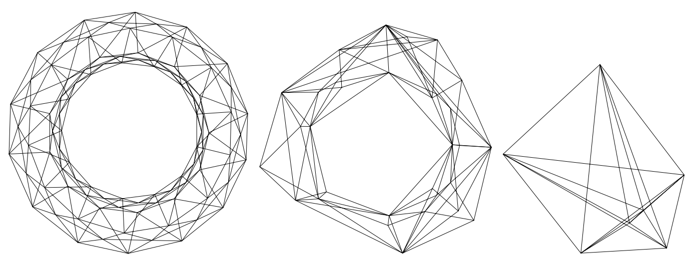
CONE STRATEGIES FOR FORMAL DEFORMATIONS OF NON-COLLAPSIBLE SIMPLICIAL COMPLEXES

JAN F. KAYATZ

Abstract. The simple homotopy equivalence class of a simplicial complex consists of complexes resulting from subsequent formal deformations. We use anticollapses to build cones over collapsible subcomplexes; depending on the properties of the cone bases, we obtain a complex with very few simplices after the cones collapse again. We analyze several strategies to compute cones that lead to an almost minimal simple homotopy equivalent complex. After we explained our ideas in theory we discuss the computational possibilities of the cone-strategies.



Key words and phrases. simple homotopy theory, collapse, anticollapse, expansion, simplicial complex, cone, formal deformation, combinatorial topology, topological combinatorics.

Contents

List of Figures

Introduction

Simplicial complexes play an important role in computer science, discrete mathematics and topology. But usually the necessary amount of data to describe them is immense and their geometric, topological and homological properties can only be analyzed by sufficient computing power and time. Therefore we want to modify a simplicial complex, reducing the number of simplices but not changing the underlying topological space.

Common approaches for such a reduction strategy include edge contractions (see for instance [2] or any book on graph theory) (usually applied on graphs) and collapses (refer to [3] for definitions). The latter method (collapses) has been analyzed in many aspects (see [8],[9] or [15]); collapses are easy to implement on a computer and at the first glance they seem to be able to simplify most simplicial complexes. Yet the main problem is to find an optimal (or at least good) sequence of collapses. Random sequences can lead to arbitrarily bad results. Such a good sequence can only be computed in special cases when one deals with a p.l. manifold and additional geometrical information is at hand: [15]. Another severe problem is the existence of simplicial complexes which are topologically rather trivial but collapses alone are unable to simplify their discretization, such as the famous dunce cap (refer to [23]).

J.H.C. Whitehead provided the necessary insights to simple homotopy theory (his work on the subject is collected in [22]) and was able to give information, when a homotopy equivalence can be realized by collapses and their inverse process. Therefore we tried to describe a strategy to simplify a simplicial complex by means of collapses and anticollapses.

Outline. We begin with the basic definitions and some easy properties of the defined objects. After we have the necessary language we state the main results in simple homotopy theory which gave rise to our idea, using anticollapses as a simplification strategy. Then we redefine the term anticollapse in order to make it algorithmically more accessible and finally come to cones, their augmentation by anticollapses and the balance of augmenting and collapsing cones.

We will then include a little detour about edge contractions and other combinatorial methods and talk about how these constructions can be realized with collapses and anticollapses.

To find out, whether there are better or worse cones we defined several cone augmenting strategies such as the “maximal cone” strategy, the

"pure cone"- and "cone over p.l. manifold" strategy and some others.
In the sections about the implementation we start by describing the used data structure and then analyze, subroutine by subroutine, the complexity of our algorithm. Afterwards we provide a different point of view on the analysis and can improve our estimates a little.
The main problems and trade-offs in our implementation will then be analyzed and improved where possible.
At last we give some examples how the algorithm behaves in practice.

Results. Our analysis shows a runtime complexity between $O(N^4)$ and $O(N)$ where N is the number of simplices needed to describe a simplicial complex. We observed that the complexity of the algorithm depends on the given simplicial complex (its dimension, genus and triangulation) and the used cone strategy but is usually close to linear. The most powerful strategy seems to be the "small cone" strategy. Cones with a larger base manage to decrease the number of simplices only in certain situations.
The algorithm is able to contract the house with two rooms [1] and the dunce cap and reach the minimal triangulation for certain topological spaces such as the torus.

1. Simplicial Complexes and Posets

Definition 1.1 (Simplicial Complex)**.** Let A be a set. A collection Δ of subsets of A is called a *simplicial complex* if for every $M \in \Delta$ and every subset $K \subset M$ we have: $K \in \Delta$. The elements of Δ of cardinality $n+1$ are called *n-simplices*, or just simplices. By *vertex* we denote a 0-simplex, ie. the elements of A.

Definition 1.2 (Subcomplex)**.** A *subcomplex* of a simplicial complex Δ is a subset $\tilde{\Delta}$ which is itself a simplicial complex.

Definition 1.3 (Dimension)**.** We define the dimension of a simplex σ as $\dim(\sigma) = k - 1$ if the cardinality of σ equals k. The dimension of a simplicial complex Δ is the maximum of the dimensions of its simplices.

Remark. An *oriented* simplicial complex contains *ordered* subsets of A. We identify an ordered set with its image under *two* transpositions of its elements (that is, a simplex is the orbit of an ordered set under the group action of the alternating group, the group of even permutations). So every simplex has two orientations, denoted by $-\sigma$ and $+\sigma$.

Before we continue with further definitions about complexes we introduce the notion of categories and functors. This concept is quite usual and helpful in algebraic topology as it can be used to generalize several concepts such as tensor products, homology theory etc. We use functors for the geometric realization of simplicial complexes, the face poset construction, simple homotopy and also for homology.

Definition 1.4 (Category)**.** A category is a pair $(\mathscr{O}, \mathscr{M})$ of classes. $\mathscr{O}$ is the class of objects and $\mathscr{M}$ is the class of morphisms (or arrows), which contains the identity for every object and the composition of morphisms.

Example. Here are some examples of categories:

- **Sets** $= (S, M)$, the category of sets. The objects are all sets and the morphisms are set maps.
- **Grp** $= (\mathscr{G}, \mathscr{H})$, the category of groups. Objects are groups and morphisms are group homomorphisms.
- **Top** $= (T, C^0)$, the category of topological spaces. Morphisms are continuous maps.
- **Posets** $= (P, M_o)$, the category of partially ordered sets. The morphisms are order preserving set maps.
- **Simp** $= (D, C)$, the category of simplicial complexes. The objects are simplicial complexes and the morphisms are cellular maps, ie. maps which take simplices to simplices.

Definition 1.5 (Functor)**.** A functor is a map from a category to a category which maps objects to objects, morphism to morphism and preserves the identity and compositions. That is, a map

$$\mathscr{F} : (\mathscr{O}_1, \mathscr{M}_1) \to (\mathscr{O}_2, \mathscr{M}_2)$$

is a functor, if for objects $X, Y, Z \in \mathscr{O}_1$ and morphisms $g : X \to Y$ and $h : Y \to Z$, we have

- $\mathscr{F}(id_X) = id_{\mathscr{F}(X)}$
- $\mathscr{F}(h \circ g) = \mathscr{F}(h) \circ \mathscr{F}(g)$

A functor is called *covariant* if it preserves the direction of arrows and *contravariant* if it flips arrows.

Example. Here are some examples of functors

- The trivial functor $F : \mathbf{C} \to \mathbf{C}$ for a category $\mathbf{C}$.
- The dual vector space functor $C : \mathbf{Vect}_{\mathbb{R}} \to \mathbf{Vect}_{\mathbb{R}}$ is a contravariant functor which sends a vector space to its dual space and a linear map to its dual.
- The homology $H_\star : \mathbf{Top} \to \mathbf{Grp}$ is a covariant functor and the cohomology $H^\star : \mathbf{Top} \to \mathbf{Grp}$ is a contravariant functor. Refer to [16] for an introduction to homology theory.

Further examples can be found throughout this article. We return now to simplicial complexes.

Definition 1.6 (Boundary Maps)**.** Let K be a simplicial complex, $K^n \subset K$ the set of all n-simplices. We define the i^{th} boundary (or face-) map as

$$\begin{gathered} \partial_i^n : K^n \longrightarrow K^{n-1} \\ \sigma = \{v_0, \dots, v_n\} \mapsto \partial_i^n(\sigma) = \{v_0, \dots, v_{i-1}, v_{i+1}, \dots, v_n\} \end{gathered}$$

Definition 1.7 (k-Skeleta)**.** Let $k \in \mathbb{N}$ be a number and Δ a simplicial complex. We define the k-skeleton of Δ as the set of all j-simplices with $j \leq k$.

Definition 1.8 (Geometric Realization of a Simplicial Complex [19])**.** The geometric realization of a simplicial complex is a covariant functor

$$|\cdot| : \mathbf{Simp} \to \mathbf{Top}$$

taking a simplicial complex to a topological space and a cellular map to a continuous map. Formally:

$$|K| = \coprod_{n \in \mathbb{N}} K^n \times \mathcal{S}^n / \sim$$

where K^n is defined as in 1.6 and $\mathcal{S}^n$ is the standard unit simplex in $\mathbb{R}^{n+1}$:

$$\mathcal{S}^n := \Big\{(x_0, \dots, x_n) \in \mathbb{R}^{n+1} \Big| x_i \geq 0 \text{ and } \sum_{j=0}^{n} x_j = 1\Big\}$$

and $\sim$ is the glueing relation:
Let $\varepsilon_i^n : \mathcal{S}^n \hookrightarrow \mathcal{S}^{n+1}$ be the inclusions of the faces of the unit simplex. Then $\sim$ is generated by

$$K^{n-1} \times \mathcal{S}^{n-1} \ni \big(\partial_i^n(\sigma), \overline{x}\big) \sim \big(\sigma, \varepsilon_i^{n-1}(\overline{x})\big) \in K^n \times \mathcal{S}^n$$

for an n-simplex σ and $\overline{x} \in \mathcal{S}^{n-1}$.

Remark. We can embed a finite simplicial complex over k vertices into $\mathbb{R}^k$;

$$\begin{array}{rcl}
\Psi : \Delta & \longrightarrow & \mathbb{R}^k \\
v_i \in K_0 & \mapsto & (\underbrace{0, \dots, 0}_{i \text{ times}}, 1, 0, \dots, 0) \\
e \in K_1 & \mapsto & \big\{\lambda \cdot \Psi(\partial_0^1(e)) + (1-\lambda) \cdot \Psi(\partial_1^1(e)) \big| \lambda \in [0,1]\big\} \\
 & \vdots &
\end{array}$$

that is, we map a simplex to the convex hull of the image under Ψ of its vertices.

In this paper we started with an abstract simplicial complex as a combinatorial construction and then defined its geometric realization. In real life situations however one deals with simplicial complexes which are derived from a discretization of a manifold; that is, the usual approach is the other way around. As the concepts in this article deal only with the discretization, that is, with the combinatorial properties of a complex, the underlying geometric properties are of no particular interest.

Definition 1.9 (Star, Costar and Link of a Simplex [19], pages 11 ff.)**.** Let Δ be a simplicial complex and $\sigma \in \Delta$ a simplex. We define the star of σ to be the following subcomplex of Δ:

$$\operatorname{star}_\Delta(\sigma) := \big\{\tau \in \Delta | \tau \cup \sigma \in \Delta\big\}$$

and the costar of σ

$$\operatorname{costar}_\Delta(\sigma) := \{\tau \in \Delta | \tau \cap \sigma = \emptyset\}$$

ie. the largest subcomplex of Δ that does not contain the vertices of σ. Finally the link of a simplex is

$$\mathrm{link}_\Delta(\sigma) := \mathrm{star}_\Delta(\sigma) \cap \mathrm{costar}_\Delta(\sigma)$$

See fig.1 for an example.

