Equivariant Simplicial Reconstruction[*]

Lisa Carbone[†], Vidit Nanda[‡], and Yusra Naqvi[§]

Abstract. We introduce and analyze parallelizable algorithms to compress and accurately reconstruct finite simplicial complexes that have nontrivial automorphisms. The compressed data—called a complex of groups—amounts to a functor from (the poset of simplices in) the orbit space to the 2-category of groups, whose higher structure is prescribed by isomorphisms arising from conjugation. Using this functor, we show how to algorithmically recover the original complex up to equivariant simplicial isomorphism. Our algorithms are derived from generalizations (by Bridson–Haefliger, Carbone–Rips, and Corson, among others) of the classical Bass–Serre theory for reconstructing group actions on trees.

Key words. complexes of groups, higher categories, simplicial group actions

AMS subject classifications. 05E18, 20F65, 18D05

DOI. 10.1137/20M1327483

1. Introduction.

The ongoing proliferation of large, complicated, and vital datasets has sparked considerable activity focused on rendering hitherto-abstract branches of mathematics applicable to processing complex data. Examples of this phenomenon include, but are by no means restricted to, the recent employment of Laplacian eigenfunctions for spectral embedding [3], rough paths in machine learning [9], sheaf theory for linear programming [23], Morse theory for image processing [13], and (co)homology for data analysis [17]. In each case, efficient algorithms have catalyzed the percolation of theory to application. And, particularly in the last two examples, these algorithms accept as input some cell complex structure (often simplicial or cubical) imposed on the constituent elements of a given dataset.

Our work here continues these efforts: we adapt a framework originally devised for the study of infinite group actions on nonpositively curved spaces [6, 33] to the concrete task of efficiently exploiting symmetries to compress and reconstruct finite simplicial complexes. The central contribution of this paper therefore consists of two parallelizable algorithms. The first, called `Compress`, accepts as input a finite simplicial complex $\mathbb{X}$ along with a subgroup G of its automorphism group, and outputs a compressed structure $\boldsymbol{A}$ called a *complex of groups* [19, 10]. This $\boldsymbol{A}$ may, for the purposes of these introductory remarks, be regarded as a (typi-

[†]Department of Mathematics, Rutgers University, Piscataway, NJ 08854 USA (carbonel@math.rutgers.edu).
[‡]Mathematical Institute, University of Oxford, Oxford, UK, OX2 6GG (nanda@maths.ox.ac.uk).
[§]School of Mathematics and Statistics F07, University of Sydney, Sydney, NSW, 2006 Australia (yusra.naqvi@sydney.edu.au).

cally much smaller) group-weighted simplicial complex. The second algorithm, `Reconstruct`, inverts the first by using the overlaid algebraic data to correctly unfold the complex of groups $\boldsymbol{A}$ so that $\mathbb{X}$ is recovered up to G-equivariant isomorphism.

Assume, for the purposes of this introduction, that $\mathbb{X}$ is a triangulated bow-tie:
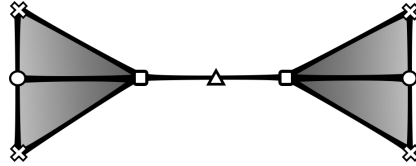
and consider the action of the Klein four-group $G = \langle \sigma, \tau \mid \sigma^2 = \tau^2 = (\sigma\tau)^2 = 1 \rangle$, where σ and τ act by reflection across the horizontal and vertical axes through the central vertex Δ, respectively (vertices lying in the same orbit have been decorated similarly). One possible choice of quotient $\mathbb{X}/G$ is the following *fundamental domain*, i.e., a subcomplex of $\mathbb{X}$ which intersects each G-orbit exactly once:
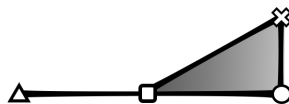
It is not too difficult to construct an infinite family of other group actions on different simplicial complexes which produce the same quotient. Clearly, $\mathbb{X}/G$ does not carry sufficient information to recover either $\mathbb{X}$ or the G-action. One therefore seeks the minimal amount of additional data necessary for such a reconstruction. Not surprisingly, the extra machinery required varies (in complexity, if not nature) depending on whether $\mathbb{X}$ is a tree [33], a graph [2, 14], a smooth manifold [18], a Coxeter complex [12], or the classifying space of a small loopfree category [19].

In any event, the complex of groups $\boldsymbol{A}$ for the action described above assigns to each simplex y of $\mathbb{X}/G$ a stabilizer subgroup $\boldsymbol{A}_y \leq G$ which fixes some simplex of $\mathbb{X}$ lying in the corresponding orbit class:
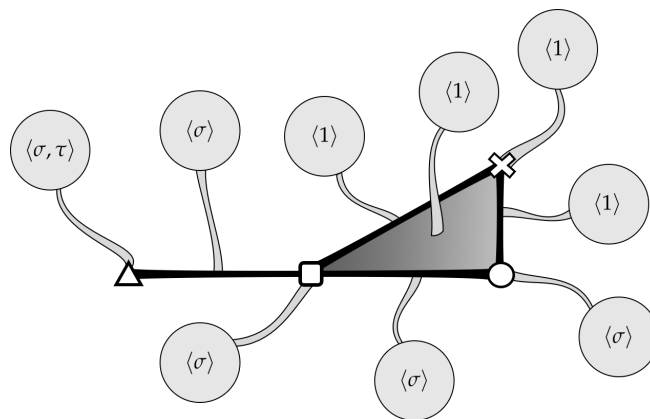

The group assigned to each simplex includes the groups assigned to simplices in its boundary, so our $\boldsymbol{A}$ is a group-valued *cellular cosheaf* [11], i.e., a functor from the poset of simplices

in $\mathbb{X}/G$ to the lattice of subgroups of G. It should be noted that—at least in our simple example—G is the colimit of $\boldsymbol{A}$. An appeal to the orbit-stabilizer theorem also guarantees that the number of simplices in $\mathbb{X}$ whose orbit class is represented by a simplex y in $\mathbb{X}/G$ equals the usual index $[G : \boldsymbol{A}_y]$, which counts cosets of $\boldsymbol{A}_y$ in G. Our next task, therefore, is to determine the correct face relations among these simplices.

At this stage, the scope of the difficulty starts to become apparent: How should one glue the coset-indexed simplices below so that the result is (isomorphic to) $\mathbb{X}$?
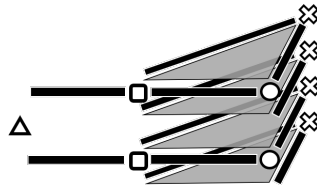
Knowledge of the subgroup indices $[G : \boldsymbol{A}_y]$ across all simplices y in $\mathbb{X}/G$ does not uniquely specify the desired face relations between our coset-simplices. Making unfortunate choices of attachments, even between vertices and edges, could easily backfire:
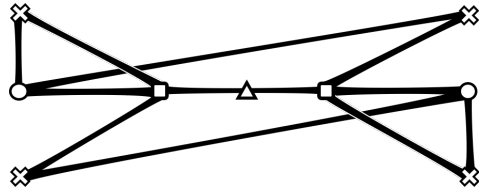
Here is a partial summary of the challenges which must be overcome before one can reconstruct more general group actions on more general complexes than our G and $\mathbb{X}$:

(1) The quotient $\mathbb{X}/G$ may not be a simplicial complex.
(2) Even if $\mathbb{X}/G$ is simplicial, it may not be a subcomplex of $\mathbb{X}$.
(3) If $\mathbb{X}$ is not simply connected, then G may not be the colimit of $\boldsymbol{A}$.
(4) The groups $\boldsymbol{A}_y$ are only determined up to conjugation in G.
(5) It is unclear how one should glue coset-simplices to recover $\mathbb{X}$.

Obstacle (1) above is bypassed by imposing a mild regularity constraint on the G-action, which is always satisfied after barycentric subdivision. In order to address the remaining challenges, one must carefully keep track of how the various $\boldsymbol{A}_y$ embed within each other inside G. Perhaps the most streamlined way to accomplish this is to view $\boldsymbol{A}$ as a *pseudofunctor*[1] by enhancing its target into a 2-category as follows. Given two homomorphisms ϕ and ϕ' between the same pair of groups, the set of 2-morphisms $\phi \Rightarrow \phi'$ is given by all elements g in the codomain (if any) which satisfy $\phi(\bullet) = g \cdot \phi'(\bullet) \cdot g^{-1}$. The remarkable advantage of this pseudofunctorial perspective, which we exploit in section 4, is that knowledge of the 2-morphisms lying in the image of $\boldsymbol{A}$ solves *all* the problems (2)–(5).

The following result motivates and underlies much of our work. In its statement, $\underline{G}^{\mathbb{Y}}$ denotes the constant complex of groups over $\mathbb{Y}$, which assigns the group G to all simplices and identity maps to all face relations in sight.

---

[1]Roughly, this means that $\boldsymbol{A}$ is associative only up to inner automorphisms. See Definition 2.6 for details.

Theorem (Thm. 2.18 and Cor. 2.19 of sect. III.C in [6]). *For each finite simplicial complex* $\mathbb{Y}$ *and finite group* G, *there is an equivalence:*

$$\begin{bmatrix} \textit{Regular } \mathrm{G}\textit{-actions on finite} \\ \textit{simplicial complexes } \mathbb{X} \textit{ with} \\ \textit{quotient } \mathbb{X}/\mathrm{G} = \mathbb{Y} \end{bmatrix} \xleftrightarrow{\;\simeq\;} \begin{bmatrix} \textit{Certain equivalence classes} \\ \textit{of complexes of groups over } \mathbb{Y} \\ \textit{that map injectively into } \underline{\mathrm{G}}^{\mathbb{Y}} \end{bmatrix}.$$

Related work and outline. Much of the theoretical machinery presented here (including the theorem above) will be quite familiar, or at least unsurprising, to those with expertise in certain areas of geometric group theory and equivariant algebraic topology. To the best of our knowledge, however, none of the existing literature on reconstructing group actions seriously confronts its algorithmic aspects. To accomplish that task here, we have adapted the treatment in the comprehensive text of Bridson and Haefliger [6, Chap. III.C], a recent framework developed by the first author with Rips [7], and the pseudofunctorial viewpoint mentioned above (which was also employed in Fiore–Lück–Sauer [15, sect. 8] for a different purpose). In an effort to avoid impeding the progress of readers unfamiliar with this body of work, we have also provided a summary of the basic constructions and fundamental results which are required to verify the correctness of our algorithms.

