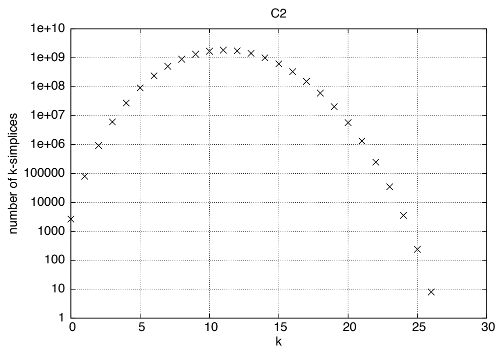

Fig. 7. Number of k-simplices in $\mathcal{R}(P,r)$ as a function of k for a point set P that samples C_2. We used a logarithmic scale for the y-axis. Experimentally, we observe that the number of k-simplices is above but close to $\frac{n}{k+1}\binom{N}{k}$ for $N = 25$. The number of simplices is thus at least $\frac{n}{N+1}2^N \approx 3.4\ 10^9$. This should be compared with our data structure which has only to store 2646 vertices and 80304 edges to represent the same complex.

5.2. *Datasets*

Let us introduce the datasets we use to present our experiment findings. Each dataset is a set of points P that samples a d-dimensional manifold X embedded in $\mathbb{R}^D$. For each dataset, we try to find a scale parameter r, so that $|\mathcal{R}(P,r)| \simeq X$. In this we are helped by Ref. 3 which describes conditions guaranteeing that the Rips complex $\mathcal{R}(P,r)$ recovers the homotopy type of X. Table 1 gives for each point set P, its size, the dimension d of the sampled manifold, the ambient dimension D, the smallest degree $N_{\min}$ in $\mathcal{R}(P,r)$, and the largest degree $N_{\max}$ in $\mathcal{R}(P,r)$. The first group in the table contains synthetic datasets and the second group contains real datasets.

5.2.1. *Synthetic data*

We consider three scenarios for generating P. First, we sample the boundary of a $(d+1)$-dimensional cube, $C_d = \partial[-1,1]^{d+1}$ embedded in $\mathbb{R}^{d+1}$. Specifically, we sample each of the $2d+2$ facets of C_d using a regular grid of k^d points. Applying this sampling method for $k = 21$ and $d = 2$, we get a first point set $\mathtt{C}_2$ whose size is

Table 1. The seven point sets considered in our experiments.

Dataset	♯points	d	D	N_{min}	N_{max}
$\texttt{C}_2$	2646	2	3	50	74
$\texttt{S}_2$	2646	2	3	64	151
$\texttt{C}_3$	46656	3	4	91	153
$\texttt{S}_3$	46656	3	4	38	333
$\texttt{SO}_3$	50000	3	9	33	55
Ramses	193252	2	3	6	38
Lucky_cat	72	1	128^2	7	44

$6 \times 21^2 = 2646$. Applying again this method for $k = 18$ and $d = 3$, we get a second point set $\texttt{C}_3$ whose size is $8 \times 18^3 = 46656$. For $d \in \{2, 3\}$, we then normalize $\texttt{C}_\texttt{d}$ and get a point set $\texttt{S}_\texttt{d}$ which samples the d-sphere $S_d = \{x \in \mathbb{R}^{d+1} \mid \|x\| = 1\}$. Finally, we sample the special orthogonal group SO_3 using the method described in Ref. 21 and get a point set $\texttt{SO}_3 \subset \mathbb{R}^9$ whose size is 50000. We recall that the special orthogonal group SO_3 is diffeomorphic to the real projective space $\mathbb{RP}^3$ and can be embedded in $\mathbb{R}^9$ by representing each rotation in 3D by a 3×3 matrix.

