Bounds, Boundaries, and Betti Numbers, Oh My! The Role of Hodge Theory in Laplacian Matrices and Computational Homology

Amy Hu, Guy Wilson

February 2017

1 Introduction

Computational homology has emerged as a heavily discussed area of applied mathematics within the past decades. The realization that traditionally pure fields such as algebraic topology have practical applications has led to an interest in calculating topological properties of data for use in clustering, statistical inference, and feature selection for biology, network analysis, machine learning, and other fields. As a result, much focus has been poured into methods of computing the homology of spaces algorithmically.

In this paper, we discuss some theoretical foundations drawn from differential geometry and topology culminating in a proof of the Hodge theorem. Using the lens of Hodge theory, we discuss the Laplacian operator and its discrete version, the Laplacian matrix. We then show the utility of this matrix in studying graph homology followed by a discussion on algorithmic complexity.

2 Notation

We begin with some notation prior to proceeding with preliminaries. Most notation will be blatantly copied over from Wikipedia, but with occasional modifications for ease of reading.

$C^k(\mathbb{R})$ k-times differentiable functions over the reals
C_k module of k-chains in a chain complex
$H_k(C)$ k^{th} homology group of a chain complex
$H^k(C)$ k^{th} cohomology group of a chain complex
$H^k_\Delta(M)$ kernel of Laplacian on k-forms over a smooth manifold M

In most of the cases below (especially those pertaining to differential geometry and topology), the notions explored can be extended to smooth manifolds but we will restrict ourselves to $\mathbb{R}^n$.

3 Differential Operators and Forms

We introduce an important tool used in differential geometry and more broadly analysis, the differential form. The first introduction into differential forms that most people remember is from calculus when taking the area under a curve over a certain interval, namely, integration. Examples of differential operators include the integral in one dimensional space and divergence in a vector field. These differential operators can be further defined in k dimensional space. We will be using differential operators to understand the De Rham complex.

We start by building some definitions that will help us better understand De Rham cohomology. The following section will largely consist of an attempt to introduce the language of linear algebra into calculus while simultaneously trying to adopt a coordinate-free approach to the notion of derivatives and differentials. Let's dive in.

Suppose we have some polynomial $p(x) \in \mathbb{P}_n$ (the space of polynomials of degree $\leq n$) with coefficients in $\mathbb{R}$ such that $p(x) = a_1 y + a_2 y^{'} + a_3 y^{''} + ... + a_n y^{n-1} + y^n$ where the y^i are linearly independent in $\mathbb{P}_n$. Note that, for all polynomials in this space, we can actually cast it in this form by taking $y = x^n$ and using the a_i to scale the coefficients appropriately. Hence the following construction is not an arbitrary notion for some small subset of polynomials but instead an applicable description for any polynomial.

Since differentiation is a linear map we can represent it via an operator $D : \mathbb{P}_n \to \mathbb{P}_{n-1}$. Therefore:

$$p(x) = (a_1 + a_2 D + a_3 D^2 + ... + a_n D^{n-1} + D^n)y$$
$$p(x) = [P(D)](y)$$

where $P(D) = a_1 + a_2 D + a_3 D^2 + ... + a_n D^{n-1} + D^n$ is our first example of a *differential operator*, a sort of abstraction of the notion of differentiation.

Definition. A *differential operator* is a function of the form:

$$f : C^k(\mathbb{R}) \to C^k(\mathbb{R})$$
$$f(D) = \sum a_i(x) D^i$$

where the $a_i(x)$ are generalized functions instead of constant coefficients now.

Since our building block D is linear, it should come as no surprise that differential operators exhibit linearity as well. In addition, the composition is defined as usual for functions assuming nitpicky conditions are met (e.g if coefficients are functions then they should be appropriately differentiable or smooth).

Example: Gradient

Suppose we have $\mathbb{F}^n$ ($\mathbb{F}$ a field with some metric) and assign a vector at each point via a function $f(\overrightarrow{x_1}, \overrightarrow{x_2}, ..., \overrightarrow{x_n})$ where the x_i can be thought of as unit vectors that act as a basis for our space.