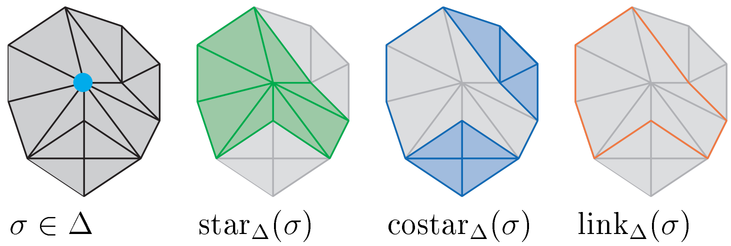


FIGURE 1. An example of *star*, *costar* and *link* for a vertex and an easy two dimensional simplicial complex.

Definition 1.10 (Star, Costar and Link of a Subcomplex)**.** Let $\tilde{\Delta} \subset \Delta$ be a subcomplex. We define the star of $\tilde{\Delta}$ to be the complex

$$\mathrm{star}_\Delta(\tilde{\Delta}) := \bigcup_{\sigma \in \tilde{\Delta}} \mathrm{star}_\Delta(\sigma)$$

the costar

$$\mathrm{costar}_\Delta(\tilde{\Delta}) := \bigcap_{\sigma \in \tilde{\Delta}} \mathrm{costar}_\Delta(\sigma)$$

and the link

$$\mathrm{link}_\Delta(\tilde{\Delta}) := \mathrm{star}_\Delta(\tilde{\Delta}) \cap \mathrm{costar}_\Delta(\tilde{\Delta})$$

Lemma 1.11. *The* costar *is inclusion reversing: If $A, B \subset \Delta$ are two subcomplexes, then*

$$A \subseteq B \Rightarrow \mathrm{costar}_\Delta(A) \supseteq \mathrm{costar}_\Delta(B)$$

Proof. Let $A \subseteq B$. Then

$$\begin{aligned} \mathrm{costar}_\Delta(B) &= \bigcap_{b \in B} \mathrm{costar}_\Delta(b) \\ &= \bigcap_{a \in A} \mathrm{costar}_\Delta(a) \ \cap \bigcap_{b \in B \setminus A} \mathrm{costar}_\Delta(b) \subseteq \mathrm{costar}_\Delta(A) \end{aligned}$$

□

Definition 1.12 (Induced Subcomplexes)**.** A subcomplex $A \subset \Delta$ is called *induced*, if every $\sigma \in \Delta$, $\sigma = \{v_0, \dots, v_n\}$ with $v_0, \dots, v_n \in A$ is also a simplex in A.

Lemma 1.13. *If (in 1.11) A, B are* induced *subcomplexes, the implication works in both directions:*

$$A \subseteq B \Leftrightarrow \operatorname{costar}_\Delta(A) \supseteq \operatorname{costar}_\Delta(B)$$

Proof. Let $\operatorname{costar}_\Delta(A) \supseteq \operatorname{costar}_\Delta(B)$. We only need to show that the vertex set of A is contained in the vertex set of B. From the definition of costar it follows that if $b \in B$: $b \notin \operatorname{costar}_\Delta(B)$ and vice versa, and the same holds for the vertices of A. Together with 1.11 this proves the lemma. □

Lemma 1.14. *For a simplicial complex Δ and a subcomplex $\tilde{\Delta} \subset \Delta$, we have*

$$\tilde{\Delta} \subseteq \operatorname{costar}_\Delta\left(\operatorname{costar}_\Delta(\tilde{\Delta})\right) \tag{1}$$

Proof. For every $b \in \operatorname{costar}_\Delta(\tilde{\Delta})$ and $a \in \tilde{\Delta}$ we have $a \in \operatorname{costar}_\Delta(b)$, by the definition of costar. Taking the intersection over all b proves the inclusion. □

Remark. If $\tilde{\Delta} \subset \Delta$ is induced in 1.14, the inclusion 1.14.(1) becomes an identity. The proof follows by passing over to the costar on both sides of the inclusion, the order reversing property of 1.11 and the definition of costar.

We analyzed the properties of the costar complexes deeper in order to understand the following definition better. We wish to access those simplices of a subcomplex, whose star contains simplices both inside and outside of the subcomplex.

Definition 1.15 (Boundary of an Induced Subcomplex)**.** Let $\tilde{\Delta} \subset \Delta$ be an induced subcomplex. The simplicial complex

$$\partial(\tilde{\Delta}) := \operatorname{link}_\Delta\left(\operatorname{costar}_\Delta(\tilde{\Delta})\right)$$

is called the boundary of $\tilde{\Delta}$.

Definition 1.16 (Poset)**.** A poset is a set P with a strict partial order $\prec$.

- We also call the set of minimal elements the *atoms* and write $\mathcal{A}(P)$
- We define the *levels* of P as $P_0 := \mathcal{A}(P)$, $P_1 := \mathcal{A}(P \setminus P_0)$, …, $P_n := \mathcal{A}(P \setminus \bigcup_{i=0}^{n-1} P_i)$ and call a poset *leveled* if $\prec$ is the trivial relation on every level.
- Let $y \in P$, we write $P_{\succ y} := \{x \in P | x \succ y\}$ for the *subposet above* y and analogously $P_{\prec y}$ for the *subposet below* y.

- If $z \in \mathcal{A}(P_{\succ x} \cap P_{\succ y})$ exists and is unique for two elements $x, y \in P$, we call z the *join* of x and y and write $z = x \vee y$.
- If $z \in P_{\prec x} \cap P_{\prec y}$ exists and is a unique maximal element of $P_{\prec x} \cap P_{\prec y}$, we call it the *meet* of x and y and write $z = x \wedge y$.

Definition 1.17 (The Face Poset $\mathcal{F}(\Delta)$)**.** Given a simplicial complex Δ, we define its face poset $\mathcal{F}(\Delta)$ to be the set of simplices of Δ, partially ordered by inclusion. Join and meet are naturally defined by union and intersection, atoms correspond to vertices (see fig.2 for an example). It is convenient to add an imaginary level -1 below the atoms, containing the empty set which is dominated by all elements.

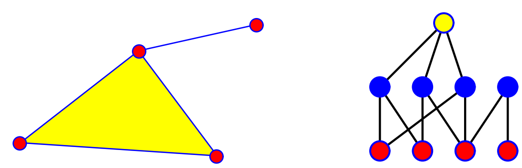


FIGURE 2. A complex and its face poset (without the level -1)

This poset has a few nice properties: first of all it is leveled. Then, from the simplicial structure, it follows that an element on the level $k-1$ has $\binom{k}{l}$ smaller elements on the level $l-1$. This information is crucial when we anticollapse; we have to maintain this simplicial structure when adding new elements.

Definition 1.18 (Pure Complexes)**.** We call a simplicial complex *pure* or to be of *homogeneous dimension*, if every maximal simplex (that is, simplices which are not a subset of another simplex) is of the same dimension.

Definition 1.19 (Order Complex)**.** Let P be a poset. We define its order complex $\Delta(P)$ as the set of chains in P where a chain $\{c_1, \dots, c_k\} \subset P$ is a totally ordered subset.

Remark. The face poset and order complex constructions are functorial: $\mathcal{F}$: **Simp** $\to$ **Posets** and Δ : **Posets** $\to$ **Simp** are functors.

The face poset construction allows us to use either the language of simplicial complexes or posets, whichever suits our needs better.

2. Some Simple Homotopy Theory

Before we talk about simple homotopy theory (which is a combinatorial concept for simplicial complexes) we define some important objects from homotopy theory and topology.

Definition 2.1 (Homotopy equivalent maps)**.** Let $f, g : A \to B$ be two continuous maps of topological spaces. We say f is homotopic to g if there is a continuous map

$$\begin{aligned} H : A \times [0,1] &\longrightarrow B \\ H(a,0) &= f(a) \\ H(a,1) &= g(a) \end{aligned}$$

and write $f \overset{H}{\simeq} g$.

Definition 2.2 (Homotopy equivalent spaces)**.** We call two topological spaces A, B homotopy equivalent, if there are maps $f : A \to B$ and $g : B \to A$ such that $f \circ g \simeq id_B$ and $g \circ f \simeq id_A$. We also say that A is a deformation of B and vice versa. In this case the maps f and g are called homotopy equivalences.

Definition 2.3 (h-Cobordant Spaces (refer to [17]))**.** Let A and B be topological spaces and $\Gamma = A \cup B$ their disjoint union. If there is a manifold M which has the boundary $\partial M = \Gamma$, then A is called cobordant to B and vice versa; M is called a cobordism of A and B. If the inclusions $M \to A$ and $M \to B$ are homotopy equivalences, A and B are called h-cobordant.

Definition 2.4 (The Homotopy Extension Property)**.** Given two topological spaces A and Ω. A map $i : A \to \Omega$ is said to have the *homotopy extension property*, if for every topological space Σ and every homotopy $H : A \times I \to \Sigma$ and map $f : \Omega \to \Sigma$ with $f \circ i = H(\cdot, 1)$, there is a homotopy $\tilde{H} : \Omega \times I \to \Sigma$ which equals to H on the image of (i, id_I) and $\tilde{H}(\cdot, 0) = f$. That is, the diagram

$$\begin{tikzcd}
A \arrow[r, "i"] \arrow[d, "\times 0"] & \Omega \arrow[d, "\times 0"] \arrow[ddr, "f"] & \\
A \times [0,1] \arrow[r, "{(i,id_I)}"] \arrow[drr, "H"'] & \Omega \times [0,1] \arrow[dr, dotted, "\exists \tilde{H}"] & \\
 & & \Sigma
\end{tikzcd}$$

commutes. A map with the homotopy extension property is also called a *cofibration*.

Definition 2.5 (Punctured spaces and maps)**.** We fix a base point in every topological space. A punctured space is a pair (K, k_0) where $k_0 \in K$ is the base point, and a punctured map $f : (K, k_0) \to (L, l_0)$ is a map with $f(k_0) = l_0$: "base points map to base points".

Definition 2.6 (Homotopy Equivalence Class)**.** Let A and B be topological spaces with the base points a_0 and b_0 and $f : A \to B$ be a punctured map. We define the homotopy equivalence class of f as

$$\langle f \rangle := \{ g : A \to B | g \simeq f \text{ and } g(a_0) = b_0 \}$$

that is, all maps which are homotopy equivalent to f relative to the base points.

Definition 2.7 (Homotopy groups π_k)**.** Let (K, k_0) be a topological space. We define

$$\pi_k(K) := \{ \langle f \rangle \, | f : (S^k, x_0) \to (K, k_0) \text{ a punctured map} \}$$

the k^{th} homotopy group or, if $k = 1$, we call it the fundamental group.

Now we get to simple homotopy theory. As mentioned, this is a combinatorial concept for simplicial complexes - hence we will use the language of posets for definitions and then explain the correspondence between the combinatorial approach and homotopy theory.
Let P be the poset of a simplicial complex.

Definition 2.8 (Collapses and Anticollapses)**.** Let $\sigma \in P$ be maximal, say on the level k. If $\tau < \sigma$ is on the level $k - 1$ (later we call such elements *premaximal*), and there is no $\tilde{\sigma} \neq \sigma$ above τ, then the removal of σ and τ yields an elementary collapse; we write $P \searrow_{(\tau,\sigma)} \tilde{P}$ and say P *collapses to* $\tilde{P}$. Furthermore, if $P \searrow_{(\tau,\sigma)} \tilde{P}$, we also say that $\tilde{P}$ *anticollapses to* P and write $\tilde{P} \nearrow_{(\tau,\sigma)} P$. (See fig.3 for a few examples.)

