Spectral Graph Theory and the Cheeger Inequality

0. Notation

Symbol	Meaning
$G = (V, E)$	Graph; vertex set V, edge set E, $n = \|V\|$
$W \in \mathbb{R}^{n\times n}$	Adjacency matrix, $W_{ij} = 1$ if $(i, j) \in E$, else 0
$D = \mathrm{diag}(d_1, \dots, d_n)$	Degree matrix, $d_i = \sum_j W_{ij}$
$L = D - W$	Graph Laplacian
$L_{sys} = I - D^{-1/2}WD^{-1/2}$	Normalized Laplacian
$\mu_1 \geq \mu_2 \geq \cdots \geq \mu_n$	Eigenvalues of W (descending)
$\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$	Eigenvalues of L (ascending)
$u_1, u_2, \dots, u_n$	Corresponding eigenvectors (of W or L depending on context)
$v \in \mathbb{R}^n$	Arbitrary vector on the graph (one scalar per node)
$S \subseteq V$	Vertex subset, $\bar{S} = V \setminus S$
∂S	Cut: the set of edges between S and $\bar{S}$
$\mathrm{vol}(S) = \sum_{i\in S} d_i$	Volume of S

1. Eigenvectors Describe Directions of Motion

1.1 Direct Observation of Wv

$$Wv = \left(\sum_{j=1}^{n} W_{1j} v_j,\ \sum_{j=1}^{n} W_{2j} v_j,\ \ldots,\ \sum_{j=1}^{n} W_{nj} v_j \right)^T$$

The i-th component $(Wv)_i$ is the sum of all messages node i receives from its neighbors.

Continuing the iteration:

$$W^2 v = \left(\sum_{j,j'} W_{1j} W_{jj'} v_{j'},\ \ldots,\ \sum_{j,j'} W_{nj} W_{jj'} v_{j'} \right)^T$$

Expanding $W^k v$ directly is **useless**: the indices grow increasingly complex, and no structure emerges.

1.2 Changing the Representation of v: Expansion in the Eigenvector Basis

Let $\{u_i\}_{i=1}^{n}$ be the eigenvector basis of W, with $Wu_i = \mu_i u_i$. Expand v in this basis:

$$v = \sum_{i=1}^{n} \alpha_i u_i$$

By orthonormality $u_i^T u_j = \delta_{ij}$, the coefficients are read off directly:

$$\alpha_i = u_i^T v$$

Then:

$$Wv = W \sum_i \alpha_i u_i = \sum_i \alpha_i W u_i = \sum_i \mu_i \alpha_i u_i$$

Iterating:

$$W^k v = \sum_{i=1}^{n} \mu_i^k \alpha_i u_i$$

Key point: In the eigenbasis, W simply multiplies each component by its eigenvalue μ_i. The dynamics become completely transparent.

1.3 Physical Interpretation

Each eigenvector u_i is a **direction of pure motion**: a perturbation along u_i remains along u_i after applying W, scaled by μ_i. All complex dynamics decompose into motion along these independent directions.

2. Simplified Temperature Problem (Motivating Discussion)

The simplest setting: an unweighted graph with no self-loops,

$$W_{ii} = 0, \qquad W_{ij} = \begin{cases} 1 & (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

Interpret $v = [v_1, v_2, \ldots, v_n]^T$ as a temperature distribution on the nodes.

2.1 One Step

The new temperature of node i is the sum of temperatures of its neighbors:

$$(Wv)_i = \sum_{j:(i,j)\in E} v_j$$

2.2 Multiple Steps

Iteration rule $v^{(k+1)} = Wv^{(k)}$; by Section 1.2:

$$v^{(k)} = W^k v^{(0)} = \sum_i \mu_i^k \alpha_i u_i$$

2.3 Long-time Behavior

For large k, the term with the largest $|\mu_i|$ dominates:

$$v^{(k)} \ \propto \ u_{i^*}, \qquad i^* = \arg\max_i |\mu_i|$$

Conclusion: after a long time, the temperature distribution is proportional to the eigenvector with the largest eigenvalue.

> **Remark**: This is the core idea of power iteration, and also underlies PageRank and Markov chain stationary distributions.
>
> This simplified system is not true physical heat diffusion (genuine heat diffusion obeys $dv/dt = -Lv$ and decays to the zero mode), but it builds the intuition that the largest eigenvalue governs long-time behavior.

