The Graph Laplacian and Determining the Connectivity of Meshes

Eli Griffiths

November 2024

Abstract

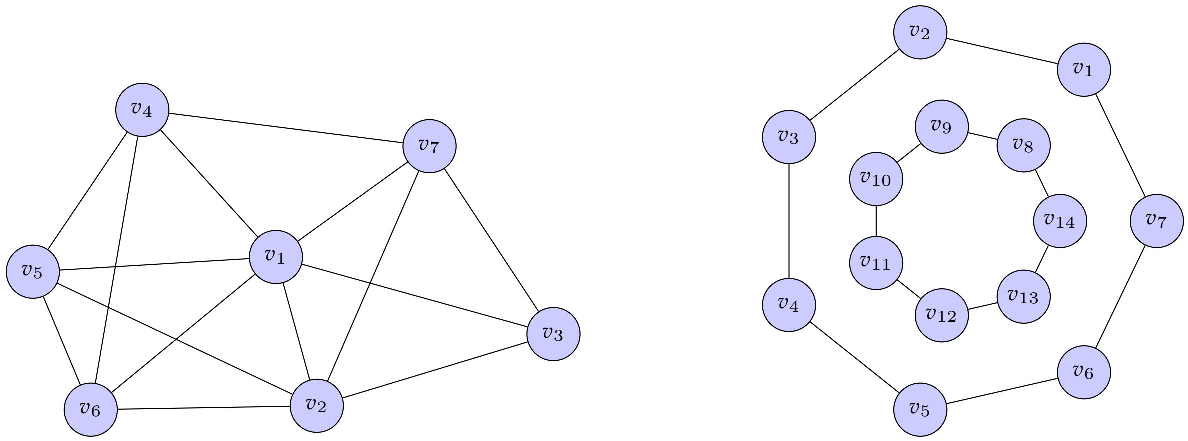
Graphs lend themselves naturally to matrices that encode properties such as which vertices are connected to others and how many vertices are connected to a given vertex. These matrix representations allow us to analyze graphs outside of traditional combinatoric approaches by considering their eigenvalues and eigenvectors. Key to our study is the Laplacian matrix representation of a graph. We first show that the eigenvalues of the Laplacian matrix give insight into if every vertex of a graph is connected to each other via some path through edges, and if not how many separate components there are in which this is the case. We then frame this result within the context of of computational geometry and discuss how more advanced problems in computational geometry can be approached with this spectral framework.

1 Introduction

Graphs at their core encode connectivity. A graph is simply vertices and edges where vertices are objects and edges are connections between these objects. While simple in concept, this means that anywhere there are pairwise relationships between objects, a graph structure can be assigned. We see graphs appear in situations such as modeling friendship networks on social media platform, determining flight schedules for optimal transport, modeling the structure of chemical compounds, and of importance to us computational geometry. Because of the fundamental nature of connectivity in many differing fields and applications, it is of great interest to develop theories and algorithms for graphs.

Graph theory as a field has always been deeply tied to combinatorics. One of the earliest problems in graph theory is Euler's famous Königsberg bridge problem. In the town of Königsberg, there were 7 bridges connecting 4 land masses. The question is if if there was a possible route one could walk that would cross each bridge exactly once. Euler reframed the problem by considering the bridges as edges and each land mass as vertices. By employing the fundamental concept of counting from combinatorics, Euler solved the problem in terms of the parity of the number of edges going into a vertex for each land mass [WW13].

In this paper, we will explore an alternative way of viewing and understanding graphs. We will seek to construct a bridge between the discrete graph structure and a more algebraic description. The question is what algebraic description or structure should we use? Note that if we limit the number of vertices of a graph to be finite, we can index its vertices as $v_1, \ldots, v_n$. We can then lay out in a 2D grid all possible pairs of vertices and mark each cell when the corresponding vertices are connected. Quite naturally, this same grid could be expressed as a matrix where each row and column represents some vertex and the entries are 1 if two vertices are connected and 0 otherwise. We will find that such possible matrix representations of graphs will have well behaved spectra



(a) A graph with 7 vertices and 15 edges

(b) A graph with 2 connected components

Figure 1: Two example graphs in the plane

(that is well behaved eigenvalues and eigenvectors). We will then show that the connectedness of a graph is encoded in the eigenvalues of a certain matrix representation of a graph, a result we can then reframe in computational geometry.

