A Computationally Intractable Problem on Simplicial Complexes *

Ömer Eğecioğlu and *Teofilo F. Gonzalez*

Department of Computer Science
University of California Santa Barbara, CA 93106

Abstract

We analyze the problem of computing the minimum number $er(\mathcal{C})$ of internal simplexes that need to be removed from a simplicial 2-complex $\mathcal{C}$ so that the remaining complex can be nulled by deleting a sequence of external simplexes. This is equivalent to requiring that the resulting complex be collapsible to a 1-complex. By reducing a restricted version of SAT, we show that this problem is $\mathcal{NP}$-complete and therefore computationally intractable. This implies that there is no simple formula for $er(\mathcal{C})$ terms of the Betti numbers of the complex. The problem remains $\mathcal{NP}$-complete for higher dimensional complexes, but can be solved in polynomial time for graphs.

Keywords: Simplicial complex, collapsing, Betti number, algorithmic complexity, intractable problem.

1 Introduction

We consider finite connected simplicial 2-complexes, all of whose maximal simplexes are 2-dimensional. We view such a complex as a collection of 2-simplexes $\mathcal{C} = \{s_1, s_2, \ldots, s_n\}$ modulo an equivalence relation that identifies a pair of simplexes s_i and s_j with $i \neq j$ along a common edge or a vertex. In particular $s_i \cap s_j$ is a subsimplex of both. It is known that a simplicial 2-complex $\mathcal{C}$ has a geometric realization as a subset of the Euclidean 5-space in which each s_i is a closed triangular plane region. We refer the reader to the texts [3], [4], [6], [7], and [8] for more information.

We will study the properties of complexes in terms of the subcomplexes obtained by removal of a subset of its 2-simplexes. A 2-simplex $s \in \mathcal{C}$ is called an *external* simplex of $\mathcal{C}$ if s has at least one proper face which not shared with any other simplex in $\mathcal{C}$; otherwise s is called *internal*. Given a 2-complex $\mathcal{C}$ and a 2-simplex $s_i \in \mathcal{C}$, we denote by $\mathcal{C} - s_i$ the 2-complex obtained by restricting the given identifications defining $\mathcal{C}$ to $\{s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n\}$. We say that $\mathcal{C} - s_i$ is obtained from $\mathcal{C}$ by *removing* (*erasing*) the internal (external) simplex s_i. If $\mathcal{C}'$ is obtained from $\mathcal{C}$ by erasing an external simplex of $\mathcal{C}$, then we denote this by $\mathcal{C} \rightsquigarrow \mathcal{C}'$. More generally if two complexes $\mathcal{C}$ and $\mathcal{C}_m$ are related by a sequence of erasures of external simplexes $\mathcal{C} \rightsquigarrow \mathcal{C}_1 \rightsquigarrow \cdots \rightsquigarrow \mathcal{C}_m$, then we denote this also by $\mathcal{C} \rightsquigarrow \mathcal{C}_m$. We say that the complex $\mathcal{C}$ is *erasable* (or *nullable*) if $\mathcal{C} \rightsquigarrow \phi$. As examples, the segment of a *pipe* in Figure 1 (a) is erasable. However the triangulation of the 2-dimensional sphere

*Section 2 of this Technical Report is intended for a non-CS audience.

S^2 in Figure 1 (b), and the complex (c) are not erasable since these have no external simplexes. Note that the operation $\rightsquigarrow$ is not a topological invariant, since it can destroy the fundamental group.

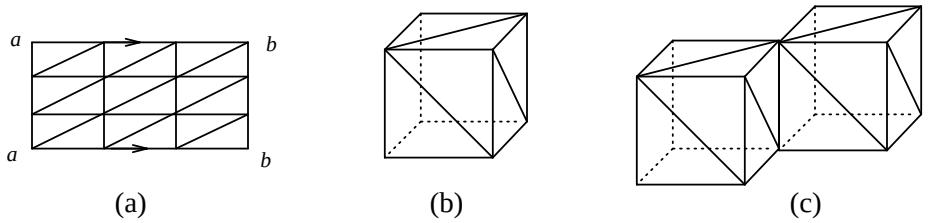

Figure 1: (a) A pipe segment, (b) The sphere S^2, (c) Two spheres with a common edge.

Given a 2-complex $\mathcal{C}$ we define $er(\mathcal{C})$ to be the minimum number of internal 2-simplexes that need to be removed from $\mathcal{C}$ so that the resulting complex is erasable. For example, for the complexes in Figure 1 (a), (b), and (c), we have $er(\mathcal{C}) = 0, 1,$ and 2, respectively. The quantity $er(\mathcal{C})$ also gives the minimum number of internal 2-simplexes that need to be removed from $\mathcal{C}$ so that the resulting complex can be *collapsed* to a 1-dimensional subcomplex (see [3] for the notion of collapsing of complexes, and Appendix II for a proof of this fact). If $\mathcal{C}$ collapses to a d or lower dimensional subcomplex, this is denoted by $\mathcal{C} \searrow d$.

In this paper we show that the problem of computing $er(\mathcal{C})$ for a given 2-complex is intractable. More precisely we show that the decision problem defined below is $\mathcal{NP}$-complete.

Erasability Problem :
INSTANCE : A pair $(\mathcal{C}, k)$ where $\mathcal{C}$ is a 2-complex and k is a non-negative integer.
QUESTION : Is $er(\mathcal{C}) = k$? i.e., does $\mathcal{C}$ contain a subset $\mathcal{K}$ of 2-simplexes of cardinality k such that $\mathcal{C} - \mathcal{K} \rightsquigarrow \phi$?

The Erasability Problem can be paraphrased as the following decision problem involving collapsibility

Collapsibility Problem :
INSTANCE : A pair $(\mathcal{C}, k)$ where $\mathcal{C}$ is a 2-complex and k is a non-negative integer.
QUESTION : Is $er(\mathcal{C}) = k$? i.e., does $\mathcal{C}$ contain a subset $\mathcal{K}$ of 2-simplexes of cardinality k such that $\mathcal{C} - \mathcal{K} \searrow 1$?

The outline of this paper is as follows. In Section 2, we give an introduction to deterministic and nondeterministic algorithms and the theory of $\mathcal{NP}$-completeness. In Section 3, we present the reduction for proving the $\mathcal{NP}$-completeness of the Erasability Problem. In Section 4 we conclude by considering the relationship of $er(\mathcal{C})$ with Betti numbers and the computational complexity of these quantities for more general complexes, and the special case of graphs.