Recall that the gradient of a vector field is a multidimensional analog to the notion of a derivative, measuring the partial derivative at a given point with respect to multiple basis vectors (the gradient of a single variable function is its derivative). In other words:

$$\nabla f = \frac{\partial f}{\partial \overrightarrow{x_1}} \cdot \overrightarrow{x_1} + \frac{\partial f}{\partial \overrightarrow{x_2}} \cdot \overrightarrow{x_2} + ... + \frac{\partial f}{\partial \overrightarrow{x_n}} \cdot \overrightarrow{x_n}$$

With differential operators now introduced, let's hastily handwave some intuition about *differential forms* and state a nice property for a specific operator acting on them (hint: starts with "exterior" and ends with "derivative").

A differential form can be thought of as infinitesimal, oriented patches of $\mathbb{R}^n$ that allow for a coordinate-free approach to calculus over nicely behaved manifolds [1]. For example, a differential 1-form looks like:

$$\omega = a_1 dx_1 + ... + a_n dx_n$$

Differentials of single variable functions are clearly a type of k-form but not all k-forms are simple differentials (most aren't). The advantage of using differential forms is that the abstraction from regular differentials allows for a range of simpler operations when, for instance, integrating over some region of a high-dimensional space or encapsulating Stoke's theorem in a very elegant manner (the intrepid reader is highly encouraged to check this out).

We can measure the instantaneous change of a differential form through differentiation. Defining this operator on k-forms gets a bit technical (actually, very much so) but we'll just note two properties of differentiation that will get us through the rest of this paper [1].

Claim 0.1.
Let ω be a k-form. Then $d(\omega)$ is a $(k+1)$-form.

In the above example, $d(\omega)$ is called the *exterior derivative* of ω. This "bumping up" in dimension by d will be important when we join together various vector spaces in a later section. Now let's throw in our last fact (we'll take this as a given as well) about differential forms before we move on.

Claim 0.2.
$d(d\omega) = 0$

4 The Laplacian Operator

Continuing from our previous example of the gradient, we'll now introduce the Laplacian operator (a differential operator!). The Laplacian operator is used in a variety of fields and settings such as the heat equation in physics or fluid dynamics in chemistry.

Definition. The *Laplacian operator* is defined as the sum of second partial derivatives where n is the number of variables in the twice-differentiable function. It can also be defined as the divergence of the gradient, otherwise known as the dot product of the gradient with itself.

$$\Delta : C^k(\mathbb{R}) \to C^{k-2}(\mathbb{R})$$
$$\Delta = \sum_{i=1}^{n} \frac{\partial^2 f}{\partial x_i^2} = \nabla \cdot \nabla f$$

Example: The Laplacian operator on a 2-Dimensional Space

The Laplacian on a two dimension space in Cartesian coordinates is given by:

$$\Delta = \sum_{i=1}^{2} \frac{\partial^2 f}{\partial x_i^2} = (\frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2})$$

or equivalently

$$\Delta = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \end{bmatrix} = (\frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2})$$

As with many other equations in mathematics, we can look at a specific instance of the Laplacian operator when $\Delta f = 0$. Solutions to this are called *harmonic functions* and lie in $\ker(\Delta)$.

5 De Rham Cohomology

We now introduce both our first chain complex and (simultaneously) instance of an algebraic structure equipped with a somewhat intimidating European name. In spite of its formal garb, the De Rham complex is constructed rather simply given the information in section 3.

An observant reader will notice that claim 0.2 is part of the definition of a chain complex. Using this fact, one might intuitively hope to string together a series of vector spaces with the differential operator in such a way as to yield information about the underlying topological space. This approach is precisely the one taken in the construction of the De Rham complex, a series of vector spaces composed of various k-forms that will give us some insight into the structure of our manifold.