2 Background

We first outline the formal structure of a graph and related definitions that will be used throughout the paper, following the formalism and notation from [Die17]. We denote $[S]^n$ as the set of all n-element sized subsets of S.

Definition 2.1 (Graph Structure)**.** A *graph* is a pair $G = (V, E)$ of sets where $E \subseteq [V]^2$. The elements of V are *vertices* and the elements of E are *edges*. A vertex v is said to be *incident* to an edge e if $v \in e$. Two vertices v_1 and v_2 are *adjacent* or *neighbors* if $\{v_1, v_2\} \in E$. We denote $\{v_1, v_2\} \in E$ by $v_1 \sim v_2$. The *set of neighbors* of a vertex v is denoted by $N(v) := \{w \in V : v \sim w\}$. The *degree* of a vertex v is $\deg(v) := |N(v)|$. A *subgraph* H of G, denoted by $H \subseteq G$, is a graph whose vertex and edge sets are subsets of G's.

Remark 2.2. Edges importantly are defined here as two element sets and not as ordered pairs. This makes the graph *undirected*.

For the purposes of this paper, we will assume that every graph has finitely many vertices (and hence finitely many edges). Consider the illustrations of two graphs in Figure 1. Notice that they differ in terms of how connected their vertices are to one other. If each vertex was a city and each edge a road, a car driving on 1a could get to any city, but a car on the outer ring of 1b would be stuck on the outer ring. That is, there is some path that the car can take from any city to any other city in 1a but not in 1b. We formalize this notion of paths and connectedness as follows.

Definition 2.3 (Connectedness)**.** A *path* is a non-empty graph $P = (V, E)$ where

$$V = \{v_0, v_1, \ldots, v_k\} \qquad E = \{\{v_0, v_1\}, \ldots, \{v_{k-1}, v_k\}\}.$$

A graph G is then *connected* if between any two vertices v_0 and v_f there exists a path $P \subseteq G$ starting at v_0 and ending at v_f. A *connected component* of a graph is a subgraph $H \subseteq G$ that is connected and is not contained in any larger connected subgraph.

Example 2.4. Consider the graphs in Figure 1. A possible path in 1a is $\{v_5, v_2, v_7, v_3\}$ which means that v_5 and v_3 are connected. In 1b, the subgraphs $\{v_1, \ldots, v_7\}$ and $\{v_8, \ldots, v_{14}\}$ are both connected components.

As previously discussed we will attempt to gain insight about graphs, and specifically their connectivity, by representing them algebraically as matrices. Of importance to us are the adjacency matrix and degree matrix which both capture in different ways the connectivity between vertices.

Definition 2.5. The *adjacency matrix* A_G and *degree matrix* D_G of a graph G with n vertices are the $n \times n$ matrices such that

$$(A_G)_{ij} = \begin{cases} 1 & v_i \sim v_j \\ 0 & \text{otherwise} \end{cases} \qquad (D_G)_{ij} = \begin{cases} \deg(v_i) & i = j \\ 0 & i \neq j \end{cases}$$

If G is understood via context, we simply refer to them as A and D.

Example 2.6. The adjacency and degree matrices for the graph in Figure 1a are

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \qquad D = \begin{bmatrix} 6 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{bmatrix}$$

By observation, we can see that both the matrices in Example 2.6 are symmetric matrices. One may recall from linear algebra important results about symmetric matrices and their spectra. Symmetric matrices are guaranteed to have real eigenvalues, an orthonormal basis of eigenvectors that diagonalize them, and many other properties. Generally speaking then, the spectra of a symmetric matrix is well behaved and numerically friendly. This begs the question of (1) are the adjacency and degree matrices always symmetric for any graph, and (2) if they are symmetric what insight could we possibly ascertain from their spectra? It turns out the answer to the first question is yes (as the next theorem demonstrates) and we will attempt to answer the second question in the next two sections.