2 Nondeterministic Algorithms and $\mathcal{NP}$-completeness

We begin our discussion of $\mathcal{NP}$-complete problems by formally defining the optimization problem known as the traveling salesperson problem.

Traveling Salesperson Problem (TSP) :
INSTANCE : A finite set $C = \{c_1, c_2, ..., c_n\}$ of *cities*, and a *distance* $d(c_i, c_j) \in Z^+$ for each (ordered) pair of cities $c_i, c_j \in C$, where Z^+ denotes the positive integers.
PROBLEM : Find a least length *tour* of all the cities in C, that is, an ordering $< c_{\pi(1)}, c_{\pi(2)}, ..., c_{\pi(n)} >$ of C such that $(\sum_{i=1}^{n-1} d(c_{\pi(i)}, c_{\pi(i+1)})) + d(c_{\pi(n)}, c_{\pi(1)})$ is minimum.

The theory of $\mathcal{NP}$-completeness was developed for decision problems rather than for optimization problems. The corresponding decision version of the TSP optimization problem includes an additional input B (positive integer) and the question then is to decide whether or not there is a tour of length at most B. Clearly, any algorithm that solves the TSP optimization problem can be easily adapted to solve the corresponding decision problem, but the converse does not seem to be true. That is why we say that the TSP optimization problem is computationally harder than its corresponding decision problem. Since in the theory of $\mathcal{NP}$-completeness one establishes that the decision version of the TSP problem is computationally intractable, the above observation implies that the optimization problem is also computationally intractable.

We denote by $\mathcal{P}$ the set of all decision problems solvable by a deterministic algorithm in polynomial time, i.e., the time (or the elementary number of steps) required to solve every problem instance is bounded above by a polynomial function of the length of the input. For uniformity we assume that all problem instances are coded in binary by some translation process, and the length of the input refers to the number of bits in this representation.

Given an instance of the TSP decision problem the corresponding *verification* problem consists of determining whether a given sequence of n integers represents a tour with length at most B in some given problem instance. A nondeterministic algorithm for the TSP decision problem consists of two stages: a guessing stage and a verifying stage. The guessing stage finds a tour with length at most B when one exists, otherwise it stops or generates a a sequence of n integers that may or may not represent a tour. The subsequent verification step checks whether or not the sequence represents a tour of length at most B. The time complexity of a nondeterministic algorithm takes into account only the time required to solve the verification problem when the decision problem has a YES answer.

The set of decision problems $\mathcal{NP}$ consists of all decision problems solvable by a nondeterministic algorithm in polynomial time with respect to the input length. Obviously $\mathcal{P} \subseteq \mathcal{NP}$, since the existence of a polynomial time solution implies polynomial time verification. However, the converse does not seem true, and at present it is not known whether or not $\mathcal{P} = \mathcal{NP}$. The conjecture is that $\mathcal{P} \neq \mathcal{NP}$.

Next we define the class of $\mathcal{NP}$-complete problems. We start with the definition of the *satisfiability* (SAT) problem. Let $X = \{x_1, x_2, ..., x_n\}$ be a set of Boolean variables. A *truth assignment* for X is a function $t : X \rightarrow \{T, F\}$. If $t(x) = T$ we say that x is *true* under t; if $t(x) = F$ we say that x is *false*. If x is a variable in X, then x and its negation $\overline{x}$ are called *literals* over X. The literal x is true under t if and only if the variable x is true under t; the literal $\overline{x}$ is true if and only if the variable x is false. A truth assignment t for X can be represented an n-tuple of symbols $t(X) = (x_1^{t(x_1)}, x_2^{t(x_2)}, \ldots, x_n^{t(x_n)})$.

A *clause* over X is a set of literals over X, such as $\{x_1, \overline{x}_3, x_8\}$. It represents the disjunction of

those literals and is satisfiable by a truth assignment if and only if at least one of its members is set to true under that assignment. The clause above will be satisfied by t unless $t(x_1) = F$, $t(x_3) = T$, and $t(x_8) = F$. A collection C of clauses over X is *satisfiable* if and only if there exists some truth assignment for X that simultaneously satisfies all of the clauses in C. Such a truth assignment is called a *satisfying truth assignment* for C. The problem SAT is defined as follows.

Satisfiability Problem (SAT) :
INSTANCE : A pair (X, C) where X is a finite set of variables and C is a collection of clauses over X.
QUESTION : Is there a satisfying truth assignment for C?

For example, the pair (X, C) with $X = \{x_1, x_2\}$ and $C = \{\{x_1, \overline{x}_2\}, \{\overline{x}_1, x_2\}\}$ provides an instance of SAT for which the answer is YES. A satisfying truth assignment is given by $t(X) = (x_1^T, x_2^T)$. On the other hand, replacing C by $C' = \{\{x_1, x_2\}, \{x_1, \overline{x}_2\}, \{\overline{x}_1\}\}$ yields an instance of SAT for which the answer is NO; C' is not satisfiable.

We say that decision problem Q_1 polynomially reduces to decision problem Q_2 if there exists a deterministic algorithm to convert a given instance q_1 of Q_1 to an instance q_2 of Q_2 using only polynomial time in the length of q_1, such that q_2 is a YES instance if and only if q_1 is a YES instance. Polynomial time reducibility is a transitive relation expressed symbolically by $Q_1 \, \alpha \, Q_2$. At the heart of the question of whether $\mathcal{P} = \mathcal{NP}$ is the notion of *hardest* problems for $\mathcal{NP}$: a decision problem Q is called $\mathcal{NP}$*-hard* if $Q' \, \alpha \, Q$ for every $Q' \in \mathcal{NP}$; and $\mathcal{NP}$*-complete* if Q is $\mathcal{NP}$-hard, and $Q \in \mathcal{NP}$. Thus if Q is an $\mathcal{NP}$-complete problem which has a deterministic polynomial time algorithm, this would imply that $\mathcal{P} = \mathcal{NP}$. Cook's theorem states that SAT is an $\mathcal{NP}$-complete problem (see [2]). The class of $\mathcal{NP}$-complete problems can then be characterized as the set of all decision problems $Q \in \mathcal{NP}$ such that SAT $\alpha \, Q$. For example it can be shown that SAT α TSP. Since TSP is trivially in $\mathcal{NP}$, the decision version of TSP is also $\mathcal{NP}$-complete. The class of $\mathcal{NP}$-complete problems is very rich. We refer the reader to [1], [5], and especially to [2].