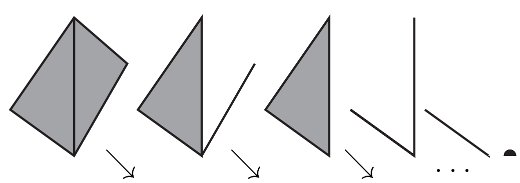


FIGURE 3. A sequence of five collapses leading from two triangles to a vertex.

Definition 2.9. We call a simplicial complex *collapsible*, if there is a sequence of collapses which remove every one its simplices except for one vertex.

Definition 2.10 (Free Faces)**.** We call the simplices which are face of exactly one maximal simplex *free face*. A simplicial complex is definitely not collapsible if it does not have any free faces.

Remark. Collapses and anticollapses (sometimes referred to as "expansions") together are called *formal deformations*.

Remark. A classical question is "Do collapses alone suffice to contract a 0-homotopic space?" The answer is 'no' and standard counterexamples are the *dunce hat* (see [23]) and the *house with two rooms* (see [1] or fig.4).

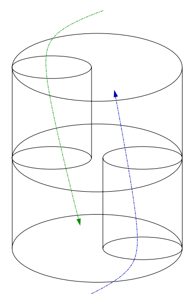


FIGURE 4. Bing's "House with two rooms": Homotopy equivalent to a point but not collapsible - There are no free edges.

Remark. The inclusion $i : |K| \to |L|$ of a subcomplex $K \subset L$ has the homotopy extension property.

Lemma 2.11. *If* $A \searrow_{(\partial_1^n(\sigma),\sigma)} B$ *then* $|A| \simeq |B|$.

We need to show the existence of maps $f : |A| \to |B|$ and $g : |B| \to |A|$ whose compositions are homotopic to the identities in the particular domains.

See fig.5 as an illustration for the proof.

Proof. We assume that $|A| \subset \mathbb{R}^k$ (see the remark on page 3) and identify σ with its "image" under the geometric realization. We transform our spaces such that the simplices σ and $\partial_1^n(\sigma)$ coincide with the standard unit simplex $\mathcal{S}^n$ or $\mathcal{S}^{n-1}$ respectively embedded into $\mathbb{R}^k$.
The vector $y := (\underbrace{-1, \ldots, -1}_{n \text{ times}}, 1, 0, \ldots, 0\}$ is orthogonal on $\partial_1^n(\sigma)$. Let $g : |B| \to |A|$ be the inclusion and $f : |A| \to |B|$ be defined as

$$f(x) := \begin{cases} x & \text{if } x \text{ is not in } \sigma \\ x + h(x) \cdot y & \text{otherwise} \end{cases}$$

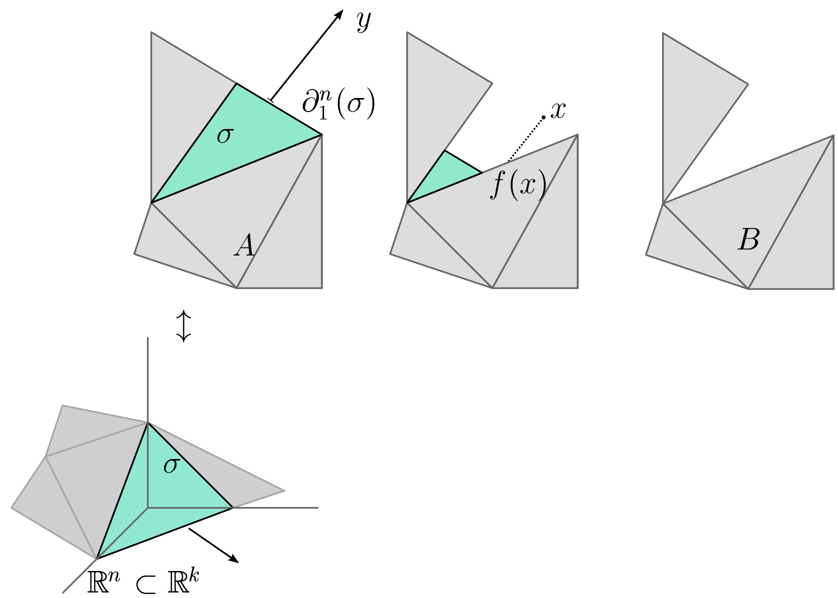


FIGURE 5. A homotopy that leads to a collapse.

where

$$h(x) = \sup\{\lambda \in \mathbb{R} | x + \lambda \cdot y \in \partial_k^n(\sigma) \text{ for some } k \neq 1\}$$

Then $f \circ g = id_{|B|}$ and $g \circ f \overset{G}{\simeq} id_{|A|}$ where the homotopy G translates the points $g(f(x))$ by $-t \cdot h(x) \cdot y$, $t \in [0,1]$ $\square$

Simple homotopy theory discusses the possibilities of formal deformations instead of general homotopies. Due to this limitation the homotopy equivalence class of a simplicial complex splits into multiple disjoint simple homotopy equivalence classes. J.H.C. Whitehead provided a theorem giving conditions for two CW-complexes to be simple homotopy equivalent.

Theorem 2.12 (In [3], §22: (22.2)). *A homotopy equivalence $g : L \to K$ of CW-complexes is a simple homotopy equivalence if the Whitehead group*

$$Wh\big(\pi_1(K)\big) = Gl_\infty\Big(\mathbb{Z}\big(\pi_1(K)\big)\Big) / \pm \pi_1(K)$$

vanishes. The group ring $\mathbb{Z}(G)$ of a group G is the ring whose elements are of the form $\sum_{g \in G} a_g \cdot g$ where the coefficients $a_g \in \mathbb{Z}$. The matrix group $Gl_\infty(R)$ of a ring R consists of those non-singular matrices which differ from the unit matrix only in finitely *many entries.*

This is an obstruction for h-corbodant spaces. We refer the interested reader to [3], especially §22 and §24 and literature on higher K-groups for more information. The following lemma is an easy consequence:

Lemma 2.13. *Let Δ be a finite simplicial complex. The topological space $|\Delta|$ is contractible if and only if there is a sequence of collapses and anticollapses leading from Δ to one of its vertices.*

Proof. The inclusion of a subcomplex $A \hookrightarrow B$ has the homotopy extension property and because $B \searrow A$ the inclusion is a homotopy equivalence (as we have shown in 2.11). So, if $\Delta \nearrow\searrow *$, we have $|\Delta| \simeq *$.
The other implication follows from theorem 2.12: In our case $g : |\Delta| \to K$ is a contraction, ie. K is a vertex. Then

$$Wh(\pi_1(K)) = Wh(\pi_1(*)) = 0$$

because in [7] it is shown that $Gl_\infty(\mathbb{Z})/\langle -1, 1\rangle = 0$, for every $k > 0$ and as there is only one way to map a sphere to a single point, $\mathbb{Z}\big(\pi_1(*)\big) = \mathbb{Z}$. $\square$

T. Chapman provided a relation between simple homotopies and homeomorphisms:

Theorem 2.14 (In [3], Appendix (pp.102), "Main Theorem")**.** *Let X, Y be finite CW-complexes. The map $f : X \to Y$ is a simple homotopy equivalence if and only if $f \times id_Q : X \times Q \to Y \times Q$ is homotopic to a homeomorphism $X \times Q \twoheadrightarrow Y \times Q$ where $Q = \prod_{j=1}^{\infty}[0, 1]$ is the Hilbert cube.*

These statements unfortunately provide no strategy for an algorithm. The following refinement of 2.14 from C.T.C. Wall [21] makes sure we do not have to anticollapse up to a very high level in order to reduce a simplicial complex.

Theorem 2.15. *Let Δ_1 and Δ_2 be two simplicial complexes of the same simple homotopy type. If $\dim(\Delta_1) = \dim(\Delta_2) = n$ and $n \neq 2$, then $\Delta_1 \nearrow\searrow \ldots \nearrow\searrow \Delta_2$ and we never have to go up more than one dimension. If $n = 2$ we do not have to go up more than two dimensions.*

There are conditions for the case $n = 2$ to make sure that anticollapses of dimension 3 suffice (see for example [13]), however these conditions depend on the fundamental groups of Δ_1 and Δ_2 - and we do not want to compute them.

3. Combinatorics of Anticollapses

We provide a different construction for anticollapses which suits our needs better than "searching for inverse collapses".

Construction 3.1 (Anticollapses (on a high level))**.** Let P be a face poset, $\tau \in P$ on the level $k \geq 0$, and assume there are at least two larger elements $\sigma_{1,2} > \tau$ on the level $k+1$ such that $\sigma_1 \vee \sigma_2$ does not exist (see fig.6 for a $k=1$ example). Then, for $j = 1, 2$: $\sigma_j \setminus \tau = \{w_j\}$ are the two vertices outside τ. By the simplicial structure we know that τ is above $k+1$ elements τ_i $(i = 0, \ldots, k)$ on the level $k-1$.
The simplex that is to be added to the poset is

$$\theta := \tau \cup \{w_1, w_2\}$$

on the level $k+2$. Define

$$M_\tau^{w_1,w_2} := \big\{\tau_i | i \in \{0, \ldots, k\}, \tau_i \vee \{w_1, w_2\} \text{ does not exist}\big\}$$

If $M_\tau^{w_1,w_2} = \{\tau_j\}$, we add θ and $\tau_j \cup \{w_1, w_2\} =: \hat{\tau}$ to the poset, thus producing an anticollapse:

$$P \nearrow_{(\hat{\tau},\theta)} \tilde{P} := P \cup \{\sigma_1 \vee \sigma_2 := \theta, \hat{\tau}\}$$

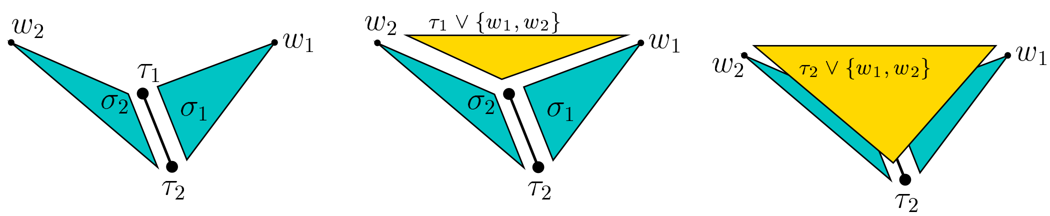


Figure 6. Anticollapsing: An explosion of the simplices whose presence in the complex is to be checked.

Remark. In the case that $M_\tau^{w_1,w_2} = \emptyset$ we know that the boundary of the simplex θ is entirely in the complex, but θ is not. Hence we must not anticollapse. If $|M_\tau^{w_1,w_2}| > 1$ we go down one level: we select a new $\tilde{\tau} \in M_\tau^{w_1,w_2}$ and set $\tilde{\sigma}_i := \tilde{\tau} \cup \{w_i\}$, $i = 1, 2$, then we compute $M_{\tilde{\tau}}^{w_1,w_2}$. If it has only one element, we anticollapse and go up again, if it has more than one element, we go further down. If it has no element, we can not proceed.
If $M_\tau^{w_1,w_2}$ can be reduced to have only one element by anticollapses on lower levels, we also write $P \nearrow_{(\hat{\tau},\theta)} \tilde{P}$, eventhough in this case we have to add more than θ and $\hat{\tau}$ to P.

Construction 3.2 (Anticollapses (on the lowest level))**.** Let $a \in P$ be an atom. Adding a new atom b and an edge $\{a, b\}$ yields an anticollapse on the lowest level. We also write $P \nearrow_{(b,a\vee b)} \tilde{P} = P \cup \{b, \{a, b\}\}$.

Remark. We assumed for anticollapses on a high level the existence of σ_1 and σ_2. Lowest level anticollapses make it possible to construct these elements if necessary.