5.2.2. *Real data*

Finally, we consider two real datasets. The first one, referred to as Ramses, is a 3D scan data consisting of points measured on the surface of a statue representing Ramses II. The surface of the statue is homeomorphic to S_2 and its scan is available in the Aim@Shape repository. The second dataset, referred to as Lucky_cat, is a collection of 72 images of a toy cat placed on a turntable and observed by a fixed camera.[15] Images of the toy are taken at pose interval of 5 degrees. Each image has size $128^2 = 16384$. Since the degree of freedom of the acquisition system is 1, the collection of images can be interpreted as a point cloud that samples a curve in $\mathbb{R}^{16384}$. Some of the images in the dataset Lucky_cat are shown in Figure 8. The Rips complex of Lucky_cat we use in our experiments is presented in Figure 9.

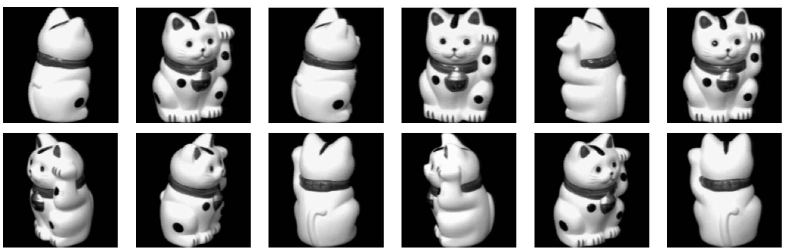

Fig. 8. Twelve of the 72 points in the dataset Lucky_cat. Each point represents a 128 pixel by 128 pixel image of a toy cat taken during its rotation around a fixed axis.

5.3. *Simplification*

Once the Rips complex of P is built, we simplify it by iteratively contracting edges. After each edge contraction, we delete poppable blockers. Initially, all edges are stored in a priority queue $\mathcal{Q}$. We use the length of the edges to prioritize them, so that the shortest edge has highest priority. We then remove the edge ab with highest priority from the priority queue. If ab satisfies the link condition, we contract ab to a new vertex $c = \frac{a+b}{2}$ and update the data structure, which includes the removal of edges from $\mathcal{Q}$ and the insertion of new edges into $\mathcal{Q}$. We also remove poppable blockers during that step. We let the process continue until no edges remain in $\mathcal{Q}$. Each edge contraction decreases the number of vertices by one. We call K_i the simplicial complex obtained after i edge contractions and set $n_i = \sharp\,\mathrm{Vert}(K_i) = \sharp\,\mathrm{Vert}(K_0) - i$.

5.4. *Results and discussion*

For each point set $P \in \{\mathtt{C}_2, \mathtt{S}_2, \mathtt{C}_3, \mathtt{S}_3, \mathtt{SO}_3, \mathtt{Ramses}\}$, we plot with respect to the number i of edge contractions the number of blockers, the cumulated number of poppable blockers that have been removed at step i and the average number of neighbors per vertex; see Figure 10. For `Lucky_cat`, the simplicial complex obtained after simplification is drawn in Figure 9.

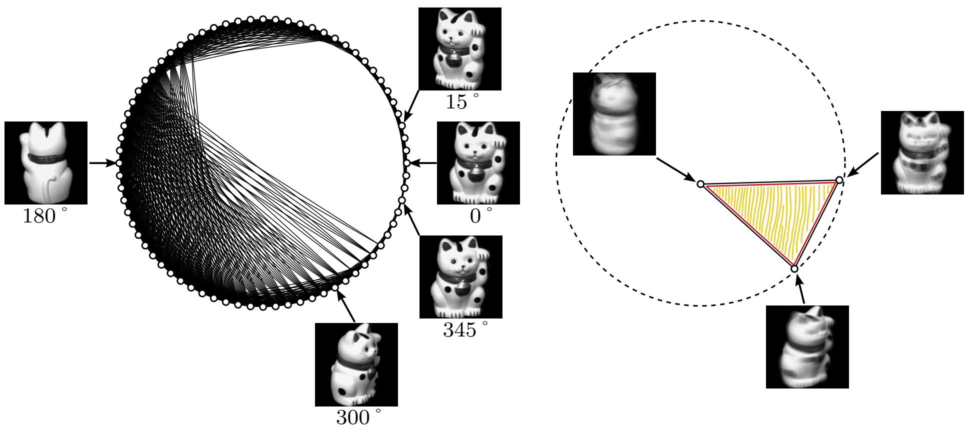

Fig. 9. Left: Rips complex of the point set `Lucky_cat`. Right: After simplification, we obtain a simplicial complex which consists of three vertices, three edges and has a unique blocker.