Since it is widely believed that $\mathcal{P} \neq \mathcal{NP}$, $\mathcal{NP}$-hard and $\mathcal{NP}$-complete problems are referred to as *computationally intractable* problems. An optimization problem whose corresponding decision problem is intractable is therefore also intractable, and unlikely to be solvable by a (deterministic) polynomial time algorithm.

3 $\mathcal{NP}$-completeness of the Erasability Problem

To establish our intractability results, we present a polynomial time reduction from a variant of SAT called (3,2)-SAT to the Erasability Problem. The problem (3,2)-SAT is formally defined below, and shown to be $\mathcal{NP}$-complete in Appendix I.

(3,2)-SAT :
INSTANCE : An instance (X, C) of SAT in which each clause contains either two or three distinct literals, and each literal appears in exactly two distinct clauses.
QUESTION : Is there a satisfying truth assignment for C?

An example of (3,2)-SAT is the pair $(\{x_1, x_2, x_3\}, C)$ where C consists of the clauses

$$C_1 = \{x_1, x_2, \overline{x}_3\},\ \ C_2 = \{x_1, x_3\},\ \ C_3 = \{\overline{x}_1, x_2\},\ \ C_4 = \{\overline{x}_2, x_3\},\ \ C_5 = \{\overline{x}_1, \overline{x}_2, \overline{x}_3\}\ .$$

First we consider certain complexes that we shall use in our reductions. The simplest of these are the Klein bottle Figure 2 (a), and the complex called an AND gate as shown in Figure 2 (b). The complex in Figure 2 (b) serves the function of a logical AND gate in the following sense. The vertical portion of the complex indicated by Z can be erased (without removing a simplex from Z) only when both pipes X and Y are erased. However if only X or only Y is erased, Z is left intact. Note that the interior of the common circle that is shared by X, Y, and Z is not part of the AND gate in Figure 2 (b), just as the interior of the top circle that appears in the Klein bottle in Figure 2 (a) is not part of the Klein bottle.

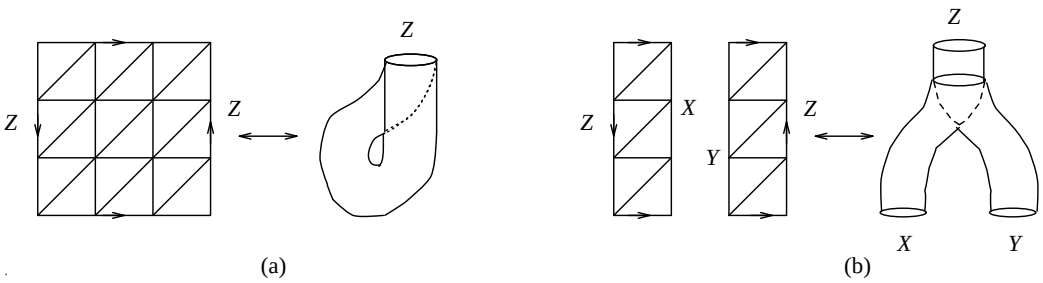

Figure 2: (a) The Klein bottle, (b) An AND gate: Z is erasable if both X and Y are erasable.

The Klein bottle is a nonorientable surface which has no embedding in 3-dimensional space. It is clear that it is necessary and sufficient to remove a single 2-simplex from the Klein bottle to make it erasable. Thus $er(\textit{Klein bottle}) = 1$.

We now construct a special complex in 4-dimensional space which will be denoted by $B(x)$. It appears in Figure 3. $B(x) = x^T \cup x^F \cup P^T \cup P^F$ consists of two Klein bottles x^T and x^F, identified by removing a simplex from each at C and gluing the pieces that remain along their boundaries. Thus this portion of $B(x)$ is the so called connected sum of two Klein bottles ([8], p. 38). $B(x)$ also contains two pipes of simplexes: P^T connecting C to U, and P^F connecting C to D. Locally, three pipes of simplexes come together at U in x^T and in D in x^F, in the form of the AND gate given in Figure 2. The local picture at C is that of two Klein bottles as in Figure 2, identified at Z. We fix a simplex E^T on the pipe from C to U, and a simplex E^F on the pipe from C to D, to be used later in our construction. The complex $B(x)$ has the following property:

Lemma 1

1. *If $s \in x^T$, then $B(x) - s \rightsquigarrow x^F \cup P^F$. Thus the removal of a 2-simplex from x^T erases $x^T \cup P^T$. Similarly, if $s \in x^F$, then $B(x) - s \rightsquigarrow x^T \cup P^T$.*

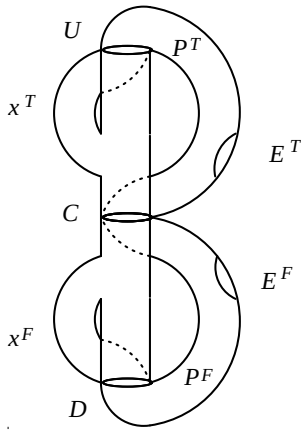

Figure 3: A building block $B(x)$.

2. *$B(x)$ can only be erased by removing at least two 2-simplexes, one of which must be in x^T or in x^F. In particular $er(B(x)) = 2$.*

Proof Part 1 is trivial. To prove part 2, suppose we remove a single simplex $s \in x^T \cup P^T$ from $B(x)$. If $s \in x^T$, then $B(x) - s \rightsquigarrow x^F \cup P^F$. If $s \in P^T$, then the removal of p only erases P^T, and $B(x) - s \rightsquigarrow x^T \cup x^F \cup P^F$. In either case $B(x) - s \not\rightsquigarrow \phi$. The situation for a single simplex removed from $x^F \cup P^F$ is similar. It follows that $er(B(x)) \geq 2$. Now even if the pipes P^T and P^F are completely removed from $B(x)$ (say by removing the 2-simplexes E^T and E^F), we would be left with two Klein bottles $x^T \cup x^F$ joined at C. This remaining configuration still requires the removal of a simplex to be erasable. On the other hand, given two 2-simplexes r and s with $r \in x^T$ and $s \in x^F \cup P^F$, we have $B(x) - r - s \rightsquigarrow \phi$. Similarly for any two 2-simplexes r and s with $r \in x^T \cup P^T$ and $s \in x^F$, we have $B(x) - r - s \rightsquigarrow \phi$. □

Another basic building block that we shall make use of in addition to $B(x)$ is given in Figure 4. It is

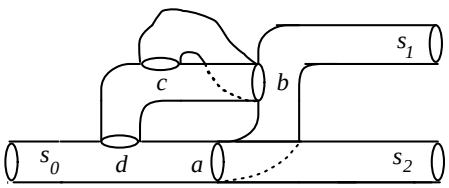

Figure 4: Another building block: One-way AND gate G.

also constructed in 4-dimensional space by gluing pipes of 2-simplexes in the following manner: the junctures at a and b shown in Figure 4 are exactly as in the AND gate of Figure 2 where the ends of three pipes are identified at a circle. In particular, the interiors of the circles at these points are not part of the structure. The two junctures at points marked c and d are obtained by removing a 2-simplex from the bottom pipe and identifying the end of the pipe on the top with the boundary of the hole left by the removed simplex. We refer to this structure as a *one-way AND gate* and denote it by G.