Our hope is that these algorithms will eventually be used for compressing simplicial complexes built around data points sampled from symmetric manifolds. It is, however, an unfortunate by-product of almost all discretization techniques that symmetries of smooth objects are not directly inherited by their finite approximations. For instance, a dense point cloud sampled uniformly at random from the unit sphere (viewed as a submanifold of Euclidean space) will not be fixed by any nontrivial element of the orthogonal group. Any reasonable framework for inferring automorphisms of symmetric manifolds from random samples will, in all probability, require methods to efficiently discover, quantify, and analyze *approximate* symmetries of finite data. While that grail-quest lies far beyond the scope of our work here, we direct interested parties to promising geometric [29] and statistical [16, sect. 3] efforts in its general direction.

The rest of this paper is organized as follows. Section 2 contains preliminary material regarding complexes of groups, sections 3 and 4 describe the voyage from a simplicial group action to the associated complex of groups and back, while section 5 contains the two promised algorithms `Compress` and `Reconstruct` along with their detailed complexity estimates, which are recorded in Theorems 5.5 and 5.6, respectively.

2. Background. For combinatorial and homological perspectives on simplicial group actions, we invite the reader to consult [1] and [5, Chap. III.1], respectively. General introductions to simplicial complexes, group actions and 2-categories may be found in [34, Chap. III], [22, Chap. II.4], and [24], respectively.

2.1. Simplicial group actions and quotients. Fix a finite simplicial complex $\mathbb{X}$ and a finite group G which acts on $\mathbb{X}$ via simplicial automorphisms. For each simplex x in $\mathbb{X}$, we have an *orbit* $\mathrm{G}x = \{g \cdot x \mid g \in \mathrm{G}\}$, which is a subset of $\mathbb{X}$, and a *stabilizer* $\mathrm{G}_x = \{g \in \mathrm{G} \mid g \cdot x = x\}$, which is a subgroup of G. The following definition is adapted from [5, Chap. III.1].

Definition 2.1. *The action of* G *on* $\mathbb{X}$ *is called* regular *if the following two properties hold for every simplex x of* $\mathbb{X}$. *Letting* $(v_0, \ldots, v_d)$ *denote the vertices of x:*

(1) *every g in G_x must satisfy $g \cdot v_i = v_i$ for all i in $\{0, \ldots, d\}$, and*

(2) *given $g_0, \ldots, g_d \in G$, if $x' = (g_0 \cdot v_0, \ldots, g_d \cdot v_d) \in \mathbb{X}$, then x' must lie in Gx.*

Regular actions are actions *without inversion* [2, 33] in the special case when $\mathbb{X}$ is a graph (i.e., $\dim \mathbb{X} = 1$), since the first requirement of the definition above prohibits the G-action from interchanging the two vertices of a given edge. We will assume henceforth that G acts regularly[2] in order to avail ourselves of three pleasant consequences. First, if y is a face of x (written $x \succ y$), then the corresponding stabilizers satisfy the subgroup relation $G_x \leq G_y$ because any g which fixes x is forced to fix all the vertices of x (and hence, y) pointwise. Second, the orbits assemble to form a quotient simplicial complex so that the natural projection from $\mathbb{X}$ is a simplicial map. And third, G acts transitively on the fibers of this map.

Definition 2.2. *The* orbit space *or* quotient $\mathbb{X}/G$ *associated to the action of* G *on* $\mathbb{X}$ *is the simplicial complex defined as follows. Its vertices are the* G-*orbits of vertices in* $\mathbb{X}$*, and a* d-*simplex in* $\mathbb{X}/G$ *is spanned by* $(d+1)$ *distinct orbits* $(Gv_0, \ldots, Gv_d)$ *if and only if there exists some simplex* $(u_0, \ldots, u_d)$ *in* $\mathbb{X}$ *with* $u_i \in Gv_i$ *for each* i.

By construction, there is a canonical surjective simplicial map $\boldsymbol{p}_G : \mathbb{X} \twoheadrightarrow \mathbb{X}/G$, called the *orbit map*, which sends each simplex of $\mathbb{X}$ to its G-orbit in $\mathbb{X}/G$. And by the second regularity requirement, if two simplices x and x' of $\mathbb{X}$ satisfy $\boldsymbol{p}_G(x) = \boldsymbol{p}_G(x')$, then some $g \in G$ satisfies $g \cdot x = x'$.

Definition 2.3. *A* fundamental domain *for the action of* G *on* $\mathbb{X}$ *is a subcomplex* $\mathbb{X}' \subset \mathbb{X}$ *which contains exactly one simplex from each orbit.*

Whenever a fundamental domain exists for a given action (as with the bow-tie from the introduction), the quotient space $\mathbb{X}/G$ may be viewed as a subcomplex of $\mathbb{X}$. However, not every action admits a fundamental domain.

Example 2.4. The cyclic group (on three elements) $G = C_3$ acts by rotation on the subdivided standard 2-simplex $\mathbb{X}$ shown below:
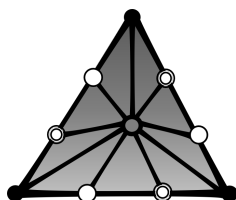
Since this action is regular, $\mathbb{X}/G$ inherits a natural simplicial complex structure (given by a coarser subdivision of the 2-simplex):

---
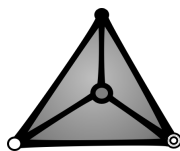
[2]Fortunately, regularity is not a severe requirement on group actions—any automorphism can be made regular via passage to the second barycentric subdivision of $\mathbb{X}$ (see condition (B) in [5, Chap. III.1]).

One can further illustrate $\boldsymbol{p}_G$ as a simplicial map from $\mathbb{X}$ down to $\mathbb{X}/G$, so that the fiber $\boldsymbol{p}_G^{-1}$ over a simplex in $\mathbb{X}/G$ is the collection of all simplices from $\mathbb{X}$ in the associated orbit:
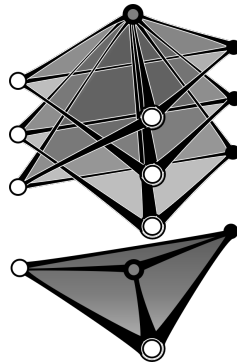

On the other hand, G also acts by rotation on the quotient $\mathbb{Y} = \mathbb{X}/G$, but this action fails to satisfy the second requirement of regularity. As a consequence, the successive quotient of $\mathbb{Y}$ by G is no longer a simplicial complex.

2.2. The 2-category of groups. Consider a pair of groups G_0, G_1 and group homomorphisms $\phi, \phi' : G_0 \to G_1$. We write $g : \phi \Rightarrow \phi'$ if some group element $g \in G_1$ relates ϕ and ϕ' by conjugation in the sense that

$$\phi = \mathrm{Ad}(g) \circ \phi', \ \text{i.e.,} \ \phi(h) = g \cdot \phi'(h) \cdot g^{-1} \ \text{for all} \ h \in G_0.$$

The homomorphisms $G_0 \to G_1$ thus form the objects of a groupoid,[3] which we will denote by $\mathbf{Grp}(G_0, G_1)$—its morphisms are given by such $g : \phi \Rightarrow \phi'$ and the composition law is inherited from multiplication in G_1 as follows. Given

$$\phi \overset{g}{\Rightarrow} \phi' \overset{g'}{\Rightarrow} \phi'',$$

where ϕ'' is yet another homomorphism $G_0 \to G_1$, a straightforward calculation reveals that the product $g \cdot g'$ in G_1 satisfies $g \cdot g' : \phi \Rightarrow \phi''$, and hence that $g^{-1} : \phi' \Rightarrow \phi$ serves as the inverse to g. In order to formally distinguish a product in the groupoid $\mathbf{Grp}(G_0, G_1)$ from the corresponding (contravariant) product in the group G_1, we will denote the former by $g' * g$ and call it the *vertical* composite of g with g'.

Given $g : \phi \Rightarrow \phi'$ in $\mathbf{Grp}(G_0, G_1)$ and $h : \psi \Rightarrow \psi'$ in $\mathbf{Grp}(G_1, G_2)$, define their *horizontal* composite $h \circ g$ as the group element $\psi(g) \cdot h$ in G_2. Two straightforward calculations reveal the following:

(1) across any triple of groups G_0, G_1, and G_2, horizontal composition yields a map of groupoids

$$\circ : \mathbf{Grp}(G_1, G_2) \times \mathbf{Grp}(G_0, G_1) \to \mathbf{Grp}(G_0, G_2),$$

which extends the ordinary composition for group homomorphisms, and

---

[3]By *groupoid* we mean a small category all of whose morphisms are invertible.

(2) the horizontal and vertical compositions satisfy the *interchange law*, meaning that across any diagram of the form
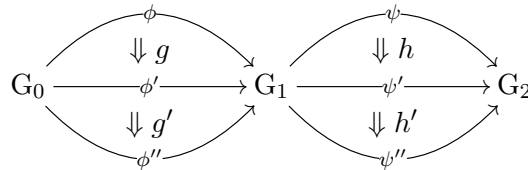
we have an equality

$$(h' \circ g') * (h \circ g) = (h' * h) \circ (g' * g)$$

in the groupoid $\mathbf{Grp}(\mathrm{G}_0, \mathrm{G}_2)$.

These two facts allow us to define a higher category of groups [28, Ex. 1.3], [15, Def. 8.1].

Definition 2.5. *The* 2-category of groups, *denoted* **Grp**, *is defined as follows:*
(1) *its objects are all groups,*
(2) *the 1-morphisms $\phi : \mathrm{G} \to \mathrm{G}'$ are the usual group homomorphisms, and*
(3) *the 2-morphisms $g : \phi \Rightarrow \phi'$ are given by conjugation, i.e., $\phi = \mathrm{Ad}(g) \circ \phi'$.*
The 1-morphisms are endowed with the usual composition $\circ$ for group homomorphisms, while the horizontal and vertical composition of 2-morphisms is given by $\circ$ and $*$, respectively.