Definition. The *De Rham complex* is a co-chain complex:

$$0 \to \Omega^0(M) \xrightarrow{\partial_0} \Omega^1(M) \xrightarrow{\partial_1} ... \xrightarrow{\partial_{n-1}} \Omega^n(M) \xrightarrow{\partial_n} 0$$

where $\Omega^k(M)$ is a vector space of k-forms on a closed smooth manifold (for our purposes, we will be working with k-forms over R^n equipped with the euclidean metric) and $\partial_k : \Omega^k(M) \to \Omega^{k+1}(M)$ the exterior derivative.

For those unfortunate enough to be acquainted with abstract algebra, vector spaces are modules defined over a field and linear maps between such spaces are in fact homomorphisms. Hence the De Rham complex satisfies the definition of a co-chain complex when paired with with the observation that $\partial_{i+1} \circ \partial_i = 0$ (a co-chain complex is a chain complex that "runs the other way" in some sense- the boundary operator just maps to *higher* dimensions).

For the De Rham complex, we can define a notion of *cohomology*, a topological invariant that is quite similar to homology in some ways (spoiler: the Laplacian will be deployed as a tool to calculate cohomology).

Definition. The i^{th} *cohomology group* of a co-chain complex $(C_\bullet, \partial_\bullet)$ is defined as the quotient space:

$$H^i \; = \ker \partial_i / \operatorname{im}(\partial_{i-1})$$

where ∂_i maps C_i to C_{i+1}.

By De Rham's theorem (which we omit for brevity), the cohomology of the manifold M underlying our vector spaces can be studied by examining the cohomology of the associated De Rham complex, providing a powerful link between differential and algebraic topology (sidenote: Stoke's theorem can be viewed as a statement on the relation between associated co-chain and chain complexes on a smooth manifold).

This link between analytic and algebraic notions becomes further apparent in Hodge theory, which we now have the tools to properly address.

6 The Hodge Theorem

Before we dive in, we'll define an abstracted form of the usual Laplacian that operates on a variety of co-chain complexes. This may seem a bit artificial at first but it will reduce to our more familiar friend in a later example. The abstraction will also lend itself to our desire to create a discrete Laplacian for graph analysis.

Definition. Let $(C_\bullet, d_\bullet)$ be a co-chain complex. We define the *Hodge Laplacian* to be the map:

$$d\delta + \delta d$$

where δ is the conjugate transpose of d i.e. $\delta = d^*$.

Theorem 6.1. (Hodge)
$\forall k$, there exists a canonical isomorphism $H^k_\Delta(M) \simeq H^k(M)$. Equivalently, each harmonic form in $H^k_\Delta(M)$ gives rise to an equivalence class in $H^k(M)$ and such equivalence classes are represented by only a single harmonic form.

Proof.

By De Rham's theorem, $H^k(M) \simeq H^k_{dR}(M) = \operatorname{im} d_i / ker(d_{i-1})$ hence it suffices to show isomorphism between the set of harmonic k-forms over M and the k^{th} De Rham cohomology group. Note that both the exterior derivative and the Laplacian are differential operators and therefore linear. This implies that $H^k_\Delta(M)$ and $H^k(M)$ are vector spaces since the kernel of a linear operator is a subspace of the domain and the quotient of vector spaces is a vector space (recall that the space of k-forms is a vector space).

Let's gather some intuition before we construct our isomorphism (partly so that it doesn't seem too artificial). Pick $v \in H^k_\Delta(M)$. Then:

$$\Delta(v) = 0$$
$$\Rightarrow (d\delta + \delta d)v = 0$$
$$\Rightarrow d(\delta(v)) + \delta(d(v)) = 0$$

We'll use a result from another paper to obtain a nice property about the elements in the kernel[2].

Claim. For a matrix with values in $\mathbb{R}$, $x \in \ker(A^*A + AA^*) \Leftrightarrow x \in \ker(A) \cap \ker(A^*)$

Proof. :

Note $\ker(A) \cap \ker(\mathcal{A}^*) \subseteq \ker(AA^* + A^*A)$. Now pick $v \in \ker(AA^* + A^*A)$. Since A is real-valued, then $A^* = A^T$ and hence AA^* and A^*A are positive semi-definite $\Rightarrow v \in \ker(AA^*) \cap ker(A^*A)$

Note $\ker(A) \in \ker(A^*A)$. Then $A^*Ax = 0 \Rightarrow x^*A^*Ax = 0 \Rightarrow ||Ax||^2 = 0 \Rightarrow x \in \ker A$. By similar logic, one finds that $x \in \ker(A^*)$.