In our experiments, we observe that the total amount of blockers present in the data structure remains notably small at all times. For `Lucky_cat`, none other than the penultimate edge contraction produces a blocker. For $\mathtt{C}_2$ and $\mathtt{S}_2$, all blockers produced before the third from the last step are poppable. It follows that for `Lucky_cat`, $\mathtt{C}_2$, and $\mathtt{S}_2$, the data structure contains no blockers almost until the end of the simplification. This should be compared with results in Ref. 2 in which we were not removing poppable blockers. This slight modification in the simplification process dramatically decreases the number of blockers in the data structure and also accelerates computations since the cost of an edge contraction increases with the number and dimension of blockers around that edge.

Except for $\mathtt{SO}_3$, the complex remaining at the end of the simplification process possesses a unique blocker whose boundary is precisely the simplified complex. The dimension of this unique remaining blocker is $d+1$, one more than the dimension d of the sampled surface. Results are thus consistent with the topology of the sampled shape which is a topological d-sphere for all our datasets but $\mathtt{SO}_3$. The complex we obtain after simplifying the Rips complex of $\mathtt{SO}_3$ possesses 12 vertices and has dimension 4. We computed its homology groups and checked that they coincide with the homology groups of $\mathbb{RP}^3$. Interestingly, this complex is close to the smallest triangulation of SO_3 which contains 11 vertices.[20]

We also observe that, in all our experiments, the size of our data structure decreases during the simplification. This can be explained by the fact that every edge contraction decreases the number of vertices and edges and that the additional cost of storing blockers remains negligible compared to the cost of storing the 1-skeleton of the complex.

These very first illustrations of our data structure and simplification procedure are quite promising. Indeed, in these preliminary experiments we have only tested one of the simplest criteria for ordering edge contractions, namely the edge length, and restricted ourselves to a strict application of edge contractions. In fact, we believe that together with Ref. 3 this theoretical work lays theoretical foundations and opens a new field of design and experimentation of simplification strategies or computation of topological invariants in our representation. In future work, we plan to revisit in this context usual simplification operations including, beyond edge contraction, point cloud filtering, simplex collapse and anti-collapse.

Acknowledgements

Authors have been supported in part by ANR Project *GIGA* ANR-09-BLAN-0331-01.

References

1. D. Attali and A. Lieutier. Reconstructing shapes with guarantees by unions of convex sets. In *26th Ann. Sympos. Comput. Geom.*, pages 344–353, Snowbird, Utah, June 13-16 2010.
2. D. Attali, A. Lieutier, and D. Salinas. Efficient Data Structure for Representing and Simplifying Simplicial Complexes in High Dimension. In *Proc. 27th Ann. Sympos. Comput. Geom.*, pages 501–509, Paris, France, June 13-15 2011.
3. D. Attali, A. Lieutier, and D. Salinas. Vietoris-Rips complexes also provide topologically correct reconstructions of sampled shapes. In *Proc. 27th Ann. Sympos. Comput. Geom.*, pages 491–500, Paris, France, June 13-15 2011.