Theorem 2.7. *The adjacency and degree matrices A and D of a graph G are symmetric.*

Proof. By the definition of the degree matrix, the only non zero entries lie on the main diagonal of D. Therefore D is a diagonal matrix and hence symmetric. Consider two vertices v_i, v_j of G in which $v_i \sim v_j$. Since the edges of G are undirected we also have $v_j \sim v_i$. Hence $(A)_{ij} = (A)_{ji} = 1$. Likewise, if $v_i \nsim v_j$ then $v_j \nsim v_i$ meaning $(A)_{ij} = (A)_{ji} = 0$. Therefore A is symmetric. ∎

3 The Spectra of a Graph

Definition 3.1. The *Laplacian Matrix* of a graph G is $L_G := D_G - A_G$. If G is understood via context, we simply refer to it as L.

Notice that since L is the difference of two symmetric matrices it is also symmetric. Therefore we know that it has eigenvalues that are all real. Of importance to us is that amongst these eigenvalues, we are guaranteed to have zero as an eigenvalue for any Laplacian matrix.

Theorem 3.2. *The Laplacian matrix L of a graph G has zero as an eigenvalue.*

Proof. Let n denote the number of vertices in G and take $x \in \mathbb{R}^n$ to be the vector of all 1's. By the definition of the Laplacian matrix, we have $L = D - A$. Consider the product Dx. Since D is diagonal, the i^{th} component will be the i^{th} component of x times the i^{th} entry along the diagonal of D, which by the definition of D gives $(Dx)_i = \deg v_i$. Now consider the i^{th} component of the product Ax. We can express it as the sum

$$(Ax)_i = \sum_{j=1}^{n} (A)_{ij} x_j = \sum_{j=1}^{n} (A_{ij}).$$

This sum is simply adding up the entries of the i^{th} row of A. Since the entries are only 0 and 1, and are only 1 when v_i is adjacent to another vertex, the sum is $\deg v_i$. Hence $(Ax)_i = \deg v_i$. Thus $Ax = Dx$ from which we have $Lx = (D - A)x = Dx - Ax = 0$. Therefore zero is an eigenvalue of L. ∎

We will find in further analysis more utility in an alternative description of the Laplacian matrix. If we assign a real number to each vertex of a graph with n vertices, we can view this as a vector in $x \in \mathbb{R}^n$ where each component corresponds to a vertex. Therefore the Laplacian matrix can be used to in a sense operate on x like in Lx. This gives rise to the *quadratic form* of the Laplacian. We will not provide a derivation of it here, but one can be found in [Moh04].

Theorem 3.3. *Let L be the Laplacian matrix of a finite graph G with n vertices. Then for a given vector $x \in \mathbb{R}^n$,*

$$x^T L x = \sum_{v_i \sim v_j} (x_i - x_j)^2.$$

Remark 3.4. Sometimes the Laplacian matrix is defined as the unique symmetric matrix L with this quadratic form and the matrix definition is then a derived representation [Wat94]. The quadratic form also helps motivate the choice to name L the Laplacian matrix. In continuous settings, the Laplacian operator ∇ "smooths" functions and averages locally large differences. In the matrix case, if we viewed a vector $x \in \mathbb{R}^n$ as a list of temperatures at each vertex of a graph, minimizing the quadratic form of its Laplacian matrix would act to diffuse heat in a smooth manner that like wise would average locally large differences.

We are almost equipped with what we need to draw the connection between the Laplacian matrix's spectra and the connectivity of its corresponding graph. The following result about block diagonal matrices and their determinant will be needed, but we shall omit the proof of it here. A proof of this result can be found in [Sil00].