Hence $v \in \ker(A) \cap \ker(A^*)$. $\square$

Then, by our claim, $v \in \ker(d) \cap \ker(\delta)$. Now recall:

$$d : \Omega^k(M) \to \Omega^{k+1}(M)$$
$$\delta : \Omega^k(M) \to \Omega^{k-1}(M)$$

Then $v \in \ker d_k$ such that $\delta(v) = 0 \in \Omega^{k-1}(M)$ or equivalently $d_{k-1}(\delta(v)) = 0 \in \Omega^k$. It seems like v must be all elements in the kernel that aren't really in the image (save for zero) so let's try relating the resultant quotient space to our set of harmonic forms.

Note that the homology groups are in this case quotients of vector spaces and hence behave nicely as vector spaces with the following operations for scalar multiplication and addition:

$$a(x + N) = (ax) + N$$
$$(x + N) + (y + N) = (x + y) + N$$

for scalars a and equivalence classes $x + N$ and $y + N$. Let us construct a map:

$$\varphi : H^k_\Delta(M) \to H^k_{dR}(M)$$
$$v \mapsto v + \operatorname{im} d_{k-1}$$

This is a well-defined map. Furthermore φ inherits quite a bit of structure from its codomain and so it also satisfies linearity as a property:

$$\varphi(av_1 + v_2) = (av_1 + v_2) + \operatorname{im} d_{k-1}$$
$$= a(v_1 + \operatorname{im} d_{k-1}) + (v_2 + \operatorname{im} d_{k-1}) = a\varphi(v_1) + \varphi(v_2)$$

We'll refer to the image of d_{k-1} as N to cut down on verbosity from now on. Let's see if this is injective. Pick $\varphi(v_1)$, $\varphi(v_2) \in H_{k,dR}(M)$ and suppose $\varphi(v_1) = \varphi(v_2)$ i.e. $v_1 + N = v_2 + N$.

Then, by definition of the left coset relation (and since φ is linear), we have:

$$(v_1 - v_2) = k \in N$$
$$\Rightarrow \varphi^{-1}((v_1 - v_2) + N) = (v_1 - v_2) \in H^k_\Delta(M)$$

But we know that $\forall v \in H^k_\Delta(M)$, $v \in \ker \delta$. Since δ maps between vector spaces with coefficients in $\mathbb{R}$, then $\delta = d^* = d^T$. Additionally, from some linear algebra one can obtain:

$$\text{im}(A)^\perp = \ker(A^T) \tag{1}$$
$$\Rightarrow < (v1 - v2), (v_1 - v_2) >= 0$$
$$\Rightarrow v_1 - v_2 = 0 \Leftrightarrow v_1 = v_2$$

Then φ is injective. Let's now show it's surjective as well and hence an isomorphism.

Choose an equivalence class $(v + N) \in H^k$. Select $v' \in (v + N)$ such that v is orthogonal to $\text{im}(d_{k-1})$ [2]. Then $< v, x >= 0 \ \forall x \in \text{im}(d_{k-1}) \Rightarrow x \in im(d_{k-1})^\perp = \ker(\delta_k)$ by (1). Then $v' = (v + n) \in \ker(d_k)$ since $v \in \ker(d_k)$ by definition and $n \in \text{im}(d_{k-1}) \Rightarrow n \in \ker(d_k)$ since $d_k d_{k-1} = 0$ by definition of a co-chain complex.

Then, for such v', we have that $v' \in \ker(d_k) \cap \ker(\delta_k) = \ker(dd^* + d^*d)$. Then $v' \in H^k_\Delta(M)$. Hence φ is surjective and thus an isomorphism.

□

This is a powerful statement that ties together the cohomology classes of our manifold with the kernel of the Laplacian operating on subspaces of it. By studying the behavior of the Laplacian, we can extract topological information about our space without resorting to algebraic methods. Let's try out an example in something more familiar, $\mathbb{R}^3$.