4. D. Attali, A. Lieutier, and D. Salinas. Vietoris-Rips complexes also provide topologically correct reconstructions of sampled shapes. *Computational Geometry: Theory and Applications (CGTA)*, 2011.
5. A. Bjorner. Topological methods. In *Handbook of combinatorics (vol. 2)*, page 1850. MIT Press, 1996.
6. K. Clarkson. Nearest-neighbor searching and metric space dimensions. *Nearest-Neighbor Methods for Learning and Vision: Theory and Practice*, pages 15–59, 2006.
7. T. Dey, H. Edelsbrunner, S. Guha, and D. Nekhayev. Topology preserving edge contraction. *Publ. Inst. Math. (Beograd) (N.S.)*, 66:23–45, 1999.
8. T. K. Dey, H. Edelsbrunner, and S. Guha. Computational topology. In B. Chazelle, J. E. Goodman, and R. Pollack, editors, *Advances in Discrete and Computational Geometry*, volume 223 of *Contemporary Mathematics*. AMS, Providence, 1999.
9. H. Edelsbrunner and J. Harer. *Computational topology: an introduction*. Amer Mathematical Society, Providence, Rhode Island, 2010.
10. M. Garland and P. Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 209–216. ACM Press/Addison-Wesley Publishing Co., 1997.
11. T. Gieng, B. Hamann, K. Joy, G. Schussman, and I. Trotts. Constructing hierarchies for triangle meshes. *IEEE Transactions on Visualization and Computer Graphics*, 4(2):145–161, 2002.
12. H. Hoppe. Progressive meshes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 99–108. ACM, 1996.
13. J. Leray. Sur la forme des espaces topologiques et sur les points fixes des représentations. *J. Math. Pures Appl.*, 24:95–167, 1945.
14. J. Munkres. *Elements of algebraic topology*. Perseus Books, 1993.
15. S. A. Nene, S. K. Nayar, and H. Murase. Columbia Object Image Library (COIL-100). Technical Report No. CUCS-006-96., Department of Computer Science, Columbia University, New York, 1996.
16. P. Perry and V. de Silva. Plex: Simplicial complexes in MATLAB, 2006.
17. J. Rossignac and P. Borrell. Multi-resolution 3D approximations for rendering complex scenes. In B. Falcidieno and T. Kunii, editors, *Modeling in Computer Graphics: Methods and Applications*, pages 455–465. Springer-Verlag, 1993.
18. W. J. Schroeder, J. A. Zarge, and W. E. Lorensen. Decimation of triangle meshes. *Computer Graphics (SIGGRAPH '92 proceedings)*, 26(2):65–70, 1992.
19. D. M. Thomas, V. Natarajan, and G.-P. Bonneau. Link conditions for simplifying meshes with embedded structures. *IEEE Transactions on Visualization and Computer Graphics*, 99, 2010.
20. D. Walkup. The lower bound conjecture for 3-and 4-manifolds. *Acta Mathematica*, 125(1):75–107, 1970.
21. A. Yershova, S. Jain, S. LaValle, and J. Mitchell. Generating Uniform Incremental Grids on $SO(3)$ using the Hopf Fibration. *The International journal of robotics research*, 29(7):801–812, 2010.
22. A. Zomorodian. The tidy set: a minimal simplicial set for computing homology of clique complexes. In *Proceedings of ACM symposium of computational geometry*, 2010.

