Efficient 1-Laplacian Solvers for Well-Shaped Simplicial Complexes: Beyond Betti Numbers and Collapsing Sequences

Ming Ding
ETH Zürich, Switzerland

Peng Zhang
Rutgers University, New Brunswick, NJ, USA

Abstract

We present efficient algorithms for solving systems of linear equations in 1-Laplacians of well-shaped simplicial complexes. 1-Laplacians, or higher-dimensional Laplacians, generalize graph Laplacians to higher-dimensional simplicial complexes and play a key role in computational topology and topological data analysis. Previously, nearly-linear time solvers were developed for simplicial complexes with known collapsing sequences and bounded Betti numbers, such as those triangulating a three-ball in $\mathbb{R}^3$ (Cohen, Fasy, Miller, Nayyeri, Peng, and Walkington [SODA'2014], Black, Maxwell, Nayyeri, and Winkelman [SODA'2022], Black and Nayyeri [ICALP'2022]). Furthermore, Nested Dissection provides quadratic time solvers for more general systems with nonzero structures representing well-shaped simplicial complexes embedded in $\mathbb{R}^3$.

We generalize the specialized solvers for 1-Laplacians to simplicial complexes with additional geometric structures but *without collapsing sequences and bounded Betti numbers*, and we improve the runtime of Nested Dissection. We focus on simplicial complexes that meet two conditions: (1) each individual simplex has a bounded aspect ratio, and (2) they can be divided into "disjoint" and balanced regions with well-shaped interiors and boundaries. Our solvers draw inspiration from the Incomplete Nested Dissection for stiffness matrices of well-shaped trusses (Kyng, Peng, Schwieterman, and Zhang [STOC'2018]).

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms; Mathematics of computing → Computations on matrices; Mathematics of computing → Algebraic topology

Keywords and phrases 1-Laplacian Solvers, Simplicial Complexes, Incomplete Nested Dissection

Digital Object Identifier 10.4230/LIPIcs.ESA.2023.41

Related Version *Full Version*: `https://arxiv.org/abs/2302.06499` [20]

Funding This work was supported in part by NSF Grant CCF-2238682.

Acknowledgements We thank Rasmus Kyng for valuable discussions and the reviewers for their insightful comments.

1 Introduction

Combinatorial Laplacians generalize graph Laplacian matrices to higher dimensional simplicial complexes – a collection of 0-simplexes (vertices), 1-simplexes (edges), 2-simplexes (triangles), and their higher dimensional counterparts. Simplicial complexes encode higher-order relations between data points in a metric space. By studying the topological properties of these complexes using Combinatorial Laplacians, one can capture higher-order features that go beyond connectivity and clustering.

31st Annual European Symposium on Algorithms (ESA 2023).
Editors: Inge Li Gørtz, Martin Farach-Colton, Simon J. Puglisi, and Grzegorz Herman; Article No. 41; pp. 41:1–41:19
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Given an oriented d-dimensional simplicial complex $\mathcal{K}$, for each $0 \leq i \leq d$, let $\mathcal{C}_i$ be the vector space generated by the i-simplexes in $\mathcal{K}$ with coefficients in $\mathbb{R}$. We can define a sequence of boundary operators:

$$\mathcal{C}_d \xrightarrow{\partial_d} \mathcal{C}_{d-1} \xrightarrow{\partial_{d-1}} \cdots \xrightarrow{\partial_2} \mathcal{C}_1 \xrightarrow{\partial_1} \mathcal{C}_0,$$

where each ∂_i is a linear map that maps every i-simplex to a signed sum of its boundary $(i-1)$-faces. We define the i-Laplacian $\boldsymbol{L}_i : \mathcal{C}_i \to \mathcal{C}_i$ to be

$$\boldsymbol{L}_i = \partial_{i+1}\partial_{i+1}^\top + \partial_i^\top \partial_i. \tag{1}$$

In particular, ∂_1 is the vertex-edge incidence matrix, and $\boldsymbol{L}_0$ is the graph Laplacian (following the convention, we define $\partial_0 = \mathbf{0}$). One can assign weights to each simplex in $\mathcal{K}$ and define weighted Laplacians.

It is well-known that linear equations in graph Laplacians can be approximately solved in nearly-linear time in the number of nonzeros of the system [50, 30, 31, 29, 36, 46, 16, 34, 32, 27]. These fast Laplacian solvers have led to significant developments in algorithm design for graph problems such as maximum flow [41, 42, 10], minimum cost flow and lossy flow [37, 18], and graph sparsification [49], known as "the Laplacian Paradigm" [52].

Inspired by the success of graph Laplacians, Cohen, Fasy, Miller, Nayyeri, Peng, and Walkington [13] initiated the study of fast solvers for 1-Laplacian linear equations. They designed a nearly-linear time solver for simplicial complexes *with zero Betti numbers*[1] *and known collapsing sequences*. Later, Black, Maxwell, Nayyeri and Winkelman [5], and Black and Nayyeri [6] generalized this algorithm to subcomplexes of such a complex *with bounded first Betti numbers*[2]. One concrete example studied in these papers is convex simplicial complexes that piecewise linearly triangulate a convex ball in $\mathbb{R}^3$, for which a collapsing sequence exists and can be computed in linear time [11, 12]. However, deciding whether a simplicial complex has a collapsing sequence is NP-hard in general [51]; computing the Betti numbers is as hard as computing the ranks of general $\{0, 1\}$ matrices [23]. In addition, 1-Laplacian systems for general simplicial complexes embedded in $\mathbb{R}^4$ are as hard to solve as general sparse linear equations [19], for which the best-known algorithms need super-quadratic time [47, 44]. All the above motivates the following question:

> Can we efficiently solve 1-Laplacian systems for other classes of *structured* simplicial complexes, e.g., *without known collapsing sequences and with arbitrary Betti numbers*?

In addition to the specialized solvers for 1-Laplacian systems mentioned above, Nested Dissection can solve 1-Laplacian systems in quadratic time for simplicial complexes in $\mathbb{R}^3$ with additional geometric structures [25, 39, 43] such as bounded aspect ratios[3] of individual tetrahedrons. Furthermore, iterative methods such as Preconditioned Conjugate Gradient approximately solve 1-Laplacian systems in time $\widetilde{O}(n\sqrt{\kappa})$, where n is the number of simplexes and κ is the condition number of the coefficient matrix.

Inspired by solvers that leverage geometric structures and spectral properties, we develop efficient 1-Laplacian solvers for well-shaped simplicial complexes embedded in $\mathbb{R}^3$ *without known collapsing sequences and with arbitrary Betti numbers*. Our solver adapts the Incomplete Nested Dissection algorithm, proposed by Kyng, Peng, Schwieterman, and Zhang [33]

---

[1] Informally, the ith Betti number is the number of i-dimensional holes on a topological surface. For example, the zeroth, first, and second Betti numbers represent the numbers of connected components, one-dimensional "circular" holes, and two-dimensional "voids" or "cavities," respectively.

[2] The solver has cubic dependence on the first Betti number.

[3] The aspect ratio of a geometric shape S is the radius of the smallest ball containing S divided by the radius of the largest ball contained in S.

for solving linear equations in well-shaped 3-dimensional truss stiffness matrices. These matrices represent another generalization of graph Laplacians; however, they differ quite from the 1-Laplacians studied in this paper. A primary distinction is that the kernel of a truss stiffness matrix has an explicit and well-understood form, while computing a 1-Laplacian's kernel is as hard as that for a general matrix.

1.1 Our Results

We say a simplex is *stable* if it has $O(1)$ aspect ratio and $\Theta(1)$ weight. We focus on a *pure* simplicial complex[4] $\mathcal{K}$ embedded in $\mathbb{R}^3$. We require $\mathcal{K}$ admits a nice division parameterized by $r \in \mathbb{R}_+$, called *r-hollowing*. We adopt and adapt the concept of r-hollowing introduced in [33] to suit our 1-Laplacian solvers. Informally, our r-hollowing for a simplicial complex containing n simplexes divides $\mathcal{K}$ into $O(n/r)$ "separated" regions where each region has $O(r)$ simplexes and $O(r^{2/3})$ boundary simplexes. Only boundary simplexes can appear in multiple regions. Additionally, we mandate that each region's boundary triangulates a spherical shell in $\mathbb{R}^3$, exhibiting a *"hop" diameter* of $O(r^{1/3})$ and a *"hop" shell width* of at least 5. The formal definition is given in Definition 2.9. The bounded aspect ratio of each tetrahedron allows us to employ Nested Dissection for the interior simplexes within every region. The boundary shape requirement facilitates preconditioning the sub-system, derived from partial Nested Dissection, by the boundaries themselves and solving this sub-system using Preconditioned Conjugate Gradient.

Below, we present our main results informally. Firstly, we assume that an r-hollowing of a pure 3-complex is provided, which offers the broadest applicability of our algorithm. This assumption is justifiable when one can determine the construction of the simplicial complex; for instance, one can decide how to discretize a continuous topological space or how to triangulate a space given a set of points. Subsequently, we establish sufficient conditions for 3-complexes that allow us to compute r-hollowings in linear time.

▶ **Theorem 1.1** (Informal statement). *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes. Given an r-hollowing for $\mathcal{K}$, for any $\epsilon > 0$, we can approximately solve a system in the 1-Laplacian of $\mathcal{K}$ within error ϵ in time $O\left(nr + n^{4/3}r^{5/18}\log(n/\epsilon) + n^2r^{-2/3}\right)$. The runtime is $o(n^2)$ if $r = o(n)$ and $r = \omega(1)$. In particular, when $r = \Theta(n^{3/5})$, the runtime is minimized (up to constant) and equals $O(n^{8/5}\log(n/\epsilon))$.*

Our runtime in Theorem 1.1 does not depend on the Betti numbers of $\mathcal{K}$ and does not require collapsing sequences. When $r = o(n)$ and $r = \omega(1)$, the runtime is $o(n^2)$, asymptotically faster than Nested Dissection [43]. The solver in [6] for a 1-Laplacian system for the $\mathcal{K}$ stated in Theorem 1.1 is $\widetilde{O}(\beta^3 m)$[5], where m is the number of simplexes in $\mathcal{X} \supset \mathcal{K}$ with a known collapsing sequence and β is the first Betti number of $\mathcal{K}$. In the worst-case scenario, m can be as large as $\Omega(n^2)$. But [6] does not require a known r-hollowing.

Without assuming prior knowledge about r-hollowing, the following theorem presents a solver with the same runtime as Theorem 1.1 when the 3-complex $\mathcal{K}$ satisfies additional geometric restrictions: First, the convex hull of $\mathcal{K}$ has $O(1)$ aspect ratio, and each tetrahedron of $\mathcal{K}$ has $\Theta(1)$ volume. Second, all but one the boundary components of $\mathcal{K}$, which correspond to "holes inside" $\mathcal{K}$, satisfy the following conditions: (1) every boundary component of $\mathcal{K}$ has

[4] A simplicial complex is *pure* if every maximal simplex (i.e., a simplex that is not a proper subset of any other simplex in the complex) has the same dimension. For example, a pure 3-complex is a tetrahedron mesh that consists of tetrahedrons and their sub-simplexes.