6.1 Gradient, Curl, Divergence

Consider the following structure:

$$C^k(\mathbb{R}^3) \xrightarrow{\nabla(f)} \mathbb{X}(\mathbb{R}^3) \xrightarrow{\nabla\times} \mathbb{X}(\mathbb{R}^3) \xrightarrow{\nabla\cdot} C^k(\mathbb{R}^3)$$

where C^k is our space of differentiable functions and $\mathbb{X}(\mathbb{R}^3)$ denotes the space of vector fields over $\mathbb{R}^3$ (which forms a vector space [3]). The operators are, from left to right, the gradient, curl, and divergence. Those with a background in multivariable calculus might recall that:

$$\nabla \times \nabla(f) = 0$$
$$\nabla \cdot (\nabla \times V) = 0$$

Hence this structure is a co-chain complex. Furthermore, studying the kernel of our Hodge Laplacian on $C^k(\mathbb{R}^3)$ we return to an old friend:

$$dd^* + d^*d = \nabla^*\nabla f \quad \textit{the other direction is a map into 0}$$
$$= (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}) \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial z} \end{bmatrix} = \sum_{i=1}^{3} \frac{\partial f}{\partial x_i} = \Delta$$

The Hodge Theorem proof also didn't use very much information about our k-forms at all. So, assuming the co-chain complex structure is preserved in a more exotic context, we can still derive a lot of value from the method assuming there's an equivalent "Laplacian" in the new setting. Let's consider graphs.

7 Discrete Laplacian

The discrete Laplacian is a specific case of the negative Laplacian operator as defined on a discrete grid. We use the discrete Laplacian when there is limited data. Let the function, f, be a smooth, continuous function. The Laplacian is defined as above. Using the limit definition, we can redefine the Laplacian as follows:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

$$\Delta = -\sum_{i=1}^{2} \frac{\partial^2 f}{\partial x_i^2} = -\lim_{h \to 0} \sum_{i=1}^{2} \frac{f'(x_i + h) - f'(x_i)}{h} = -\lim_{h \to 0} \sum_{i=1}^{2} \frac{f(x_i + 2h) - 2f(x_i + h) + f(x_i)}{h^2}$$

Let's look at a brief example with (x_1, x_2). Then we know its Laplacian:

$$\Delta = -(\frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2}) = -(\lim_{h \to 0} \frac{f(x_1 + 2h) - 2f(x_1 + h) + f(x_1)}{h^2} + \frac{f(x_2 + 2h) - 2f(x_2 + h) + f(x_2)}{h^2})$$

We can visualize this as a discrete grid with all equal distance, h, points from (x_1, x_2). Hence, the discrete Laplacian.

Now given the discrete graph, we can generalize this for all (undirected) graphs. Let's continue with our previous example. Using the limit definition of the Laplacian, we take the function value at each of the points to find the Laplacian:

$$\Delta = -(f(x_1-h, x_2) - f(x_1, x_2) + f(x_1+h, x_2) - f(x_1, x_2)) - (f(x_1, x_2-h) - f(x_1, x_2) + f(x_1, x_2+h) - f(x_1, x_2))$$

Let's simplify each point to a vertex like so,

$$v_1 = (x_1, x_2)$$

$$v_2 = (x_1 + h, x_2)$$
$$v_3 = (x_1 - h, x_2)$$
$$v_4 = (x_1, x_2 + h)$$
$$v_5 = (x_1, x_2 - h)$$

Then the Laplacian simplifies to be:

$$\begin{aligned}\Delta &= -(f(v_3) - f(v_1) + f(v_2) - f(v_1)) - (f(v_5) - f(v_1) + f(v_4) - f(v_1)) \\ &= 4f(v_1) - f(v_2) - f(v_3) - f(v_4) - f(v_5) \\ &= \begin{bmatrix} 4 & -1 & -1 & -1 & -1 \end{bmatrix} \begin{bmatrix} f(v_1) \\ f(v_2) \\ f(v_3) \\ f(v_4) \\ f(v_5) \end{bmatrix}\end{aligned}$$

Given the above example, we see that the functions can be written as a linear combination. As any fun math student knows, linear combinations can be written as matrices. This gives way to what we call the Laplacian matrix.