3. The Laplacian Matrix

3.1 Definition

$$\boxed{L = D - W}$$

Name	Symbol	Meaning
Degree Matrix	D	Number of edges at each vertex ($D_{ii} = d_i$)
Adjacency Matrix	W	Whether an edge exists between two nodes

3.2 Why Introduce L

W describes how nodes interact, but to measure **differences** (how much a node differs from its neighbors), the form $L = D - W$ is more natural. Sections 4.2 and 5.1 will show that L naturally produces $\sum(v_i - v_j)^2$, which is precisely the measure of **roughness** on the graph.

3.3 Normalized Laplacian

$$D^{-1/2}LD^{-1/2} = D^{-1/2}DD^{-1/2} - D^{-1/2}WD^{-1/2} = I - D^{-1/2}WD^{-1/2}$$

Define:

$$\boxed{L_{sys} = I - D^{-1/2}WD^{-1/2}}$$

The middle term $D^{-1/2}WD^{-1/2}$ is called the **standardized adjacency matrix**, denoted A:

$$A_{ij} = \frac{W_{ij}}{\sqrt{d_i d_j}} \quad \Longleftrightarrow \quad A_{ij} = \frac{1}{d_{ij}}, \;\; d_{ij} := \sqrt{\deg(i)\,\deg(j)}$$

3.4 Action Formulas for A and L_{sys}

$$(Av)_i = \sum_{j:(i,j)\in E} \frac{1}{d_{ij}}\, v_j$$

$$(L_{sys}\, v)_i = v_i - \sum_{j:(i,j)\in E} \frac{1}{d_{ij}}\, v_j$$

Intuition: L_{sys} subtracts from each node's value the degree-weighted average influence of its neighbors.

3.5 Spectral Decomposition (Power Form)

If $Lu_i = \lambda_i u_i$, expand $v = \sum_i \alpha_i u_i$:

$$L^k v = \sum_i \lambda_i^k \, \alpha_i \, u_i$$

The same structure as for W: spectral decomposition makes high powers of L transparent.

4. Rayleigh Quotient

4.1 Motivation

Can we find λ_k without solving $Lu = \lambda u$ directly?

Yes — via a variational (minimization) formulation.

4.2 Derivation

Step 1: Expand v in the eigenbasis of L:

$$v = \sum_k \alpha_k u_k, \qquad \alpha_k = u_k^T v$$

Step 2: Compute $v^T L v$:

$$v^T L v = \left(\sum_k \alpha_k u_k\right)^T L \left(\sum_j \alpha_j u_j\right) = \sum_{k,j} \alpha_k \alpha_j \lambda_j (u_k^T u_j) = \sum_k \alpha_k^2 \lambda_k$$

Step 3: Compute $v^T v$:

$$v^T v = \sum_k \alpha_k^2$$

Step 4: Take the ratio to obtain a bound:

$$\min_k \lambda_k \;\leq\; \frac{\sum_k \alpha_k^2 \lambda_k}{\sum_k \alpha_k^2} \;=\; \frac{v^T L v}{v^T v} \;\leq\; \max_k \lambda_k$$

This middle quantity is the **Rayleigh Quotient**:

$$\boxed{R(v) := \frac{v^T L v}{v^T v}}$$

4.3 Variational Characterization of λ_1 and λ_2

Step 4 (continued):

$$\lambda_1 = \min_{v \neq 0} R(v)$$

To find λ_2, we must "avoid u_1" by adding an orthogonality constraint:

$$\lambda_2 = \min_{v \perp u_1} R(v)$$

4.4 Courant–Fischer Theorem (Generalization)

Step 5: Generalize to all λ_k.

Observation:

$$\text{Find } \lambda_1 \leftrightarrow R(u_1) = \frac{1 \cdot \lambda_1 + 0 \cdot \lambda_2 + \cdots}{1^2 + 0^2 + \cdots}$$

$$\text{Find } \lambda_2 \leftrightarrow R(u_2) = \frac{0 \cdot \lambda_1 + 1 \cdot \lambda_2 + \cdots}{0^2 + 1^2 + \cdots}$$

The condition for the second one is $v \perp u_1$, forcing $\alpha_1 = 0$.