[5] We use $\widetilde{O}(\cdot)$ to hide polylog factors on the number of simplexes and the inverse of error parameter.

1-skeleton diameter $O(r^{1/3})$; (2) the total size of boundary components within any $\mathbb{X} \subset \mathbb{R}^3$ of volume r is at most $O(r^{2/3})$, and the total size of boundary components of $\mathcal{K}$ is $O(nr^{-1/3})$; (3) the triangle distance between any two boundary components of $\mathcal{K}$ is greater than 5. These geometric conditions allow us to find an r-hollowing of $\mathcal{K}$ in linear time.

▶ **Theorem 1.2** (Informal statement). *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes; assume $\mathcal{K}$ satisfies the aforementioned additional geometric structures with parameter r. Then, for any $\epsilon > 0$, we can approximately solve a system in the 1-Laplacian of $\mathcal{K}$ within error ϵ in time $O\left(nr + n^{4/3}r^{5/18}\log(n/\epsilon) + n^2 r^{-2/3}\right)$. In particular, when $r = \Theta(n^{3/5})$, the runtime is minimized (up to constant) and equals $O(n^{8/5}\log(n/\epsilon))$.*

We then examine unions of pure 3-complexes glued together by identifying certain subsets of simplexes on the boundary components (called exterior simplexes) of 3-complex chunks. Moreover, each 3-complex chunk admits a $\Theta(n_i^{3/5})$-hollowing with n_i being the number of simplexes in this chunk. We remark that such a union of 3-complexes, called $\mathcal{U}$, *may not be embeddable in* $\mathbb{R}^3$. So, the previously established methods from [13, 5, 6] and Nested Dissection are unsuitable for this scenario. Building on our algorithm for Theorem 1.1, we design an efficient algorithm for $\mathcal{U}$ whose runtime depends sub-quadratically on the size of $\mathcal{U}$ and polynomially on the number of chunks and the number of simplexes shared by more than one chunk.

▶ **Theorem 1.3** (Informal statement). *Let $\mathcal{U}$ be a union of h pure 3-complexes that are glued together by identifying certain subsets of their exterior simplexes. Each 3-complex chunk is embedded in $\mathbb{R}^3$, contains n_i stable simplexes, and has a known $\Theta(n_i^{3/5})$-hollowing. For any $\epsilon > 0$, we can solve a system in the 1-Laplacian of $\mathcal{U}$ within error ϵ in time $\widetilde{O}(n^{8/5}k + h^2k^2 + k^3)$ where n is the number of simplexes in $\mathcal{U}$, k is the number of exterior simplexes shared by more than one complex chunk.*

When $h = \widetilde{O}(1)$ and $k = \widetilde{O}(n^{1/2})$, the solver in Theorem 1.3 has the same runtime as Theorem 1.1. When $h = o(n^{2/5}), k = o(n^{3/5})$, the runtime is $o(n^2)$, asymptotically faster than Nested Dissection.

1.2 Motivations and Applications

In the past decade, combinatorial Laplacians have played a crucial role in the development of computational topology and topological data analysis in various domains, such as statistics [28, 45], graphics and imaging [40, 53], brain networks [35], deep learning [8], signal processing [3], and cryo-electron microscope [54]. We recommend readers consult accessible surveys [26, 9, 22, 38] for more information.

Combinatorial Laplacians have their roots in the study of discrete Hodge decomposition [21], which states that the kernel of the i-Laplacian $\boldsymbol{L}_i$ is isomorphic to the ith homology group of the simplicial complex. Among the many applications of combinatorial Laplacians, a central problem is determining the Betti numbers – the ranks of the homology groups – which are important topological invariants. Additionally, discrete Hodge decomposition allows for the extraction of meaningful information from data by decomposing them into three mutually orthogonal components: gradient (in the image of $\partial_i^\top$), curl (in the image of ∂_{i+1}), and harmonic (in the kernel of $\boldsymbol{L}_i$) components. For instance, the three components of edge flows in a graph capture the global trends, local circulations, and "noise".

The computation of both Betti numbers and discrete Hodge decomposition of higher-order flows can be achieved by solving systems of linear equations in combinatorial Laplacians [24, 38]. The rank of a matrix $\boldsymbol{L}_i$ can be determined by solving a logarithmic number of linear equation systems in $\boldsymbol{L}_i$ [2]. The discrete Hodge decomposition can be calculated by solving least square problems involving boundary operators or combinatorial Laplacians, which in turn reduces to solving linear equations in these matrices.

Furthermore, an important question in numerical linear algebra concerns whether the nearly-linear time solvers for graph Laplacian linear equations can be generalized to larger classes of linear equations. Researchers have achieved success with elliptic finite element systems [7], Connection Laplacians [32], directed Laplacians [15, 14], well-shaped truss stiffness matrices [17, 48, 33]. It would be intriguing to determine what structures of linear equations facilitate faster solvers. Another theoretically compelling reason for developing efficient solvers for 1-Laplacians stems from the "equivalence" of time complexity between solving 1-Laplacian systems and general sparse systems of linear equations [19]. If one can solve all 1-Laplacian systems in time $\widetilde{O}((\text{\# of simplexes})^c)$ where $c \geq 1$ is a constant, then one can solve all general systems of linear equations in time $\widetilde{O}((\text{\# of nonzero coefficients})^c)$.

2 Preliminaries

2.1 Background of Linear Algebra

Given a vector $\boldsymbol{x} \in \mathbb{R}^n$, for $1 \leq i \leq n$, we let $\boldsymbol{x}[i]$ be the ith entry of $\boldsymbol{x}$; for $1 \leq i < j \leq n$, let $\boldsymbol{x}[i:j]$ be $(\boldsymbol{x}[i], \boldsymbol{x}[i+1], \ldots, \boldsymbol{x}[j])^\top$. The Euclidean norm of $\boldsymbol{x}$ is $\|\boldsymbol{x}\|_2 \stackrel{\text{def}}{=} \sqrt{\sum_{i=1}^n \boldsymbol{x}[i]^2}$. Given a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, for $1 \leq i \leq m, 1 \leq j \leq n$, we let $\boldsymbol{A}[i,j]$ be the (i,j)th entry of $\boldsymbol{A}$; for $S_1 \subseteq \{1,\ldots,m\}, S_2 \subseteq \{1,\ldots,n\}$, let $\boldsymbol{A}[S_1,S_2]$ be the submatrix with row indices in S_1 and column indices in S_2. Furthermore, we let $\boldsymbol{A}[S_1,:] = \boldsymbol{A}[S_1,\{1,\ldots,n\}]$ and $\boldsymbol{A}[:,S_2] = \boldsymbol{A}[\{1,\ldots,m\},S_2]$. The operator norm of $\boldsymbol{A}$ (induced by the Euclidean norm) is $\|\boldsymbol{A}\|_2 \stackrel{\text{def}}{=} \max_{\boldsymbol{v}\in\mathbb{R}^n} \frac{\|\boldsymbol{A}\boldsymbol{x}\|_2}{\|\boldsymbol{x}\|_2}$. The image of $\boldsymbol{A}$ is the linear span of the columns of $\boldsymbol{A}$, denoted by $\mathrm{Im}(\boldsymbol{A})$, and the kernel of $\boldsymbol{A}$ to be $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{A}\boldsymbol{x} = \boldsymbol{0}\}$, denoted by $\mathrm{Ker}(\boldsymbol{A})$. A fundamental theorem of Linear Algebra states $\mathbb{R}^m = \mathrm{Im}(\boldsymbol{A}) \oplus \mathrm{Ker}(\boldsymbol{A}^\top)$.

▶ **Fact 2.1.** [6] *For any matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $Im(\boldsymbol{A}) = Im(\boldsymbol{A}\boldsymbol{A}^\top)$.*

Pseudo-inverse and Projection Matrix

The pseudo-inverse of $\boldsymbol{A}$ is defined to be a matrix $\boldsymbol{A}^\dagger$ that satisfies all the following four criteria: (1) $\boldsymbol{A}\boldsymbol{A}^\dagger\boldsymbol{A} = \boldsymbol{A}$, (2) $\boldsymbol{A}^\dagger\boldsymbol{A}\boldsymbol{A}^\dagger = \boldsymbol{A}^\dagger$, (3) $(\boldsymbol{A}\boldsymbol{A}^\dagger)^\top = \boldsymbol{A}\boldsymbol{A}^\dagger$, (4) $(\boldsymbol{A}^\dagger\boldsymbol{A})^\top = \boldsymbol{A}^\dagger\boldsymbol{A}$. The orthogonal projection matrix onto $\mathrm{Im}(\boldsymbol{A})$ is $\boldsymbol{\Pi}_{\mathrm{Im}(\boldsymbol{A})} = \boldsymbol{A}(\boldsymbol{A}^\top\boldsymbol{A})^\dagger\boldsymbol{A}^\top$.

Eigenvalues and Condition Numbers

Given a square matrix $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, let $\lambda_{\max}(\boldsymbol{A})$ be the maximum eigenvalue of $\boldsymbol{A}$ and $\lambda_{\min}(\boldsymbol{A})$ the minimum *nonzero* eigenvalue of $\boldsymbol{A}$. The condition number of $\boldsymbol{A}$, denoted by $\kappa(\boldsymbol{A})$, is the ratio between $\lambda_{\max}(\boldsymbol{A})$ and $\lambda_{\min}(\boldsymbol{A})$. A symmetric matrix $\boldsymbol{A}$ is *positive semi-definite (PSD)* if all eigenvalues of $\boldsymbol{A}$ are non-negative. Let $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ be another square matrix. We say $\boldsymbol{A} \succcurlyeq \boldsymbol{B}$ if $\boldsymbol{A} - \boldsymbol{B}$ is PSD. The *condition number of* $\boldsymbol{A}$ *relative to* $\boldsymbol{B}$ is

$$\kappa(\boldsymbol{A},\boldsymbol{B}) \stackrel{\text{def}}{=} \min\left\{\frac{\alpha}{\beta} : \beta\boldsymbol{\Pi}_{\mathrm{Im}(\boldsymbol{A})}\boldsymbol{B}\boldsymbol{\Pi}_{\mathrm{Im}(\boldsymbol{A})} \preccurlyeq \boldsymbol{A} \preccurlyeq \alpha\boldsymbol{B}\right\}.$$

[6] All the facts in this section are well-known. For completeness, we include their proofs in the Appendix of the full paper [20].

▶ **Fact 2.2.** *Let $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$ be symmetric matrices such that $\boldsymbol{A} \preccurlyeq \boldsymbol{B}$. Then, for any $\boldsymbol{V} \in \mathbb{R}^{m\times n}$, $\boldsymbol{V}\boldsymbol{A}\boldsymbol{V}^\top \preccurlyeq \boldsymbol{V}\boldsymbol{B}\boldsymbol{V}^\top$.*

Schur Complement

Let $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, and let $F \cup C$ be a partition of $\{1, \dots, n\}$. We write $\boldsymbol{A}$ as a block matrix:

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}[F,F] & \boldsymbol{A}[F,C] \\ \boldsymbol{A}[C,F] & \boldsymbol{A}[C,C] \end{pmatrix}. \tag{2}$$

We define the (generalized) Schur complement of $\boldsymbol{A}$ onto C to be

$$\mathrm{Sc}[\boldsymbol{A}]_C = \boldsymbol{A}[C,C] - \boldsymbol{A}[C,F]\boldsymbol{A}[F,F]^\dagger \boldsymbol{A}[F,C].$$

The Schur complement appears in performing a block Gaussian elimination on matrix $\boldsymbol{A}$ to eliminate the indices in F.