Lemma 3.3. *The above definition of anticollapses is equivalent to our initial definition.*

Proof. Suppose $\Delta \searrow_{(\tilde{\tau},\theta)} \tilde{\Delta}$ and $\tilde{\tau}$ is on the level $k \geq 1$. Then θ has at least two further faces $\tau_1 \neq \tilde{\tau} \neq \tau_2$. Let $\sigma := \tau_1 \wedge \tau_2$. Using our new definition of anticollapses we get

$$\tilde{\Delta} \nearrow_{(\sigma_j \vee (\theta \backslash \sigma),\theta)} \Delta$$

□

Next we give a first glance of how anticollapses could help simplifying a simplicial complex:

3.1. **Elementary Anticollapses.** In the definition of anticollapses, if τ was chosen to have *exactly* two *maximal* parents $\sigma_{1,2}$, and if we can anticollapse $P \nearrow_{(\delta,\theta)} \tilde{P}$ directly (without work on lower levels), then

$$\tilde{P} \searrow_{(\sigma_1,\theta)} \searrow_{(\tau,\sigma_2)} Q$$

and we removed 2 simplices in total. This looks good, but unfortunately these choices are rarely possible. So we anticollapse on lower levels to make them become possible - this again increases the number of simplices by more than 2 and the following collapses probably will not contervail that. To get an idea how to work around these difficulties we need to apply our anticollapses on very special places. We introduce the concept of cones.

4. Cones

Definition 4.1 (Cones)**.** A cone with apex x_0 over a simplicial complex Δ is a simplicial complex $C_{x_0}(\Delta)$ with the simplex set

$$\Delta \cup \{\sigma \cup x_0 | \sigma \in \Delta\} \cup \{x_0\}$$

Cones are collapsible - in fact, ordering the elements of Δ in dimension decreasing order gives a collapsing sequence. Hence the inclusion of a complex Δ into a cone is a homotopy equivalence if Δ is contractible. Alternatively one can span a cone over a contractible *sub*complex.

Construction 4.2 (Augmenting a Cone)**.** Let Δ_0 be a simplicial complex and $\tilde{\Delta}$ a collapsible subcomplex, $\tilde{v} \in \tilde{\Delta}$ a vertex. We can anticollapse on the lowest level

$$\Delta_0 \nearrow_{(v,\{v,\tilde{v}\})} \Delta$$

by adding a vertex v and an edge $(v, \tilde{v})$. Obviously $|\Delta_0| \simeq |\Delta|$. Furthermore $C := \Delta \cup C_v(\tilde{\Delta})$ is a cone spanned over a collapsible subcomplex. See fig.7 for a visualization of the following construction. If $w \in \partial(\mathrm{star}_\Delta(\tilde{\Delta}))$ is a vertex on the boundary[1] of $\mathrm{star}_\Delta(\tilde{\Delta})$, then there are simplices $\sigma_1, \dots, \sigma_n$ in $\tilde{\Delta}$ such that $\sigma_i \cup \{w\} \in \Delta$. For any of these simplices we have

$$C \nearrow_{(\sigma_i \cup \{w\}, \sigma_i \cup \{v,w\})} C^i$$

(where sometimes this also includes anticollapses on lower levels). Furthermore, if $B := \{\sigma_1, \dots, \sigma_n\}$ is collapsible, ie. there is a permutation p such that

$$B \searrow_{(\sigma_{p(1)}, \sigma_{p(2)})} \searrow_{(\sigma_{p(3)}, \sigma_{p(4)})} \cdots \searrow_{(\sigma_{p(k-1)}, \sigma_{p(k)})} \{\sigma_{p(k+1)}\}$$

(where $\sigma_{p(k+1)}$ is a vertex) then the sequence $p(k), p(k-2), \dots, p(2)$ yields a sequence of anticollapses on C:

$$\begin{array}{lll} C & \nearrow_{(\sigma_{p(k+1)} \cup \{w\}, \sigma_{p(k+1)} \cup \{w,v\})} & C^{p(k+1)} \\ & \nearrow_{(\sigma_{p(k-1)} \cup \{w,v\}, \sigma_{p(k)} \cup \{w,v\})} & \nearrow_{(\sigma_{p(k-3)} \cup \{w,v\}, \sigma_{p(k-2)} \cup \{w,v\})} \\ & \vdots & \\ & \nearrow_{(\sigma_{p(1)} \cup \{w,v\}, \sigma_{p(2)} \cup \{w,v\})} & \bigcup_{i=1}^{k+1} C^{p(i)} =: \tilde{C} \end{array}$$

So $\tilde{C}$ is again a cone, a larger one than C because it is spanned over a larger collapsible subcomplex than $\tilde{\Delta}$. If the simplicial complex B is not collapsible, we proceed in the same way with a collapsible subcomplex $\tilde{B} \subset B$. We mention here that our algorithm chooses the subcomplex containing *a vertex of minimal valency*[2] in the 1-skeleton of Δ. We call this the "least valency first"-method and explain it later. One can iteratively increase a cone until there is no $w \in \partial(\mathrm{star}_\Delta(\tilde{\Delta}))$.

Remark. It should be clear by now why we whished for an alternative definition of anticollapses. During the cone augmentation we apply our new definition over and over again whilst the search for inverse collapses would be less determined.

[1]we will call such vertices "augmenting vertices"

[2]Valency of a vertex v: The cardinality of the 0-skeleton of $\mathrm{link}_\Delta(v)$; alternatively: the number of edges incident to v.

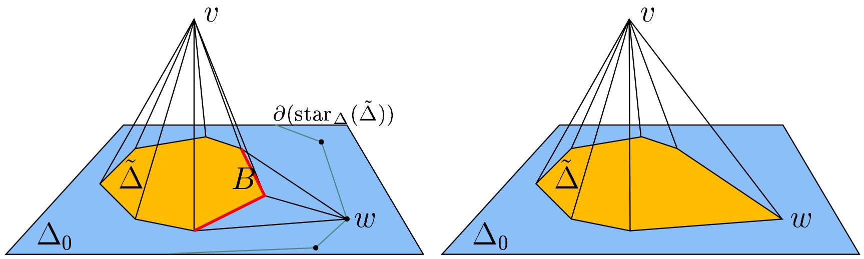


FIGURE 7. The objects involved in an augmentation.

4.1. **Flat Cones.** Above we started with an anticollapse on the lowest level to add the apex v. We can omit this first step and take any vertex of $\tilde\Delta$ as the apex of the cone (refer to fig.8 for a simple visualization of regular and flat cones. The figure also holds the main ideas for the following construction). The advantage is that the initial cone is already given by the star of v in Δ_0.

The further augmentation of this *flat cone* works exactly in the same way as above. Topologically both constructions are equally powerful:

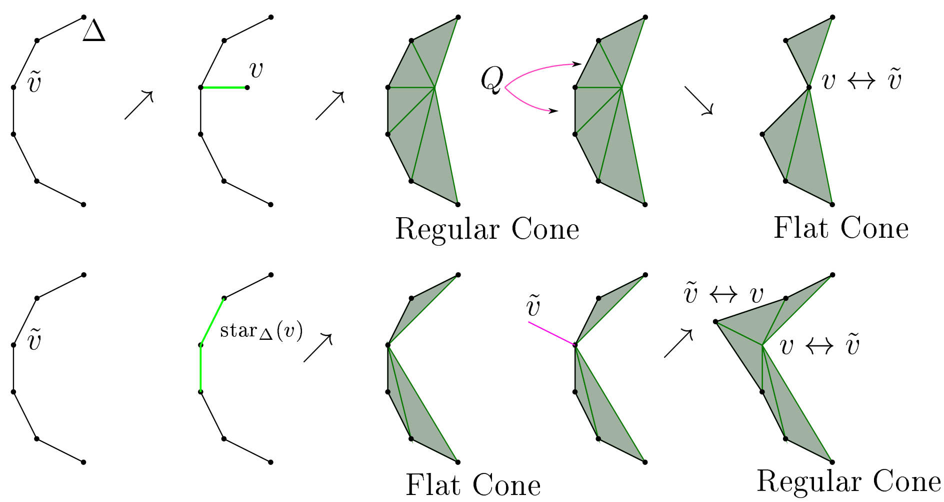


FIGURE 8. First row: The first three pictures describe the construction of a regular cone, the following pictures show how it collapses to a flat cone. Second row: The first three pictures show the construction of a flat cone and the following pictures describe the anticollapses to a regular cone.

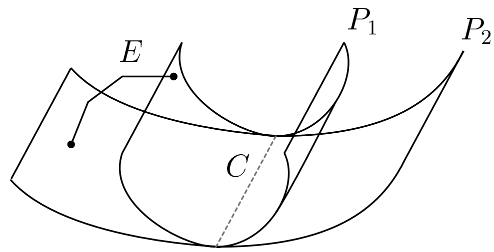


FIGURE 9. Two subcomplexes P_1 and P_2 which have a $dim(P_1) - 1$ dimensional subcomplex C in common and are connected by a path E.

Let $v \in \tilde{C}$ be the apex of the regularly spanned cone, initially added by $\Delta_0 \nearrow_{(v,\{v,\tilde{v}\})} \Delta$, and $Q := \{\sigma_0, \ldots, \sigma_k\}$ the simplices in $\mathrm{star}_{\tilde{C}}(\tilde{v}) \setminus \mathrm{costar}_{\tilde{C}}(\tilde{v})$, ordered in decreasing dimension. By collapsing

$$\tilde{C} \searrow_{(\sigma_0,\sigma_0 \vee v)} \searrow_{(\sigma_1,\sigma_1 \vee v)} \cdots \searrow_{(\sigma_k,\sigma_k \vee v)} \searrow_{(v,\{v,\tilde{v}\})} \hat{C}$$

and relabeling $\tilde{v}$ to v we built a flat cone out of a regular cone. The other direction works the same way, by changing order and vertically flipping arrows.

4.2. **Various Cone Augmenting Strategies.** If we chose to augment the cone only by simplices having certain properties, ie. to let the base of the cone have more nice properties, the shape of the cone and also its properties change. We will now discuss such strategies for the augmentation of the cone and its resulting properties. We apply our strategies on the descriptive example in fig.9.

Construction 4.3 (Pure Cones). Suppose $\mathrm{star}_\Delta(\tilde{v})$ is a pure subcomplex of dimension d. We span a cone over $\mathrm{star}_\Delta(\tilde{v})$ with apex v but chose now only to anticollapse

$$C_i \nearrow_{(\sigma,\sigma \vee v)} C_{i+1}$$

if $\sigma \subset \tau \in \mathrm{star}_\Delta(C_i)$, τ is maximal and of dimension d. Thus our cone is of pure dimension $d+1$, see fig.10

Construction 4.4 (Cones over Manifolds). We call a vertex v critical if $|\,\mathrm{link}_\Delta(v)| := |\,\mathrm{star}_\Delta(v) \cap \mathrm{costar}_\Delta(v)|$ is not homeomorphic to a sphere. Note that the star of a non critical vertex is pure and homeomorphic to a precompact subset of $\mathbb{R}^n$. Our strategy is the following: We augment the cone only by neighbours of non critical vertices. If the entire boundary of the cone base is critical, we stop augmenting,

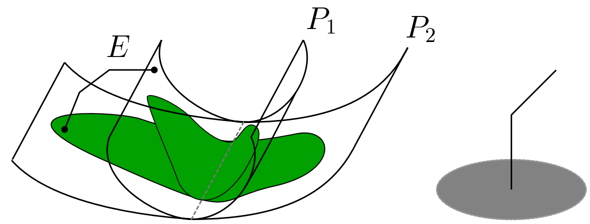


FIGURE 10. A pure cone base: It grows over P_1 and P_2 but will never contain parts of E since there are vertices having a non homogeneous star.

see fig.11. These cones are very clean and easy to control. Unfortunately we will see examples where they are not as powerful as the less restrictive strategies.