Key observation: In any 2D subspace V, there is always a vector orthogonal to u_1 (avoiding one direction requires only one constraint, so 2D is sufficient). Using this, λ_2 can be re-expressed as:

$$\boxed{\lambda_2 = \min_{\dim(V)=2} \max_{v \in V} R(v)}$$

This is the **Courant–Fischer Theorem**. The general form:

$$\lambda_k = \min_{\dim(V)=k} \max_{v \in V} R(v)$$

5. Laplacian + Rayleigh Quotient

5.1 Graph-Theoretic Form of $v^T L v$

$$\begin{aligned} v^T L v &= v^T (D - W) v \\ &= \sum_i d_i v_i^2 - \sum_{(i,j)\in E} 2\, v_i v_j \\ &= \sum_{(i,j)\in E} (v_i^2 + v_j^2 - 2 v_i v_j) \\ &= \sum_{(i,j)\in E} (v_i - v_j)^2 \end{aligned}$$

Note: The second line uses $\sum_i d_i v_i^2 = \sum_{(i,j)\in E} (v_i^2 + v_j^2)$, because node i appears d_i times in the sum (once per incident edge).

5.2 Graph-Theoretic Form of the Rayleigh Quotient

$$\boxed{R(v) = \frac{v^T L v}{v^T v} = \frac{\sum_{(i,j)\in E} (v_i - v_j)^2}{\sum_i v_i^2}}$$

5.3 Intuitive Interpretation

- **Numerator**: Sum of squared differences across all edges — **the roughness of v on the graph**
- **Denominator**: Squared norm of v — normalization

The meaning of $\lambda_2 = \min_{v \perp \mathbf{1}} R(v)$ is:

> "Among all non-constant assignments, how smooth can the smoothest one be?"

A small λ_2 means there exists a very smooth non-constant function — which is only possible when the graph has a bottleneck (the two sides of the bottleneck can take different values, with only a few edges crossing to produce variation).

5.4 $\lambda_1 = 0$ Is Trivial

Substituting $v = \mathbf{1}$:

$$R(\mathbf{1}) = \frac{\sum_{(i,j)\in E} (1-1)^2}{\sum_i 1^2} = \frac{0}{n} = 0$$

This holds for **any graph** — so $\lambda_1 = 0$ carries no structural information. λ_2 is the first meaningful quantity.

6. Spectral Filters (Derived from Heat Flow)

6.1 Motivation: What Is "Operating on a Signal"?

Apply a linear operation M to a graph signal v_{in}, obtaining output $v_{\text{out}} = M\, v_{\text{in}}$.

Inspecting M entry by entry is opaque. But in the **eigenbasis**, the operation becomes transparent — this is the core idea of spectral filtering.

In this section we derive the concept of a spectral filter concretely from heat flow.

6.2 Starting from the Heat Equation

The heat diffusion equation on a graph:

$$\frac{dv(t)}{dt} = -L\, v(t)$$

where $v(t)$ is the temperature distribution at time t.

Step 1: Single Mode

If the initial condition is exactly one eigenvector, $v(0) = u_k$, then since $Lu_k = \lambda_k u_k$:

$$\frac{dv(t)}{dt} = -\lambda_k\, v(t)$$

This scalar ODE has the solution:

$$v(t) = e^{-\lambda_k t}\, u_k$$

Conclusion: each eigenvector is an independently decaying mode, with decay rate λ_k.

Step 2: General Initial Condition

Any $v(0)$ expands in the eigenbasis:

$$v(0) = \sum_k \alpha_k u_k, \qquad \alpha_k = u_k^T v(0)$$

Since the heat equation is linear, each mode evolves independently:

$$\boxed{v(t) = \sum_k \alpha_k\, e^{-\lambda_k t}\, u_k}$$

6.3 Extracting the Operator Form from the Solution

What is the linear operator that maps $v(0)$ to $v(t)$?