▶ **Fact 2.3.** *Let $\boldsymbol{A}$ be a PSD matrix defined in Equation (2). Then,*

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{I} & \\ \boldsymbol{A}[C,F]\boldsymbol{A}[F,F]^\dagger & \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{A}[F,F] & \\ & Sc[\boldsymbol{A}]_C \end{pmatrix} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{A}[F,F]^\dagger \boldsymbol{A}[F,C] \\ & \boldsymbol{I} \end{pmatrix}.$$

▶ **Fact 2.4.** *Let $\boldsymbol{A}$ be a PSD matrix defined in Equation (2). Let $\boldsymbol{A} = \boldsymbol{B}\boldsymbol{B}^\top$, and we decompose $\boldsymbol{B} = \begin{pmatrix} \boldsymbol{B}_F \\ \boldsymbol{B}_C \end{pmatrix}$ accordingly. Then, $Sc[\boldsymbol{A}]_C = \boldsymbol{B}_C \boldsymbol{\Pi}_{Ker(\boldsymbol{B}_F)} \boldsymbol{B}_C^\top$, where $\boldsymbol{\Pi}_{Ker(\boldsymbol{B}_F)}$ is the projection onto the kernel of $\boldsymbol{B}_F$.*

Solving Linear Equations

We will need Fact 2.5 for relations between different error notations for linear equations and Theorem 2.6 for Preconditioned Conjugate Gradient.

▶ **Fact 2.5.** *Let $\boldsymbol{A}, \boldsymbol{Z} \in \mathbb{R}^{n\times n}$ be two symmetric PSD matrices, and let $\boldsymbol{\Pi}$ be the orthogonal projection onto $Im(\boldsymbol{A})$.*

1. *If $(1-\epsilon)\boldsymbol{A}^\dagger \preccurlyeq \boldsymbol{Z} \preccurlyeq (1+\epsilon)\boldsymbol{A}^\dagger$, then $\|\boldsymbol{A}\boldsymbol{Z}\boldsymbol{b} - \boldsymbol{b}\|_2 \le \epsilon\sqrt{\kappa(\boldsymbol{A})}\,\|\boldsymbol{b}\|_2$ for any $\boldsymbol{b} \in Im(\boldsymbol{A})$.*
2. *If $\|\boldsymbol{A}\boldsymbol{Z}\boldsymbol{b} - \boldsymbol{b}\|_2 \le \epsilon\,\|\boldsymbol{b}\|_2$ for any $\boldsymbol{b} \in Im(\boldsymbol{A})$, then $(1-\epsilon)\boldsymbol{A}^\dagger \preccurlyeq \boldsymbol{\Pi}\boldsymbol{Z}\boldsymbol{\Pi} \preccurlyeq (1+\epsilon)\boldsymbol{A}^\dagger$.*

▶ **Theorem 2.6** (Preconditioned Conjugate Gradient [1])**.** *Let $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$ be two symmetric PSD matrices, and let $\boldsymbol{b} \in \mathbb{R}^n$. Each iteration of Preconditioned Conjugate Gradient multiplies one vector with $\boldsymbol{A}$, solves one system of linear equations in $\boldsymbol{B}$, and performs a constant number of vector operations. For any $\epsilon > 0$, the algorithm outputs an $\boldsymbol{x}$ satisfying $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{\Pi}_{\boldsymbol{A}}\boldsymbol{b}\|_2 \le \epsilon\,\|\boldsymbol{\Pi}_{\boldsymbol{A}}\boldsymbol{b}\|_2$ in $O(\sqrt{\kappa}\log(\kappa/\epsilon))$ such iterations, where $\boldsymbol{\Pi}_{\boldsymbol{A}}$ is the orthogonal projection matrix onto the image of $\boldsymbol{A}$ and $\kappa = \kappa(\boldsymbol{A}, \boldsymbol{B})$.*

2.2 Background of Topology

Simplex and Simplicial Complexes

We consider a *d-simplex* (or d-dimensional simplex) σ as an ordered set of $d+1$ vertices, denoted by $\sigma = [v_0, \dots, v_d]$. A *face* of σ is a simplex obtained by removing a subset of vertices from σ. A *simplicial complex* $\mathcal{K}$ is a finite collection of simplexes such that (1) for every $\sigma \in \mathcal{K}$ if $\tau \subset \sigma$ then $\tau \in \mathcal{K}$, and (2) for every $\sigma_1, \sigma_2 \in \mathcal{K}$, $\sigma_1 \cap \sigma_2$ is either empty or a face of both σ_1, σ_2. The *dimension* of $\mathcal{K}$ is the maximum dimension of any simplex in $\mathcal{K}$. A *d-complex* is a d-dimensional simplicial complex. For $1 \le i \le d$, the *i-skeleton* of a d-complex $\mathcal{K}$ is the subcomplex consisting of all the simplexes of $\mathcal{K}$ of dimensions at most i. In particular, the 1-skeleton of $\mathcal{K}$ is a graph.

A *piecewise linear embedding* of a 3-complex in $\mathbb{R}^3$ maps a 0-simplex to a point, a 1-simplex to a line segment, a 2-simplex to a triangle, and a 3-simplex to a tetrahedron. In addition, the interior of the images of simplices are disjoint and the boundary of each simplex is mapped to the appropriate simplices. Such an embedding of a simplicial complex $\mathcal{K}$ defines an *underlying topological space* $\mathbb{K}$ – the union of the images of all the simplexes of $\mathcal{K}$. We say $\mathcal{K}$ is *convex* if $\mathbb{K}$ is convex. We say $\mathcal{K}$ *triangulates* a topological space $\mathbb{X}$ if $\mathbb{K}$ is homeomorphic to $\mathbb{X}$. A simplex σ of $\mathcal{K}$ is a *exterior* simplex if σ is contained in the boundary of $\mathbb{K}$, and σ is an *interior* simplex otherwise. A connected component of exterior simplexes is called a *boundary component* of $\mathcal{K}$.

The *aspect ratio* of a set $S \subset \mathbb{R}^3$ is the radius of the smallest ball containing S divided by the radius of the largest ball contained in S. The aspect ratio of S is always greater than or equal to 1. We say a simplex σ is *stable* if it has $O(1)$ aspect ratio and $\Theta(1)$ weight. Miller and Thurston proved the following lemma. As a corollary, the numbers of the vertices, the edges, the triangles, and the tetrahedrons of a 3-complex $\mathcal{K}$ that is composed of stable tetrahedrons are all equal up to a constant factor.

▶ **Lemma 2.7** (Lemma 4.1 of [43]). *Let $\mathcal{K}$ be a 3-complex in $\mathbb{R}^3$ in which each tetrahedron has $O(1)$ aspect ratio. Then, each vertex of $\mathcal{K}$ is contained in at most $O(1)$ tetrahedrons.*

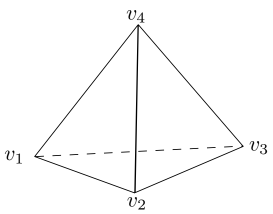
Boundary Operators

An *i-chain* is a weighted sum of the oriented i-simplexes in $\mathcal{K}$ with the coefficients in $\mathbb{R}$. Let $\mathcal{C}_i$ denote the ith chain space. The *boundary operator* is a linear map $\partial_i : \mathcal{C}_i \to \mathcal{C}_{i-1}$ such that for an oriented i-simplex $\sigma = [v_0, v_1, \ldots, v_i]$,

$$\partial_i(\sigma) = \sum_{j=0}^{i} (-1)^j [v_0, \ldots, \hat{v}_j, \ldots, v_i],$$

where $[v_0, \ldots, \hat{v}_j, \ldots, v_i]$ is the oriented $(i-1)$-simplex obtained by removing v_j from σ.

The operator ∂_i can be written as a matrix in $|\mathcal{C}_{i-1}| \times |\mathcal{C}_i|$ dimensions, where the (r, l)th entry of ∂_i is ± 1 if the rth $(i-1)$-simplex is a face of the lth i-simplex and 0 otherwise. See Figure 1 for an example.



$$\partial_2 = \begin{array}{c} \\ {[v_1,v_2]} \\ {[v_1,v_3]} \\ {[v_1,v_4]} \\ {[v_2,v_3]} \\ {[v_2,v_4]} \\ {[v_3,v_4]} \end{array} \begin{array}{c} \begin{array}{cccc} [v_1,v_4,v_2] & [v_2,v_4,v_3] & [v_1,v_3,v_4] & [v_1,v_2,v_3] \end{array} \\ \left[\begin{array}{cccc} -1 & 0 & 0 & 1 \\ 0 & 0 & 1 & -1 \\ 1 & 0 & -1 & 0 \\ 0 & -1 & 0 & 1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \end{array}\right] \end{array}$$

Figure 1 An example of boundary operator. The left side is a 3-simplex (a tetrahedron) with vertices v_1, v_2, v_3, v_4. The right side is the corresponding second boundary operator ∂_2, where each column corresponds to an oriented 2-simplex (a triangle) and each row corresponds to an oriented 1-simplex (an edge).

An important property of boundary operators is $\partial_i \partial_{i+1} = \mathbf{0}$, which implies $\mathrm{Im}(\partial_{i+1}) \subseteq \mathrm{Ker}(\partial_i)$. So, we can define the quotient space $H_i = \mathrm{Ker}(\partial_i) \backslash \mathrm{Im}(\partial_{i+1})$, called the *ith homology space* of $\mathcal{K}$. The rank of H_i is called the ith Betti number of $\mathcal{K}$. If the ith Betti number of $\mathcal{K}$ is 0, then $\mathrm{Im}(\partial_i^\top) \oplus \mathrm{Im}(\partial_{i+1}) = \mathbb{R}^{|\mathcal{C}_i|}$. The first and second Betti numbers of a triangulation of a three-ball are both 0.

Hodge Decomposition and Combinatorial Laplacians

Combinatorial Laplacians arise from the discrete Hodge decomposition.

▶ **Theorem 2.8** (Hodge decomposition [38]). *Let $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ and $\boldsymbol{B} \in \mathbb{R}^{n \times p}$ be matrices satisfying $\boldsymbol{AB} = \mathbf{0}$. Then, there is an orthogonal direct sum decomposition*

$$\mathbb{R}^n = Im(\boldsymbol{A}^\top) \oplus Ker(\boldsymbol{A}^\top \boldsymbol{A} + \boldsymbol{B}\boldsymbol{B}^\top) \oplus Im(\boldsymbol{B}).$$

Since $\partial_i \partial_{i+1} = \mathbf{0}$, it is valid to set $\boldsymbol{A} = \partial_i$ and $\boldsymbol{B} = \partial_{i+1}$. The matrix we get in the middle term is the *combinatorial Laplacian*: $\boldsymbol{L}_i \stackrel{\text{def}}{=} \partial_i^\top \partial_i + \partial_{i+1} \partial_{i+1}^\top$.