In modern parlance, **Grp** is a *strict (2,1)-category*, which is to say that all compositions are associative, all identity elements are unique (rather than being defined only up to coherent isomorphisms), and all 2-morphisms are invertible. Dispensing with these subtleties in the interest of brevity, we will simply call **Grp** a 2-category henceforth.

2.3. Complexes of groups. Variants of the following definition appeared simultaneously in the work of Haefliger [19] and Corson [10].

Definition 2.6. *A complex of groups $\boldsymbol{A}$ over a simplicial complex $\mathbb{Y}$ assigns to each*
(1) *simplex x a finite group $\boldsymbol{A}_x$,*
(2) *pair $x \succ y$ an injective homomorphism $\boldsymbol{A}_{x \succ y} : \boldsymbol{A}_x \hookrightarrow \boldsymbol{A}_y$, and*
(3) *triple $x \succ y \succ z$ a 2-morphism $\boldsymbol{A}_{x \succ y \succ z} : \boldsymbol{A}_{y \succ z} \circ \boldsymbol{A}_{x \succ y} \Rightarrow \boldsymbol{A}_{x \succ z}$,*
subject to the following constraints:
(1) *$\boldsymbol{A}_{x \succ x}$ is the identity $\boldsymbol{A}_x \to \boldsymbol{A}_x$,*
(2) *$\boldsymbol{A}_{x \succ x \succ y}$ and $\boldsymbol{A}_{x \succ y \succ y}$ are identities $\boldsymbol{A}_{x \succ y} \Rightarrow \boldsymbol{A}_{x \succ y}$, and*
(3) *for any simplices $w \succ x \succ y \succ z$ in $\mathbb{Y}$, the following relation holds in the group $\boldsymbol{A}_z$:*

$$\boldsymbol{A}_{y \succ z}(\boldsymbol{A}_{w \succ x \succ y}) \cdot \boldsymbol{A}_{w \succ y \succ z} = \boldsymbol{A}_{x \succ y \succ z} \cdot \boldsymbol{A}_{w \succ x \succ z}.$$

The last constraint is called the cocycle condition *[6, Chap. III.C.2].*

In other words, a complex of groups is a pseudofunctor $\boldsymbol{A} : \mathbf{Fac}(\mathbb{Y}) \to \mathbf{Grp}$ from the poset $\mathbf{Fac}(\mathbb{Y})$ of simplices in $\mathbb{Y}$ (ordered by the is-a-face-of relation) to the 2-category of groups from Definition 2.5. This means that the homomorphism $\boldsymbol{A}_{x \succ z}$ only equals the expected composite $\boldsymbol{A}_{y \succ z} \circ \boldsymbol{A}_{x \succ y}$ up to conjugation as prescribed by the 2-morphism $\boldsymbol{A}_{x \succ y \succ z}$ (these 2-morphisms are called *twisting elements* in [6]). Similarly, $\boldsymbol{A}$ need not send identity 2-morphisms in its domain to identities in its codomain. The cocycle condition, however, ensures associativity

in the assignment of 2-morphisms by insisting that the following diagram commutes in the groupoid $\mathbf{Grp}(\boldsymbol{A}_w, \boldsymbol{A}_z)$:

$$
\begin{array}{ccc}
\boldsymbol{A}_{y \succ z} \circ \boldsymbol{A}_{x \succ y} \circ \boldsymbol{A}_{w \succ x} & \xrightarrow{\ \mathrm{id} \circ \boldsymbol{A}_{w \succ x \succ y}\ } & \boldsymbol{A}_{y \succ z} \circ \boldsymbol{A}_{w \succ y} \\[2pt]
\Big\Downarrow {\scriptstyle \boldsymbol{A}_{x \succ y \succ z} \circ \mathrm{id}} & & \Big\Downarrow {\scriptstyle \boldsymbol{A}_{w \succ y \succ z}} \\[2pt]
\boldsymbol{A}_{x \succ z} \circ \boldsymbol{A}_{w \succ x} & \xrightarrow[\ \boldsymbol{A}_{w \succ x \succ z}\ ]{} & \boldsymbol{A}_{w \succ z}
\end{array}
$$

We have restricted our attention here to complexes of *finite* groups for algorithmic reasons; the preceding definition and subsequent ones (can be made to) extend almost verbatim to bifunctors from the poset of open sets in a reasonable topological space to the 2-category of arbitrary groups. The finite complexes of groups defined above may thus be viewed as concrete, combinatorial incarnations of orbifolds [30, 31] and topological Deligne–Mumford stacks [32, sect. 19.5].

Example 2.7. Given a finite group G (with identity element 1_G) and simplicial complex $\mathbb{Y}$, the *constant* G-valued complex of groups over $\mathbb{Y}$ is denoted $\underline{\mathrm{G}}^{\mathbb{Y}} : \mathbf{Fac}(\mathbb{Y}) \to \mathbf{Grp}$, and defined as follows:
   (1) each simplex y is assigned the same group $\underline{\mathrm{G}}^{\mathbb{Y}}_y = \mathrm{G}$,
   (2) each pair $x \succ y$ is assigned the identity homomorphism $\underline{\mathrm{G}}^{\mathbb{Y}}_{x \succ y} = \mathrm{id} : \mathrm{G} \to \mathrm{G}$, and
   (3) each triple $x \succ y \succ z$ is assigned the identity $\underline{\mathrm{G}}^{\mathbb{Y}}_{x \succ y \succ z} = 1_\mathrm{G} : \mathrm{id} \Rightarrow \mathrm{id}$.

In order to compare complexes of groups, we require a convenient notion of morphisms between them (see [19, Def. 2.5] and [26, Def. 12]).

Definition 2.8. *A morphism* $\Phi : \boldsymbol{A} \to \boldsymbol{B}$ *of complexes of groups over (the same simplicial complex)* $\mathbb{Y}$ *assigns the following:*
   (1) *to each simplex y a 1-morphism $\Phi_y : \boldsymbol{A}_y \to \boldsymbol{B}_y$ in* $\mathbf{Grp}$, *and*
   (2) *to each pair $x \succ y$ a 2-morphism $\Phi_{x \succ y} : \Phi_y \circ \boldsymbol{A}_{x \succ y} \Rightarrow \boldsymbol{B}_{x \succ y} \circ \Phi_x$ in* $\mathbf{Grp}$*:*

$$
\begin{array}{ccc}
\boldsymbol{A}_x & \xrightarrow{\ \Phi_x\ } & \boldsymbol{B}_x \\[2pt]
\Big\downarrow {\scriptstyle \boldsymbol{A}_{x \succ y}} & {\scriptstyle \Phi_{x \succ y}} & \Big\downarrow {\scriptstyle \boldsymbol{B}_{x \succ y}} \\[2pt]
\boldsymbol{A}_y & \xrightarrow[\ \Phi_y\ ]{} & \boldsymbol{B}_y
\end{array}
$$

so that the following two axioms hold:
   (1) *the* identity *axiom requires $\Phi_{y \succ y}$ to be the identity $\Phi_y \Rightarrow \Phi_y$ for each simplex y, whereas*
   (2) *the* coherence *axiom imposes a relation*

$$
\Phi_z(\boldsymbol{A}_{x \succ y \succ z}) \cdot \Phi_{x \succ z} = \Phi_{y \succ z} \cdot \boldsymbol{B}_{y \succ z}(\Phi_{x \succ y}) \cdot \boldsymbol{B}_{x \succ y \succ z}
$$

in the group $\boldsymbol{B}_z$.

Thus, Φ is a pseudonatural transformation [25, sect. 1.2] between pseudofunctors $\boldsymbol{A}, \boldsymbol{B}$: $\mathbf{Fac}(\mathbb{Y}) \to \mathbf{Grp}$. The coherence axiom requires the commutativity of a certain pentagon in the groupoid $\mathbf{Grp}(\boldsymbol{A}_x, \boldsymbol{B}_z)$ for any three simplices $x \succ y \succ z$. We indicate its vertices below:

$$\Phi_z \circ \boldsymbol{A}_{y \succ z} \circ \boldsymbol{A}_{x \succ y} \Rrightarrow \boldsymbol{B}_{y \succ z} \circ \Phi_y \circ \boldsymbol{A}_{x \succ y}$$

$$\Phi_z \circ \boldsymbol{A}_{x \succ z} \qquad\qquad \boldsymbol{B}_{y \succ z} \circ \boldsymbol{B}_{x \succ y} \circ \Phi_x$$

$$\boldsymbol{B}_{x \succ z} \circ \Phi_x$$

and encourage the reader to accurately decorate its edges. We call $\Phi : \boldsymbol{A} \to \boldsymbol{B}$ *injective* if each $\Phi_y : \boldsymbol{A}_y \to \boldsymbol{B}_y$ is an injective group homomorphism. And if each Φ_y is an isomorphism between $\boldsymbol{A}_y$ and $\boldsymbol{B}_y$, then one can easily construct an inverse pseudofunctor $\boldsymbol{B} \to \boldsymbol{A}$, so in this case Φ itself is an *isomorphism* between $\boldsymbol{A}$ and $\boldsymbol{B}$.

Remark 2.9. For our purposes here, the most important morphisms of complexes of groups over a simplicial complex $\mathbb{Y}$ will be the *injective ones from an arbitrary domain to a constant codomain*, i.e., $\Phi : \boldsymbol{A} \to \underline{\mathrm{G}}^{\mathbb{Y}}$ for some fixed group G. In this special case, Φ assigns to each

(1) simplex y an injective group homomorphism $\Phi_y : \boldsymbol{A}_y \hookrightarrow \mathrm{G}$, and
(2) pair $x \succ y$ a 2-morphism $\Phi_{x \succ y} : \Phi_y \circ \boldsymbol{A}_{x \succ y} \Rightarrow \Phi_x$ in $\mathbf{Grp}$,[4]

so that every $\Phi_{y \succ y}$ is the identity, and the following relation holds in G across all triples $x \succ y \succ z$:

$$(1) \qquad\qquad \Phi_z(\boldsymbol{A}_{x \succ y \succ z}) \cdot \Phi_{x \succ z} = \Phi_{y \succ z} \cdot \Phi_{x \succ y}.$$

The existence of such a Φ implies that all the $\boldsymbol{A}_y$ are simultaneously and coherently realizable as subgroups of G. In other words, G forms a *cocone* [27, Chap. III.4] for the diagram in $\mathbf{Grp}$ parametrized by the functor $\boldsymbol{A}$ over the poset $\mathbf{Fac}(\mathbb{Y})$.