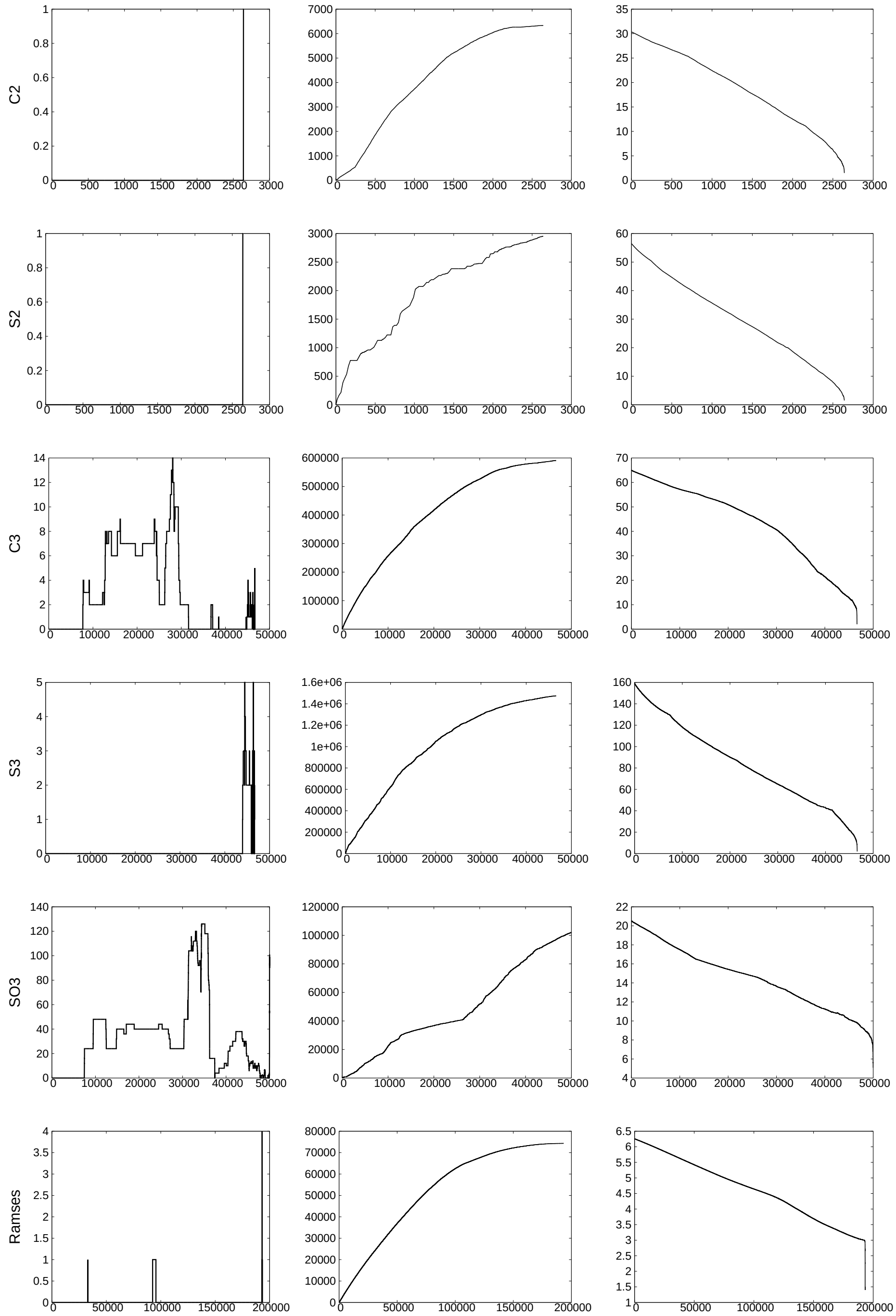

Fig. 10. We plotted as functions of i the number of blockers in the simplicial complex K_i on the left, the cumulated number of poppable blockers removed from the data structure up to step i in the middle and the average number of neighbors per vertex on the right. From top to bottom: the initial Rips complex has vertex set $\mathtt{C}_2$, $\mathtt{S}_2$, $\mathtt{C}_3$, $\mathtt{S}_3$, $\mathtt{SO}_3$, and `Ramses`.

Appendix A. Pseudo-codes and running times

We provide pseudo-codes for testing the link condition, computing the link of a simplex α, testing whether a vertex belongs to a complex, testing whether a complex is a cone and computing the blockers passing through c after the edge contraction $ab \mapsto c$. To express time complexities, let $N_\sigma = \max_{v\in\sigma} \sharp\mathcal{N}(v)$, $B_\sigma = \max_{v\in\sigma} \sharp\mathcal{B}(v)$ and $Q_\sigma = \max_{v\in\sigma} \max_{\beta\in\mathcal{B}(v)} \sharp\beta$. Moreover, let A_σ be the number of blockers contained in $\sigma \cup \mathcal{N}(\sigma)$. We also suppose that we are able to answer whether $v \in \sigma$ in $O(\log \sharp\sigma)$ time and whether $\tau \subset \sigma$ in $O(\sharp\sigma \log \sharp\sigma)$ time. Furthermore, we suppose that removing a vertex from σ takes $O(\log \sharp\sigma)$ time.