Definition. The Laplacian matrix as defined for a simple graph, G with n vertices:

$$L = D \text{ - } A$$

where D is the degree matrix and the A is the adjacency matrix. As a refresher from linear algebra, D is the diagonal matrix with the degree (number of incident edges)of each vertex in the graph. The adjacency matrix, A, is the (0, 1)-matrix that describes the relationship between all the vertices (i.e. whether or not they are connected via an edge). With this in mind, we further generalize this graph Laplacian for simplicial complexes as used previously in the proof of Hodge theorem.

8 Combinatorial Laplacian Method

We continue to use the Laplacian as a tool to understand cohomology groups and cochain complexes. A keen reader might notice that previously we had investigated $\ker(dd^* + d^*d)$ when proving the Hodge theorem. We shall give this operator (in the context of simplicial complexes) a name.

Definition. The higher order combinatorial Laplacian is an extension of the graph Laplacian.

$$L_i = d_{i+1}\delta^{i+1} + \delta^i d_i$$

Let's look at another example of how the combinatorial Laplacian for simplicial complexes gives the same result as the graph Laplacian of simplicial complexes. Consider the following graph:

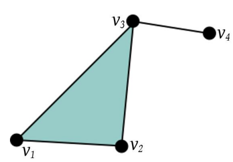

Figure 1: An undirected graph with one simplicial complex.

Let's find a graph Laplacian of this graph.

$$L = D - A$$

$$L = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

Let's write out the cohomology spaces and what they represent. The first cohomology space, C_0, gives us the vertices of the graph. The second cohomology space, C_1, gives us the edges connecting each of the vertices in the graph. We'll stop at the third cohomology space, C_2, which gives us the simplicial complex in the graph. Visualizing these relationships:

$$C_2 \longrightarrow C_1 \longrightarrow C_0$$

$$[v_1, v_2, v_3] \longrightarrow \begin{bmatrix} v_1, v_2 \\ v_1, v_3 \\ v_2, v_3 \\ v_3, v_4 \end{bmatrix} \longrightarrow \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$$

Let's look specifically at $d_1 : C_1 \to C_0$, from the edge space to the vertex space. Based on the picture above, the matrix for d_1 looks like:

$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Then we know that $\delta_1 : C_0 \to C_1$ will look like the transpose of d_1.

$$\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

Plugging these two matrices into the higher order combinatorial Laplacian,

$$L_1 = d_1\delta_1 + 0 \tag{2}$$

$$= \begin{bmatrix} 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & -1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} + 0 = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix} \tag{3}$$

Hooray! This result from the combinatorial Laplacian matches our previous answer from the graph Laplacian! Now what can we do with this tool?

As with any matrix, we can find the eigenvalues and eigenvectors of each of these Laplacians [4]. The eigenvalues, λ, give interesting information about the connectivity of the graph, particularly the multiplicity of the 0 eigenvalue. The multiplicity of $\lambda = 0$ gives the dimension of the null space or the kernel. Now reviewing the Hodge theorem, recall there is an isomorphism from $\ker L_i$ to the cohomology group, $H^i(X)$. We can see that the multiplicity of $\lambda = 0$ would correspond to the dimension of $H^i(X)$. Then we can use the Laplacian to compute homology!

Definition. The ith *Betti number* of a topological space, X, written as $b_i(X)$, is the rank of the homology group H^i. We can also interpret this as $b_i(x) = \dim(H^i) = \dim(\ker d_i)$ - $\dim(\operatorname{im} d_{i+1})$. (As a refresher, $H^i = \ker \partial_i / \operatorname{im}(\partial_{i-1})$) where ∂_i maps C_i to C_{i+1}.

Betti numbers are interesting because they are topological invariants. For example, the zeroth Betti number corresponds to the number of connected components in a graph. Poincare also proved Betti numbers' utility in developing the Poincare polynomial which is the generating function of the Betti numbers of a space.