The weighted combinatorial Laplacian generalizes combinatorial Laplacian. For each $1 \le i \le d$, we assign each i-simplex of $\mathcal{K}$ with a *positive* weight, and let $\boldsymbol{W}_i : \mathcal{C}_i \to \mathcal{C}_i$ be a diagonal matrix where $\boldsymbol{W}_i[\sigma, \sigma]$ is the weighted of the i-simplex σ. Then the weighted *i-Laplacian* of $\mathcal{K}$ is a linear operator $\boldsymbol{L}_i : \mathcal{C}_i \to \mathcal{C}_i$ defined as

$$\boldsymbol{L}_i \stackrel{\text{def}}{=} \partial_i^\top \boldsymbol{W}_{i-1} \partial_i + \partial_{i+1} \boldsymbol{W}_{i+1} \partial_{i+1}^\top.$$

Note that Hodge decomposition also applies to weighted combinatorial Laplacian (by setting $\boldsymbol{A} = \boldsymbol{W}_{i-1}^{1/2} \partial_i$ and $\boldsymbol{B} = \partial_{i+1} \boldsymbol{W}_{i+1}^{1/2}$, we have $\boldsymbol{AB} = \mathbf{0}$). We call $\boldsymbol{L}_i^{\text{down}} \stackrel{\text{def}}{=} \partial_i^\top \boldsymbol{W}_{i-1} \partial_i$ the ith *down-Laplacian* operator and $\boldsymbol{L}_i^{\text{up}} \stackrel{\text{def}}{=} \partial_{i+1} \boldsymbol{W}_{i+1} \partial_{i+1}^\top$ the ith *up-Laplacian* operator. Sometimes, we use subscripts to specify the complex on which these operators are defined: $\partial_{i,\mathcal{K}}$, $\boldsymbol{W}_{i,\mathcal{K}}$, $\boldsymbol{L}_{i,\mathcal{K}}^{\text{down}}$, $\boldsymbol{L}_{i,\mathcal{K}}^{\text{up}}$.

r-Hollowings

Let $\mathcal{K}$ be a pure 3-complex with n simplexes. A set of triangles $\hat{\triangle}_1, \ldots, \hat{\triangle}_k$ form a *triangle path* of length $k-1$ if for any $1 \le i \le k-1$, $\hat{\triangle}_i$ and $\hat{\triangle}_{i+1}$ share an edge. The *triangle distance* between two triangles $\triangle_1$ and $\triangle_2$ is the shortest triangle path length between $\triangle_1$ and $\triangle_2$. The *triangle diameter* of $\mathcal{K}$ is the longest triangle distance between any two triangles. A *spherical shell* is $\{\boldsymbol{x} \in \mathbb{R}^3 : R_1 \le \|\boldsymbol{x}\|_2 \le R_2\}$ where $R_1 < R_2$. If $\mathcal{K}$ triangulates a spherical shell, we define the *shell width* to be the shortest triangle distance between any two triangles where one is on the outer sphere and one is on the inner sphere.

▶ **Definition 2.9** (r-hollowing). *Let $\mathcal{K}$ be a 3-complex with n simplexes, and let $r = o(n)$ be a positive number. We divide $\mathcal{K}$ into $O(n/r)$* regions *each of $O(r)$ simplexes and $O(r^{2/3})$ boundary simplexes. Only boundary simplexes can appear in more than one region. The boundary of each region triangulates a spherical shell in $\mathbb{R}^3$ and has triangle diameter $O(r^{1/3})$ and shell width at least 5. The union of all boundary simplexes of each region is referred to as an* r-hollowing *of $\mathcal{K}$.*

In addition, this paper also examines sufficient conditions for 3-complexes that enable us to compute an r-hollowing in linear time (Algorithm 2 and refer to Figure 2 for an illustration). Specifically, we consider a pure 3-complex $\mathcal{K}$ embedded in $\mathbb{R}^3$ with n stable simplexes possessing the following additional geometric structures:

1. The aspect ratio of the convex hull of $\mathcal{K}$ is $O(1)$, and the volume of each tetrahedron in $\mathcal{K}$ is $\Theta(1)$.
2. All but one boundary component has 1-skeleton diameter $O(r^{1/3})$.
3. The total number of exterior simplexes of $\mathcal{K}$ within any $\mathbb{X} \subset \mathbb{R}^3$ of volume r is $O(r^{2/3})$; the total number of exterior simplexes of $\mathcal{K}$ is $O(nr^{-1/3})$.
4. The triangle distance between any two boundary components of $\mathcal{K}$ is greater than 5.

It is worth noting that fulfilling the aforementioned assumptions is not excessively challenging. On one end of the spectrum, there are scenarios where $\mathcal{K}$ contains at most $O(n/r)$ 2-dimensional holes, each with an interior volume of $O(r)$. On the other end, there are instances where $\mathcal{K}$ encompasses $O(nr^{-1/3})$ uniformly distributed small holes, each with a constant interior volume. Moreover, it is likely that all scenarios lying between these extremes would also meet these assumptions.

3 Main Theorems

We formally state our main results as follows.

▶ **Theorem 3.1.** *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ consisting of n stable simplexes and with a known r-hollowing. Let $\boldsymbol{L}_1$ be the 1-Laplacian operator of $\mathcal{K}$, and let $\boldsymbol{\Pi}_1$ be the orthogonal projection matrix onto the image of $\boldsymbol{L}_1$. For any vector $\boldsymbol{b}$ and $\epsilon > 0$, we can find a solution $\tilde{\boldsymbol{x}}$ such that $\|\boldsymbol{L}_1\tilde{\boldsymbol{x}} - \boldsymbol{\Pi}_1\boldsymbol{b}\|_2 \leq \epsilon\|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2$ in time $O\left(nr + n^{4/3}r^{5/18}\log(n/\epsilon) + n^2r^{-2/3}\right)$.*

We will overview our algorithm for Theorem 3.1 in Section 4 and prove in Section 5.

▶ **Theorem 3.2.** *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ consisting of n stable simplexes. Suppose $\mathcal{K}$ satisfies the additional geometric structures 1-4 with parameter $r = o(n)$. Let $\boldsymbol{L}_1$ be the 1-Laplacian operator of $\mathcal{K}$, and let $\boldsymbol{\Pi}_1$ be the orthogonal projection matrix onto the image of $\boldsymbol{L}_1$. For any vector $\boldsymbol{b}$ and $\epsilon > 0$, we can find a solution $\tilde{\boldsymbol{x}}$ such that $\|\boldsymbol{L}_1\tilde{\boldsymbol{x}} - \boldsymbol{\Pi}_1\boldsymbol{b}\|_2 \leq \epsilon\|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2$ in time $O\left(nr + n^{4/3}r^{5/18}\log(n/\epsilon) + n^2r^{-2/3}\right)$.*

The known r-hollowing assumption is replaced with geometric structures in Theorem 3.2, and a linear time algorithm for finding r-hollowing is presented in Section 6. It is worth mentioning that the additional geometric structures are introduced to ensure the feasibility of finding an r-hollowing in linear time. However, the algorithm for solving the system of linear equations remains the same.

▶ **Theorem 3.3.** *Let $\mathcal{U}$ be a union of h pure 3-complexes glued together by identifying certain subsets of their exterior simplexes. Each 3-complex chunk is embedded in $\mathbb{R}^3$ and comprises n_i stable simplexes, and has a known $\Theta(n_i^{3/5})$-hollowing. Let $\boldsymbol{L}_1$ be the 1-Laplacian operator of $\mathcal{U}$, and let $\boldsymbol{\Pi}_1$ be the orthogonal projection matrix onto the image of $\boldsymbol{L}_1$. For any vector $\boldsymbol{b}$ and $\epsilon > 0$, we can find a solution $\tilde{\boldsymbol{x}}$ such that $\|\boldsymbol{L}_1\tilde{\boldsymbol{x}} - \boldsymbol{\Pi}_1\boldsymbol{b}\|_2 \leq \epsilon\|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2$ in time $\widetilde{O}(n^{8/5}k + h^2k^2 + k^3)$, where n is the number of simplexes in $\mathcal{U}$, k is the number of exterior simplexes shared by more than one chunk.*

Due to space constraints, the proof of Theorem 3.3 can be found in the full version of the paper [20].

4 Algorithm Overview

Cohen, Fasy, Miller, Nayyeri, Peng, and Walkington [13] observed that

$$\boldsymbol{L}_1^{\dagger} = \left(\boldsymbol{L}_1^{\mathrm{down}}\right)^{\dagger} + (\boldsymbol{L}_1^{\mathrm{up}})^{\dagger},$$

where $\boldsymbol{L}_1^{\mathrm{down}} = \partial_1^{\top} \boldsymbol{W}_0 \partial_1$ is the down-Laplacian and $\boldsymbol{L}_1^{\mathrm{up}} = \partial_2 \boldsymbol{W}_2 \partial_2^{\top}$ is the up-Laplacian. The orthogonal projection matrices onto $\mathrm{Im}(\partial_1^{\top})$ and $\mathrm{Im}(\partial_2)$ are:

$$\boldsymbol{\Pi}_1^{\mathrm{down}} \stackrel{\mathrm{def}}{=} \partial_1^{\top}(\partial_1\partial_1^{\top})^{\dagger}\partial_1, \ \boldsymbol{\Pi}_1^{\mathrm{up}} \stackrel{\mathrm{def}}{=} \partial_2(\partial_2^{\top}\partial_2)^{\dagger}\partial_2^{\top}.$$

▶ **Lemma 4.1** (Lemma 4.1 of [13]). *Let $\boldsymbol{b}$ be a vector. Consider the systems of linear equations: $\boldsymbol{L}_1\boldsymbol{x} = \boldsymbol{\Pi}_1\boldsymbol{b}$, $\boldsymbol{L}_1^{up}\boldsymbol{x}^{up} = \boldsymbol{\Pi}_1^{up}\boldsymbol{b}$, $\boldsymbol{L}_1^{down}\boldsymbol{x}^{down} = \boldsymbol{\Pi}_1^{down}\boldsymbol{b}$. Then, $\boldsymbol{x} = \boldsymbol{\Pi}_1^{up}\boldsymbol{x}^{up} + \boldsymbol{\Pi}_1^{down}\boldsymbol{x}^{down}$.*

Lemma 4.1 implies that four operators are needed to approximate $\boldsymbol{L}_1^{\dagger}$: (1) an approximate projection operator $\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}} \approx \boldsymbol{\Pi}_1^{\mathrm{down}}$, (2) an approximate projection operator $\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}} \approx \boldsymbol{\Pi}_1^{\mathrm{up}}$, (3) a down-Laplacian solver $\boldsymbol{Z}_1^{\mathrm{down}}$ such that $\boldsymbol{L}_1^{\mathrm{down}}\boldsymbol{Z}_1^{\mathrm{down}}\boldsymbol{b} \approx \boldsymbol{b}$ for any $\boldsymbol{b} \in \mathrm{Im}(\boldsymbol{L}_1^{\mathrm{up}})$, and (4) an up-Laplacian solver $\boldsymbol{Z}_1^{\mathrm{up}}$ such that $\boldsymbol{L}_1^{\mathrm{up}}\boldsymbol{Z}_1^{\mathrm{up}}\boldsymbol{b} \approx \boldsymbol{b}$ for any $\boldsymbol{b} \in \mathrm{Im}(\boldsymbol{L}_1^{\mathrm{up}})$.