Lemma 3.5. *For a block diagonal matrix A with blocks A_i, $\det(A) = \det(A_1)\det(A_2)\cdots\det(A_n)$*

The following lemma gives a very strict characterization of zero as an eigenvalue for connected graphs. We will then generalize this characterization from a connected graph to any graph G by decomposing it into its connected components.

Lemma 3.6. *A graph G is connected if and only if the algebraic multiplicity of zero is 1 for L_G.*

Proof. We proceed with the contrapositive of the reverse direction. Suppose that G is not connected. If k is the number of connected components of G then $k > 1$. Denote each connected component as H_i. Note that each components vertex set must be pairwise disjoint with any other component then itself. Therefore we can group the vertices of G by which subgraph H_i they are in. If we then consider the adjacency matrix A of G with such an ordering, we will have a block diagonal matrix where each block is the adjacency matrix A_i corresponding to each H_i. Thus the Laplacian matrix for G is also block diagonal with each block being the Laplacian matrix L_i for each H_i. Consider the characteristic polynomial $\det(L - I\lambda)$ of L. Since L is block diagonal, by Lemma 3.5 we can expand this as $\det(L - I\lambda) = \det(L_1 - I\lambda) \cdots \det(L_k - I\lambda)$. For each L_i we know from Lemma 3.2 that each L_i has zero as an eigenvalue and hence has the linear factor $(\lambda - 0)$ in its characteristic polynomial. Therefore $\det(L - I\lambda)$ is the product of at least two terms that both have $(\lambda - 0)$ as a factor, hence the multiplicity of zero for L is greater than 1.

Suppose towards contradiction that G is connected but the algebraic multiplicity of zero is not equal to 1. Then the algebraic multiplicity of zero must be greater than 1. Therefore the associated eigenspace must have a dimension greater than 1, meaning we can pick two linearly independent eigenvectors u and w. Since both are eigenvectors with eigenvalue 0, we have

$$\sum_{v_i \sim v_j} (u_i - u_j)^2 = u^T L u = u^T \mathbf{0} = 0$$

which is true for w as well. Since the quadratic form is a sum of non-negative terms, the only way it is zero is if every term is itself 0. Therefore if $v_i \sim v_j$ we have $u_i = u_j$ and $w_i = w_j$. Consider two vertices v_m and v_n of G. Since G is connected, there must exist a path between them. Note then that $u_m = u_n$ and $w_m = w_n$ since the equality of each component holds across each edge in the path. But since we can choose any two arbitrary vertices v_m and v_n, every component of u and w are just constants. However, this means both u and w are scalar multiples of the vector of all 1's and hence are linearly dependent, a contradiction. ∎

Theorem 3.7. *A graph G has m connected components if and only if the algebraic multiplicity of zero is m for L_G.*

Proof. Suppose G has m connected components. Then the Laplacian matrix L can be expressed as a block diagonal matrix since both the adjacency and degree matrix can be expressed as block diagonal. Each block L_i represents the corresponding Laplacian matrix for the subgraph induced by each component. From Lemma 3.5, we then have $\det(L - I\lambda) = \det(L_1 - I\lambda) \cdots \det(L_m - I\lambda)$ meaning the algebraic multiplicity of zero for L is the sum of the algebraic multiplicities of zero for each L_i. Each L_i is connected so by Lemma 3.6 we know the algebraic multiplicity of zero for each L_i is one. Therefore the algebraic multiplicity of zero for L is one summed up m times which is just m. Now suppose the algebraic multiplicity of L is m. Since G is finite, we can let k denote the number of connected components of G. By the same logic as above, k must be the sum of the algebraic multiplicities of 0 for each component. But this sum is simply m meaning $k = m$. ∎

4 Application to Computational Geometry

In computer graphics, modeling, simulation, etc. we often want a representation of some real world geometry that we can perform computations on. A common way to of doing so is via a mesh. Consider the example mesh of a vase to the right. From some basic observations, the mesh appears to be comprised of points connected by segments/edges which outline faces. We formalize these observations in Definition 4.1.