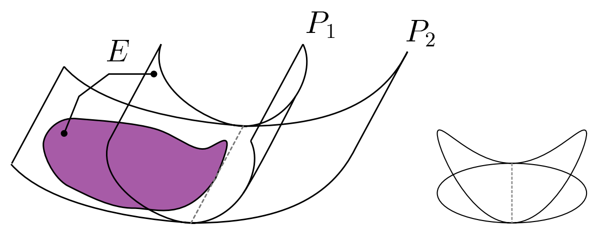


FIGURE 11. A manifold cone base: In C the link of a vertex is not homoeomorphic to a sphere.

Construction 4.5 (Small Cones)**.** Instead of augmenting a cone as far as possible over those simplices which have "a certain property", as above, we just stop as soon as there is a possibility that the following collapsing phase removes more simplices than we added. The insights we will gain in 4.4 tell us that the ratio of boundary / total simplices of the cone base has to be $\lesssim \frac{1}{2}$. We wish to find an 'as general as possible' rule to describe cone bases having this property.
Let x be the apex of the cone. We observe the augmentation:

$$
\begin{aligned}
\Delta \nearrow \Delta \cup C_x(\{v\}) \quad & \nearrow \overset{\text{a)}}{\cdots} \nearrow \quad \Delta \cup C_x(\mathrm{star}_\Delta(v)) \\
& \nearrow \overset{\text{b)}}{\cdots} \nearrow \quad C_x\big(\mathrm{star}_\Delta(\mathrm{star}_\Delta(v))\big) \\
& \nearrow \ldots \nearrow \quad C_x\Big(\mathrm{star}_\Delta\left(\mathrm{star}_\Delta(\mathrm{star}_\Delta(v))\right)\Big) \\
& \quad \vdots
\end{aligned}
$$

where we assumed that the simplicial neighbourhood of v is collapsible. Without eplicitely knowing the complex we can neither compute the number of interior nor boundary simplices - however we know, that in

a) and b) the ratio increases, and after that it may in- or decrease. Hence the strategy for *small cones* is to augment the cone base only to $\mathrm{star}_\Delta(\mathrm{star}_\Delta(v))$ and then stop.

Construction 4.6 (Smart Cones)**.** We will later (see 4.4) have a condition for the shape of a cone such that the cone spanning / collapsing process reduces the number of simplices. Thus another method for spanning a cone would be to check these conditions after every anticollapse and stop the augmentation as soon as the conditions hold. If we do not succeed to satisfy the condition, we reverse our action and remove the cone again.

See fig.12 for an example for a smart and a not so smart cone base

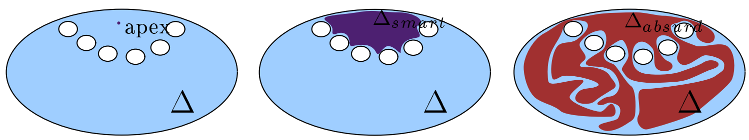


FIGURE 12. Smart cones: "Holes" in the topological space can cause absurd cone augmenting; smart cones (here over the base Δ_{smart}) only grow as long as it is reasonable.

over a 2 dimensional simplicial complex

4.3. **Collapsing a Cone.** Let $\Delta := \Delta_0 \cup C_v(\tilde{\Delta})$ be a simplicial complex with a cone having apex v and base $\tilde{\Delta}$, which is a collapsible subcomplex of Δ_0. The boundary of $\tilde{\Delta}$ is

$$\partial\tilde{\Delta} := \mathrm{star}_\Delta(\mathrm{costar}_\Delta(\tilde{\Delta})) \cap \Delta = \mathrm{link}_\Delta(\mathrm{costar}_\Delta(\tilde{\Delta}))$$

ie. $\partial\tilde{\Delta}$ contains simplices having parents inside and outside of $\tilde{\Delta}$. We can order the simplices of $\tilde{\Delta} = \{\sigma_1, \ldots, \sigma_k, \sigma_{k+1}, \ldots, \sigma_l\}$ in such a way, that the first k simplices are sorted in decreasing dimension and the remaining simplices are exactly the ones of $\partial\tilde{\Delta}$. Then

$$\Delta \searrow_{(\sigma_1, \sigma_1 \vee v)} \searrow_{(\sigma_2, \sigma_2 \vee v)} \cdots \searrow_{(\sigma_k, \sigma_k \vee v)} \hat{\Delta}$$

which means, we collapse the cone from below.

4.4. **The Balance of Building and Collapsing Cones.** Suppose the base of the spanned cone C has n_B simplices, among these there are $n_{\partial B}$ simplices on the boundary. While building the cone we had to add $1 + n_B$ simplices (in the case of a non-flat cone). When collapsing, we remove $2 \cdot (n_B - n_{\partial B})$ simplices again. Hence a cone procedure reduces the number of simplices if

$$1 + n_B - 2 \cdot (n_B - n_{\partial B}) = 1 - n_B + 2 \cdot n_{\partial B} < 0$$

Therefore, another strategy for spanning cones is to track the numbers $n_{\partial B}$ and n_B while augmenting, check the above condition and collapse if it is satisfied.

4.5. **The "Least Valency First" Strategy.** When we augment the cone $C = \Delta \cup C_x(\Delta_B)$ (Δ_B is the current cone base), we pick a vertex $w \in \partial(\mathrm{star}_\Delta(\Delta_B))$, where $w \neq x$ and try to expand the cone by w.

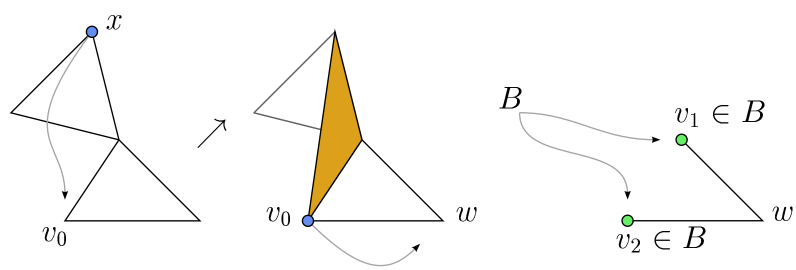


FIGURE 13. A flat cone (apex x, cone base $\{v_0, v_1, (v_0, v_1)\}$) with an ambiguous augmentation: We could either augment via $\nearrow_{(\{w,x\},\{w,x,v_1\})}$ or via $\nearrow_{(\{w,x\},\{w,x,v_2\})}$

We saw explicitely how to do so if

$$B := \mathrm{star}_\Delta(w) \cap \Delta_B$$

was collapsible. If this is not the case we mentioned that we chose that specific collapsible subcomplex of B which contains the simplices of least valency. There are three reasons why we think this is a good choice:

First reason: A simplex σ of the base of the cone will be collapsed if none of its vertices is on the boundary of the base, ie. all its subsimplices are faces of another simplex $\tau \in \Delta_B$. If a simplex is adjacent to many simplices (that is, a simplex whose vertices have a high valency), then we need to do a lot of anticollapses in the augmentation to span a cone over its entire neighbourhood. Many anticollapses mean a higher complexity of the cone and an interim higher number of simplices.
Second reason: The cone method produces vertices of high valency,

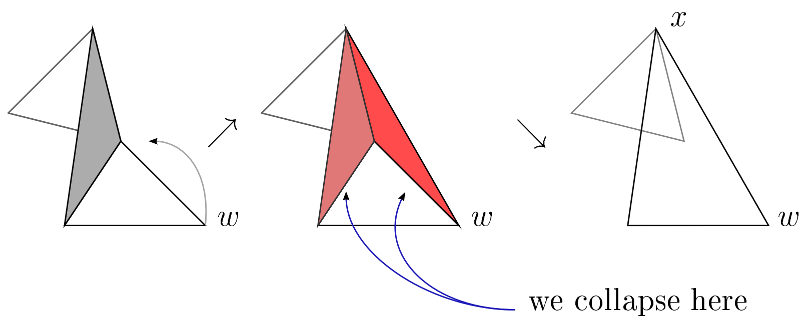


FIGURE 14. We augment the cone via the vertex v_1 which has a higher valency (4) than v_2 (3): The following collapses do not lead to an improvement and we end up with exactly the same number of simplices as we started.

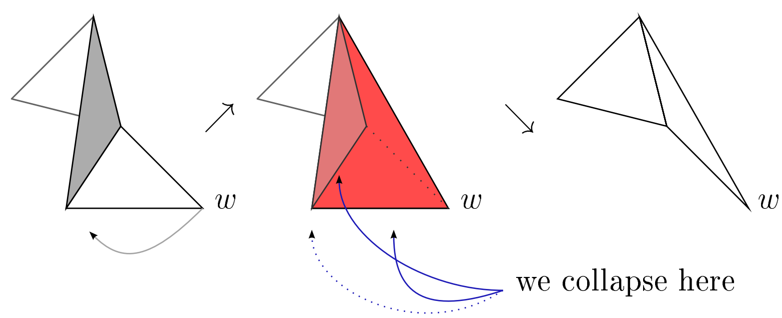


FIGURE 15. We augment the cone using the "least valency first" strategy: The following collapses simplified the graph by one egde and one vertex.

namely the apices of the cones. Suppose we started with a flat cone over the cone base Λ_1 and the collapsed

$$\Delta_0 \cup C_{x_1}(\Lambda_1) \searrow \dots \searrow \Delta_1$$

and then proceeded with the second cone with apex x_2:

$$\Delta_1 \nearrow \dots \nearrow \Delta_1 \cup C_{x_2}(\Lambda_2)$$

and $x_1 \in \Lambda_2$, the cone base of the second cone. The next augmentation is to add the vertex $w \in \Delta_1 \setminus \Lambda_2$, chosen as above. Assume w is incident to x_1 and to some other vertex v in Λ_2, then it is better to proceed with the anticollapse $\nearrow_{(w \vee x_2, w \vee v \vee x_2)}$, because otherwise ($\nearrow_{(w \vee x_2, w \vee x_1 \vee x_2)}$) we would repeat a similar augmentation as in the first cone (which, obviously, did not lead to success).
Third reason: The least valency first strategy adjusts imbalances of the degree in graphs (one dimensional simplicial complexes). Every

graph is homotopy equivalent to some complete graph K_n which has a constant degree, hence is perfectly balanced.

4.6. **The "Maximum Dimension First" Strategy.** Concerning the choice of the augmenting vertex chosen in $\mathrm{link}_\Delta(\tilde{\Delta})$ we just want to mention that it usually is a good idea to pick that one with the highest dimensional star. If the dimension of the star of the augmenting vertex is lower than the cone base, the cone can not be pure after the augmentation. The purity of a cone has shown to be a promising property.

5. Further Combinatorial Methods

5.1. **Shellings.** Let $\sigma_1, \dots, \sigma_n$ be the maximal simplices of a simplicial complex Δ. If there is a permutation $p \in S^n$ of these simplices such that

$$B_k := \sigma_{p(k)} \cap \bigcup_{i=1}^{k-1} \sigma_{p(i)}$$

is pure and $\dim(\sigma_k) - 1$ dimensional, for every $k = 2, \dots, n$, then Δ is called shellable and p a shelling order.

There are combinatorial ways to compute a shelling order. In [12] shellability is derived from a labeling of a poset.

If B_k is collapsible, for every $k \geq 2$, then the shelling order provides a sequence of anticollapses: We can inductively span cones:

$$\Delta = C_{v_n}(B_n) \cup \big(\bigcup_{i=1}^{n-1} \sigma_{p(i)} \big)$$

where v_n is a vertex of $\sigma_{p(n)}$ such that $\sigma_{p(n)} \subset C_{v_n}(B_n)$; such a vertex always exists. For more complicated B_k's, there is probably only little relation between collapses and shellings.

Shellings can also be used to analyze p.l. homeomorphisms, see [20].