In what follows, we will compare not only complexes of groups, but also their morphisms. We achieve the latter by using the 2-categorical structure of $\mathbf{Grp}$ locally over each simplex y in $\mathbb{Y}$ as follows.

Definition 2.10. *Given two morphisms* $\Phi, \Psi : \boldsymbol{A} \to \boldsymbol{B}$ *of complexes of groups over* $\mathbb{Y}$, *a homotopy* $\theta : \Phi \Rightarrow \Psi$ *is a (contravariant) collection of 2-morphisms* $\{\theta_y : \Psi_y \Rightarrow \Phi_y\}$ *in* $\mathbf{Grp}$, *indexed by simplices* y *of* $\mathbb{Y}$, *so that for each face relation* $x \succ y$ *the following square commutes:*

$$\begin{array}{ccc}
\Psi_y \circ \boldsymbol{A}_{x \succ y} & \xRightarrow{\Psi_{x \succ y}} & \boldsymbol{B}_{x \succ y} \circ \Psi_x \\
{\scriptstyle \theta_y \circ id} \Downarrow & & \Downarrow {\scriptstyle id \circ \theta_x} \\
\Phi_y \circ \boldsymbol{A}_{x \succ y} & \xRightarrow[\Phi_{x \succ y}]{} & \boldsymbol{B}_{x \succ y} \circ \Phi_x
\end{array}$$

In other words, we have a relation

$$\Psi_{x \succ y} \cdot \boldsymbol{B}_{x \succ y}(\theta_x) = \theta_y \cdot \Phi_{x \succ y}$$

---

[4]As in Definition 2.5, $\Phi_{x \succ y}$ is an element of G satisfying $\mathrm{Ad}(\Phi_{x \succ y})\Phi_x = \Phi_y \circ \boldsymbol{A}_{x \succ y}$.

in the group $\boldsymbol{B}_y$.

Horizontal and vertical compositions for homotopies are defined simplexwise. Since such homotopies relate pseudonatural transformations between pseudofunctors, they form examples of *modifications* [25, sect. 1.3]. Complexes of groups over $\mathbb{Y}$, injective morphisms between them, and homotopies between those morphisms also form a higher category.

Definition 2.11. *The* 2-category of complexes of groups over $\mathbb{Y}$, *written* $\mathbf{Com}(\mathbb{Y})$, *is described by the following data:*
(1) *its objects are all complexes of groups* $\boldsymbol{A} : \mathbf{Fac}(\mathbb{Y}) \to \mathbf{Grp}$ *over* $\mathbb{Y}$,
(2) *its* 1-*morphisms are all* injective *morphisms* $\Phi : \boldsymbol{A} \to \boldsymbol{B}$, *and*
(3) *its* 2-*morphisms are all homotopies* $\theta : \Phi \Rightarrow \Psi$.

3. Compression. Throughout this section, $\mathbb{X}$ is a finite, connected simplicial complex while G is a fixed subgroup of its regular automorphisms. Let $\mathbb{Y}$ denote the orbit space $\mathbb{X}/\mathrm{G}$ and $\boldsymbol{p}_G : \mathbb{X} \to \mathbb{Y}$ the orbit map. Here we construct a pair $(\boldsymbol{A}, \Phi)$, where
(1) $\boldsymbol{A} : \mathbf{Fac}(\mathbb{Y}) \to \mathbf{Grp}$ is a complex of groups over $\mathbb{Y}$, and
(2) $\Phi : \boldsymbol{A} \to \underline{\mathrm{G}}^{\mathbb{Y}}$ is an injective morphism to the constant complex of groups.
Following [6, sect. III.C.2.9], we explicitly describe $\boldsymbol{A}$ and Φ by making certain ad-hoc local choices. Fortunately, it turns out that different choices lead to isomorphic constructions.

3.1. Lifts and transfers. The surjectivity of $\boldsymbol{p}_G$ guarantees that each simplex y of $\mathbb{Y}$ admits a *lift* in the sense that some simplex x in $\mathbb{X}$ satisfies $\boldsymbol{p}_G(x) = y$. We select such lifts arbitrarily for all such y, emphasizing that face relations of the form $y \succ y'$ in $\mathbb{Y}$ may not ascend to relations of the form $x \succ x'$ in $\mathbb{X}$ among lifts. As a result, the collection of lifts (which is sometimes called a G-*transversal* [14, sect. I.1.3]) might be quite far from forming a subcomplex of $\mathbb{X}$. The black simplices below form a complete and valid choice of lifts for the action from Example 2.4, but they manifestly do not constitute a simplicial subcomplex:
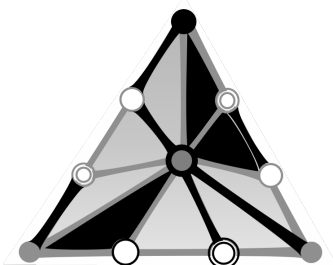
In order to address the defect illustrated above, one seeks to relate the lifts x and x' associated to pairs of distinct adjacent simplices $y \succ y'$ in $\mathbb{Y}$. Let z be the unique face of x in $\mathbb{X}$ which satisfies $\boldsymbol{p}_G(z) = y'$, i.e., z is the face whose vertices lie in orbits determined by y'. Since G acts regularly on $\mathbb{X}$, there exists an element $g \in \mathrm{G}$, not necessarily unique, so that $g \cdot z = x'$. Thus, the best that one can expect, in general, is $x \succ g^{-1}x'$ in $\mathbb{X}$ for some g whenever $y \succ y'$ holds in $\mathbb{Y}$. We arbitrarily select one such $g = g_{y \succ y'}$, and call it the *transfer*[5] associated to the relation $y \succ y'$. On the other hand, if the lifts did happen to satisfy $x \succ x'$ in

---

[5]These transfers give rise to the monodromy elements of [7].

$\mathbb{X}$, then one could simplify (computational) matters greatly by selecting the identity transfer $g_{y \succ y'} = 1_{\mathrm{G}}$.

3.2. Constructing A and Φ. Given a choice of lifts and transfers corresponding to simplices and face relations in $\mathbb{Y}$, one may construct a complex of groups $\boldsymbol{A} : \mathbf{Fac}(\mathbb{Y}) \to \mathbf{Grp}$ as follows:
   (1) for each simplex y, the group $\boldsymbol{A}_y$ is the stabilizer G_x, where $x = x_y$ is chosen the lift of y,
   (2) for each pair $y \succ y'$, the map $\boldsymbol{A}_{y \succ y'} : \boldsymbol{A}_y \hookrightarrow \boldsymbol{A}_{y'}$ is $\mathrm{Ad}(g)$, where $g = g_{y \succ y'}$ is the transfer selected for $y \succ y'$, and
   (3) for each triple $y \succ y' \succ y''$, the map $\boldsymbol{A}_{y \succ y' \succ y''} : \boldsymbol{A}_{y' \succ y''} \circ \boldsymbol{A}_{y \succ y'} \Rightarrow \boldsymbol{A}_{y \succ y''}$ is given by the following product of transfers in G:

$$g_{y' \succ y''} \cdot g_{y \succ y'} \cdot g_{y \succ y''}^{-1}.$$

To confirm that the above data prescribe a bona fide complex of groups, one must check that conjugation by $g_{y \succ y'}$ maps G_x injectively to a subgroup of $\mathrm{G}_{x'}$ (it does), and that the 2-morphisms satisfy the cocycle condition from Definition 2.6 (they do).[6]

Remark 3.1. If G is abelian, then its conjugate subgroups are necessarily equal; in this case, all the 1-morphisms $\boldsymbol{A}_{y \succ y'}$ are just inclusions.

Turning to the matter of constructing an injective morphism $\Phi : \boldsymbol{A} \to \underline{\mathrm{G}}^{\mathbb{Y}}$, we assign
   (1) to each simplex y, the inclusion $\Phi_y : \boldsymbol{A}_y \hookrightarrow \mathrm{G}$ (noting that $\boldsymbol{A}_y$ is the stabilizer of y's lift, and hence naturally a subgroup of G), and
   (2) to each face relation $y \succ y'$, the 2-morphism $\Phi_{y \succ y'}$ given by the transfer $g_{y \succ y'} \in \mathrm{G}$.
We leave it to the reader to confirm that these assignments successfully produce an injective morphism $\Phi : \boldsymbol{A} \to \underline{\mathrm{G}}^{\mathbb{Y}}$ as described in Remark 2.9.

3.3. Context. In this section we quantify the impact of lifts and transfers on the construction of $(\boldsymbol{A}, \Phi)$. Let $\mathbf{Com}(\mathbb{Y})$ be the 2-category from Definition 2.11. The following *fiber 2-category* [8, sect. 3] forms a natural home for pairs like the $(\boldsymbol{A}, \Phi)$ constructed above.

Definition 3.2. By $\mathbf{Com}(\mathbb{Y})/\underline{\mathrm{G}}^{\mathbb{Y}}$ *we denote the fiber 2-category whose*
   (1) *objects are all pairs* $(\boldsymbol{B}, \Psi)$, *where* $\boldsymbol{B}$ *is an object of* $\mathbf{Com}(\mathbb{Y})$ *while* $\Psi : \boldsymbol{B} \to \underline{\mathrm{G}}^{\mathbb{Y}}$ *is a morphism,*
   (2) *1-morphisms* $(\boldsymbol{B}, \Psi) \to (\boldsymbol{B}', \Psi')$ *are all pairs* (Δ, η) *where* $\Delta : \boldsymbol{B} \to \boldsymbol{B}'$ *is a 1-morphism in* $\mathbf{Com}(\mathbb{Y})$ *while* $\eta : \Psi' \circ \Delta \Rightarrow \Psi$ *is a homotopy:*

$$
\begin{array}{ccc}
\boldsymbol{B} & \xrightarrow{\ \ \Delta\ \ } & \boldsymbol{B}' \\
& \overset{\eta}{\Longleftarrow} & \\
{\scriptstyle \Psi} \searrow & & \swarrow {\scriptstyle \Psi'} \\
& \underline{\mathrm{G}}^{\mathbb{Y}} &
\end{array}
$$

---

[6] After performing straightforward manipulations, both sides of the cocycle equation evaluate to the pleasant expression $g_{y'' \succ y'''} \cdot g_{y' \succ y''} \cdot g_{y \succ y'} \cdot g_{y \succ y'''}^{-1}$.