Substituting $\alpha_k = u_k^T v(0)$:

$$v(t) = \sum_k e^{-\lambda_k t}\,(u_k^T v(0))\,u_k = \sum_k e^{-\lambda_k t}\,u_k u_k^T\,v(0)$$

Factoring out the sum:

$$v(t) = \left(\sum_k e^{-\lambda_k t}\,u_k u_k^T\right) v(0)$$

In matrix notation ($U = [u_1, \ldots, u_n]$, $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$):

$$\sum_k e^{-\lambda_k t}\,u_k u_k^T = U\,e^{-\Lambda t}\,U^T$$

Therefore:

$$\boxed{v(t) = \underbrace{U\,e^{-\Lambda t}\,U^T}_{\text{Heat Kernel } K(t)}\;v(0)}$$

This is the **heat kernel**: a matrix that maps the initial condition to the solution at time t.

6.4 Generalization: Definition of a Spectral Filter

The heat flow result reveals: **a linear operator can be written as "multiply each mode by some scalar in the eigenbasis."**

For heat flow the scalar is $e^{-\lambda_k t}$. In general, choose any function h:

$$\boxed{v_{\text{out}} = U\,h(\Lambda)\,U^T\,v_{\text{in}}}$$

where:

$$h(\Lambda) = \mathrm{diag}\big(h(\lambda_1),\, h(\lambda_2),\, \ldots,\, h(\lambda_n)\big)$$

This operation is called a **spectral filter**; h is the **filter function**.

Equivalently (summing over modes):

$$v_{\text{out}} = \sum_k h(\lambda_k)\,(u_k^T v_{\text{in}})\,u_k$$

That is: decompose v_{in} into modes (coefficients $u_k^T v_{\text{in}}$), scale each mode by $h(\lambda_k)$, and reassemble.

6.5 Identifying the Filter of a Given Operator

Given a linear operator M, how do we find its filter function $h(\lambda)$?

Rule: If M and L share eigenvectors $\{u_k\}$, then M is a spectral filter. Its filter function is determined by:

$$M\,u_k = h(\lambda_k)\,u_k$$

That is, $h(\lambda_k)$ is the eigenvalue of M at eigenvector u_k.

Example 1: Heat Kernel

$$M = e^{-Lt}$$

Acting on u_k: $e^{-Lt}u_k = e^{-\lambda_k t}u_k$ (since $Lu_k = \lambda_k u_k$).

So $h(\lambda) = e^{-\lambda t}$, consistent with Section 6.3.

Example 2: Normalized Adjacency

$$M = A = D^{-1/2}WD^{-1/2} = I - L_{sys}$$

Acting on u_k: $Au_k = (I - L_{sys})u_k = (1 - \lambda_k)u_k$.

So $h(\lambda) = 1 - \lambda$.

Example 3: p-th Power of the Laplacian

$$M = L^p$$

Acting on u_i: $L^p u_i = \lambda_i^p u_i$.

So $h(\lambda) = \lambda^p$.

6.6 Physical Meaning of Different Filters

Each filter function treats different frequency modes differently:

Filter $h(\lambda)$	Low freq. (λ small)	High freq. (λ large)	Type
$e^{-\lambda t}$	≈ 1 (preserved)	≈ 0 (attenuated)	**Low-pass**
$1-\lambda$	≈ 1 (preserved)	Small or negative	**Low-pass** (approx.)
λ^p	≈ 0 (suppressed)	Large	**High-pass**
1 (identity)	1	1	All-pass

Meaning of "low frequency" on graphs:

- $\lambda_1 = 0$: constant function $u_1 \propto \mathbf{1}$, no spatial variation
- λ_2 small: slowly varying function, varying at the scale of the whole graph
- λ_n large: rapidly oscillating function, adjacent nodes have opposite signs

Thus **low-pass filters preserve large-scale structure** (which nodes belong to the same cluster), while **high-pass filters preserve local differences** (which nodes differ from their neighbors).

6.7 Physical Intuition Revisited

Returning to the heat equation solution with $h(\lambda) = e^{-\lambda t}$:

- $t = 0$: $h(\lambda) = 1$ for all λ — no operation (just the initial condition)
- t increases: high-frequency modes (large λ) decay first; low-frequency modes (small λ) decay later
- $t \to \infty$: only the $\lambda_1 = 0$ mode survives — the system reaches thermal equilibrium (uniform temperature)

Choosing an appropriate t controls which structural scale is preserved. For example, choosing t^* such that $e^{-\lambda_2 t^*} \sim O(1)$ but $e^{-\lambda_k t^*} \approx 0$ for $k \geq 3$ leaves only the mode u_2 corresponding to λ_2 — i.e., the Fiedler vector dominates the temperature distribution.