We will apply the same approximate orthogonal projection $\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}}$ given in [13], which does not depend on Betti numbers. Our solver for the down 1-Laplacian is a slight modification of the one in [13] to incorporate the simplex weights. We state the two lemmas below.

▶ **Lemma 4.2** (Down-projection operator, Lemma 3.2 of [13]). *Let $\mathcal{K}$ be a 3-complex with n simplexes. For any $\epsilon > 0$, there exists a linear operator $\widetilde{\boldsymbol{\Pi}}_1^{down}$ such that*

$$(1-\epsilon)\boldsymbol{\Pi}_1^{down} \preccurlyeq \widetilde{\boldsymbol{\Pi}}_1^{down}(\epsilon) \preccurlyeq \boldsymbol{\Pi}_1^{down}.$$

▶ **Lemma 4.3** (Down-Laplacian solver). *Let $\mathcal{K}$ be a weighted simplicial complex, and let $\boldsymbol{b} \in Im(\boldsymbol{L}_1^{down})$. There exists an operator $\boldsymbol{Z}_1^{down}$ such that $\boldsymbol{L}_1^{down}\boldsymbol{Z}_1^{down}\boldsymbol{b} = \boldsymbol{b}$. In addition, we can compute $\boldsymbol{Z}_1^{down}\boldsymbol{b}$ in linear time.*

4.1 Solver for Up-Laplacian

One of our primary technical contributions is the development of an efficient solver for the up-Laplacian system, stated in Lemma 4.4. We will describe the key idea behind our solver in this section.

▶ **Lemma 4.4** (Up-Laplacian solver). *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes. Suppose we are given an r-hollowing for $\mathcal{K}$. Then for any $\epsilon > 0$, there exists an operator $\boldsymbol{Z}_1^{up}$ such that*

$$\forall \boldsymbol{b} \in Im(\boldsymbol{L}_1^{up}), \ \left\|\boldsymbol{L}_1^{up}\boldsymbol{Z}_1^{up}\boldsymbol{b} - \boldsymbol{b}\right\|_2 \leq \epsilon \left\|\boldsymbol{b}\right\|_2.$$

In addition, $\boldsymbol{Z}_1^{up}\boldsymbol{b}$ can be computed in time $O\left(nr + n^{4/3}r^{5/18}\log(n/\epsilon) + n^2r^{-2/3}\right)$.

We remark that Lemma 4.4 can be improved to $\widetilde{O}(n^{3/2})$ by using a slightly different r-hollowing (proved in the full version [20]), which might be of independent interest. Since the bottleneck of our solver for 1-Laplacians is from the projection for up 1-Laplaicans, we use the same r-hollowing here.

The given $O(n^{3/5})$-hollowing suggests a partition of the edges in $\mathcal{K}$ into $F \cup C$. We will explain the concrete partition shortly. We have the following matrix identity:

$$\boldsymbol{L}_1^{\mathrm{up}} = \begin{pmatrix} \boldsymbol{I} & \\ \boldsymbol{L}_1^{\mathrm{up}}[C,F]\boldsymbol{L}_1^{\mathrm{up}}[F,F]^{\dagger} & \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{L}_1^{\mathrm{up}}[F,F] & \\ & \mathrm{Sc}[\boldsymbol{L}_1^{\mathrm{up}}]_C \end{pmatrix} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{L}_1^{\mathrm{up}}[F,F]^{\dagger}\boldsymbol{L}_1^{\mathrm{up}}[F,C] \\ & \boldsymbol{I} \end{pmatrix},$$

where

$$\mathrm{Sc}[\boldsymbol{L}_1^{\mathrm{up}}]_C = \boldsymbol{L}_1^{\mathrm{up}}[C,C] - \boldsymbol{L}_1^{\mathrm{up}}[C,F]\boldsymbol{L}_1^{\mathrm{up}}[F,F]^{\dagger}\boldsymbol{L}_1^{\mathrm{up}}[F,C].$$

The following Lemma 4.5 reduces (approximately) solving a system in $\boldsymbol{L}_1^{\mathrm{up}}$ to (approximately) solving two systems in $\boldsymbol{L}_1^{\mathrm{up}}[F,F]$ and one system in $\mathrm{Sc}[\boldsymbol{L}_1^{\mathrm{up}}]_C$, whose proof can be found in the Appendix of the full version [20]. It is worth noting that Lemma 4.5 holds if we replace $\boldsymbol{L}_1^{\mathrm{up}}$ with an arbitrary symmetric PSD matrix, and we will apply it or its variants for different PSD matrices in our solvers. To avoid introducing additional notations, we state the lemma below in terms of $\boldsymbol{L}_1^{\mathrm{up}}$.

▶ **Lemma 4.5.** *Suppose we have two operators (1) UpLapFSolver$(\cdot)$ such that given any $\boldsymbol{b} \in Im(\boldsymbol{L}_1^{up}[F,F])$, UpLapFSolver$(\boldsymbol{b})$ returns a vector $\boldsymbol{x}$ satisfying $\boldsymbol{L}_1^{up}[F,F]\boldsymbol{x} = \boldsymbol{b}$, and (2) SchurSolver$(\cdot,\cdot)$ such that for any $\boldsymbol{h} \in Im(Sc[\boldsymbol{L}_1^{\mathrm{up}}]_C)$ and $\delta > 0$, SchurSolver$(\boldsymbol{h},\delta)$ returns $\tilde{\boldsymbol{x}}$ satisfying $\|Sc[\boldsymbol{L}_1^{up}]_C\tilde{\boldsymbol{x}} - \boldsymbol{h}\|_2 \le \delta\|\boldsymbol{h}\|_2$. Given any $\boldsymbol{b} = \begin{pmatrix}\boldsymbol{b}_F\\ \boldsymbol{b}_C\end{pmatrix} \in Im(\boldsymbol{L}_1^{up})$ and any $\epsilon > 0$, let*

$$\begin{aligned}
&\boldsymbol{h} = \boldsymbol{b}_C - \boldsymbol{L}_1^{up}[C,F]\cdot \textsc{UpLapFSolver}(\boldsymbol{b}_F),\\
&\tilde{\boldsymbol{x}}_C = \textsc{SchurSolver}(\boldsymbol{h},\delta),\\
&\tilde{\boldsymbol{x}}_F = \textsc{UpLapFSolver}\left(\boldsymbol{b}_F - \boldsymbol{L}_1^{up}[F,C]\tilde{\boldsymbol{x}}_C\right),
\end{aligned} \tag{3}$$

where $\delta \le \frac{\epsilon}{1+\left\|\boldsymbol{L}_1^{up}[C,F]\boldsymbol{L}_1^{up}[F,F]^{\dagger}\right\|_2}$. Then,

$$\|\boldsymbol{L}_1^{up}\tilde{\boldsymbol{x}} - \boldsymbol{b}\|_2 \le \epsilon\|\boldsymbol{b}\|_2,$$

where $\tilde{\boldsymbol{x}} = \begin{pmatrix}\tilde{\boldsymbol{x}}_F\\ \tilde{\boldsymbol{x}}_C\end{pmatrix}$. Let $m_F = |F|$ and $m_C = |C|$, and let UpLapFSolver have runtime $t_1(m_F)$ and SchurSolver have runtime $t_2(m_C)$. Then, we can compute $\tilde{\boldsymbol{x}}$ in time $O(t_1(m_F) + t_2(m_C) + m_F + m_C)$.

4.1.1 Partitioning the Edges

As suggested by Lemma 4.5, we want to partition the edges of $\mathcal{K}$ into $F \cup C$ so that both systems in $\boldsymbol{L}_1^{\mathrm{up}}[F,F]$ and the Schur complement $\mathrm{Sc}[\boldsymbol{L}_1^{\mathrm{up}}]_C$ can be efficiently solved. The given $O(n^{3/5})$-hollowing divides $\mathcal{K}$ into *"disjoint" and balanced* regions with *small* boundary. Let F be the set of the "interior" edges of the regions and C be the set of the "boundary" edges.

We first show the interiors of different regions are "disjoint" in the sense that $\boldsymbol{L}_1^{\mathrm{up}}[F,F]$ is a block diagonal matrix where each diagonal block corresponds to the interior of a region. We can write $\boldsymbol{L}_1^{\mathrm{up}}$ as the sum of rank-1 matrices that each corresponds to a triangle in $\mathcal{K}$:

$$\boldsymbol{L}_1^{\mathrm{up}} = \partial_2\boldsymbol{W}_2\partial_2^{\top} = \sum_{\sigma:\text{triangle in }\mathcal{K}} \boldsymbol{W}_2[\sigma,\sigma]\cdot\partial_2[:,\sigma]\partial_2[:,\sigma]^{\top}. \tag{4}$$

For any two edges e_1, e_2, $\boldsymbol{L}_1^{\mathrm{up}}[e_1,e_2] = 0$ if and only if no triangle in $\mathcal{K}$ contains both e_1, e_2. By our definition of r-hollowing in Definition 2.9, for different regions R_1, R_2 of $\mathcal{K}$ w.r.t. an r-hollowing, no triangle contains both an edge from R_1 and an edge from R_2.

In addition, the following lemma shows that the boundaries of the regions well approximate the Schur complement onto the boundaries. We give a formal proof of Lemma 4.6 in the full version [20].

▶ **Lemma 4.6** (Spectral bounds for r-hollowing)**.** *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ composed of stable simplexes. Let $\mathcal{T}$ be an r-hollowing of $\mathcal{K}$, and let C be the edges of $\mathcal{T}$. Then, $\boldsymbol{L}_{1,\mathcal{T}}^{up} \preccurlyeq Sc[\boldsymbol{L}_1^{up}]_C \preccurlyeq O(r)\boldsymbol{L}_{1,\mathcal{T}}^{up}$.*

4.1.2 Proof of Lemma 4.4 for Up-Laplacian Solver

Algorithm 1 sketches a pseudo-code for our up-Laplacian solver.

Algorithm 1 UpLapSolver($\mathcal{K}, \mathcal{T}, \boldsymbol{b}, \epsilon$).