Lemma 2 *Suppose s_0, s_1, and s_2 are three 2-simplexes on a one-way AND gate G, as shown in Figure 4. Then*

1. $G - s_0 \rightsquigarrow \phi$,

2. $G - s_1 \not\rightsquigarrow \phi$, *and* $G - s_2 \not\rightsquigarrow \phi$, *but* $G - s_1 - s_2 \rightsquigarrow \phi$.

Proof To prove part 1, we note that the removal of s_0 creates a sequence of external 2-simplexes that first erases the portion of the bottom horizontal pipe up to a. After this d and subsequently c and b become external 2-simplexes, first erasing the two pipes sharing c, and then erasing s_1 through b. After the removal of s_1, the rest of the pipe on the top can be erased down to a. Following this, a becomes external and the rest of the bottom pipe from a to s_2 can also be erased.

The removal of s_1 from G only erases the pipe that contains s_1 from G, and the remaining configuration has no external 2-simplexes, and therefore it is not erasable. Similarly, the removal of s_2 from G erases only the right hand side portion of the pipe containing s_2 to a, and the remaining configuration G' has no external 2-simplexes. Now removing s_1 from G' creates a sequence of external 2-simplexes that first erases the pipe from s_1 to s_0 through a. Subsequently d becomes external, and the remaining 2-simplexes can be erased. Therefore part 2 holds. □

To paraphrase lemma 2, the removal of s_0 makes the whole configuration G erasable and the removal of both s_1 and s_2 makes G erasable. However the removal of a single one of s_1 and s_2 alone is not sufficient to erase the remaining set of 2-simplexes.

Now we construct what we refer to as a truth-testing component $C(x)$ by putting together $B(x)$ and two copies of the one-way AND gate G. To do this we first remove the 2-simplexes E^T and E^F from $B(x)$ and sew the ends of two disjoint copies of G to these holes. The resulting structure is shown in Figure 5. Combining what we know about $B(x)$ and G from lemmas 1 and 2, we obtain

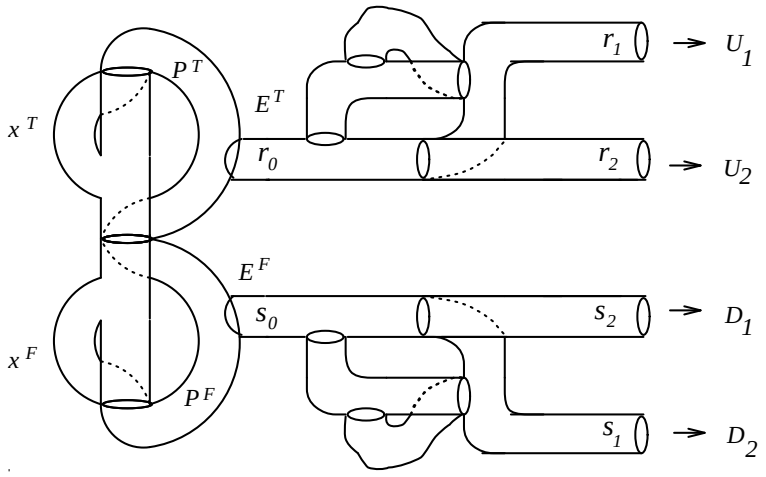

Figure 5: The truth-testing component $C(x)$ for a variable x.

Proposition 1

1. *$C(x)$ can only be erased by removing at least two 2-simplexes one of which must be in x^T or in x^F. Thus $er(C(x)) = 2$.*

2. *$C(x) - r_1 - r_2 - s_1 - s_2 \rightsquigarrow x^T \cup x^F$. In particular $C(x) - r_1 - r_2 - s_1 - s_2 \not\rightsquigarrow \phi$.*

By proposition 1, even if all four 2-simplexes r_1, r_2, s_1, s_2 are removed from $C(x)$, to erase the rest requires the removal of one more simplex from $x^T \cup x^F$.

Now suppose that we are given an instance $\mathcal{S}$ of (3,2)-SAT with Boolean variables $x_1, x_2, \ldots, x_n$ and clauses $C_1, C_2, \ldots, C_k$. We construct an instance of the Erasability Problem in 4-dimensional space first by constructing n disjoint copies $C(x_1), C(x_2), \ldots, C(x_n)$ of the truth-testing component $C(x)$ shown in Figure 5, using a constant number of 2-simplexes per component. Schematically, each $C(x_i)$ is of the form shown in Figure 6 (a). Next, for each clause C_i of $\mathcal{S}$ with three literals, we take a set of 2-simplexes in the form of a sphere with three holes (i.e. three disjoint 2-simplexes removed), and three pipes of 2-simplexes attached to the resulting holes. This clause structure is shown in Figure 6 (b). For each clause C_i of $\mathcal{S}$ with two literals, we remove two 2-simplexes from a sphere an attach two pipes of 2-simplexes to the resulting holes, as shown in Figure 6 (c).

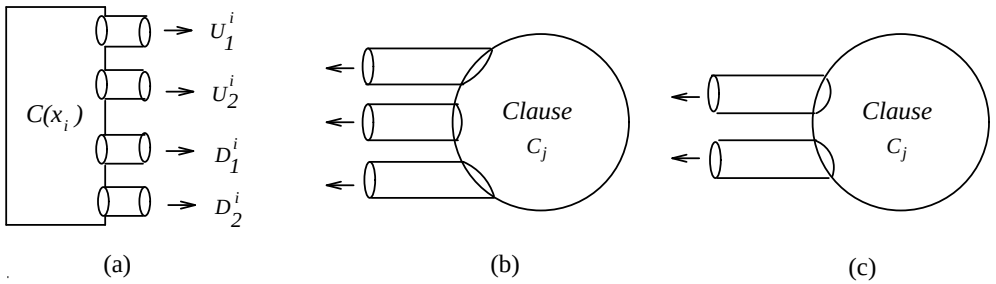

Figure 6: (a) Component for variable x_i, (b), (c) Possible clause structures for clause C_j.