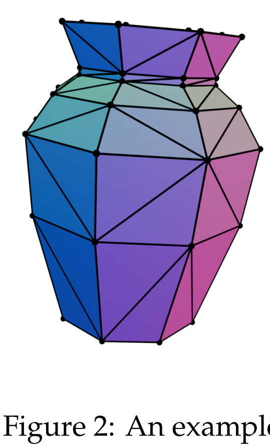


Figure 2: An example mesh

Definition 4.1. A *triangular mesh* is a triple $K = (V, E, F)$ such that

- $V \subseteq \mathbb{R}^3$ is a finite set representing the vertices
- $E \subseteq [V]^2$ is a set representing edges
- $F \subseteq [E]^3$ is a set representing faces such that for any $f = \{e_1, e_2, e_3\} \in F$,

$$e_1 \cap e_2 = \{v_1\} \qquad e_2 \cap e_3 = \{v_2\} \qquad e_3 \cap e_1 = \{v_3\}.$$

  for $v_1 \neq v_2 \neq v_3$ and there are no three faces $f_1, f_2, f_3 \in F$ such that $f_1 \cap f_2 \cap f_3 = \{e\}$.

Notice that a triangular mesh lends itself to some very natural graph structures. One is simply using the vertices as the graph vertices and the edges as graph edges. The other one which is of use to us is using the faces as graph vertices and faces sharing a common edge as the edge set. This is often referred to as the dual graph of a mesh and appears in computational geometry problems such as mesh based signal processing [Tau02] and refinement of meshed implicit surfaces [OB02].

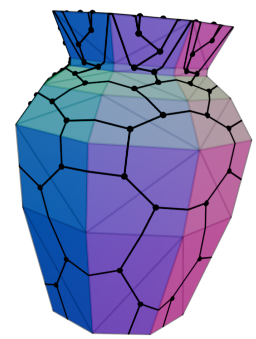


Figure 3:
Dual mesh of Fig. 2

Definition 4.2. The *dual mesh* of a triangular mesh $K = (V_K, E_K, F_K)$ is the graph $G = (V, E)$ such that $V = F_K$ and $f_1 \sim f_2$ if $f_1 \cap f_2 = \{e\}$ for some in $e \in E_K$.

Example 4.3. We can superimpose the dual mesh visually on the vase mesh in Figure 2 by placing vertices on each face and connecting them on the surface of the vase itself, which can be seen in Figure 3.

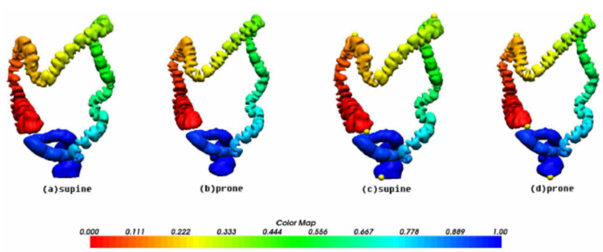
When dealing with meshes, we are usually interested in capturing the surface geometry of some object. It is very natural to ask questions about traversing or simulating heat dispersion on the surface of a mesh. Both of these tasks necessitate some concept of when parts of the mesh are connected to ensure paths don't jump between different unconnected parts or heat spreads through empty space.

Imagine living on the surface of some face of a mesh. Then one can freely walk around the surface of that face. Furthermore, one can walk across edges to adjacent faces in well defined manner as there can only be at most 1 possible choice due to the last restriction on faces in Definition 4.1. Passing through a vertex is less well defined because there can be an arbitrary number of possible faces to go to. We will therefore define (quite informally) that a mesh is *connected* if for any two points (starting on faces) can be connected via a path on the mesh surface that crosses between faces via edges only. In the same vain as a graph, a *connected component* of a mesh is a maximal submesh that is connected. This quickly lends itself to a reframing of Theorem 3.7.