(3) *2-morphisms* $(\Delta, \eta) \Rightarrow (\Delta', \eta')$ *are given by all those homotopies* $\theta : \Delta' \Rightarrow \Delta$ *in* $\mathbf{Com}(\mathbb{Y})$ *which make the following diagram commute:*

$$\Psi' \circ \Delta \xLeftarrow{\quad id \circ \theta \quad} \Psi' \circ \Delta'$$
$$\eta \searrow \qquad \swarrow \eta'$$
$$\Psi$$

Proposition 3.3. *Let* $(\boldsymbol{A}, \Phi)$ *and* $(\boldsymbol{A}', \Phi')$ *be produced by different choices of lifts and transfers. Then, the following hold in* $\mathbf{Com}(\mathbb{Y})/\underline{\mathrm{G}}^{\mathbb{Y}}$:

(1) *there exists an isomorphism* $(\Delta, \eta) : (\boldsymbol{A}, \Phi) \to (\boldsymbol{A}', \Phi')$ *so that each* $\Delta_y : \boldsymbol{A}_y \to \boldsymbol{A}'_y$ *is given by conjugation by some element of* G, *and*

(2) *given any two such isomorphisms, there exists a unique homotopy* $\theta : (\Delta, \eta) \Rightarrow (\Delta', \eta')$.

Proof. Let $\{x_y, g_{y \succ y'}\}$ and $\{z_y, h_{y \succ y'}\}$ be two choices of lifts and transfers used to construct $\boldsymbol{A}$ and $\boldsymbol{A}'$, respectively. For each simplex y of $\mathbb{Y}$, the regularity of the G-action on $\mathbb{X}$ guarantees the presence of some $k_y \in \mathrm{G}$ so that $k_y \cdot x_y = z_y$. Given such k_y, define an isomorphism $\Delta : \boldsymbol{A} \to \boldsymbol{A}'$ in $\mathbf{Com}(\mathbb{Y})$ as follows: the group isomorphism $\Delta_y : \boldsymbol{A}_y \to \boldsymbol{A}'_y$ is $\mathrm{Ad}(k_y)$, while the 2-morphism $\Delta_{y \succ y'}$ is prescribed by the element

$$k_{y'} \cdot g_{y \succ y'} \cdot k_y^{-1} \cdot h_{y \succ y'}^{-1}$$

in the group $\boldsymbol{A}'_{y'}$. A homotopy $\eta : \Phi' \circ \Delta \Rightarrow \Phi$ is now obtained by setting $\eta_y = k_y^{-1}$ for each simplex y. Given another isomorphism $(\Delta', \eta') : (\boldsymbol{A}, \Phi) \to (\boldsymbol{A}', \Phi')$ generated by different choices $\{\ell_y\}$ instead of $\{k_y\}$, the desired unique homotopy $\theta : (\Delta, \eta) \Rightarrow (\Delta', \eta')$ is determined completely by setting $\theta_y = k_y \cdot \ell_y^{-1}$. ∎

4. Reconstruction. We assume throughout this section that $\mathbb{Y}$ is a fixed finite simplicial complex and that $(\boldsymbol{A}, \Phi)$ is a distinguished object in the fiber 2-category $\mathbf{Com}(\mathbb{Y})/\underline{\mathrm{G}}^{\mathbb{Y}}$ from Definition 3.2. Our goal here is to construct a simplicial complex $\mathbb{X}$ and a regular action of G on $\mathbb{X}$ that satisfies three natural criteria:

(1) the quotient $\mathbb{X}/\mathrm{G}$ is isomorphic to $\mathbb{Y}$,

(2) compressing this G-action produces $(\boldsymbol{A}, \Phi)$, and

(3) $\mathbb{X}$ is unique up to G-equivariant simplicial isomorphism.

The last criterion above is a universal property: it asserts that any other $\mathbb{X}'$ satisfying the first two properties admits an invertible simplicial map $\boldsymbol{s} : \mathbb{X} \to \mathbb{X}'$ so that

$$\boldsymbol{s}(g \cdot x) = g \cdot \boldsymbol{s}(x)$$

for each simplex x in $\mathbb{X}$ and group element g in G.

The process of unfolding $(\boldsymbol{A}, \Phi)$ in order to extract both $\mathbb{X}$ and the concomitant G-action is of central importance in the theory of complexes of groups, to the extent that it has been called *the Basic Construction* in [6, Thm. III.C.2.13] and also in [12, Chap. 5].

4.1. The basic construction. For each simplex y of $\mathbb{Y}$, let $\boldsymbol{c}_y : \mathrm{G} \to \mathrm{G}/\Phi_y(\boldsymbol{A}_y)$ be the canonical surjective map (of sets) that sends each group element g to the corresponding left coset $g \cdot \Phi_y(\boldsymbol{A}_y)$. Consider the set P of pairs

$$P = \{(y, \boldsymbol{c}_y(g)) \mid y \text{ is a simplex of } \mathbb{Y} \text{ and } g \in \mathrm{G}\},$$

equipped with a binary relation $\blacktriangleright$, defined as follows:

$$(y, \boldsymbol{c}_y(g)) \blacktriangleright (y', \boldsymbol{c}_{y'}(g')) \quad \text{if} \quad y \succ y' \text{ and } \boldsymbol{c}_{y'}(g') = \boldsymbol{c}_{y'}(g \cdot \Phi_{y \succ y'}^{-1}).$$

Proposition 4.1. *The relation $\blacktriangleright$ forms a well-defined partial order on P.*

Proof. To see that $\blacktriangleright$ is well-defined on P, note by Remark 2.9 that for any k in $\boldsymbol{A}_y$, we have an equality between $\mathrm{Ad}(\Phi_{y \succ y'}) \circ \Phi_y(k)$ and $\Phi_{y'} \circ \boldsymbol{A}_{y \succ y'}(k)$. Recalling that $\boldsymbol{A}_{y \succ y'}$ takes values in the group $\boldsymbol{A}_{y'}$, it follows that

$$\boldsymbol{c}_{y'}(\Phi_{y \succ y'}^{-1}) = \boldsymbol{c}_{y'}(\Phi_y(k) \cdot \Phi_{y \succ y'}^{-1}).$$

Therefore, for any g in G, we have an equality of cosets

$$\boldsymbol{c}_{y'}(g \cdot \Phi_{y \succ y'}^{-1}) = \boldsymbol{c}_{y'}(g \cdot \Phi_y(k) \cdot \Phi_{y \succ y'}^{-1}),$$

and so $\blacktriangleright$ does not depend on our choice of k. Since both the reflexivity and antisymmetry of $\blacktriangleright$ are straightforward, it remains to establish its transitivity. To this end, assume we have a string of two $\blacktriangleright$-relations:

$$(y_0, \boldsymbol{c}_{y_0}(g_0)) \blacktriangleright (y_1, \boldsymbol{c}_{y_1}(g_1)) \blacktriangleright (y_2, \boldsymbol{c}_{y_2}(g_2)).$$

By definition of $\blacktriangleright$, we obtain $y_0 \succ y_1$ and $y_1 \succ y_2$ in $\mathbb{Y}$, whence $y_0 \succ y_2$. Moreover, using the second and first $\blacktriangleright$-relations above (in that order), we have

$$\begin{aligned} \boldsymbol{c}_{y_2}(g_2) &= \boldsymbol{c}_{y_2}(g_1 \cdot \Phi_{y_1 \succ y_2}^{-1}), \\ &= \boldsymbol{c}_{y_2}(g_0 \cdot \Phi_{y_0 \succ y_1}^{-1} \cdot \Phi_{y_1}(h) \cdot \Phi_{y_1 \succ y_2}^{-1}) \end{aligned}$$

for some $h \in \boldsymbol{A}_{y_1}$. By the well-definedness of $\blacktriangleright$, the coset above does not depend on h and we can simplify it to

$$\boldsymbol{c}_{y_2}(g_2) = \boldsymbol{c}_{y_2}(g_0 \cdot \Phi_{y_0 \succ y_1}^{-1} \cdot \Phi_{y_1 \succ y_2}^{-1}).$$

By (1) we have

$$\Phi_{y_0 \succ y_1}^{-1} \cdot \Phi_{y_1 \succ y_2}^{-1} = \Phi_{y_0 \succ y_2}^{-1} \quad \mathrm{mod}\ \Phi_{y_2}(\boldsymbol{A}_{y_2}),$$

so we obtain the desired relation $(y_0, \boldsymbol{c}_{y_0}(g_0)) \blacktriangleright (y_2, \boldsymbol{c}_{y_2}(g_2))$. ∎

In fact, $(P, \blacktriangleright)$ is the poset of simplices $\mathbf{Fac}(\mathbb{X})$ associated to our desired simplicial complex $\mathbb{X}$. Each pair $x = (y, \boldsymbol{c}_y(g))$ of P constitutes a simplex of dimension $\dim(y)$ in $\mathbb{X}$, and its faces are given by all pairs of the form $x' = (y', \boldsymbol{c}_{y'}(g \cdot \Phi_{y \succ y'}^{-1}))$, where y' is a face of y in $\mathbb{Y}$. We leave it to the reader to verify that the assignments $(y, \boldsymbol{c}_y(g)) \mapsto (y, \boldsymbol{c}_y(h \cdot g))$ parametrized by group elements $h \in G$ yield a regular group action of G on $\mathbb{X}$ with quotient $\mathbb{Y}$. The associated orbit map $\boldsymbol{p}_G : \mathbb{X} \to \mathbb{Y}$ is simply given by projecting onto the first factor, i.e., $\boldsymbol{p}_G(y, \boldsymbol{c}_y(g)) = y$.

4.2. Context. To establish that the G-action on $\mathbb{X}$ described above will produce $(\boldsymbol{A}, \Phi)$ when compressed along the lines of section 3, one chooses

(1) the lift $(y, \boldsymbol{c}_y(1_G))$ for each simplex y (where 1_G is the identity element of G), and
(2) the transfer $\Phi_{y \succ y'}$ for each face relation $y \succ y'$.