The construction of all clause structures corresponding to the clauses in $\mathcal{S}$ can be done in three dimensions using a constant number of 2-simplexes for each given clause. After this, each of the pipes U_1^i and U_2^i are extended and attached to the pipes emanating from the clause structures corresponding to the the two clauses that contain the literal x_i. Similarly, each of the pipes D_1^i and D_2^i are extended and attached to the pipes emanating from the clause structures that contain the literal $\overline{x}_i$. Doing this for every i results in a set of 2-simplexes in 4-dimensional space that we will denote by $\mathcal{K}_{\mathcal{S}}$. Next we argue that the pipe connections from the truth-satisfying components to clause structures can be done by using a polynomial number of 2-simplexes in n. To this end we note that by reserving four levels of pipes for each $C(x_i)$, all of the connections can be done in three dimensional space by using about $4n$ levels of pipes. Therefore all of the connections between the truth-satisfying components and clause structures can be contained in cubic set of lattice points in 3-dimensional space, whose sides have lengths which are polynomially bounded as a function of n. Since the total number of 2-simplexes in such a set itself is a polynomial in n, the construction uses only polynomially many

2-simplexes in n. It follows from the construction that the transformation that takes $\mathcal{S}$ to $\mathcal{K}_{\mathcal{S}}$ is polynomial time computable. The following example illustrates our transformation.

Example
Suppose the instance of (3,2)-SAT is $\mathcal{S} = (\{x_1, x_2, x_3\}, C)$, where C consists of the five clauses given below

$$C_1 = \{x_1, x_2, \overline{x}_3\}, \quad C_2 = \{x_1, x_3\}, \quad C_3 = \{\overline{x}_1, x_2\}, \quad C_4 = \{\overline{x}_2, x_3\}, \quad C_5 = \{\overline{x}_1, \overline{x}_2, \overline{x}_3\} \ .$$

The corresponding set of 2-simplexes $\mathcal{K}_{\mathcal{S}}$ is given in Figure 7.

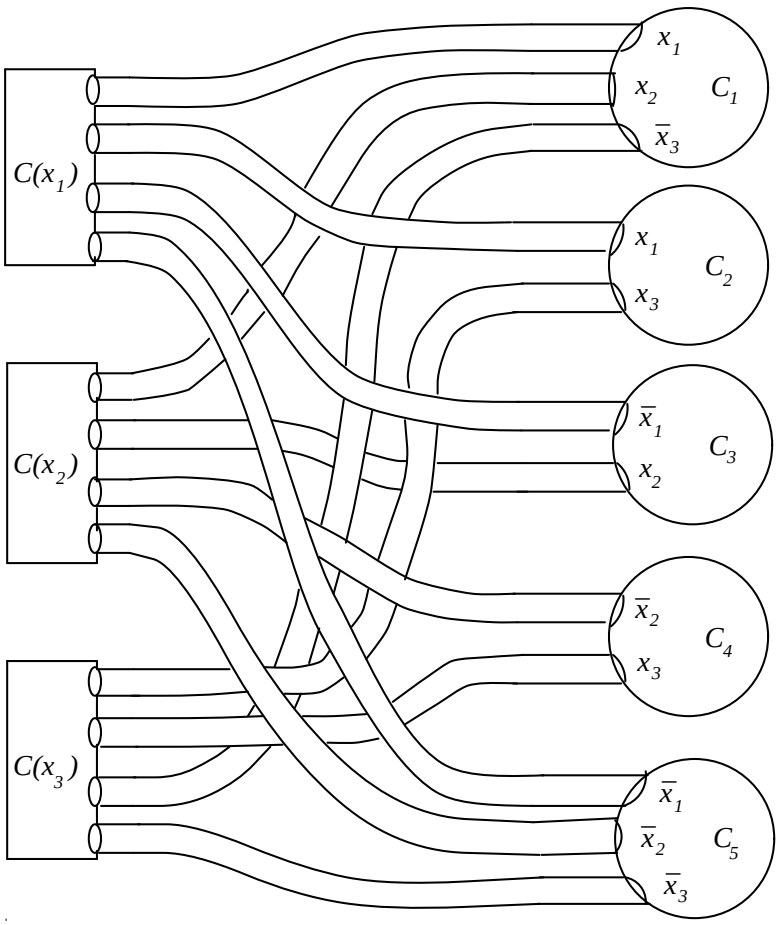

Figure 7: *Example* : $\mathcal{K}_{\mathcal{S}}$. The reduction from (3,2)-SAT.

Theorem 1 *(3,2)-SAT is polynomial time reducible to the Erasability Problem. More precisely, let* $\mathcal{S} = (\{x_1, \ldots, x_n\}, C)$ *be an instance of (3,2)-SAT, and let* $\mathcal{K}_{\mathcal{S}}$ *be the set of 2-simplexes constructed as above. Then there is a satisfying truth assignment for* C *if and only if* $er(\mathcal{K}_{\mathcal{S}}) = n$.

Proof We first show that if C is satisfiable, then the answer to the instance of the Erasability Problem $(\mathcal{K}_{\mathcal{S}}, n)$ is YES. Suppose $C = \{C_1, \ldots, C_k\}$ and $t(X) = (x_1^{v_1}, x_2^{v_2}, \ldots, x_n^{v_n})$ is a satisfying truth assignment for C, where each $v_i = t(x_i) \in \{T, F\}$. We show that in this case

1. $er(\mathcal{K}_\mathcal{S}) \geq n$,

2. The removal of one simplex from each of the Klein bottles $x_i^{v_i}$ in $\mathcal{K}_\mathcal{S}$ makes $\mathcal{K}_\mathcal{S}$ erasable.