Input: A pure 3-complex $\mathcal{K}$ of n stable simplexes with up-Laplacian $\boldsymbol{L}_1^{\mathrm{up}}$, an $O(n^{3/5})$-hollowing $\mathcal{T}$, a vector $\boldsymbol{b} \in \mathrm{Im}(\boldsymbol{L}_1^{\mathrm{up}})$, an error parameter $\epsilon > 0$

Output: An approximate solution $\tilde{\boldsymbol{x}}$ such that $\|\boldsymbol{L}_1^{\mathrm{up}}\tilde{\boldsymbol{x}} - \boldsymbol{b}\|_2 \leq \epsilon \|\boldsymbol{b}\|_2$

1 $F \leftarrow$ the interior edges of regions of $\mathcal{K}$ w.r.t. $\mathcal{T}$, $C \leftarrow$ the boundary edges of regions.
2 UpLapFSolver($\cdot$)$\leftarrow$ a solver by Nested Dissection that satisfies the requirement in Lemma 4.5.
3 SchurSolver($\cdot,\cdot$)$\leftarrow$ a solver by Preconditioned Conjugate Gradient with the preconditioner being the up-Laplacian of $\mathcal{T}$ that satisfies the requirement in Lemma 4.5.
4 $\tilde{\boldsymbol{x}} \leftarrow$ computed by Equation (3)
5 **return** *solution* $\tilde{\boldsymbol{x}}$

By Lemma 4.5, the $\tilde{\boldsymbol{x}}$ returned by Algorithm 1 satisfies $\|\boldsymbol{L}_1^{\mathrm{up}}\tilde{\boldsymbol{x}} - \boldsymbol{b}\|_2 \leq \epsilon \|\boldsymbol{b}\|_2$. To bound the runtime of Algorithm 1, we need the following lemmas for lines 2 and 3.

▶ **Lemma 4.7** (Solver for the "F" part). *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes. Let $\mathcal{T}$ be an r-hollowing of $\mathcal{K}$, and let F be the set of interior edges in each region of $\mathcal{K}$ w.r.t. $\mathcal{T}$. Then, with a pre-processing time $O(nr)$, there exists a solver UpLapFSolver($\cdot$) such that given any $\boldsymbol{b}_F \in im(\boldsymbol{L}_1^{up}[F,F])$, UpLapFSolver($\boldsymbol{b}_F$) returns an $\boldsymbol{x}_F$ such that $\boldsymbol{L}_1^{up}[F,F]\boldsymbol{x}_F = \boldsymbol{b}_F$ in time $O(nr^{1/3})$.*

By our choice of F, the matrix $\boldsymbol{L}_1^{\mathrm{up}}[F,F]$ can be written as a block diagonal matrix where each block corresponds to a region of $\mathcal{K}$ w.r.t. the r-hollowing $\mathcal{T}$. Since each region is a 3-complex in which every tetrahedron has an aspect ratio $O(1)$, we can construct the solver UpLapFSolver by Nested Dissection [43]. However, since each row or column of $\boldsymbol{L}_1^{\mathrm{up}}[F,F]$ corresponds to an edge, we need to turn the good *vertex separators* in [43] into good *edge separators* for regions of $\mathcal{K}$. The proof of Lemma 4.7 can be found in the full version [20].

▶ **Lemma 4.8** (Solver for the Schur complement). *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes. Let $\mathcal{T}$ be an r-hollowing of $\mathcal{K}$, and let C be the set of boundary edges of each region of $\mathcal{K}$ w.r.t. $\mathcal{T}$. Then, with a pre-processing time $O(nr + n^2r^{-2/3})$ there exists a solver SchurSolver($\cdot,\cdot$) such that for any $\boldsymbol{h} \in Im(Sc[\boldsymbol{L}_1^{up}]_C)$ and $\delta > 0$, SchurSolver($\boldsymbol{h},\delta$) returns an $\tilde{\boldsymbol{x}}_C$ such that $\|Sc[\boldsymbol{L}_1^{up}]_C\tilde{\boldsymbol{x}}_C - \boldsymbol{h}\|_2 \leq \delta \|\boldsymbol{h}\|_2$ in time $\widetilde{O}\left(nr^{5/6} + n^{4/3}r^{5/18}\right)$.*

Our solver SchurSolver is based on the Preconditioned Conjugate Gradient (PCG) with the preconditioner $\boldsymbol{L}_{1,\mathcal{T}}^{\mathrm{up}}$, the up-Laplacian operator of $\mathcal{T}$. By Theorem 2.6 and Lemma 4.6, the number of PCG iterations is $\widetilde{O}(\sqrt{r})$. In each PCG iteration, we solve the system in $\boldsymbol{L}_{1,\mathcal{T}}^{\mathrm{up}}$ via Nested Dissection. Again, the proof of Lemma 4.8 can be found in the full version of the paper [20].

Given the above lemmas, we prove Lemma 4.4.

Proof of Lemma 4.4. The correctness of Algorithm 1 is by Lemma 4.5. By Lemma 4.7 and 4.8, the total runtime of the algorithm is

$$\widetilde{O}\left(nr^{5/6} + n^{4/3}r^{5/18} + nr + n^2r^{-2/3}\right). \quad ◀$$

4.2 Projection for Up 1-Laplacian

As the first Betti number of $\mathcal{K}$ can be arbitrary, the approximate projection operators for the up 1-Laplacian provided in [13, 5, 6] are not applicable here. Our approximate projection operator follows a similar approach to our up 1-Laplacian solver, which is based on an incomplete Nested Dissection for *triangles*, instead of edges.

▶ **Lemma 4.9** (Up-projection operator). *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes. Suppose we are given an r-hollowing for $\mathcal{K}$. Then, for any $\epsilon > 0$, there exists an operator $\widetilde{\boldsymbol{\Pi}}_1^{up}$ such that*

$$\forall \boldsymbol{b}, \left\| \widetilde{\boldsymbol{\Pi}}_1^{up} \boldsymbol{b} - \boldsymbol{\Pi}_1^{up} \boldsymbol{b} \right\|_2 \leq \epsilon \left\| \boldsymbol{\Pi}_1^{up} \boldsymbol{b} \right\|_2 .$$

In addition, $\widetilde{\boldsymbol{\Pi}}_1^{up} \boldsymbol{b}$ can be computed in time $O\left(nr + n^{4/3} r^{5/18} \log(n/\epsilon) + n^2 r^{-2/3}\right)$.

The proof of Lemma 4.9 can be found in the full paper [20]. The subsequent Lemma offers a helpful formula for $\boldsymbol{\Pi}_1^{\mathrm{up}}$, the orthogonal projection matrix onto the image of $\boldsymbol{L}_1^{\mathrm{up}}$.

▶ **Lemma 4.10.** *Let $\mathcal{K}$ be a simplicial complex with boundary operator ∂_2. For any partition $F \cup C$ of the 2-simplexes of $\mathcal{K}$, the orthogonal projection $\boldsymbol{\Pi}_1^{up}$ for $\mathcal{K}$ can be decomposed as*

$$\boldsymbol{\Pi}_1^{up} = \boldsymbol{\Pi}_{Im(\partial_2[:,F])} + \boldsymbol{\Pi}_{Ker(\partial_2^\top[F,:])} \partial_2[:,C] (Sc[\boldsymbol{L}_2^{down}]_C)^\dagger \partial_2^\top [C,:] \boldsymbol{\Pi}_{Ker(\partial_2^\top [F,:])},$$

where $\boldsymbol{L}_2^{down}$ is the down 2-Laplacian.

Once more, an r-hollowing offers a natural partition of the triangles within $\mathcal{K}$. We assign all the "interior" triangles to F and all the "boundary" triangles to C. As such, Nested Dissection can be utilized to compute $\boldsymbol{\Pi}_{\mathrm{Im}(\partial_2[:,F])}$ and $\boldsymbol{\Pi}_{\mathrm{Ker}(\partial_2^\top[F,:])}$. The primary technical challenge arises when solving a system in the Schur complement $\mathrm{Sc}[\boldsymbol{L}_2^{\mathrm{down}}]_C$. We precondition it using the boundary $\partial_2^\top[C,:]\partial_2[:,C]$ and apply Preconditioned Conjugate Gradient, which requires a distinct approach to bound the relative condition numbers.

5 Proof of Main Theorem 3.1

Given all the four operators in Lemma 4.2, 4.3, 4.4, and 4.9, we prove Theorem 3.1.

Proof of Theorem 3.1. Let κ be the maximum of $\kappa(\boldsymbol{L}_1^{\mathrm{down}})$ and $\kappa(\boldsymbol{L}_1^{\mathrm{up}})$. Let $\delta > 0$ be a parameter to be determined later. Let $\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}} = \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}}(\delta), \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}} = \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}}(\delta)$ be defined in Lemma 4.2 and 4.9, and let $\boldsymbol{Z}_1^{\mathrm{down}}$ be the operator in Lemma 4.3 with no error and $\boldsymbol{Z}_1^{\mathrm{up}}$ in Lemma 4.4 with error δ. Let

$$\begin{aligned}
&\tilde{\boldsymbol{b}}^{\mathrm{up}} \stackrel{\mathrm{def}}{=} \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}} \boldsymbol{b}, \ \tilde{\boldsymbol{b}}^{\mathrm{down}} \stackrel{\mathrm{def}}{=} \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}} \boldsymbol{b}, \\
&\widetilde{\boldsymbol{x}}^{\mathrm{up}} \stackrel{\mathrm{def}}{=} \boldsymbol{Z}_1^{\mathrm{up}} \tilde{\boldsymbol{b}}^{\mathrm{up}}, \ \widetilde{\boldsymbol{x}}^{\mathrm{down}} \stackrel{\mathrm{def}}{=} \boldsymbol{Z}_1^{\mathrm{down}} \tilde{\boldsymbol{b}}^{\mathrm{down}}, \\
&\tilde{\boldsymbol{x}} \stackrel{\mathrm{def}}{=} \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}} \widetilde{\boldsymbol{x}}^{\mathrm{up}} + \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}} \widetilde{\boldsymbol{x}}^{\mathrm{down}}.
\end{aligned}$$

Then,

$$\begin{aligned}
&\left\| \boldsymbol{L}_1 \tilde{\boldsymbol{x}} - \boldsymbol{\Pi}_1 \boldsymbol{b} \right\|_2 \\
&\quad \leq \left\| \boldsymbol{L}_1^{\mathrm{up}} \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}} \widetilde{\boldsymbol{x}}^{\mathrm{up}} - \tilde{\boldsymbol{b}}^{\mathrm{up}} \right\|_2 + \left\| \boldsymbol{L}_1^{\mathrm{down}} \widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}} \widetilde{\boldsymbol{x}}^{\mathrm{down}} - \tilde{\boldsymbol{b}}^{\mathrm{down}} \right\|_2 + \left\| \tilde{\boldsymbol{b}}^{\mathrm{up}} + \tilde{\boldsymbol{b}}^{\mathrm{down}} - \boldsymbol{\Pi}_1 \boldsymbol{b} \right\|_2 .
\end{aligned}$$

- For the first term,

$$\left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2 \leq \left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \boldsymbol{L}_1^{\mathrm{up}}\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}}\right\|_2 + \left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2.$$

  By Lemma 4.9,

$$\begin{aligned}\left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \boldsymbol{L}_1^{\mathrm{up}}\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}}\right\|_2 &\leq \|\boldsymbol{L}_1^{\mathrm{up}}\|_2 \left\|(\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}} - \boldsymbol{\Pi}_1^{\mathrm{up}})\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}}\right\|_2 \\ &\leq \delta \|\boldsymbol{L}_1^{\mathrm{up}}\|_2 \|\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}}\|_2 .\end{aligned}$$