It remains to show that this G-action on $\mathbb{X}$ satisfies the universal property (involving uniqueness up to equivariant isomorphism) mentioned at the beginning of this section. For our purpose, it is convenient to not work directly with the action, but rather with equivalent information contained in the orbit map $\boldsymbol{p}_{\mathrm{G}} : \mathbb{X} \to \mathbb{Y}$. By the regularity assumption from Definition 2.1, $\boldsymbol{p}_{\mathrm{G}}$ forms a *stratified cover*[7] of $\mathbb{Y}$ so that G acts transitively on each fiber $\boldsymbol{p}_{\mathrm{G}}^{-1}(y)$. We will call such maps the *stratified G-covers of* $\mathbb{Y}$, and note that they correspond bijectively with regular simplicial group actions whose orbit space is $\mathbb{Y}$.

Definition 4.2. *The category of stratified G-covers of* $\mathbb{Y}$, *denoted* $\mathbf{Str}_{\mathrm{G}}(\mathbb{Y})$, *is defined by the following data:*
   (1) *its objects are all pairs* $(\mathbb{Z}, \boldsymbol{q})$ *consisting of finite simplicial complexes* $\mathbb{Z}$ *and stratified G-covers* $\boldsymbol{q} : \mathbb{Z} \to \mathbb{Y}$,
   (2) *the morphisms* $(\mathbb{Z}, \boldsymbol{q}) \to (\mathbb{Z}', \boldsymbol{q}')$ *are all simplicial maps* $\boldsymbol{s} : \mathbb{Z} \to \mathbb{Z}'$ *which make the evident triangle*[8] *commute:*

$$\mathbb{Z} \xrightarrow{\ \boldsymbol{s}\ } \mathbb{Z}'$$
$$\boldsymbol{q} \searrow \quad \swarrow \boldsymbol{q}'$$
$$\mathbb{Y}$$

Morphisms are composed in the usual manner for simplicial maps.

The following result establishes the desired universal property of the basic construction.

Proposition 4.3. *Any stratified G-cover* $\boldsymbol{q} : \mathbb{Z} \to \mathbb{Y}$, *for which the associated G-action on* $\mathbb{Z}$ *produces the compressed data* $(\boldsymbol{A}, \Phi)$, *is isomorphic to* $\boldsymbol{p}_{\mathrm{G}} : \mathbb{X} \to \mathbb{Y}$ *in* $\mathbf{Str}_{\mathrm{G}}(\mathbb{Y})$.

Proof. Given a simplex $(y, \boldsymbol{c}_y(g))$ of $\mathbb{X}$, let $z = z_y$ be the lift of y used to generate $(\boldsymbol{A}, \Phi)$ as described in section 3, so, in particular, $\boldsymbol{q}(z) = y$ and the group $\boldsymbol{A}_y$ equals the stabilizer G_z. Let $\boldsymbol{s} : \mathbb{X} \to \mathbb{Z}$ be the assignment $(y, \boldsymbol{c}_y(g)) \mapsto g \cdot z$, which we claim is the desired isomorphism. There are several properties to check, but all follow from routine calculations.
   (1) To see that $\boldsymbol{s}$ is *well-defined*, note that if $\boldsymbol{c}_y(g) = \boldsymbol{c}_y(g')$, then $g = g' \cdot k$ for some k in the stabilizer G_z (recall that Φ_y is the inclusion $\mathrm{G}_z \hookrightarrow \mathrm{G}$). Now, $g \cdot z = g' \cdot k \cdot z = g' \cdot z$, so the image of $(y, \boldsymbol{c}_y(g))$ depends only on the coset and not on g.
   (2) Similarly, if two simplices $(y, \boldsymbol{c}_y(g))$ and $(y', \boldsymbol{c}_{y'}(h))$ of $\mathbb{X}$ are sent by $\boldsymbol{s}$ to the same simplex z' of $\mathbb{Z}$, then we must have $y = y' = \boldsymbol{q}(z')$ and $z' = g \cdot z = h \cdot z$, which implies $\boldsymbol{c}_y(g) = \boldsymbol{c}_y(h)$ and hence that $\boldsymbol{s}$ is *injective*.
   (3) Given an arbitrary simplex z' in $\mathbb{Z}$, let $y = \boldsymbol{q}(z)$ have lift z. Since $\boldsymbol{q}$ is a stratified G-cover, G acts transitively on $\boldsymbol{q}^{-1}(y)$ and so there is some h in G for which $h \cdot z = z'$. By definition, we have $\boldsymbol{s}(y, \boldsymbol{c}_y(h)) = h \cdot z = z'$, so $\boldsymbol{s}$ is *surjective*.
   (4) To see that $\boldsymbol{s}$ is G-*equivariant*, pick h in G and note that

$$h \cdot (y, \boldsymbol{c}_y(g)) = (y, \boldsymbol{c}_y(h \cdot g)),$$

whose image under $\boldsymbol{s}$ is the desired $h \cdot g \cdot z = h \cdot \boldsymbol{s}(y, \boldsymbol{c}_y(g))$.

---

[7]By this we mean that the fiber of $\boldsymbol{p}_{\mathrm{G}}$ over any simplex y of $\mathbb{Y}$ forms a nonempty finite-sheeted covering space in the sense of [20, Chap. 1.1].

[8]If we regard G as acting trivially on $\mathbb{Y}$, then this triangle commutes G-equivariantly in the category of simplicial complexes.

(5) To see that $\boldsymbol{s}$ is a *simplicial* map, consider a relation $(y, \boldsymbol{c}_y(g)) \blacktriangleright (y', \boldsymbol{c}_{y'}(g'))$ in $\mathbb{X}$, so we have $y \succ y'$ in $\mathbb{Y}$ and $g \cdot \Phi_{y \succ y'}^{-1} = g' \cdot \Phi_{y'}(k)$ for some k in $\boldsymbol{A}_{y'}$. But since $\Phi_{y'}$ is just the inclusion $\boldsymbol{A}_{y'} \hookrightarrow \mathrm{G}$, we have $\Phi_{y'}(k) = k$. Let z and z' denote the lifts of y and y', so $\boldsymbol{A}_{y'}$ is the stabilizer $\mathrm{G}_{z'}$ (whence k fixes z'). By definition, the transfer $\Phi_{y \succ y'}$ satisfies $z \succ \Phi_{y \succ y'}^{-1} \cdot z'$. Now,

$$z \succ \Phi_{y \succ y'}^{-1} \cdot z' = g^{-1} \cdot g' \cdot k \cdot z' = g^{-1} \cdot g' \cdot z',$$

so, in fact, $\boldsymbol{s}(y, \boldsymbol{c}_y(g)) = g \cdot z$ admits $\boldsymbol{s}(y', \boldsymbol{c}_{y'}(g')) = g' \cdot z'$ as a face.
And finally, since $\boldsymbol{s} : \mathbb{X} \to \mathbb{Z}$ preserves fibers (i.e., maps $\boldsymbol{p}_{\mathrm{G}}^{-1}(y)$ to $\boldsymbol{q}^{-1}(y)$ for each simplex y of $\mathbb{Y}$), we also have $\boldsymbol{q} \circ \boldsymbol{s} = \boldsymbol{p}_{\mathrm{G}}$ as simplicial maps. Thus, $\boldsymbol{s}$ is the desired isomorphism from $\boldsymbol{p}_{\mathrm{G}}$ to $\boldsymbol{q}$ in $\mathbf{Str}_{\mathrm{G}}(\mathbb{Y})$. ∎

5. Algorithms. The efficiency (and, often, even the feasibility) of group-theoretic algorithms varies enormously with the data structures used to store groups on a computer.[9] Given this dependence, we will describe the requisite group-theoretic subroutines at a high level, and hope that their prospective implementer will be able to tailor data structures to specific choices of the acting group. Naive implementations which work for all finite groups are straightforward to specify, but they may be quite far from optimal in practice.

For similar reasons, we do not fix a particular data structure for representing the G-action. There are several reasonable options, each with its own relative (dis)advantages. For instance, one could employ a hash table with key/value assignments $(g, v) \mapsto g \cdot v$ ranging over group elements g in G and vertices v of $\mathbb{X}$.

Remark 5.1. Even with extremely naive implementations, *both algorithms described below can be made to run in quadratic time* in terms of the order of G and the dimension of $\mathbb{X}$. The bounds are recorded in Corollary 5.7 below.

5.1. Subroutines. We will assume the ability to evaluate results of the two standard group operations in G:
  (1) the function `prod` accepts $(g, h) \in \mathrm{G} \times \mathrm{G}$ and returns their product $g \cdot h$, and
  (2) the function `inv` accepts $g \in \mathrm{G}$ and returns its inverse g^{-1}.
Even for these simple operations, the data structure which holds G plays an essential role in determining the computational complexity. For instance, if G has been stored via its multiplication table, then `prod` incurs a constant cost; but if G is stored as a list of permutations, then this cost may be as large as $\boldsymbol{O}(|\mathrm{G}|)$. We will denote the complexity of a single product computation in G by r and a single inversion by i.

We also require a less standard function `minrep`. Fix, once and for all, an enumeration of the elements of G

$$\iota : \mathrm{G} \to \{1, 2, \ldots, |\mathrm{G}|\},$$

where $|\mathrm{G}|$ is the cardinality of G. It will be convenient to assume that the identity has minimal index (i.e., $\iota(1_{\mathrm{G}}) = 1$). With this preamble in place, `minrep` accepts as input a

---

[9]Finite groups may be stored on computers as multiplication tables, lists of permutations, lists of invertible matrices coming from a representation, or sets of generators and relations. The standard reference [21] devotes its entire third chapter to such considerations.

subgroup H $\leq$ G along with an element $g \in$ G, and returns the ι-minimal element contained in the corresponding left-coset $g \cdot$ H. In particular, if $g \in$ H, then $\mathtt{minrep}(H, g)$ returns the identity 1_G. A naive implementation of $\mathtt{minrep}$ would involve checking the index of each group element obtained by multiplying g with each element of H—this incurs a cost of $\boldsymbol{O}(|H|p)$. We will denote by m the worst-case complexity of executing $\mathtt{minrep}$ (across all choices of input H and g). Calling $\mathtt{minrep}$ on a fixed H with all choices of $g \in$ G and removing duplicates from the resulting outputs produces a *coset transversal* for G/H in the language of [21, Chap. 4.6.7].