These two facts imply that $er(\mathcal{K}_\mathcal{S}) = n$. The Klein bottles $x_i^{v_i}$ are parts of the $B(x_i)$ blocks in each of the truth-satisfying components $C(x_i)$ that are used in the construction of $\mathcal{K}_\mathcal{S}$. The fact that $er(\mathcal{K}_\mathcal{S}) \geq n$ is an immediate consequence of lemma 1, which forces the removal of at least one simplex from each $x_i^T \cup x_i^F$ just to make $B(x_i)$ erasable. To prove part 2, we will show that the removal of a simplex from $x_i^{v_i}$ for each i is sufficient to erase $C(x_i)$, the clause structures containing the literal x_i, the clause structures containing the literal $\overline{x}_i$, as well as the pipes connecting these in $\mathcal{K}_\mathcal{S}$. Without loss of generality assume that $v_i = T$ and $x = x_i$. By lemma 1 the removal of a simplex $s \in x^T$ erases $x^T \cup P^T$. Therefore the simplex r_0 can now be erased through E^T. By lemma 2, the one-way AND gate G sharing E^T and subsequently the 2-simplexes r_1 and r_2 become erasable. Therefore through the pipes on which r_1 and r_2 are located, the two clause structures containing the literal x can be erased. Since this is true for every i, all of the clause structures are erasable after the removal of a simplex from each Klein bottle $x_i^{v_i}$: by first erasing the one-way AND gate attached to $B(x_i)$ at $E_i^{v_i}$, erasing the pipes attaching the $C(x_i)$ to the relevant clause structures, and finally the clause structures themselves. After the removal of all clause structures, both 2-simplexes s_1 and s_2 on $C(x)$ become erasable through the pipes connecting $C(x)$ to the clause structures containing $\overline{x}_i$. By lemma 2 this is sufficient to erase s_0. Using lemma 1 one more time, we conclude that all of $C(x)$ is erasable. Therefore all $C(x_i)$ are erasable and the answer to the Erasability Problem instance $(\mathcal{K}_\mathcal{S}, n)$ is YES.

Next we show that a YES answer to the Erasability Problem $(\mathcal{K}_\mathcal{S}, n)$ results in a satisfying truth assignment for C. Let $p_1, p_2, \ldots, p_n$ be n distinct 2-simplexes such that $\mathcal{K}_\mathcal{S} - p_1 - p_2 - \cdots - p_n \rightsquigarrow \phi$. From lemma 1, any set of 2-simplexes whose removal erases $\mathcal{K}_\mathcal{S}$ must include at least one simplex from each $x_i^T \cup x_i^F$. Reordering if necessary, we can assume that $p_i \in x_i^T \cup x_i^F$ for $i = 1, 2, \ldots, n$. Define $t(x_i) = T$ if $p_i \in x_i^T$ and $t(x_i) = F$ if $p_i \in x_i^F$. Then $t(X) = (x_1^{t(x_1)}, x_2^{t(x_2)}, \ldots, x_n^{t(x_n)})$ is a satisfying truth assignment for C. To see this, note that a clause structure corresponding to a clause $C_i = \{u_{i_1}, u_{i_2}, u_{i_3}\}$ (where each symbol is a literal) is connected only to the components $C(x_{i_1})$, $C(x_{i_2})$, and $C(x_{i_3})$ in $\mathcal{K}_\mathcal{S}$. Therefore the clause structure C_i cannot be erased unless at least one of $C(x_{i_1})$, $C(x_{i_2})$, and $C(x_{i_3})$ is erased. This means that in the truth assignment t, at least one of the literals $u_{i_1}, u_{i_2}, u_{i_3}$ is set to true. □

Note that in the example of the reduction given in Figure 7, $n = 3$ and the instance of the Erasability Problem to be solved is $(\mathcal{K}_\mathcal{S}, 3)$. The answer to this instance is YES, since $t(X) = (x_1^F, x_2^T, x_3^T)$ is a satisfying truth assignment for the given instance $calS$ of the (3,2)-SAT problem.

Since the Erasability Problem is trivially in $\mathcal{NP}$, we obtain

Corollary 1 *The Erasability Problem is $\mathcal{NP}$-complete.*

We prove in Appendix II that the number of internal 2-simplexes that needs to be removed from $\mathcal{C}$ so that the resulting complex is collapsible to a 1-dimensional complex is also given by $er(\mathcal{C})$. Therefore the Collapsibility Problem defined in Section 1 is also $\mathcal{NP}$-complete.

4 Remarks

The Betti numbers β_i for $0 \leq i \leq d$ of a d-dimensional simplicial complex $\mathcal{C}$ are topological invariants related to high dimensional connectivity properties of $\mathcal{C}$. The Betti number β_0 is the number of connected components of $\mathcal{C}$, and intuitively, β_i is the number of "i-dimensional holes" in $\mathcal{C}$ (see [4]). Since $er(\mathcal{C})$ appears to count the number of three dimensional regions enclosed by $\mathcal{C}$, we may expect some relationship between $er(\mathcal{C})$ and the Betti numbers of $\mathcal{C}$. To this end, we first consider 1-dimensional simplicial complexes, also referred to as *graphs*. For a connected graph $\mathcal{G}$ with n vertices and e edges (1-simplexes), the 1-dimensional Betti number has a simple interpretation: β_1 is the maximum number of linearly independent elementary cycles in $\mathcal{G}$ ([6] p. 71). Equivalently, β_1 is the dimension of the circuit space of $\mathcal{G}$. For graphs, β_1 and the rank r of the incidence matrix of $\mathcal{G}$ are related by $\beta_1 = e - r$, and thus β_1 can be found by a rank computation in polynomial time. Actually the 1-dimensional Betti number for graphs can be expressed explicitly by $\beta_1 = e - n + 1$ as a consequence of the the Euler-Poincaré relation: for an arbitrary d-dimensional complex, this relation is

$$\chi(\mathcal{C}) = \sum_{i=0}^{n} (-1)^i \alpha_i = \sum_{i=0}^{n} (-1)^i \beta_i \ ,$$

where α_i be the number of i-simplexes of $\mathcal{C}$. This common value is the Euler characteristic of $\mathcal{C}$.

It is known that all of the Betti numbers $\beta_0, \beta_1, \ldots, \beta_d$ of a general d-dimensional complex $\mathcal{C}$ can be computed from the ranks of the incidence matrices relating the i-dimensional simplexes of $\mathcal{C}$ to its $(i-1)$-dimensional simplexes, $1 \leq i \leq n$. Consequently these invariants can be computed in polynomial time in the total number of simplexes of $\mathcal{C}$ (see [8], [4], [6]).