  Let $\boldsymbol{y} \stackrel{\mathrm{def}}{=} (\boldsymbol{L}_1^{\mathrm{up}})^{\dagger}\tilde{\boldsymbol{b}}^{\mathrm{up}}$. By Lemma 4.4,

$$\|\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \boldsymbol{y}\|_2 \leq \left\|(\boldsymbol{L}_1^{\mathrm{up}})^{\dagger}\right\|_2 \left\|\boldsymbol{L}_1^{\mathrm{up}}\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2 \leq \delta \left\|(\boldsymbol{L}_1^{\mathrm{up}})^{\dagger}\right\|_2 \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2.$$

  By the triangle inequality,

$$\|\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}}\|_2 \leq \|\boldsymbol{y}\|_2 + \delta \left\|(\boldsymbol{L}_1^{\mathrm{up}})^{\dagger}\right\|_2 \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2 \leq (1+\delta) \left\|(\boldsymbol{L}_1^{\mathrm{up}})^{\dagger}\right\|_2 \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2.$$

  So, $\left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \boldsymbol{L}_1^{\mathrm{up}}\boldsymbol{\Pi}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}}\right\|_2 \leq \delta(1+\delta)\kappa \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2$.
  By Lemma 4.4, $\left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2 \leq \delta \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2$.
  So, $\left\|\boldsymbol{L}_1^{\mathrm{up}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{up}}\widetilde{\boldsymbol{x}}^{\mathrm{up}} - \tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2 \leq 3\delta\kappa \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2$.
- For the second term, the operator $\boldsymbol{Z}_1^{\mathrm{down}}$ has no error, which means $\boldsymbol{L}_1^{\mathrm{down}}\widetilde{\boldsymbol{x}}^{\mathrm{down}} = \tilde{\boldsymbol{b}}^{\mathrm{down}}$. Then,

$$\begin{aligned}\left\|\boldsymbol{L}_1^{\mathrm{down}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}}\widetilde{\boldsymbol{x}}^{\mathrm{down}} - \tilde{\boldsymbol{b}}^{\mathrm{down}}\right\|_2 &= \left\|\boldsymbol{L}_1^{\mathrm{down}}\widetilde{\boldsymbol{\Pi}}_1^{\mathrm{down}}\widetilde{\boldsymbol{x}}^{\mathrm{down}} - \boldsymbol{L}_1^{\mathrm{down}}\widetilde{\boldsymbol{x}}^{\mathrm{down}}\right\|_2 \\ &\leq \delta(1+\delta)\kappa \left\|\tilde{\boldsymbol{b}}^{\mathrm{down}}\right\|_2 .\end{aligned}$$

- For the third term,

$$\begin{aligned}\left\|\tilde{\boldsymbol{b}}^{\mathrm{up}} + \tilde{\boldsymbol{b}}^{\mathrm{down}} - \boldsymbol{\Pi}_1\boldsymbol{b}\right\|_2^2 &= \left\|(\widetilde{\boldsymbol{\Pi}}^{\mathrm{up}} - \boldsymbol{\Pi}^{\mathrm{up}})\boldsymbol{b}\right\|_2^2 + \left\|(\widetilde{\boldsymbol{\Pi}}^{\mathrm{down}} - \boldsymbol{\Pi}^{\mathrm{down}})\boldsymbol{b}\right\|_2^2 \\ &\leq \delta^2\left(\|\boldsymbol{\Pi}^{\mathrm{up}}\boldsymbol{b}\|_2^2 + \left\|\boldsymbol{\Pi}^{\mathrm{down}}\boldsymbol{b}\right\|_2^2\right) \quad \text{(by Lemma 4.2, 4.9, Fact 2.5)} \\ &= \delta^2 \|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2^2 .\end{aligned}$$

  Combining all the above inequalities,

$$\begin{aligned}\|\boldsymbol{L}_1\tilde{\boldsymbol{x}} - \boldsymbol{\Pi}_1\boldsymbol{b}\|_2 &\leq 3\delta\kappa \left\|\tilde{\boldsymbol{b}}^{\mathrm{up}}\right\|_2 + 2\delta\kappa \left\|\tilde{\boldsymbol{b}}^{\mathrm{down}}\right\|_2 + \delta \|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2 \\ &\leq 3\delta\kappa(1+\delta) \|\boldsymbol{\Pi}_1^{\mathrm{up}}\boldsymbol{b}\|_2 + 2\delta\kappa(1+\delta) \left\|\boldsymbol{\Pi}_1^{\mathrm{down}}\boldsymbol{b}\right\|_2 + \delta \|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2 \\ &\leq 11\delta\kappa \|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2 .\end{aligned}$$

  Choosing $\delta \leq \frac{\epsilon}{11\kappa}$, we have

$$\|\boldsymbol{L}_1\tilde{\boldsymbol{x}} - \boldsymbol{\Pi}_1\boldsymbol{b}\|_2 \leq \epsilon \|\boldsymbol{\Pi}_1\boldsymbol{b}\|_2 .$$ ◀

6 Computing an r-Hollowing

In this section, we describe a linear time algorithm (Algorithm 2) that finds an r-hollowing of a pure 3-complex $\mathcal{K}$ embedded in $\mathbb{R}^3$ with n stable simplexes that satisfies the additional geometric structures stated at the end of Section 2. We restate them below:

1. The aspect ratio of the convex hull of $\mathcal{K}$ is $O(1)$, and the volume of each tetrahedron in $\mathcal{K}$ is $\Theta(1)$.
2. All but one boundary component has 1-skeleton diameter $O(r^{1/3})$.
3. The total number of exterior simplexes of $\mathcal{K}$ within any $\mathbb{X} \subset \mathbb{R}^3$ of volume r is $O(r^{2/3})$; the total number of exterior simplexes of $\mathcal{K}$ is $O(nr^{-1/3})$.
4. The triangle distance between any two boundary components of $\mathcal{K}$ is greater than 5.

▶ **Lemma 6.1** (Finding an r-hollowing)**.** *Let $\mathcal{K}$ be a pure 3-complex embedded in $\mathbb{R}^3$ and composed of n stable simplexes. If $\mathcal{K}$ possesses additional geometric structures 1-4 with parameter $r = o(n)$, then we can find an r-hollowing of $\mathcal{K}$ in linear time.*

In the rest of the section, we will show Algorithm 2 satisfies Lemma 6.1. Let $\mathbb{K}$ be the convex hull of the underlying topological space of $\mathcal{K}$. Algorithm 2 first finds a *nice bounding box* – a box encompasses $\mathbb{K}$ and its volume and aspect ratio are within constant factors of those of $\mathbb{K}$. Lemma 6.3 provides a linear time algorithm for finding a nice bounding box for $\mathbb{K}$ when the aspect ratio of $\mathbb{K}$ is $O(1)$. Then, Algorithm 2 "cuts" the bounding box into $O(n/r)$ smaller boxes of equal volume using 2-dimensional planes and turns these cutting planes into an r-hollowing. Figure 2 illustrates the process of finding an r-hollowing.

We need the following lemma from [4] to construct a nice bounding box.

▶ **Lemma 6.2** (Lemma 3.4 of [4])**.** *Given a set X of points in $\mathbb{R}^3$, we can compute in linear time a bounding box $\mathcal{B}$ with $vol(\mathcal{B}) \leq 6\sqrt{6} vol(\mathcal{B}^*)$, where $vol(\cdot)$ is the volume and $\mathcal{B}^*$ is a bounding box of X with the minimum volume.*

▶ **Corollary 6.3** (Nice bounding box)**.** *Let $\mathcal{K}$ be a 3-complex embedded in $\mathbb{R}^3$ whose underlying topological space has aspect ratio $O(1)$. We can compute a nice bounding box of the convex hull of $\mathcal{K}$ in linear time.*

The proof of Corollary 6.3 can be found in the full version [20].

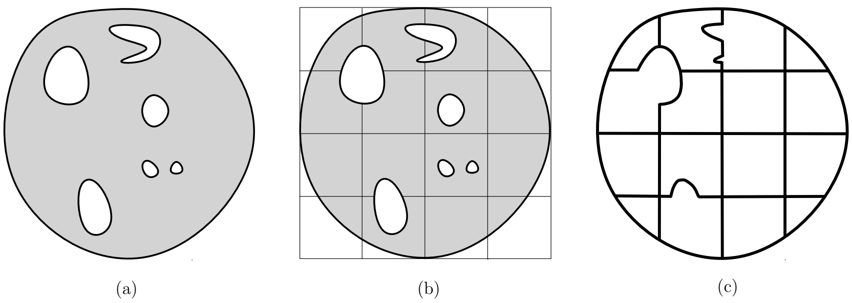


Figure 2 (a) An 2-dimensional illustration of a 3-complex $\mathcal{K}$ with several holes inside; (b) A nice bounding box of $\mathcal{K}$ with $\lfloor n^{1/3} r^{-1/3} \rfloor$ evenly-spaced 2-dimensional planes; (c) An r-hollowing $\mathcal{T}$ generated by Algorithm 2 consisting of simplexes that are "close" to the two-dimensional planes and parts of the boundaries of the intersecting holes inside with the planes.

Proof of Lemma 6.1. We can check that Algorithm 2 has linear runtime. In the rest of the proof, we will show $\mathcal{T}$ returned by Algorithm 2 is an r-hollowing of $\mathcal{K}$.

By Assumption 1, 2 and 3, the volume of the convex hull of $\mathcal{K}$, denoted by $\mathrm{CH}(\mathcal{K})$, is $\Theta(n)$; the maximum volume is attained when $\mathcal{K}$ has $\Theta(n/r)$ boundary components and each corresponds to a "hole" of volume $\Theta(r)$. By Lemma 6.3, we have $\mathrm{vol}(\mathcal{B}) = \Theta(\mathrm{vol}(\mathrm{CH}(\mathcal{K}))) = \Theta(n)$. In Algorithm 2, the 2-dimensional planes divide the box $\mathcal{B}$ into $O(n/r)$ smaller boxes

Algorithm 2 HOLLOWING($\mathcal{K}, r$).

Input: A pure 3-complex $\mathcal{K}$ embedded in $\mathbb{R}^3$ with n stable simplexes satisfying assumption 1-4 with a known parameter $r = o(n)$
Output: An r-hollowing $\mathcal{T}$

Find a nice bounding box $\mathcal{B}$ for $\mathcal{K}$ by Corollary 6.3.
For each pair of parallel faces of $\mathcal{B}$, find $\lfloor n^{1/3} r^{-1/3} \rfloor$ evenly-spaced 2-dimensional planes parallel to the face which divide $\mathcal{B}$ into equal-volume pieces. We can slightly perturb the planes so that no plane intersects with any vertex of $\mathcal{K}$ (see Figure 2(b)).