Algorithm 1 Return true if and only if ab satisfies the link condition $\mathrm{Lk}(ab) = \mathrm{Lk}(a) \cap \mathrm{Lk}(b)$.

```
for all σ ∈ B(a) do
  if b ∈ σ then return false end if
end for
return true
```

Algorithm 1 has time complexity $O(B_a \log Q_a)$ if the set of blockers $\mathcal{B}(a) \neq \emptyset$ and $O(1)$ otherwise. Algorithm 2 has time complexity $O(\sharp\,\mathrm{Vert}(L))$.

Algorithm 2 Return true if and only if the complex L is a cone.

```
for all x ∈ Vert(L) do
  if B_L(x) = ∅ then
    if ♯N_L(x) = ♯Vert(L) − 1 then
      return true
    end if
  end if
end for
return false
```

For computing the time complexity of Algorithm 3, we suppose α has constant size. Computing the vertices in the link costs $O\left(\, N_\alpha \log N_\alpha + \sharp\mathcal{N}(\alpha) B_{\mathcal{N}(\alpha)}\,\right)$. Computing the edges costs $O\left(\, (\sharp V)^2 [\log N_V + B_V \log Q_V]\,\right)$. Computing the blocker set costs $O\left(\, (\sharp V) B_V [\log Q_V + (\sharp V)\log(\sharp V)] + A_\alpha B_\alpha (\sharp V)\log(\sharp V)\,\right)$. Summing up these three costs and using $V \subseteq \mathcal{N}(\alpha)$ and $\sharp\mathcal{N}(\alpha) \le N_\alpha$, we get that the time complexity for Algorithm 3 is $O(l(\alpha))$ with:

$$l(\alpha) = (N_\alpha)^2 \left[\log N_{\mathcal{N}(\alpha)} + B_{\mathcal{N}(\alpha)}(\log N_\alpha + \log Q_{\mathcal{N}(\alpha)})\right] + A_\alpha B_\alpha N_\alpha \log N_\alpha$$

For $L = K$, the time complexity of Algorithm 4 is $O(g(\sharp\sigma, \sigma))$ where $g(x, y) = x^2 (\log N_y + B_y \log x)$. If L is the link of a vertex $v \in K$, the time complexity

Algorithm 3 Compute the 1-skeleton (V, E) and the blockers B in the link of a simplex α.

$V \leftarrow \emptyset$
$\mathcal{N}(\alpha) \leftarrow \bigcap_{u\in\alpha} \mathcal{N}(u)$
for all $v \in \mathcal{N}(\alpha)$ **do**
  newvertex $\leftarrow$ **true**
  for all $\beta \in \mathcal{B}(v)$ **do**
    if $\beta \setminus \{v\} \subset \alpha$ **then** *newvertex* $\leftarrow$ **false**; **break**; **end if**
  end for
  if *newvertex* **then** $V \leftarrow V \cup \{v\}$ **end if**
end for
$E \leftarrow \emptyset$
for all $x \in V$ **do**
  for all $y \in V$ such that $x < y$ **and** $y \in \mathcal{N}(x)$ **do**
    newedge $\leftarrow$ **true**
    for all $\beta \in \mathcal{B}(x)$ such that $y \in \beta$ **do**
      if $\beta \setminus \{x, y\} \subset \alpha$ **then** *newedge* $\leftarrow$ **false**; **break**; **end if**
    end for
    if *newedge* **then** $E \leftarrow E \cup \{x, y\}$ **end if**
  end for
end for
$B \leftarrow \emptyset$
for all $x \in V$ **do**
  for all $\beta \in \mathcal{B}(x)$ **do**
    $\sigma \leftarrow \beta \setminus \alpha$
    if $\dim \sigma \geq 2$ **and** x first vertex of σ **and** $\sigma \subset V$ **then**
      newblocker $\leftarrow$ **true**
      for all $a \in \alpha$ **do**
        for all $\eta \in \mathcal{B}(a)$ such that $\eta \subset (\sigma \cup \alpha)$ **do**
          if $(\eta \setminus \alpha) \subsetneq \sigma$ **then** *newblocker* $\leftarrow$ **false**; **break**; **end if**
        end for
      end for
      if *newblocker* **then** $B \leftarrow B \cup \{\sigma\}$ **end if**
    end if
  end for
end for