It is interesting to note that when we restrict the operation of removal and erasure to 1-dimensional simplexes, then the notions of erasability and collapsibility coincide. If $\mathcal{G}$ is a graph then $er(\mathcal{G}) = \beta_1$. Therefore the Erasability Problem for graphs is in $\mathcal{P}$. However the intractability of the Erasability Problem for 2-complexes implies that the generalization of this problem is $\mathcal{NP}$-complete for higher dimensional complexes as well. Furthermore, since the Betti numbers can be computed in polynomial time, this also implies that there can be no simple formula that relates $er(\mathcal{C})$ to the Betti numbers of the complex. More precisely, unless $\mathcal{P} = \mathcal{NP}$, there can be no polynomial time computable function $f(x_0, x_1, \ldots, x_d)$ for which $er(\mathcal{C}) = f(\beta_0, \beta_1, \ldots, \beta_d)$.

Finally we note that the Erasability Problem is solvable in polynomial time for constant k, since we can generate all l-element subsets $\mathcal{K}$ of internal simplexes of $\mathcal{C}$ for $l \leq k$, and for each subset $\mathcal{K}$ check in polynomial time whether $\mathcal{C} - \mathcal{K} \rightsquigarrow \phi$. The total number of subsets to be checked is of order n^{k+1}, and thus the resulting algorithm is in $\mathcal{P}$.

References

[1] Cook, S. A., The complexity of theorem proving procedures, *Proceedings of the 3rd Annual ACM Symposium on the Theory of Computing*, Association for Computing Machinery, New York (1971), pp. 151–158.

[2] Garey, M. J., and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, New York, 1979.

[3] Glaser, L. C., *Geometrical Combinatorial Topology*, Vol. I, Van Nostrand Reinhold Company, New York, 1970.

[4] Hocking, J. G., and G. S. Young, *Topology*, Addison-Wesley Publishing Company, Reading MA, 1961.

[5] Karp, R. M., Reducibility among combinatorial problems, In R. E. Miller and J. W. Thatcher (eds.), *Complexity of Computer Computations*, Plenum Press (1972), pp. 85–103.

[6] Lefschetz, S., *Introduction to Topology*, Princeton University Press, New Jersey, 1949.

[7] Massey, W. S., *Algebraic Topology: An Introduction*, Harcourt-Brace, New York, 1967.

[8] Munkres, J. R., *Elements of Algebraic Topology*, Addison-Wesley Publishing Company, Menlo Park, 1984.

5 Appendix I

The following version of SAT which we will call R-SAT is $\mathcal{NP}$-complete ([2], p. 259):

R-SAT :
INSTANCE : An instance (X, C) of SAT in which each clause contains at most three literals, and there are at most three clauses in C that contain either x or $\overline{x}$.
QUESTION : Is there a satisfying truth assignment for C?

We show that R-SAT α (3,2)-SAT. Given an instance (X, C) of the R-SAT problem we construct an instance (X'', C'') of the (3,2)-SAT problem in two stages as follows. First we convert the instance (X, C) of R-SAT to an instance (X', C') of R-SAT which does not contain clauses with exactly one literal, or two literals of the same variable; and no variable nor its complement appear in exactly three clauses. In other words, each variable and its complement appear in at least one clause and at most two clauses, and each clause has two or three literals. It is clear that (X, C) can be transformed into (X', C') in polynomial time. Instance (X'', C'') of (3,2)-SAT is obtained from (X', C') by introducing additional variables and clauses. For a literal x that appears in exactly one clause in C' we add a new variable x_a, and the clauses $\{x, x_a, \overline{x}_a\}$, and $\{x_a, \overline{x}_a\}$. Applying this rule for every literal that appears in only one clause, we make sure that all literals appear in exactly two clauses in C''. It is simple to show that (X'', C'') is satisfiable if and only if (X', C') is satisfiable, and the whole reduction can be done in polynomial time in the size of the input (X, C).

6 Appendix II

Suppose s is an external simplex of a 2-complex $\mathcal{C}$. Assume $s = abc$ with vertices (a, b, c) and 1-simplexes (ab, ac, bc). Assume that the face bc is not shared by any other 2-simplex of $\mathcal{C}$. The

operation of going from $\mathcal{C}$ to $\mathcal{C}' = \mathcal{C} - s + ab + ac$ (where $+$ is union) is called an *elementary collapsing*, denoted by $\mathcal{C} \searrow \mathcal{C}'$ (see [3], p. 49). Geometrically, the complex $\mathcal{C}'$ is a deformation retract of $\mathcal{C}$ obtained by "pushing in" on the free face bc of s. If ab or ac becomes an external 1-simplex in $\mathcal{C}'$ after collapsing bc, (i.e. vertex b or vertex c is not shared with any other 1-simplex in $\mathcal{C}'$) then we can collapse these as well, so that no external 1-simplexes are left in the resulting complex. Here we will use the term elementary collapsing as this composite operation of collapsing (i.e. a 2-complex followed by at most two 1-complex collapsings), and use the symbol $\mathcal{C} \searrow \mathcal{C}'$ for it. If there is a finite sequence of elementary collapsings $\mathcal{C} \searrow \mathcal{C}_1 \searrow \cdots \searrow \mathcal{C}_k$ then we say that $\mathcal{C}$ *collapses* to $\mathcal{C}_k$, written $\mathcal{C} \searrow \mathcal{C}_k$. If $\mathcal{C}$ collapses to a subcomplex of dimension d or less, then this is denoted by $\mathcal{C} \searrow d$. A complex $\mathcal{C}$ with $\mathcal{C} \searrow 0$ is called *collapsible* [3]. As examples, the complex in Figure 1 (a) collapses to the circle S^1 and thus $\mathcal{C} \searrow 1$. The complexes in Figure 1 (b) and (c) do not collapse to any of their subcomplexes.

Proposition 2 *Let $co(\mathcal{C})$ denote the minimum number of 2-simplexes $\mathcal{K}$ that need to be removed from $\mathcal{C}$ so that $\mathcal{C} - \mathcal{K} \searrow 1$. Then $co(\mathcal{C}) = er(\mathcal{C})$.*

Proof First we compare the effects of the operations of collapsing and erasing of an external 2-simplex. In Figure 8, the dark subsimplexes marked in column one are assumed to be shared with other 1 or 2-dimensional simplexes in $\mathcal{C}$. The result of erasing and collapsing the external 2-simplex in column one are given in columns two and three, respectively.