5.2. **Edge Contractions.** A common way to homotope graphs are edge contractions: Let $G = (V, E)$ be an undirected graph. If for $u, v \in V$: $(u, v) \in E$, we can modify G by adding a new vertex uv and edges (uv, x) if either $(u, x) \in E$ or $(v, x) \in E$ and then remove u and v from V and all edges $(u, \cdot), (v, \cdot)$. This process is a homotopy equivalence, if $\mathrm{link}_G(u) \cap \mathrm{link}_G(v) = \emptyset$ in the initial graph G. We denote an edge contraction by $G/(u, v)$ (G "modulo" the edge (u, v)).

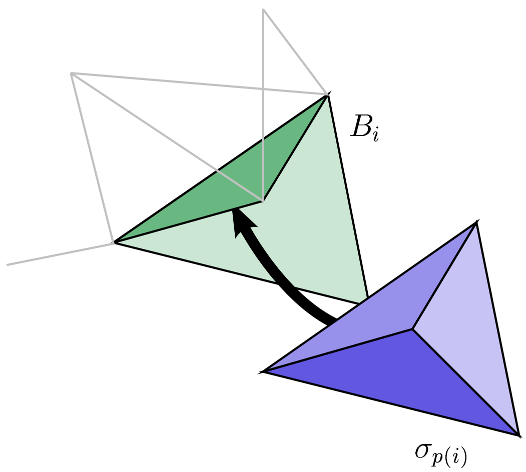


FIGURE 16. A shelling: The tetrahedron is glued to the complex on a 2 dimensional pure part of its boundary.

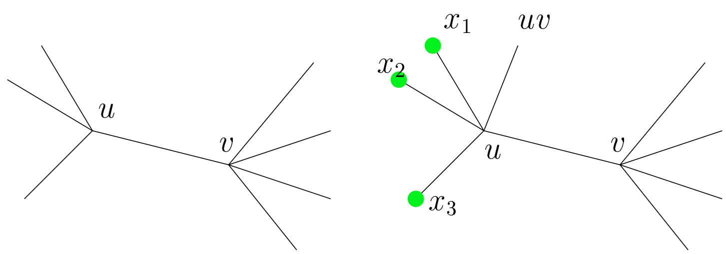


FIGURE 17. Edge contraction: The first step is to add the new vertex uv and compute the star of u.

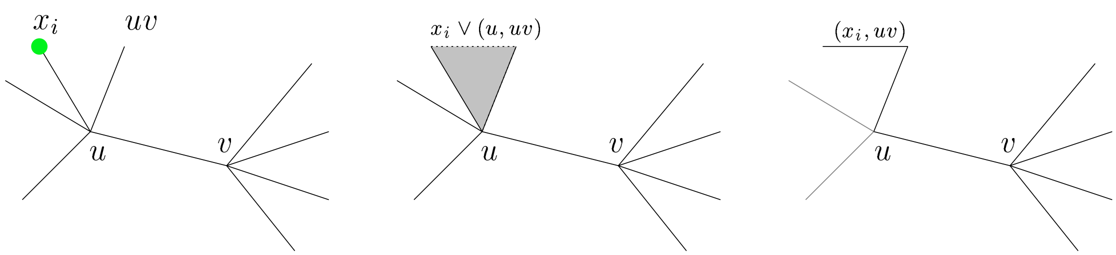


FIGURE 18. Edge contraction: For every vertex x_i of $\mathrm{star}_G(u)$ we add the triangle (u, uv, x_i) and collapse it via the edge (u, x_i).

We construct the anticollapses and collapses that lead to the above edge contraction. We start by adding the new vertex uv (fig.17) and then add all the edges (uv, x) for every $(u, x) \in E$ (fig.18). Suppose

$v, x_1, \ldots, x_k$ are the vertices of $\text{link}_G(u)$, then

$$\begin{array}{lcr} G & \nearrow_{(uv, uv \vee u)} & \tilde{G} \\ & \nearrow_{(uv \vee x_1, uv \vee (x_1, u))} \searrow_{((x_1,u), uv \vee (x_1,u))} & \tilde{G}_{x_1} \\ & \vdots & \\ & \nearrow_{(uv \vee x_k, uv \vee (x_k, u))} \searrow_{((x_k,u), uv \vee (x_k,u))} & \tilde{G}_{x_1,\ldots,x_k} \end{array}$$

we repeat the same process for the link of v, although this time we have to begin with a different step (fig.19 and 20):

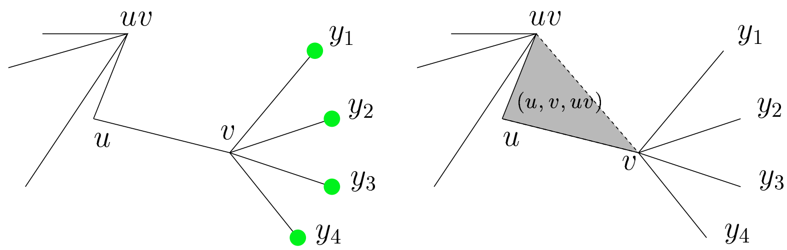


FIGURE 19. Edge contraction: In order to connect v to uv we anticollapse the triangle (u, v, uv). Then we compute the star of v.

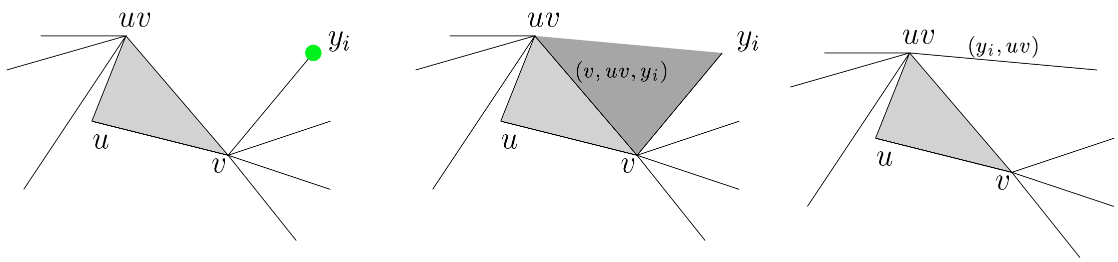


FIGURE 20. Edge contraction: We anticollapse the triangle (v, uv, y_i) for every $y_i \in \text{link}_\Delta(v)$ and collapse it via the edge (v, y_i).

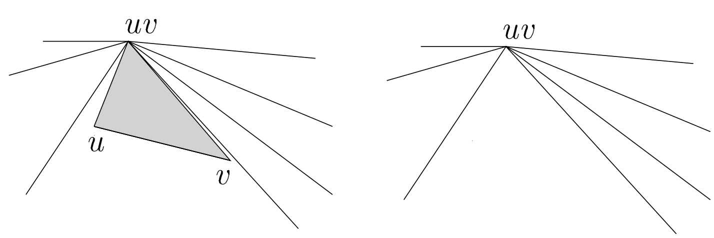


FIGURE 21. Edge contraction: The last step is to remove the triangle (u, v, uv) by collapses.

$$\begin{array}{llll}
\tilde{G}_{x_1,\dots,x_k} & & \nearrow_{(uv\vee v,uv\vee(v,u))} & \\
 & \nearrow_{(uv\vee y_1,uv\vee(y_1,v))} & \searrow_{((y_1,v),uv\vee(y_1,v))} & \tilde{G}_{y_1} \\
 & & \vdots & \\
 & \nearrow_{(uv\vee y_k,uv\vee(y_k,v))} & \searrow_{((y_k,v),uv\vee(y_k,v))} & \tilde{G}_{y_1,\dots,y_k}
\end{array}$$

The last step is to remove the triangle (u, v, uv) (fig.21):

$$\tilde{G}_{y_1,\dots,y_k} \searrow_{((u,v),(u,v,uv))} \searrow_{(u,(u,uv))} \searrow_{(v,(v,uv))} H$$

and $H = G/(u, v)$.

The concept of edge contractions can be generalized on higher dimensional simplicial complexes where similar conditions as the link intersection above lead to homotopy equivalence (see for instance [4]). The above recipe to translate edge contractions to anticollapses and collapses translates to this general case almost one to one - however one has to pay attention in which order the anticollapses are done.

Concerning the question which is more powerful, cone collapsing or edge contractions, we refer the reader to the footnote on our observations in table 4 about the torus on page 35.

5.3. **Subdivisions and Coarsening.** Given two simplicial complexes $\Delta_{1,2}$ with $|\Delta_1| = |\Delta_2|$ (up to homeomorphism), we call Δ_2 a *subdivision* of Δ_1 if there is a simplicial (cellular) map $f : \Delta_1 \to \Delta_2$ which is monic. Some subdivisions are simple homotopy equivalences, refer to [3], §25, Thm. (25.1) for a general theorem. We only provide a recipe for barycentric subdivisions of simplicial complexes using collapses and anticollapses:

(1) we glue a simplex of dimension $k+1$ onto every *maximal* simplex σ. ($k = \dim(\sigma)$)
(2) we collapse every of these new simplices via the previously maximal σ's
(3) for every remaining simplex, we span a cone over its star and collapse it from below (start with the highest dimensional remaining simplices).

The usual combinatorial way for barycentric subdivisions however is to compute the order complex of the face poset; the explicit construction is displayed in [11].

If we span cones over small bases, our algorithm behaves like a coarsening algorithm; but it does not equal the inverse of barycentrc subdivisions.

5.4. **Morse Matchings.** R. Forman introduced a Morse theory for cell-complexes in [6]. Morse functions on simplicial complexes can be computed by the linear extension of an acyclic matching on the face poset. An acyclic matching provides a sequence of collapses on a CW complex. Hence an optimal sequence of collapses can be computed by an "optimal Morse matching".

Definition 5.1 (Morse Matching)**.** A matching M on the face poset of a simplicial complex is a subset $M \subset \mathcal{F}(\Delta) \times \mathcal{F}(\Delta)$ such that

- every element $e \in \mathcal{F}(\Delta)$ appears only once in the matching: $\exists! q \in M : q = (e, \cdot)$ or $q = (\cdot, e)$.
- $(a, b) \in M$ implies $a \prec b$. We write $b := u(a)$ and $a := d(b)$.

A matching M is called a Morse matching if there is no sequence $(a_1, u(a_1)), (a_2, u(a_2)), \ldots, (a_k, u(a_k)) \in M$ of length ≥ 2 such that

$$u(a_1) \succ a_2, \ldots, u(a_{k-1}) \succ a_k, u(a_k) \succ a_1$$

Definition 5.2. A linear extension is a map $L : \mathcal{F}(\Delta) \to \mathbb{R}$, or more generally a map from a poset to a totally ordered set, which preserves the order.

Optimal Morse matchings are hard to find. See [15], where it is claimed that this computation is in general NP hard (and that every polynomial algorithm performs arbitrarily bad for certain posets); in [9] an integer program is given to compute an optimal Morse matching. Many integer programs are NP complete.
Our algorithm however starts with random collapses, just to make sure that there are no free faces - not to simplify the complex. When we span cones we can right away produce an almost optimal sequence of collapses to collapse the cone again.

6. Implementation

6.1. **Simplices as Products of Primes.** In order to motivate a certain function used later in the data structure we wish to give an alternative method of how to describe simplicial complexes. A finite simplicial complex has for example the vertices $v_1, v_2, \ldots, v_n$, and its simplices are uniquely described by their sets of vertices (modulo orientation).
Let p_j be the j^{th} prime number: $p_1 = 2$, $p_2 = 3$ and so on. Hence a simplex $\{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$ can be encoded as the number

$$s = (-1)^{\rho} \cdot \prod_{j=1}^{k} p_{i_j}$$

where we accounted for the orientation by the factor $(-1)^\rho$. Hence we can describe a finite simplicial complex as a subset $\Delta \subset \mathbb{Z}$.