The three subroutines above involved only the group G; these next three also require knowledge of how G acts on $\mathbb{X}$:

(1) $\mathtt{orb}$ takes as input a simplex x of $\mathbb{X}$ and returns the list of all simplices in the orbit Gx, which is always a subset of $\mathbb{X}$;

(2) $\mathtt{stab}$ takes as input a simplex x of $\mathbb{X}$ and returns its stabilizer G_x, which is always a subgroup of G; and

(3) $\mathtt{trans}$ takes as input two simplices x and x' of $\mathbb{X}$, and returns a group element g, if one exists, satisfying $g \cdot x = x'$.

We denote the worst-case complexity of these three algorithms by $o, s,$ and t, respectively. Again, these complexities will vary with how G and its action on $\mathbb{X}$ are stored on the computer.

5.2. The compression algorithm. Our first main algorithm, $\mathtt{Compress}$, takes as input the regular action of a finite group G on a finite simplicial complex $\mathbb{X}$. We will write $\mathbf{Sub}(G)$ to denote the set of all the subgroups of G.

Remark 5.2. The output of $\mathtt{Compress}(\mathbb{X}, G)$ is a triple $(\mathbb{Y}, S, T)$, where

(1) $\mathbb{Y}$ is the quotient simplicial complex $\mathbb{X}/G$; for each $k \geq 1$, define

$$\mathbb{Y}[k] = \{y_1 \succ y_2 \succ \cdots \succ y_k \mid y_i \in \mathbf{Fac}(\mathbb{Y}) \text{ and } \dim y_i - \dim y_j = j - i\},$$

(2) $S : \mathbb{Y}[1] \to \mathbf{Sub}(G)$ is a map that sends each simplex y of $\mathbb{Y}$ to a subgroup $S(y) \leq$ G, which is the stabilizer of a chosen lift of y, and

(3) $T : \mathbb{Y}[2] \to$ G is a map that sends each codimension one face relation $y_1 \succ y_2$ in $\mathbb{Y}$ to a transfer element $T(y_1 \succ y_2)$ in G.

As the algorithm executes, it visits all the simplices of $\mathbb{X}$ in ascending order with respect to dimension. In order to guarantee its termination, we initially mark all simplices as unvisited (via boolean variables, for instance) and allow the algorithm to mark simplices as visited once it has processed them. $\mathtt{Compress}$ also constructs two natural functions relating the input simplicial complex $\mathbb{X}$ and the output simplicial complex $\mathbb{Y}$:

(1) $p : \mathbb{X}[1] \to \mathbb{Y}[1]$ is the (simplicial) orbit map from section 2.1, while

(2) $\ell : \mathbb{Y}[1] \to \mathbb{X}[1]$ is the (not necessarily simplicial) assignment of lifts as in section 3.1.

Thus, the composite $p \circ \ell$ is always the identity map on $\mathbb{Y}$, while $\ell \circ p$ sends each simplex of

$\mathbb{X}$ to the lift chosen for its orbit class in $\mathbb{Y}$.

<div align="center">

Algorithm: Compress
Input: Regular G-action on $\mathbb{X}$
Output: Triple $(\mathbb{Y}, S, T)$

</div>

01	**for each** d in $(0, 1, \ldots, \dim \mathbb{X})$
02	**for each** unvisited d-simplex x in $\mathbb{X}$
03	**add** a new d-simplex y to $\mathbb{Y}$
04	**set** $\ell(y) = x$ and $S(y) = \texttt{stab}(x)$
05	**for each** x' in $\texttt{orb}(x)$
06	**mark** x' as visited
07	**set** $p(x') = y$
08	**for each** face $z \prec x$ in $\mathbb{X}$ of dim $d-1$
09	**set** $p(z)$ as a face of y in $\mathbb{Y}$
10	**set** $T(y \succ p(z)) = \texttt{trans}(z, \ell(p(z)))$
11	**if** $\mathbb{X}$ has no more unvisited simplices
12	**return** $(\mathbb{Y}, S, T)$

To confirm that the algorithm terminates, we note that line 06 marks every simplex in the orbit of a hitherto-unvisited simplex x as visited, and this x must lie in its own orbit. Thus, every simplex in $\mathbb{X}$ is eventually visited, at which point the *if* conditional in line 11 evaluates to true.

Remark 5.3. Before moving on to the reconstruction algorithm, we highlight two features of *compress* which significantly impact its computational complexity on distributed systems.

(1) Line 09 requires knowledge of p-images of lower-dimensional faces of x, which means that the outer *for* loop in line 01 cannot be parallelized—simplices of $\mathbb{X}$ must be processed in an order monotone with respect to their dimension.

(2) On the other hand, all simplices of a fixed dimension d can be processed in parallel provided that the $(d-1)$-simplices have already been processed. In other words, lines 02–10 may be distributed across several processors without loss of correctness.

(3) Although we will not use the following fact, we note that the inner *for* loops of lines 05–07 and 08–10 may also be parallelized—for a given simplex x, we are not required to process its orbit Gx or its faces $z \prec x$ in some prescribed serial order.

5.3. The reconstruction algorithm. Our second algorithm, `Reconstruct`, implements the basic construction of section 4.1. It accepts as input a triple $(\mathbb{Y}, S, T)$ produced by `Compress` (see Remark 5.2) along with the group G where S and T take their values. It returns a simplicial complex $\mathbb{Z}$ along with the desired regular G-action. In light of section 4, the d-simplices of $\mathbb{Z}$ will be stored as pairs of the form (y, g), where y is a d-simplex of $\mathbb{Y}$ and g is the ι-minimal representative of some left coset of $S(y)$ in G.

In addition to the subroutines already described, `Reconstruct` requires a purely combinatorial procedure, `uniqsort`. This takes in an unordered list of G-elements and returns a list sorted along the enumeration ι, with all duplicates removed. Line 03 uses this subroutine to find the ι-minimal representative of each left coset of the subgroup $S(y) \leq G$ (thus the size of M is exactly $|G|/|S(y)|$). Line 07 checks whether the group elements g' and $g \cdot T(y \succ y')$

lie in the same left coset of $S(y')$ in G—since $y \succ y'$ is enforced by the previous line, we have $(y, \boldsymbol{c}_y(g)) \blacktriangleright (y', \boldsymbol{c}_{y'}(g'))$ in the language of section 4, so we add the corresponding face relation to $\mathbb{Z}$ in line 08.

Remark 5.4. Since line 06 requires knowledge of all $(d-1)$-simplices present in $\mathbb{Z}$ before the d-simplex y can be processed, the outer *for* loop in line 01 cannot be parallelized. However, all simplices of a fixed dimension d may be processed at once, so the *for* loop of line 02 is easily parallelized.

<div align="center">

Algorithm: Reconstruct

In: $(\mathbb{Y}, S, T)$ and G

Out: Simplicial complex $\mathbb{Z}$

</div>

01	**for each** d in $(0, 1, \ldots, \dim \mathbb{Y})$
02	**for each** d-simplex y in $\mathbb{Y}$
03	M = `uniqsort`$(\{\, \texttt{minrep}(S(y), g) \mid g \in G \})$
04	**for each** g in M
05	**add** a d-simplex (y, g) to $\mathbb{Z}$
06	**for each** simplex (y', g') in $\mathbb{Z}$ with $(y \succ y')$ in $\mathbb{Y}[1]$
07	**if** `prod`$(g', \texttt{prod}(\texttt{inv}(g), T(y \succ y')))$ is in $S(y')$
08	**set** (y', c') as a face of (y, c) in $\mathbb{Z}$
09	**return** $\mathbb{Z}$

Recovering the G-action on $\mathbb{Z}$ is straightforward—since its simplices are stored as pairs of the form (y, g), for each $h \in G$ we have

$$h \cdot (y, g) = \big(y, \texttt{minrep}(S(y), \texttt{prod}(h, g))\big).$$

Thus, we recover not only $\mathbb{Z}$ but also a G-action on it. Proposition 4.3 guarantees that if the input $(\mathbb{Y}, S, T)$ to `Reconstruct` was produced by running `Compress` on a simplicial complex $\mathbb{X}$ with a regular G-action, then $\mathbb{X}$ and $\mathbb{Z}$ are G-equivariantly isomorphic.

5.4. Complexity analysis. We analyze both `Compress` and `Reconstruct` in terms of the complexity parameters p, i, m, o, s, and t associated with the six subroutines of section 5.1.

Theorem 5.5. *The computational complexity of running* `Compress` *for the regular action of a group* G *on a finite n-dimensional simplicial complex $\mathbb{X}$ is*

$$\boldsymbol{O}\Big((n+1) \cdot (s + o + f + t \cdot (n+1))\Big), \text{ where}$$

(1) *s, o, and t are the complexity parameters for* `stab`, `orb`, *and* `trans`, *while*

(2) *f is the maximal orbit-length[10] encountered among the simplices of $\mathbb{X}$,*

$$f = \max_{x \in \mathbb{X}}\{|Gx|\},$$

provided that the number of available processors exceeds the number of d-simplices in $\mathbb{X}$ for each dimension $d \in \{0, 1, \ldots, n-1, n\}$.

---

[10]Note that this f is also the maximal stabilizer index $\max_{x \in \mathbb{X}}\{[G : G_x]\}$ and the maximal fiber cardinality $\max_{y \in \mathbb{X}/G}\{|\boldsymbol{p}_G^{-1}(y)|\}$ of the orbit map.

Proof. The outer *for* loop in line 01 of `Compress` runs exactly $(n+1)$ times corresponding to d-values $\{0, 1, \ldots, n\}$. As noted in Remark 5.3, the d-simplices x of $\mathbb{X}$ can be processed independently of each other for each fixed d. So by our assumption on the number of processors, it suffices to only measure the complexity of executing lines 03–10 once. Now line 04 incurs the cost of running `stab` once, while the *for* loop in lines 05–07 runs `orb` once and iterates at most f times. Finally, the *for* loop in lines 08–09 runs once for each codimension one face of x and calls `trans` each time; there are at most $(n+1)$ such faces since $\dim x \leq \dim \mathbb{X} = n$. Combining these contributions, the cost of processing a single d-simplex x is

$$\boldsymbol{O}(s + o + f + t \cdot (n+1)).$$

The desired complexity estimate now follows from the fact that the outermost loop executes $(n+1)$ times. $\blacksquare$

Based on the preceding estimate, we expect that the cost of running `Compress` will be dominated by the calls to `trans` in line 10. It is therefore of compelling interest to optimize the implementation of `trans` to the largest extent possible. We recommend trying to find a large subset $\mathbb{T} \subset \mathbb{X}$ to populate many lifts $\ell(y)$ via a G-equivariant version of breadth-first search on $\mathbb{X}$ (when a simplex is visited, mark every simplex in its orbit as visited). The larger this $\mathbb{T}$, the more frequently we will have equalities $z = \ell(p(z))$ in line 10 of `Compress`. When its two inputs are equal, `trans` is allowed to simply return the identity 1_G and hence incur a constant cost.