Suppose $\mathcal{C}$ has n 2-simplexes, and assume $k = er(\mathcal{C})$. Suppose $\mathcal{K}$ is a cardinality k set of 2-simplexes of $\mathcal{C}$ such that $\mathcal{C} - \mathcal{K} \rightsquigarrow \phi$. Let $\mathcal{C}_1 = \mathcal{C} - \mathcal{K}$. Since $\mathcal{C}_1 \rightsquigarrow \phi$, the 2-simplexes of $\mathcal{C}_1$ can be ordered as $e_1, e_2, \ldots, e_m$ with $k+m = n$, such that e_1 is an external simplex of $\mathcal{C}_1$ and if $\mathcal{C}_{i+1} = \mathcal{C}_i - e_i$ then e_{i+1} is an external simplex of $\mathcal{C}_{i+1}$ for $i = 1, 2, \ldots, m-1$. In other words, we have a sequence

$$\mathcal{C}_1 \rightsquigarrow \mathcal{C}_2 \rightsquigarrow \cdots \rightsquigarrow \mathcal{C}_m \rightsquigarrow \mathcal{C}_{m+1} \ , \tag{1}$$

where $\mathcal{C}_{m+1} = \phi$ and the external 2-simplex e_i of $\mathcal{C}_i$ is erased in going from $\mathcal{C}_i$ to $\mathcal{C}_{i+1}$. Let $\mathcal{D}_1 = \mathcal{C}_1$. We claim that $\mathcal{D}_1 \searrow 1$ by a sequence of the form

$$\mathcal{D}_1 \searrow \mathcal{D}_2 \searrow \cdots \searrow \mathcal{D}_m \searrow \mathcal{D}_{m+1} \ , \tag{2}$$

in which e_i is an external simplex of $\mathcal{D}_i$, and we go from $\mathcal{D}_i$ to $\mathcal{D}_{i+1}$ by collapsing e_i, $i = 1, 2, \ldots, m$. Since each collapsing in such a sequence eliminates a 2-simplex from $\mathcal{C}$, the final complex $\mathcal{D}_{m+1}$ contains no 2-simplexes. Therefore we only need to show that e_i is an external simplex of $\mathcal{D}_i$, $i = 1, 2, \ldots, m$. However by induction on m, this fact is a consequence of the comparison between collapsing and erasing given in Figure 8. Therefore $er(\mathcal{C}) = k \leq co(\mathcal{C})$. The steps of going from the $\mathcal{C}_i$'s to $\mathcal{D}_i$ by a sequence of the form (2) can also be reversed. A set of 2-simplexes $\mathcal{K}$ of cardinality $k = co(\mathcal{C})$ of $\mathcal{C}$ such that $\mathcal{C} - \mathcal{K} \searrow 1$ gives rise to a sequence of erasures of the form (1) in which the 2-simplexes that are collapsed in each step of (2) are now erased instead in (1). From Figure 8, going from collapsing to erasing deletes all of the 1-simplexes left under collapsing. Therefore $\mathcal{C}_{m+1} = \phi$ and $\mathcal{C} - \mathcal{K} \rightsquigarrow \phi$. Thus $er(\mathcal{C}) \geq co(\mathcal{C})$. □

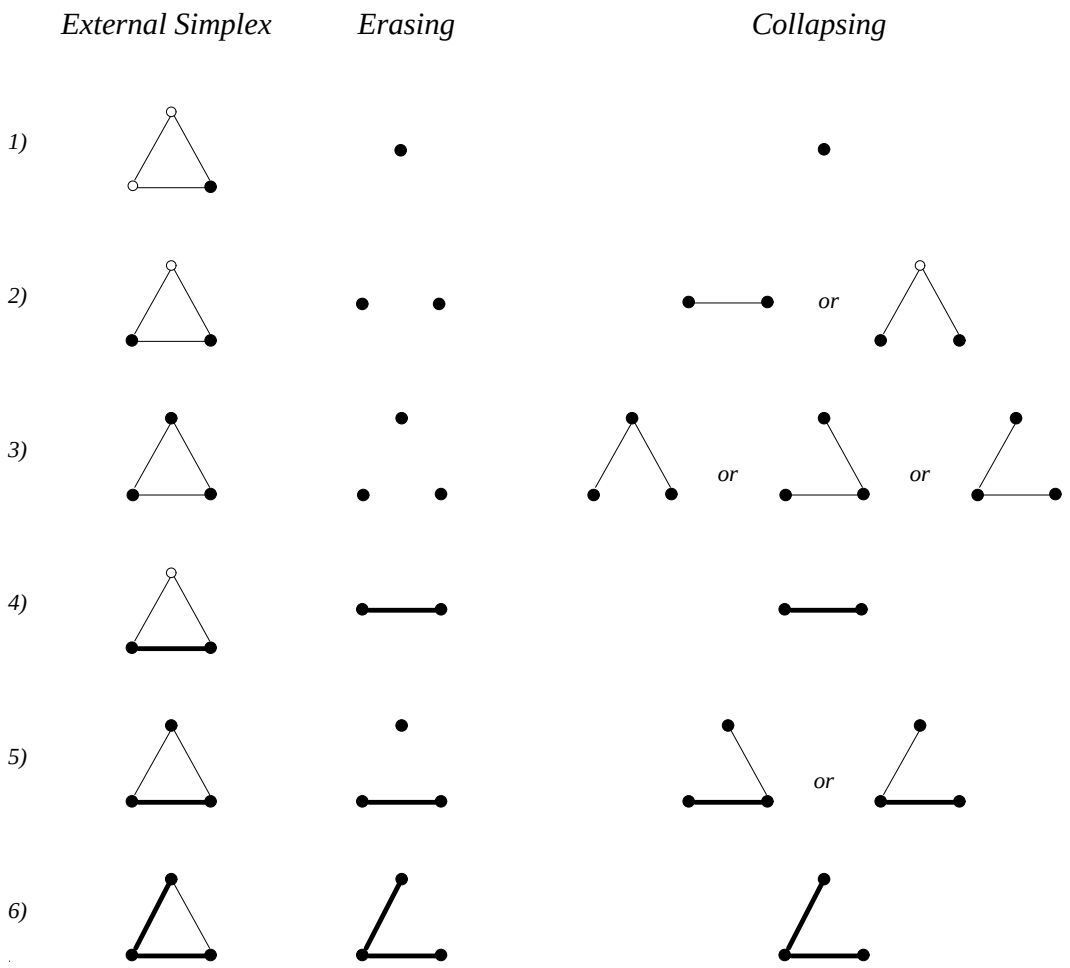

Figure 8: Erasing versus Collapsing external simplexes.