This observation is the physical foundation for Section 13 (Grain Boundary Detection) and Section 14 (GNNs).

7. Graph Conductance (Cheeger Constant)

7.1 Motivation

We ask: **how easily can this graph be cut in two?**

Intuition: find a cut $(S, \bar{S})$ and count the crossing edges $|\partial S|$. But we need normalization (otherwise a singleton S always wins).

7.2 Definition

Conductance (for a single cut):

$$\phi(S) = \frac{|\partial S|}{\min(|S|,\, |\bar{S}|)}$$

Cheeger Constant (the graph-wide optimum):

$$\boxed{h(G) = \min_{\substack{S \subseteq V \\ 0 < |S| \leq n/2}} \frac{|\partial S|}{|S|}}$$

7.3 Normalized Version (Corresponding to L_{sys})

Replace the node-count denominator with the sum of degrees:

$$\phi_{\text{norm}}(S) = \frac{|\partial S|}{\min(\text{vol}(S),\, \text{vol}(\bar{S}))}$$

$$h_{\text{norm}}(G) = \min_{S} \ \phi_{\text{norm}}(S)$$

7.4 Why $h(G)$ Matters

$h(G)$ small $\iff$ a bottleneck cut exists $\iff$ the graph has cluster structure.

Problem: computing $h(G)$ is NP-hard; one must try 2^n cuts.

8. Cheeger Inequality

8.1 Statement

For the second-smallest eigenvalue λ_2 of the normalized Laplacian L_{sys}:

$$\boxed{\frac{\lambda_2}{2} \;\leq\; h_{\text{norm}}(G) \;\leq\; \sqrt{2\lambda_2}}$$

8.2 Meaning of the Two Halves

Inequality	Direction	Meaning
$\lambda_2 \leq 2\,h(G)$	Geometry $\to$ Algebra	If a good cut exists, λ_2 must be small
$h(G) \leq \sqrt{2\lambda_2}$	Algebra $\to$ Geometry	If λ_2 is small, a good cut can be constructed

8.3 Practical Significance

- Computing $h(G)$: NP-hard
- Computing λ_2: eigenvalue problem, polynomial time
- The Cheeger Inequality guarantees that $h(G)$ and $\sqrt{\lambda_2}$ are of the same order

This is the theoretical foundation of **spectral clustering**.

9. Easy Half: $\lambda_2 \le 2\,h(G)$

Strategy: given a good cut, construct a vector with a small Rayleigh Quotient.

9.1 Setup

Let S^* be the optimal cut achieving $h(G)$, with $|S^*| \le |\bar{S}^*|$. Write $a = |S^*|$, $b = |\bar{S}^*|$, so $a \le b$ and $a + b = n$.

9.2 Constructing the Test Vector

Define $v \in \mathbb{R}^n$:

$$v_i = \begin{cases} +b & i \in S^* \\ -a & i \in \bar{S}^* \end{cases}$$

Check $v \perp \mathbf{1}$:

$$\sum_i v_i = a \cdot b + b \cdot (-a) = 0 \quad \checkmark$$

9.3 Computing the Rayleigh Quotient

Numerator: only edges crossing the cut contribute (same-side edges have $v_i = v_j$):

$$v^T L v = \sum_{(i,j)\in E} (v_i - v_j)^2 = |\partial S^*| \cdot (b - (-a))^2 = |\partial S^*| \cdot n^2$$

Denominator:

$$v^T v = a \cdot b^2 + b \cdot a^2 = ab(a+b) = abn$$

Ratio:

$$R(v) = \frac{|\partial S^*| \cdot n^2}{abn} = \frac{|\partial S^*| \cdot n}{ab}$$

9.4 Upper Bound

Since $v \perp \mathbf{1}$, the variational characterization gives:

$$\lambda_2 \le R(v) = \frac{|\partial S^*|}{a} \cdot \frac{n}{b}$$

Observe:

- $\dfrac{|\partial S^*|}{a} = \dfrac{|\partial S^*|}{|S^*|} = h(G)$
- $b \geq n/2$ (since $a \leq b$ and $a + b = n$) $\Rightarrow$ $\dfrac{n}{b} \leq 2$

Therefore:

$$\boxed{\lambda_2 \leq 2\, h(G)}$$

■

10. Hard Half: $h(G) \leq \sqrt{2\lambda_2}$ (Strategy + Black Box)

Strategy: use u_2 directly to construct a good cut via a **sweep cut**.