Remark (Join and Meet)**.** Let $\sigma_1, \sigma_2, \ldots, \sigma_l$ be products of primes, ie. simplices encoded as above. We can compute their join

$$\bigvee_{i=1}^{l} \sigma_i = lcm\{\sigma_i | i = 1, \ldots, l\}$$

and their meet

$$\bigwedge_{i=1}^{l} \sigma_i = gcd\{\sigma_i | i = 1, \ldots, l\}$$

Remark (Bit-Strings instead of Primes)**.** Yet another way to describe simplices is to write them as long strings of binary values, each position corresponds to a vertex in the complex. The position j of the simplex $\sigma = 00100\ldots00$ is 1, if $v_j \in \sigma$ and 0 otherwise. Join and meet then become logical OR and AND operations.

6.2. **Data Structure.** We will have one storage container for all simplices involved in our algorithm and handle everything "by reference". For the storage we chose a hashtable:

Values	Keys	Hashcode
Simplex 1	Vertex Array	$\prod p_j \ (mod\ 2^{32} - 1)$
Simplex 2	Vertex Array	
$\vdots$		

where the hashcode is the corresponding product of primes modulo the size of an integer. This hashtable corresponds to the elements of the face poset of a simplicial complex, currently without the order relation. Because a face poset is leveled, it suffices if we know this order relation only for each level, or element, respectively.
Every simplex is an object:

Simplex

vertices	integer array	the vertices of the simplex
parents	simplex list	simplices one level above
children	simplex list	simplices one level below

where a simplex list is

Simplex List

first	Entry	the first entry in the list
last	Entry	the last entry
count	integer	the number of elements
	Entry	
contents	Simplex	reference to that simplex
next	Entry	the next entry in the list
previous	Entry	previous entry

So we have the elements of $\mathcal{F}(\Delta)$ stored in a hashtable and the order relations $\prec$ (for only the level directly above and below) included with each entry.

6.3. **Collapsing and Apex Queue.** Finally we need to coordinate the collapsing / anticollapsing sequence. In order to do so, we introduce two queues, the apex queue and the collapsing queue. Each of which is of the type simplex list and thus is stored sorted. We implement the possibilities to add new elements both at the end and at the beginning of the list. This is necessary, because we need to be able to strictly control the sequence of collapses and not let the algorithm collapse cones on the wrong way.
The collapsing queue is a list of free faces. Free faces uniquely determine a collapse, as they have only one simplex above. The apex queue is a list of all non isolated vertices of the complex. It makes no sense to span a cone over an isolated vertex.

6.4. **The Algorithm.** We will now discuss the main subroutines of the algorithm. After some explanation we directly try to analyze its complexity. We denote the overall number of simplices N. The complexity of the algorithm does not only depend on that number but also on the grade of connectivity, genus and dimension of the simplicial complex. It is probably not possible to bound the run time exactly, however we wish to give good estimates depending on N. For the sake of simplicity we drop logarithmic expressions - which means a query in a hashtable takes constant time.

Subroutine 6.1 (Preparation)**.** We need to compute the collapsing- and apex queue. We compute them only once, at the beginning of the algorithm and later only update them as the simplicial complex changes. If we have a non empty collapsing queue it is reasonable to collapse, because the main algorithm starts with a cone spanning phase.

We avoid redundant anticollapses if we start with a non collapsible complex.

1 for every simplex $\sigma \in \Delta$
  1.1 if $\sigma < \tau$, $\sigma \not< \tilde{\tau} \neq \tau$ and τ is maximal, we add σ to the collapsing queue.
  1.2 if σ is a non isolated vertex, we add it to the apex queue.
2 if the collapsing queue is non empty, collapse.

Complexity Analysis 6.2. Iterating over all simplices has complexity $O(N)$. Our data structure makes both checking for collapsability or isolation a task of constant time.

Subroutine 6.3 (Collapsing)**.** If the collapsing queue is nonempty, the following method *collapse* iterates over all its elements and tries to collapse. Note that after a collapse there might be new maximal simplices and possibly new possibilities for collapses.

1 For every simplex σ in the collapsing queue:
  1.1 let $\tau > \sigma$ be the only parent of σ
  1.2 make sure τ is maximal and σ is not below any other $\eta \neq \tau$
  1.3 remove σ and τ from the hashtable
  1.4 if σ was a vertex, remove it from the apex queue
  1.5 update parent references of the children of τ and σ
  1.6 for every child θ of σ:
    1.6.1 if there is only one element above θ left, add θ to the very start[3] of the collapsing queue
  1.7 for every child θ of τ:
    1.7.1 if there is only one element above θ left, add θ to the very end[4] of the collapsing queue
  1.8 remove σ from the queue
2 break, if the queue is empty

Complexity Analysis 6.4. Assume the collapsing queue contains all N simplices of Δ. For each of them we check the parents ($O(1)$), possibly remove them from the hashtable ($O(1)$) and the apex queue ($O(\log(N)) \approx O(1)$, if we, for example, sort the queue after generation). Furthermore we check the children of the collapsed simplices for maximality ($O(d) \approx O(1)$, where d is the dimension of the collapsed maximal simplex). Finally, we update the collapsing queue: $O(1)$. This totals to an overall complexity of order $c \cdot N + k \in O(N)$.

---

[3]Refer to 6.7 for an exaplanation

[4]see footnote 3

Subroutine 6.5 (Spanning a Cone)**.** If there are non isolated vertices it is possible to span flat cones. This method tries to do so with the first entry of the apex queue.

1 Pick the next apex a from the apex queue
2 let S be the star of a
3 repeat the following construction until an entire iteration yields no change
  3.1 compute the vertices $M = \{v_1, \dots, v_k\}$ of $\text{link}_\Delta(S)$
  3.2 for $i = 1, \dots, k$:
    3.2.1 compute $\Omega := \text{star}_\Delta(v_i) \cap S$
    3.2.2 sort Ω by dimension and sort simplices of same dimension by valency, least valency first
    3.2.3 for every $w \in \Omega$ (and repeat until an entire iteration yields no success)
      3.2.3.1 anticollapse, if possible: $\nearrow_{(w \vee v_i, w \vee a \vee v_i)}$; that means, add both simplices to the hashtable.
      3.2.3.2 if we anticollapsed add the new simplices also to S and if $w \vee v_i$ was maximal in Δ add $w \vee v_i$ to the collapsing queue.
  3.3 next i
4 end repeat

Complexity Analysis 6.6. The computation of the star is a delicate subject. The complexity depends on the properties of the simplicial complex, such as maximal degree of the 1-skeleton, or more general the average number of simplices above the apices. We only know that this number is bounded above by N. Assuming worst case, we iterate over $O(N)$ vertices, trying to augment the cone. For each of them we compute their star ($O(N)$) and intersect it with the cone ($O(N)$). For every element of the intersection we try to anticollapse ($O(1)$) and add a new simplex to the collapsing queue ($O(1)$) and to the cone ($O(1)$). Finally, we update the star of the new vertex of the cone ($O(1)$). This totals to a runtime (per apex) of $O(N^2 + N)$. Note that we considered a very unrealistic worst case here.

Subroutine 6.7 (The Main Routine)**.** The main algorithm switches between collapses and anticollapses:

1 prepare
2 collapse $\Delta \searrow \Delta_0$
3 set $l = 0$ and $k = 0$
4 while $k < N$
  4.1 span a cone $\Delta_k \nearrow \Sigma$

4.2 collapse $\Sigma \searrow \Delta_{k+1}$
4.3 count the simplices: $N_{\Delta_{k+1}}$
4.4 if $N_{\Delta_{k+1}} \geq N_{\Delta_k}$ discard your work: set $\Delta_{k+1} := \Delta_k$ and increment l by 1. otherwise set $l = 0$
4.5 $k \to k+1$

Complexity Analysis 6.8. In worst case we have to iterate about N^2 times.

6.5. **Overall Complexity Analysis.** The above step by step analysis reveals a runtime of $O(N^4)$. However, we assumed worst case settings for each routine that are not compatible with each other. To get an accurate estimate on the algorithms complexity we obviously need a new point of view.
We will try to count the operations in the algorithm that involve a simplex. We use the *small cone* strategy, which is the easiest to analyze. Let τ be any simplex that is not collapsed in the preparation phase. Suppose $\tau = \{v_0, \ldots, v_d\}$

- τ is involved exactly in those cone spanning phases where $\tau \in \mathrm{star}_\Delta\left(\mathrm{star}_\Delta(x)\right)$, where x is the cone apex.
- We need to compute the star of x, a process whose complexity depends a lot on the implementation. We assume a generic complexity of this process: $O(g_{\mathrm{star}}(N))$. According to our previous analysis, $g_{\mathrm{star}}(N) \in O(N^2)$ in worst case. Usually it is a lot better.
- If $\tau \in \mathrm{star}_\Delta(x)$, there are at most $d-1$ anticollapses necessary to augment the cone over τ. Each anticollapse has to check the presence of certain simplices; this however takes constant time. In the following collapsing phase the simplex τ will be removed, which is again a task taking constant time.
- If $\tau \in \mathrm{star}_\Delta\left(\mathrm{star}_\Delta(x)\right) \setminus \mathrm{star}_\Delta(x)$, it is possible that τ will be removed in the following collapsing phase or it will be replaced by another simplex $\tilde{\tau} = \{v_0, \ldots, v_k, x, v_{k+2}, \ldots, v_d\}$. The latter possibility has a worse effect on the algorithms complexity.
  So we had to compute another star, anticollapse $d-1$ times to augment the cone over τ. In the collapsing phase, τ will collapse but $\tilde{\tau}$ survives.
- The replacement of a simplex τ by a similar simplex $\tilde{\tau}$ can at most happen L times, where L is the number of vertices in $\mathrm{link}_\Delta(\tau)$. After these L failures, if τ is contained in a cone, it *will* be removed for sure.

We add up the operations per simplex:

$$L\cdot\big(O(g_{\text{star}}(N))+(d-1)\cdot O(1)\big)+O(g_{\text{star}}(N))+O(1)\approx(L+1)\cdot O(g(N))$$

Before we get to the g_{star} analysis we want to sum up over all simplices. Multiplying our result by N and setting L to N leads to our previous result $N^2\cdot g_{\text{star}}(N)$. We have however to consider that L can not be large for every simplex. L depends on the simplex τ. We compute the set of simplices with a large L:

$$M=\{\tau\in\Delta|L(\tau)\geq c\cdot N\}$$

and derive by the properties of link and star that M has $\approx\frac{1}{c}\cdot N$ elements.
We end up with a complexity $O(N\cdot g_{\text{star}}(N))$ which equals $O(N^3)$ for a bad star computation and worst case settings.

6.6. **The Star Problem.** Every simplex in the data structure holds information about the simplices directly above and below. Given the simplicial complex Δ, its face poset $Q=\mathcal{F}(\Delta)$ and a vertex $v\in\Delta$, we have

$$\mathcal{F}\big(\operatorname{star}_\Delta(v)\big)=\bigcup_{a\in P_{\succ v}(Q)}P_{\prec a}(Q)$$

so the star of a vertex corresponds to the union of all subposets below any element above an atom . The subroutine which is to compute the star therefore needs to go up and down, simplex per simplex, level per level to assemble the star of a vertex. This is an obvious weakness of our data structure.
If we work with simplicial complexes whose vertices have a small average degree, there is a way to obviate the star problem. we change the data structure of *simplices*:

Simplex

vertices	as before
parents	as before
children	as before
maximals	a simplexlist containing all maximal simplices above this simplex

This modification uses a lot more memory for highly connected simplicial complexes but reduces the cost of the star computation. Note that going down in a face poset usually is less costy than going up since the number of children is bounded, the number of parents is not (unlike reality).