9 Complexity Analysis

Let's consider the complexity of extracting the zeroth Betti number of a graph. As we've noted, the kernel of the graph Laplacian yields our first cohomology group. So in order to obtain our zeroth Betti number we'll have to:

1. Create our Laplacian matrix.

2. Perform gaussian elimination or compute eigenvectors.
3. Find the dimension of the kernel (or 0-eigenspace).

Let's run through an example of calculating b_0 of an undirected graph.

9.1 Matrix Generation

As we know from above, the Laplacian is calculated from the chain complexes from one cohomology space to another. Specifically when calculating b_0, we use the chain complex from $d_1 : C_1 \longrightarrow C_0$ and the complex $\delta_1 : C_0 \longrightarrow C_1$. As a review, we found the matrices that corresponded to these maps, d_1 went from the edge space to the vertex space and δ_1 went the opposite way. Therefore, to find these matrices we will traverse the given graph, G.

We will use the Depth First Search (DFS) algorithm for traversing the graph; that is, finding all the vertices in the graph. Intuitively, DFS works by starting at a vertex and following a path from vertex to vertex until it reaches a dead-end. Then it will backtrack and until it finds an edge that leads to uncharted territory. The algorithm follows the iterative procedure:

```
choose starting vertex, v, mark as visited
initialize Stack
push all vertices connected to v into Stack

while Stack is not empty do:
   pop vertex, w
   for vertices, u, adjacent to w do:
      if u is not visited then:
         u = visited
         push u to Stack
```

The runtime of DFS is $O(|V| + |E|)$ where $|V|$ is the number of vertices and $|E|$ is the number of edges in the graph. Now that we've traversed the graph and found all the edges and vertices, we can create the degree and adjacency matrices, D and A. In order to find the paths of the graph, we will choose a vertex and run DFS from that vertex until it reaches a dead-end. We will repeat this process until all the vertices and edges have been accounted for. This process will find all the vertices that are connected and not connected in the graph. We store this data in the two matrices that will result in the Laplacian matrix. Therefore, we will find all the connected components in the graph, G, and the zeroth Betti number.

9.2 Subspace Extraction

Let's consider the multiplicative complexity of gaussian elimination first, meaning that we'll keep track of multiplication/division. We'll generate an upper bound for our algorithm by supposing no prior information about the matrix except that it's $n \times n$ i.e. representative of a graph with n vertices.

Let $L^{(k)} \in M_{n\times n}(\mathbb{R})$ be the k^{th} iteration of our matrix L in the row reduction algorithm where $M_{n\times n}(\mathbb{R})$ is the space of $n \times n$ matrices with real coefficients.

The algorithm we introduce [5] relies on a series of nested subproblems (our L^k). Beginning with $k = 1$, we compute ratios between the diagonal entry and values below to generate coefficients for eliminating entries. After operating on a $k \times k$ submatrix, we then expand to $L^{(k+1)}$ and perform the same procedure.

Let's get a little more formal. Given L, perform the following procedure:

```
for k = 1,.., n-1 do:
   if i ≤ k then:
      if i ≤ k-1 and j ≤ k-1 then:
         a_ij^(k) = l_ij^(k-1)
      else:
         0

   if i ≥ k and j ≥ k then:
      a_ij^(k) = l_ij^(k-1) - (l_{i,k-1}^(k-1) / l_{k-1,k-1}^(k-1)) l_{k-1,j}^(k-1)
```

What an absolute mess. Let's try to figure what exactly is going on. The recursive definition for entry i, j in our updated matrix $L^{(k)}$ is as follows:

$$a_{ij}^{(k)} = \begin{cases} l_{ij}^{(k-1)} & i \leq k-1 \\ 0 & i \leq k,\ j \leq k-1 \\ l_{ij}^{(k-1)} - \frac{l_{i,k-1}^{(k-1)}}{l_{k-1,k-1}^{(k-1)}} l_{k-1,j}^{(k-1)} & i \geq k,\ j \geq k \end{cases}$$