can also bound by $O(g(\sharp\sigma, \sigma))$ because for each vertex x in the link of v, we have $\sharp\mathcal{N}_{\mathrm{Lk}(v)}(x) \leq \sharp\mathcal{N}_K(x)$ and $\sharp\mathcal{B}_{\mathrm{Lk}(v)}(x) \leq \sharp\mathcal{B}_K(x)$.

Recall that $d_i(v)$ designates the largest dimension of order-i blockers through v and let $d = d_1(c) \leq d_0(a) + d_0(b)$. Noting that the size of $Z_a(b)$ is upper bounded by $B_b + N_a$ and that $\alpha\beta \subset \mathrm{Lk}(a) \cup \mathrm{Lk}(b)$ has size d at most, we get that the time

Algorithm 4 Return true if and only if the simplex $\sigma \subset \mathrm{Vert}(L)$ belongs to the subcomplex $L \subset K$.

for all $v \in \sigma$ **do**
  for all $w \in \sigma$ such that $v < w$ **do**
    if $w \notin \mathcal{N}_L(v)$ **then return false end if**
  end for
end for
for all $v \in \sigma$ **do**
  for all $\tau \in \mathcal{B}_L(v)$ **do**
    if $\tau \subset \sigma$ **then return false end if**
  end for
end for
return true

Algorithm 5 Compute the blockers B passing through c after the edge contraction $ab \mapsto c$.

$B \leftarrow \emptyset$; $L_a \leftarrow \mathrm{Lk}(a)$; $L_b \leftarrow \mathrm{Lk}(b)$
for all $\alpha \in Z_a(b)$ **do**
  for all $\beta \in Z_b(a)$ **do**
    if $\alpha\beta \in K$ **then**
      newblocker $\leftarrow$ **true**
      for all $\tau \subsetneq \alpha\beta$ with dimension one less than $\alpha\beta$ **do**
        if $\tau \notin L_a$ **and** $\tau \notin L_b$ **then**
          newblocker $\leftarrow$ **false**; **break**
        end if
      end for
      if *newblocker* **then** $B \leftarrow B \cup \{c\alpha\beta\}$ **end if**
    end if
  end for
end for

complexity for Algorithm 5 is $O(\ l(a) + l(b) + (N_a + B_b)(N_b + B_a)[g(d, \mathcal{N}_a \cup \mathcal{N}_b) + (d+1)(\log d + g(d-1, \mathcal{N}_a) + g(d-1, \mathcal{N}_b)]\ )$.

We conclude the appendix by computing the complexities when there are no blockers through the vertices impacted by a local operation. The time complexity corresponding to Algorithm 1, answering if an edge ab in K meets the link condition is $O(1)$. The time complexity corresponding to Algorithm 3 which builds a representation of the link of a simplex α, is $O(\ N_\alpha^2 \log N_{\mathcal{N}(\alpha)}\ )$. For Algorithm 4 which tests whether a simplex σ belongs to a subcomplex of K, we get $O(\ (\sharp\sigma)^2 \log N_\sigma\ )$ and updating the set of blockers takes $O(\ N_a N_b \log N_{\mathcal{N}(a)}\ )$ (or $O(\ N_a N_b \log N_{\mathcal{N}(b)}\ )$). These complexities give a good picture of the practical behavior of the simplification process when the number of blockers remains sufficiently small.