6.7. **Updating the Collapsing Queue.** In some cases a collapse

$$\Delta \searrow_{(\tau,\sigma)} \tilde{\Delta}$$

produces further free faces: Either faces $\tau_i \subset \tau$ or $\sigma_i \subset \sigma$, or both. Assume the collapsing queue before the collapse was

$$(\tau, \eta, \theta, \ldots)$$

Is there a better or a worse choice for where to add the new free faces? We add τ_i at the beginning of the queue and σ_i at the end. Why:

- If we continue with σ_i instead, we "dig" a hole deeper and deeper into the complex thus producing a chance of building a house with two (or more) rooms.
- If we continue with τ_i we proceed with the collapsing of the surface of the simplex τ (and σ as well) - hence, by continuing in that manner we possibly end up with a (vertex, edge) collapse and reduced the number of vertices.

Hence our collapsing queue will be updated to

$$(\tau_i, \eta, \theta, \ldots, \sigma_i)$$

7. Results

7.1. **Benchmarks and Examples.** We included an instruction counter *opc* which increments for every non trivial operation:

- method calls
- every iteration of for/while loops
- if clauses

We tested our algorithm for an array of topological spaces and triangulations to gauge its complexity and efficiency. To make sure the algorithm really does what we expect, we used the algorithm described in [5] to compute the homology of input and output of our algorithm.

7.1.1. *Collapsible Complexes.* In table 1 we display some benchmarks on collapsible complexes, or complexes consisting of collapsible connectivity components. Our benchmarks show, that collapsing is almost linear in the input size, independent of dimension and shape.
We assumed that a random order (combined with 6.7) of collapses leads to success. It worked for our examples but is not true in general.

Table 1. Random collapses on complexes with ≥ 1 collapsible connectivity components.

Dimension	3	4	4
$N^{\underline{o}}$ of vertices	258	2673	1081
$N^{\underline{o}}$ of edges	1248	26278	10210
$N^{\underline{o}}$ of triangles	1748	51044	19560
$N^{\underline{o}}$ of tretrahedra	756	33485	12570
$N^{\underline{o}}$ of pentatopes		51044	2140
op. count	31038	966800	367960
op.'s per simplex	7,7	8	8,08

7.1.2. *The House with Two Rooms.* We computed a triangulation of the house with two rooms and subdivided it, to check how the algorithm performs. In table 2 we collected our observations. Obviously the complexity is not linear in the number of simplices, but it is definitely below quadratic. We assume that there are some logarithmic factors depending on the number of cones spanned.

Table 2. Benchmarks on various triangulations of the house with two rooms.

Triangulation of	very coarse	coarse	fine	very fine
$N^{\underline{o}}$ of vertices	17	111	689	4185
$N^{\underline{o}}$ of edges	55	344	2092	12608
$N^{\underline{o}}$ of triangles	39	234	1404	8424
op. count	17325	347056	2286922	29929925
op.'s per simplex	156,08	503,709	546,45	1186,89

7.1.3. *Non Contractible Topological Spaces.* It is in general not easy to determine whether the number of simplices in a simplical complex can be reduced or not. We have of course the criterion 4.4 which is to be analyzed for every possible cone – which is not feasible for large complexes. This means, that there is no criterion for the algorithm to stop trying to span cones. In our description of the algorithm we repeated spanning cones until the current cone augmenting strategies fail for every apex.

We observed that in some cases the number of vertices does not decrease monotonically before reaching a minimal state; hence the above termination criterion would prevent the algorithm from finding an optimal triangulation. For this reason we changed the algorithm to have no termination criterion at all; we just span a cone (using whichever

Table 3. Minimal triangulations of non contractible simplicial complexes after a number of invocations. (Input / Output sizes are given as (vertices, edges, triangles, ...))

Space	Input Size	Invocations	Output Size
2-sphere	(798, 2388, 1592)	4	$(4, 6, 4)$
	(4778, 14328, 9552)	6	$(4, 6, 4)$
Torus	(31, 95, 64)	10	$(7, 21, 14)$
	(94, 282, 188)	11	$(7, 21, 14)$
	(564, 1692, 1128)	10	$(7, 21, 14)$
Surface of genus 2	(458, 1380, 920)	13	$(23, 101, 76)$
	(2758, 8280, 5520)	15	$(17, 80, 61)$
Complement of a solid torus	(1572, 9575, 15369, 7365)	5	$(5, 8, 4)$
	(33881, 224049, 368846, 178677)	5	$(5, 8, 4)$
Complement of two enlinked solid tori	(922, 5521, 8740, 4140)	13	$(11, 41, 31)$
	(37205, 250372, 421792, 208624)	14	$(9, 30, 22)$

augmenting strategy) for every element in the apex queue – and we even remove vertices from the apex queue if they were once contained in a cone base.
This version of the algorithm does not ensure to return a minimal[5] triangulation, but can be invoked repeatedly until the return value meets the expectations, for instance by a 'for loop' or by repeated calls from a user.
We had to find out how many calls to the algorithm suffice in general to reach a minimal state. Therefore we (repeatedly) barycentrically subdivided non contractible simplicial complexes and invoked the algorithm until we reached a certain minimal state. We collected our observations in table 3.

7.2. **Comparison of Augmenting Strategies.** There are some differences in the results of our algorithm if we use various augmenting strategies. We collected our observations in table 4.

The only drawback of small cones is that we need very many of them to simplify a simplicial complex. We implemented a cone counter

[5] "minimal" for the current cone strategy

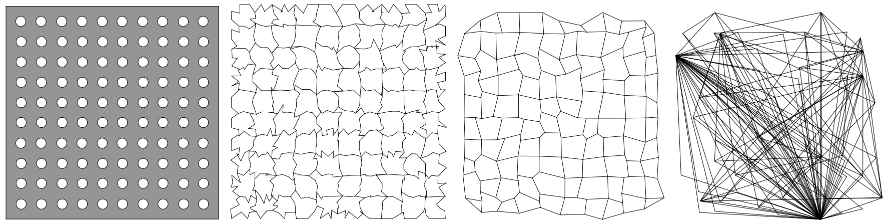


FIGURE 22. The initial complex collapses, then is simplified first by the "cone over manifold" strategy and then by maximal cones.

and computed the ${}^{cones}/_{simplex}$ ratio which we also used to analyze the complexity. Every gauging resulted in a value between 0.018 and 0.031 cones per simplex; the value seems to *decrease* if we take a finer triangulation. We have also observed that the algorithm does the most simplifications during the first invocation; the values[6] we collected in table 3 would be a lot smaller if we accepted output values which were only a bit larger than the optimal.

TABLE 4. Capabilities of different cone augmenting strategies

Strategy	Maximal	Manifold	Small
Contracts ⇧	Sometimes[a]	Sometimes	Yes
Minimizes a wedge of spheres[b]	Yes	Yes	Yes
Computes topological cliques[c]	Yes	No	Yes[d]
Minimizes a torus	No	No	Yes[e]

[a]Yes for some triangulations, No for others

[b]A topological space consisting of spheres of different dimensions which have one vertex in common

[c]A clique is a maximal complete subgraph. A topological clique is a graph homeomorphic to a complete graph.

[d]Refer to [10] for more information on topological cliques

[e]We know from [14] that "irreducible" (for edge contractions) tori have 9 or 10 vertices. The cone strategy finds vertex minimal (see [18]) triangulations with 7 vertices.

So the small cone strategy is obviously superior to all the other "greedy" strategies. Only the "smart cone" strategy is able to revert to a better cone base if it augmented in an inept way. The greedy cone strategies, especially the maximal cones, will even deteriorate a bad cone.

[6]for the number of invocations

8. Résumé

We collect our insights on the algorithm:
The cone strategy is very powerful to reduce the number of simplices and find vertex minimal triangulations for many topological spaces. It is more powerful than similar algortihms including edge contractions.
The algorithm has in theory a complexity between $O(N)$ and $O(N^4)$ depending on the assumptions taken for the simplicial complex. The gauging however shows that the algorithm performs well and its complexity is almost linear, modulo logarithmic expressions.
In order not to modify the *geometrical* properties of the complex, it is reasonable to span the cones over subcomplexes with "nice" properties. P.l. manifold cone bases are good but too restrictive. Small cones, that is, cones over $\mathrm{star}_\Delta\left(\mathrm{star}_\Delta(v)\right)$ (where v is the apex) lead to the best results.

Acknowledgements

The author would like to thank the following people for their assistance and help: Prof. Ralf Hiptmair, Dr. Alexander Engström, Ivo Dell'Ambrogio, Beat Steiner, Theo Bühler, Srdjan Johnny Micic, Peter Walser, Nora Graser, Maya and Klaus Kayatz.

References

[1] R. Bing. *Lectures in modern mathematics*, volume II. Wiley, New York, 1964.
[2] B. Bollobás. *Modern Graph Theory*, volume 183 of *Graduate Texts in Mathematics*. Springer, New York, 1998.
[3] M. Cohen. *A course in simple-homotopy theory*, volume 10 of *Graduate Texts in Mathematics*. Springer Verlag, New York - Berlin, 1973.
[4] T. Dey, H. Edelsbrunner, S. Guha, and D. Nekhayev. Topology preserving edge contractions, 1998.
[5] F. Rapetti et al. Integer matrix factorization for mesh defect detection. *C. R. Acad. Sci. Paris*, pages 717–720, 2002.
[6] R. Forman. Morse theory for cell-complexes. *Advances in Math.*, 134:90–145, 1998.
[7] G. Higman. The units of group rings. *Proc. London Math. Soc.*, 46:231–248, 1940.
[8] J.F.P. Hudson and E.C. Zeeman. On combinatorial isotopy. *Publications Mathématiques de l'IHÉS*, 19:69–94, 1964.
[9] M. Joswig and M.E. Pfetsch. Computing optimal morse matchings. *arXiv:math.CO/0408331*, 2004.
[10] J. Komlós and E. Szemerédi. Topological cliques in graphs. ii. *Combin. Probab. Comput.*, 5:79–90, 1996.

[11] D. Kozlov. Graph colorings and the Kneser conjecture. *Lecture Notes*, Lecture 4, 2004.

[12] D. Kozlov. Advanced topics in topological combinatorics. *Lecture Notes*, Lecture 18, 2005.

[13] R. Kreher and W. Metzler. Simplicial transformations of polyhedra and the Zeeman conjecture. *Topology*, 22(1):19–26, 1983.

[14] S.A. Lavrenchenko. Irreducible triangulations of a torus. *Ukrain Geom. Sb.*, 30(ii):52–62, 1987.

[15] T. Lewiner, H. Lopes, and G. Tavares. Optimal discrete morse functions for 2-manifolds. *Computational Geometry: Theory and Applications*, 26(3):221–233, 2003.

[16] W.S. Massey. *Algebraic Toplogy: An introduction*. Springer Verlag, 1997.

[17] J. Milnor. *Lectures on the h-cobordism theorem*. Notes by L. Siebenmann and J. Sondow. Princeton University Press.

[18] A.F. Möbius. Mittheilungen aus Möbius' Nachlass: I. Zur Theorie der Polyëder und der Elementarverwandtschaft. *F. Klein (Eds.): Gesammelte Werke II*, pages 515–559, 1886.

[19] J.R. Munkres. *Elements of Algebraic Topology*. Perseus Press, 1993.

[20] U. Pachner. P.l. homeomorphic manifolds are equivalent by elementary shellings. *Europ. J. Combinatorics*, 12:129–145, 1991.

[21] C. T. C. Wall. Formal deformations. *Proc. London Math. Soc.*, 16:342–352, 1966.

[22] J.H.C. Whitehead. *The mathematical works of J.H.C. Whitehead.*, volume III: Homotopy Theory. Pergamon Press, Oxford - New York - Paris, 1962.

[23] E.C. Zeeman. On the dunce hat. *Topology*, 2:341–358, 1964.