When $i \leq k-1$, then l_{ij} is above our submatrix of interest. Hence we simply copy over the value to our new matrix $L^{(k)}$. If $i \leq k$ and $j \leq k-1$ then we are in the lower triangular zero matrix (where the entries are zeros below the diagonal due to previous row reduction). These values have been knocked out already so we simply assign them to be zero. The interesting case is when we are dealing with entries that haven't been converted yet because they're in the lower right $k \times k$ submatrix that is currently being processed. When $i, j \geq k$ we calculate ratios between our pivot (in this case the entry on the diagonal) and the entries below it, then multiplying by the entry value and subtract the result from the a_{ij}. This may be a bit irritating to follow but it's simply the algorithm most people intuitively implement when asked to row reduce a matrix.

As k increases, our current submatrix decreases in size. Specifically, as we transition from $L^{(k-1)}$ to $L^{(k)}$ we must calculate $(n-k+1)$ ratios (for the $k-1$ entrries below our pivot). We then perform a multiplication for each entry in our $(n-k+1) \times (n-k+1)$ submatrix, yielding $(n-k+1)^2$ operations when updating. Reindexing and using a couple summation identities, we have:

$$\sum_{k=0}^{n} (n-k+1) + (n-k+1)^2$$

$$= \sum_{k=1}^{n-1} (n-k) + (n-k)^2 = \sum_{k=1}^{n-1} k + k^2$$

$$= \frac{n(n-1)}{2} + \frac{(n)(n-1)(2n-1)}{6} \approx n^3/3$$

Two caveats should be considered for this procedure. The first is that one should perform pivoting to ensure the division operation is stable and defined. The second is that, assuming the matrix has a nontrivial nullspace, then the algorithm should terminate when we can no longer perform the algorithm due division by zero. In practice, this means that we have obtained an upper bound for the multiplicative complexity of gaussian elimination.

For backsubstitution, we obtain roughly $\frac{n^3}{3}$ operations as well [5], yielding a rough multiplicative complexity count of $\frac{2n^3}{3}$. With a reduced echelon form available, finding the zeroth Betti number amounts to calculating $n-$ #pivots since $\dim(\ker \Delta) = b_0$.

By contrast, eigenvalue algorithms ordinarily rely on convergent approximations due to restrictions on general solutions to high degree polynomials (solving for eigenvalues of a matrix is equivalent to finding the roots of its characteristic polynomial) [6]. As a result, one can trade precision for lower runtimes.

10 Conclusion

Topological spaces can be hard to wrap one's head around (we refrain from inserting any homeomorphism jokes), which is why algebraic topology is so great: we can use abstract algebra to encapsulate and dissect these spaces without having to do the dirty work of interfacing with them directly. But at the same time, algebraic objects are annoying to work with computationally [2].

The power of Hodge theory derives from its ability to have the best of both worlds. Hodge Laplacians allow us to capture valuable data about a topological space, encoded in algebraic structures which we understand quite well. At the same time, we can retain the use of analysis and linear algebra to actually compute these features, which is easier for implementation purposes *and* thinking about the problem (at least for some people).

References

[1] Terrence Tao, *Differential Forms and Integration.*
`http://www.math.ucla.edu/ tao/preprints/forms.pdf`

[2] Lek-Heng Lim, *Hodge Laplacians on Graphs.*
`https://www.stat.uchicago.edu/~lekheng/work/psapm.pdf`.

[3] Jerrold Marsden, *Manifolds, Mappings, Vector Fields.*
`http://www.cds.caltech.edu/~marsden/wiki/uploads/cds202-08/home/Manifolds.pdf`

[4] Fan Chung, *Spectral Graph Theory: Eigenvalues and the Laplacian of a graph*, 1997.
`http://www.math.ucsd.edu/~fan/research/cb/ch1.pdf`

[5] Eugene Isaacson, Herbert Keller, *Analysis of Numerical Methods*, 1966.

[6] Steffen Borm, Christian Mehl, *Numerical Methods for Eigenvalue Problems*, 2012.