$\mathcal{T} \leftarrow$ all the tetrahedrons on the boundary of $\mathcal{K}$ that form a spherical shell.
for *each* 2*-dimensional plane* P **do**
    $\mathcal{Q} \leftarrow$ all the tetrahedrons of $\mathcal{K}$ that intersect P.
    if $\mathcal{Q}$ *is not connected (i.e.,* P *intersects some holes inside)* **then**
        $\mathcal{Q} \leftarrow \mathcal{Q} \cup \bigcup_{\mathcal{H} \in \text{intersected holes inside}}$ all the tetrahedrons on the boundary of $\mathcal{H}$, which are on one side of P and form half of a spherical shell (see Figure 2(c)).
    end
    $\mathcal{T} \leftarrow \mathcal{T} \cup \mathcal{Q}$.
end
Expand $\mathcal{T}$ such that its width reaches 5.
return $\mathcal{T}$

each of volume $O(r)$ and surface area $O(r^{2/3})$. By our construction of $\mathcal{T}$, each smaller box corresponds to a region; thus, there are $O(n/r)$ regions. By Assumption 3, each region of $\mathcal{T}$ has $O(r)$ simplexes and $O(r^{2/3})$ boundary simplexes. Moreover, the boundary of each region triangulates a spherical shell in $\mathbb{R}^3$ by construction. Additionally, the diameter of the underlying topological space of each region is upper bounded by the triangle diameter of the small box plus $\Theta(1)$ times the 1-skeleton diameter of boundary components. By Assumption 2, each region has diameter $O(r^{1/3})$.

To conclude, $\mathcal{T}$ satisfies all the conditions in Definition 2.9 and is an r-hollowing of $\mathcal{K}$. ◀

References

1. Owe Axelsson. A survey of preconditioned iterative methods for linear systems of algebraic equations. *BIT Numerical Mathematics*, 25(1):165–187, 1985.
2. Mitali Bafna and Nikhil Vyas. Optimal fine-grained hardness of approximation of linear equations. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *48th International Colloquium on Automata, Languages, and Programming (ICALP 2021)*, volume 198 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:19, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.
3. Sergio Barbarossa and Stefania Sardellitti. Topological signal processing over simplicial complexes. *IEEE Transactions on Signal Processing*, 68:2992–3007, 2020.
4. Gill Barequet and Sariel Har-Peled. Efficiently approximating the minimum-volume bounding box of a point set in three dimensions. *Journal of Algorithms*, 38(1):91–109, 2001.
5. Mitchell Black, William Maxwell, Amir Nayyeri, and Eli Winkelman. Computational topology in a collapsing universe: Laplacians, homology, cohomology. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 226–251. SIAM, 2022.
6. Mitchell Black and Amir Nayyeri. Hodge decomposition and general laplacian solvers for embedded simplicial complexes. In *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

7 Erik G Boman, Bruce Hendrickson, and Stephen Vavasis. Solving elliptic finite element systems in near-linear time with support preconditioners. *SIAM Journal on Numerical Analysis*, 46(6):3264–3284, 2008.

8 Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

9 Gunnar Carlsson. Topology and data. *Bulletin of the American Mathematical Society*, 46(2):255–308, 2009.

10 Li Chen, Rasmus Kyng, Yang P Liu, Richard Peng, Maximilian Probst Gutenberg, and Sushant Sachdeva. Maximum flow and minimum-cost flow in almost-linear time. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 612–623. IEEE, 2022.

11 David RJ Chillingworth. Collapsing three-dimensional convex polyhedra. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 63, pages 353–357. Cambridge University Press, 1967.

12 David RJ Chillingworth. Collapsing three-dimensional convex polyhedra: correction. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 88, pages 307–310. Cambridge University Press, 1980.

13 Michael B Cohen, Brittany Terese Fasy, Gary L Miller, Amir Nayyeri, Richard Peng, and Noel Walkington. Solving 1-laplacians in nearly linear time: Collapsing and expanding a topological ball. In *Proceedings of the 2014 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 204–216. SIAM, 2014.

14 Michael B Cohen, Jonathan Kelner, Rasmus Kyng, John Peebles, Richard Peng, Anup B Rao, and Aaron Sidford. Solving directed laplacian systems in nearly-linear time through sparse lu factorizations. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 898–909. IEEE, 2018.

15 Michael B Cohen, Jonathan Kelner, John Peebles, Richard Peng, Anup B Rao, Aaron Sidford, and Adrian Vladu. Almost-linear-time algorithms for markov chains and new spectral primitives for directed graphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 410–419, 2017.

16 Michael B Cohen, Rasmus Kyng, Gary L Miller, Jakub W Pachocki, Richard Peng, Anup B Rao, and Shen Chen Xu. Solving sdd linear systems in nearly $m \log^{1/2} n$ time. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, pages 343–352. ACM, 2014.

17 Samuel I Daitch and Daniel A Spielman. Support-graph preconditioners for 2-dimensional trusses. *arXiv preprint cs/0703119*, 2007.

18 Samuel I Daitch and Daniel A Spielman. Faster approximate lossy generalized flow via interior point algorithms. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 451–460, 2008.

19 Ming Ding, Rasmus Kyng, Maximilian Probst Gutenberg, and Peng Zhang. Hardness results for laplacians of simplicial complexes via sparse-linear equation complete gadgets. In Mikołaj Bojańczyk, Emanuela Merelli, and David P. Woodruff, editors, *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*, volume 229 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 53:1–53:19, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

20 Ming Ding and Peng Zhang. Weighted 1-laplacian solvers for well-shaped simplicial complexes. *arXiv preprint arXiv:2302.06499*, 2023.

21 Beno Eckmann. Harmonische Funktionen und Randwertaufgaben in einem Komplex. *Commentarii Mathematici Helvetici*, 17(1):240–255, December 1944. doi:10.1007/BF02566245.

22 Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. American Mathematical Soc., 2010.

23 Herbert Edelsbrunner and Salman Parsa. On the computational complexity of betti numbers: reductions from matrix rank. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on discrete algorithms*, pages 152–160. SIAM, 2014.

24 Joel Friedman. Computing betti numbers via combinatorial laplacians. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing*, pages 386–391, 1996.

25 Alan George. Nested dissection of a regular finite element mesh. *SIAM Journal on Numerical Analysis*, 10(2):345–363, 1973.

26 Robert Ghrist. Barcodes: the persistent topology of data. *Bulletin of the American Mathematical Society*, 45(1):61–75, 2008.

27 Arun Jambulapati and Aaron Sidford. Ultrasparse ultrasparsifiers and faster laplacian system solvers. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 540–559. SIAM, 2021.

28 Xiaoye Jiang, Lek-Heng Lim, Yuan Yao, and Yinyu Ye. Statistical ranking and combinatorial hodge theory. *Mathematical Programming*, 127(1):203–244, 2011.

29 Jonathan A Kelner, Lorenzo Orecchia, Aaron Sidford, and Zeyuan Allen Zhu. A simple, combinatorial algorithm for solving sdd systems in nearly-linear time. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 911–920, 2013.

30 Ioannis Koutis, Gary L. Miller, and Richard Peng. Approaching optimality for solving SDD linear systems. In *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, FOCS '10, pages 235–244, Washington, DC, USA, 2010. IEEE Computer Society. doi:10.1109/FOCS.2010.29.

31 Ioannis Koutis, Gary L. Miller, and Richard Peng. A nearly-$m \log n$ time solver for SDD linear systems. In *Proceedings of the 2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, FOCS '11, pages 590–598, Washington, DC, USA, 2011. IEEE Computer Society. doi:10.1109/FOCS.2011.85.

32 Rasmus Kyng, Yin Tat Lee, Richard Peng, Sushant Sachdeva, and Daniel A Spielman. Sparsified cholesky and multigrid solvers for connection laplacians. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 842–850, 2016.

33 Rasmus Kyng, Richard Peng, Robert Schwieterman, and Peng Zhang. Incomplete nested dissection. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 404–417, 2018.

34 Rasmus Kyng and Sushant Sachdeva. Approximate gaussian elimination for laplacians-fast, sparse, and simple. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 573–582. IEEE, 2016.

35 Hyekyoung Lee, Moo K Chung, Hongyoon Choi, Hyejin Kang, Seunggyun Ha, Yu Kyeong Kim, and Dong Soo Lee. Harmonic holes as the submodules of brain network and network dissimilarity. In *Computational Topology in Image Context: 7th International Workshop, CTIC 2019, Málaga, Spain, January 24-25, 2019, Proceedings 7*, pages 110–122. Springer, 2019.

36 Yin Tat Lee and Aaron Sidford. Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems. In *2013 ieee 54th annual symposium on foundations of computer science*, pages 147–156. IEEE, 2013.

37 Yin Tat Lee and Aaron Sidford. Path finding methods for linear programming: Solving linear programs in o (vrank) iterations and faster algorithms for maximum flow. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 424–433. IEEE, 2014.

38 Lek-Heng Lim. Hodge laplacians on graphs. *Siam Review*, 62(3):685–715, 2020.

39 Richard J Lipton, Donald J Rose, and Robert Endre Tarjan. Generalized nested dissection. *SIAM journal on numerical analysis*, 16(2):346–358, 1979.

40 Wenye Ma, Jean-Michel Morel, Stanley Osher, and Aichi Chien. An l 1-based variational model for retinex theory and its application to medical images. In *CVPR 2011*, pages 153–160. IEEE, 2011.

41 Aleksander Madry. Navigating central path with electrical flows: From flows to matchings, and back. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 253–262. IEEE, 2013.

42 Aleksander Madry. Computing maximum flow with augmenting electrical flows. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 593–602. IEEE, 2016.

43 Gary L Miller and William Thurston. Separators in two and three dimensions. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 300–309, 1990.

44 Zipei Nie. Matrix anti-concentration inequalities with applications. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 568–581, 2022.

45 Braxton Osting, Jérôme Darbon, and Stanley Osher. Statistical ranking using the l1-norm on graphs. *AIMS Journal on Inverse Problems and Imaging*, 7(3):907–926, 2013.

46 Richard Peng and Daniel A Spielman. An efficient parallel solver for sdd linear systems. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 333–342. ACM, 2014.

47 Richard Peng and Santosh Vempala. Solving sparse linear systems faster than matrix multiplication. In *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 504–521. SIAM, 2021.

48 Gil Shklarski and Sivan Toledo. Rigidity in finite-element matrices: Sufficient conditions for the rigidity of structures and substructures. *SIAM Journal on Matrix Analysis and Applications*, 30(1):7–40, 2008.

49 Daniel A Spielman and Nikhil Srivastava. Graph sparsification by effective resistances. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 563–568, 2008.

50 Daniel A Spielman and Shang-Hua Teng. Nearly linear time algorithms for preconditioning and solving symmetric, diagonally dominant linear systems. *SIAM Journal on Matrix Analysis and Applications*, 35(3):835–885, 2014.

51 Martin Tancer. Recognition of collapsible complexes is np-complete. *Discrete & Computational Geometry*, 55(1):21–38, 2016.

52 Shang-Hua Teng. The laplacian paradigm: Emerging algorithms for massive graphs. In *Theory and Applications of Models of Computation: 7th Annual Conference, TAMC 2010, Prague, Czech Republic, June 7-11, 2010. Proceedings 7*, pages 2–14. Springer, 2010.

53 Yiying Tong, Santiago Lombeyda, Anil N Hirani, and Mathieu Desbrun. Discrete multiscale vector field decomposition. *ACM transactions on graphics (TOG)*, 22(3):445–452, 2003.

54 Ke Ye and Lek-Heng Lim. Cohomology of cryo-electron microscopy. *SIAM Journal on Applied Algebra and Geometry*, 1(1):507–535, 2017.