Theorem 4.4. *A triangular mesh has m connected components if and only if the algebraic multiplicity of zero is m for the dual mesh's Laplacian matrix.*

Proof. The notion of connectedness for meshes we constructed is captured by the connectivity of the dual mesh. Our requirement that mesh connectivity happen across edges only is identical to connecting faces with a common edge as in the definition of the dual mesh. Therefore the question of how many connected components a mesh has is the same as how many connected components does its dual mesh have. Its dual mesh will have a corresponding Laplacian matrix which we know from Theorem 3.7 that the algebraic multiplicity of zero equals the number of connected components. Therefore since the number of connected components in the mesh is identical to the number of connected components of its dual mesh, we have the desired if and only if. ∎

(a) Mesh partitioning from [BTC23]



(b) Mesh partitioning from [Lai+10]

Figure 4: Examples of spectral partitioning of meshes

While this result may seem obvious having seen it established in a graph theory context, it serves to illustrate how simple it can be to translate results from graph theory to computational geometry. If one wanted to, they could use the previous result and find the multiplicity of a meshes zero eigenvalue numerically to determine how many components it has. This general approach we took of taking the dual, creating the Laplacian matrix of the dual, then applying some result about its spectra lends itself to some very powerful algorithms. While we were interested in a strict sense of connectivity, one may wonder if the connectivity of a graph could be represented as a more continuous quantity. This question lends itself to the realm of *spectral partitioning* in which a (possibly already connected mesh) is separated into submeshes that maximize some more continuous sense of connectedness. This separation is typically dictated by examining the unit norm eigenvector associated with the next largest eigenvalue than zero of the Laplacian matrix (called the "Fiedler Vector"). Spectral partitioning gives very natural groupings of a mesh making it useful in medical applications [Lai+10] and mesh processing [BTC23] (see Figure 4).

References

[BTC23] Xiaohan Bao, Weihua Tong, and Falai Chen. "A Spectral Segmentation Method for Large Meshes". In: *Communications in Mathematics and Statistics* 11.3 (2023), pp. 583–607.

[Die17] Reinhard Diestel. *Graph Theory*. Vol. 173. Graduate Texts in Mathematics. Berlin, Heidelberg: Springer, 2017. ISBN: 978-3-662-53621-6 978-3-662-53622-3. DOI: `10.1007/978-3-662-53622-3`. (Visited on 10/23/2024).

[Lai+10] Zhaoqiang Lai et al. "Intra-Patient Supine-Prone Colon Registration in CT Colonography Using Shape Spectrum". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010*. Ed. by Tianzi Jiang et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 332–339. ISBN: 978-3-642-15705-9.

[Moh04] Bojan Mohar. "Graph laplacians". In: *Topics in algebraic graph theory* 102 (2004), pp. 113–136.

[OB02] Yutaka Ohtake and Alexander G. Belyaev. "Dual/Primal mesh optimization for polygonized implicit surfaces". In: *Proceedings of the Seventh ACM Symposium on Solid Modeling and Applications*. SMA '02. Saarbrücken, Germany: Association for Computing Machinery, 2002, pp. 171–178. ISBN: 1581135068. DOI: `10.1145/566282.566308`. URL: `https://doi.org/10.1145/566282.566308`.

[Sil00] John R Silvester. "Determinants of block matrices". In: *The Mathematical Gazette* 84.501 (2000), pp. 460–467.

[Tau02] Gabriel Taubin. "Dual Mesh Resampling". In: *Graphical Models* 64.2 (Mar. 2002), pp. 94–113. ISSN: 1524-0703. DOI: `10.1006/gmod.2002.0571`. (Visited on 10/23/2024).

[Wat94] William Watkins. "Unimodular congruence of the Laplacian matrix of a graph". In: *Linear Algebra and its Applications* 201 (1994), pp. 43–49. ISSN: 0024-3795. DOI: `https://doi.org/10.1016/0024-3795(94)90102-3`. URL: `https://www.sciencedirect.com/science/article/pii/0024379594901023`.

[WW13] R. Wilson and J.J. Watkins. *Combinatorics: Ancient & Modern*. OUP Oxford, 2013. ISBN: 9780191630620. URL: `https://books.google.com/books?id=vj1oAgAAQBAJ`.