Finally, we turn to `Reconstruct`.

Theorem 5.6. *The computational complexity of running* `Reconstruct` *on the input* $(\mathbb{Y}, S, T;$ G$)$ *when* $\mathbb{Y}$ *is a finite n-dimensional simplicial complex is*

$$\boldsymbol{O}\Big((n+1) \cdot \big[k \cdot (m + \log_2 k) + f \cdot (n+1) \cdot (2r + i + h)\big]\Big)$$

where

(1) $k = |G|$ *is the order of* G,
(2) $m, r,$ *and* i *are the complexity parameters for* `minrep`, `prod`, *and* `inv`, *while*
(3) $f = \max_{y \in \mathbb{Y}}\{k/|S(y)|\}$ *and* $h = \max_{y \in \mathbb{Y}}\{|S(y)|\}$ *(both are* $\leq k$*),*

provided that the number of available processors exceeds the number of d-dimensional simplices in $\mathbb{Y}$ for each d in $\{0, 1, \ldots, n-1, n\}$.

Proof. We examine the nested *for* loops from the inside out, starting with the innermost loop of lines 06–08. This loop runs once for each $(d-1)$-dimensional face y' of y, and since $\dim \mathbb{Y} = n$ there are at most $(n+1)$ such faces. For each such face, `prod` is invoked at most twice and `inv` at most once in line 07. We then check whether an element of G lies in $S(y')$, a list of size bounded by h. Thus, our innermost *for* loop incurs an $\boldsymbol{O}((n+1) \cdot (2r + i + h))$ cost per iteration. The intermediate *for* loop spanning lines 04–08 runs at most f times since f bounds from above the index $[G : S(y)]$ and hence the size of M. Thus, the cost of running this intermediate loop is $\boldsymbol{O}(f \cdot (n+1) \cdot (2r + i + h))$. By our assumption on the number of processors, it suffices to run the *for* loop of lines 02–08 only once per dimension d, and it remains to account for line 03. Here we first call `minrep` exactly k times (cost $\boldsymbol{O}(k \cdot m)$), and

then sort the resulting list of k outputs and remove duplicates (cost $\boldsymbol{O}(k \log_2 k)$). Thus, when processing simplices of a fixed dimension d, `Reconstruct` incurs a computational cost of

$$\boldsymbol{O}(k \cdot (m + \log_2 k) + f \cdot (n+1) \cdot (2r + i + h)).$$

Since the outermost *for* loop of lines 01–08 runs exactly $(n+1)$ times, we obtain the desired estimate. ∎

By overestimating most of the auxiliary complexity parameters described above in terms of the order of G, it becomes evident that both `compress` and `reconstruct` will run in polynomial time. As before, let k be the order of the acting group G. We can estimate the complexity parameters used in Theorems 5.5 and 5.6 as follows.

(1) Both r and i are bounded by k, since in the worst case we have to search every element of G to find a product or an inverse. Similarly, m is bounded above by k since in the worst case one may search all of G to find the minimal representative of a given coset.

(2) The parameter o is bounded by k since one only has to compute the action of every group element on a given simplex. For the same reason, the orbit-length parameters f (from both theorems) and h (from Theorem 5.6) are bounded by k.

(3) Meanwhile, both s and t are bounded by k since in the worst case one has to test every element of G to check whether it stabilizes a simplex or sends one simplex to another.

The following result follows directly from using these overestimates in the complexities from Theorems 5.5 and 5.6 along with the elementary properties of $\boldsymbol{O}(\cdot)$.

Corollary 5.7. *Let* G *be a group of order k that acts regularly on a finite simplicial complex* $\mathbb{X}$ *of dimension n. Assume that the number of available processors exceeds the number of simplices in* $\mathbb{X}$. *Then the computational complexity of running* `compress` *for this action is no worse than*

$$\boldsymbol{O}\Big(k \cdot (n+1)^2\Big).$$

And the complexity of running `reconstruct` *on the output of* `compress` *is no worse than*

$$\boldsymbol{O}\Big(k^2 \cdot (n+1)^2\Big).$$

REFERENCES

[1] E. BABSON AND D. N. KOZLOV, *Group actions on posets*, J. Algebra, 285 (2005), pp. 439–450.

[2] H. BASS, *Covering theory for graphs of groups*, J. Pure Appl. Algebra, 89 (1993), pp. 3–47.

[3] M. BELKIN AND P. NIYOGI, *Laplacian eigenmaps for dimensionality reduction and data representation*, Neural Comput., 15 (2003), pp. 1373–1396.

[4] W. BOSMA, J. CANNON, AND C. PLAYOUST, *The Magma algebra system. I. The user language*, J. Symbolic Comput., 24 (1997), pp. 235–265, https://doi.org/10.1006/jsco.1996.0125.

[5] G. E. BREDON, *Introduction to Compact Transformation Groups*, Pure Appl. Math. 46, Academic Press, New York, London, 1972.

[6] M. R. Bridson and A. Haefliger, *Metric Spaces of Non-positive Curvature*, Grundlehren Math. Wiss. 319, Springer-Verlag, Berlin, 1999.

[7] L. Carbone and E. Rips, *Reconstructing group actions*, Internat. J. Algebra Comput., 23 (2013), pp. 255–323.

[8] A. M. Cegarra, *Homotopy fiber sequences induced by 2-functors*, J. Pure Appl. Algebra, 215 (2011), pp. 310–334.

[9] I. Chevyrev and A. Kormilitzin, *A Primer on the Signature Method in Machine Learning*, preprint, https://arxiv.org/abs/1603.03788, 2016.

[10] J. M. Corson, *Complexes of groups*, Proc. London Math. Soc. (3), 65 (1992), pp. 199–224.

[11] J. Curry, *Sheaves, Cosheaves and Applications*, preprint, https://arxiv.org/abs/1303.3255, 2013.

[12] M. W. Davis, *The Geometry and Topology of Coxeter Groups*, Princeton University Press, Princeton, NJ, 2008.

[13] O. Delgado-Friedrichs, V. Robins, and A. Sheppard, *Skeletonization and partitioning of digital images using discrete Morse theory*, IEEE Trans. Pattern Anal. Mach. Intell., 37 (2014), pp. 654–666.

[14] W. Dicks and M. J. Dunwoody, *Groups Acting on Graphs*, Cambridge Stud. Adv. Math. 17, Cambridge University Press, Cambridge, 1989.

[15] T. Fiore, W. Lück, and R. Sauer, *Euler characteristics of categories and homotopy colimits*, Doc. Math., 16 (2011), pp. 301–354.

[16] T. Gao, J. Brodzki, and S. Mukherjee, *The Geometry of Synchronization Problems and Learning Group Actions*, preprint, https://arxiv.org/abs/1610.09051, 2016.

[17] R. Ghrist, *Barcodes: The persistent topology of data*, Bull. Amer. Math. Soc. (N.S.), 45 (2009), pp. 61–75.

[18] K. Grove and W. Ziller, *Polar manifolds and actions*, J. Fixed Point Theory Appl., 11 (2012), pp. 279–313.

[19] A. Haefliger, *Extensions of complexes of groups*, Ann. Inst. Fourier (Grenoble), 42 (1992), pp. 275–311.

[20] A. Hatcher, *Algebraic Topology*, Cambridge University Press, Cambridge, 2002.

[21] D. F. Holt, B. Eick, and E. A. O'Brien, *Handbook of Computational Group Theory*, Chapman and Hall/CRC, Boca Raton, FL, 2005.

[22] T. W. Hungerford, *Algebra*, Grad. Texts in Math. 73, Springer-Verlag, New York, Berlin, 1980.

[23] S. Krishnan, *Flow-Cut Dualities for Sheaves on Graphs*, preprint, https://arxiv.org/abs/1409.6712, 2014.

[24] S. Lack, *A 2-categories companion*, in Towards Higher Categories, J. C. Baez and J. P. May, eds., Springer, New York, 2010, pp. 105–191, https://doi.org/10.1007/978-1-4419-1524-5_4.

[25] T. Leinster, *Basic Bicategories*, preprint, https://arxiv.org/abs/math/9810017, 1998.

[26] S. Lim and A. Thomas, *Covering theory for complexes of groups*, J. Pure Appl. Algebra, 212 (2008), pp. 1632–1663, https://doi.org/10.1016/j.jpaa.2007.10.012.

[27] S. Mac Lane, *Categories for the Working Mathematician*, Grad. Texts in Math. 5, Springer-Verlag, New York, Berlin, 1971.

[28] N. Martins-Ferreira, *Pseudo-categories*, J. Homotopy Relat. Struct., 1 (2006), pp. 47–78.

[29] N. J. Mitra, L. Guibas, and M. Pauly, *Partial and approximate symmetry detection for 3D geometry*, in ACM SIGGRAPH 2006 Papers, Association for Computing Machinery, New York, 2006, pp. 560–568.

[30] I. Moerdijk and D. A. Pronk, *Orbifolds, sheaves and groupoids*, K-Theory, 12 (1997), pp. 3–21.

[31] I. Moerdijk and D. A. Pronk, *Simplicial cohomology of orbifolds*, Indag. Math. (N.S.), 10 (1999), pp. 269–293.

[32] B. Noohi, *Foundations of Topological Stacks* I, preprint, https://arxiv.org/abs/math/0503247, 2005.

[33] J.-P. Serre, *Trees*, translated from the French by John Stillwell, Springer-Verlag, Berlin, 1980.

[34] E. H. Spanier, *Algebraic Topology*, McGraw-Hill, New York, Toronto, London 1966.