The structure is described here; the complete rigorous proof is in Appendix B.

10.1 Setup

Take $f = u_2$, normalized so $\sum_i f_i^2 = 1$. We have:

$$\sum_i f_i = 0, \qquad \sum_i f_i^2 = 1, \qquad \sum_{(i,j)\in E} (f_i - f_j)^2 = \lambda_2$$

10.2 Sweep Cut

For threshold $t \geq 0$, define:

$$S_t = \{i : f_i > t\}$$

Since $\sum_i f_i = 0$, f has both positive and negative entries, so at most half the nodes have positive values $\Rightarrow |S_t| \leq n/2$ for all $t \geq 0$.

10.3 Averaging Argument

Lemma: For any non-negative functions $a(t), b(t)$ with $b(t) > 0$:

$$\min_t \frac{a(t)}{b(t)} \leq \frac{\int a(t)\,dt}{\int b(t)\,dt}$$

Proof: If not, then $a(t) > r \cdot b(t)$ for all t (where $r = \int a / \int b$), integrating gives $\int a > r \int b$, a contradiction. $\square$

Applying this:

$$h(G) \leq \min_{t\geq 0} \frac{|\partial S_t|}{|S_t|} \leq \frac{\int_0^\infty |\partial S_t|\,dt}{\int_0^\infty |S_t|\,dt}$$

10.4 Bounding the Numerator

Edge (i,j) crosses S_t iff exactly one endpoint has f-value above t. Assuming $f_i \geq f_j$, the crossing condition is $f_j \leq t < f_i$.

Each edge contributes at most $f_i - f_j = |f_i - f_j|$ to the integral. Summing over all edges:

$$\int_0^\infty |\partial S_t|\, dt \leq \sum_{(i,j)\in E} |f_i - f_j|$$

10.5 Bounding the Denominator

Node i contributes $\max(f_i, 0) =: (f_i)^+$ to the integral:

$$\int_0^\infty |S_t|\, dt = \sum_i (f_i)^+$$

10.6 Combining (So Far)

$$h(G) \leq \frac{\sum_{(i,j)\in E} |f_i - f_j|}{\sum_i (f_i)^+}$$

10.7 Bounding the Numerator: Cauchy–Schwarz

$$\left(\sum_{(i,j)\in E} |f_i - f_j|\right)^2 \leq \underbrace{|E|}_{=\sum_{(i,j)\in E} 1^2} \cdot \underbrace{\sum_{(i,j)\in E} (f_i - f_j)^2}_{=\lambda_2}$$

So $\sum_{(i,j)\in E} |f_i - f_j| \leq \sqrt{|E| \cdot \lambda_2}$.

10.8 Black Box: Handling the Denominator

To obtain the clean bound $\sqrt{2\lambda_2}$, one needs:

1. Switch to the normalized Laplacian L_{sys}; replace the denominator with $\sum_i d_i (f_i)^+$
2. Use $\sum_i d_i = 2|E|$
3. Apply a weighted Cauchy–Schwarz variant so that $\sqrt{|E|}$ appears in both numerator and denominator and cancels

The key insight is to sweep f^2 rather than f (details in Appendix B).

Final result:

$$\boxed{h_{\text{norm}}(G) \leq \sqrt{2\lambda_2}}$$

11. Overall Structure

```
                 Cheeger Inequality
              lambda_2 / 2  <=  h(G)  <=  sqrt(2 * lambda_2)
                               |
         +-----------------------+-----------------------+
         |                                               |
    Easy Half                                       Hard Half
    lambda_2 <= 2 h(G)                              h(G) <= sqrt(2 lambda_2)
         |                                               |
    given optimal cut S*                            take u_2 (Fiedler vector)
         |                                               |
    construct v in {+b, -a}                         Sweep Cut S_t
         |                                               |
    plug into Rayleigh Quotient                     Averaging Argument
         |                                               |
    direct calculation                              numerator: sum |f_i - f_j|
                                                    denominator: sum (f_i)+
                                                         |
                                                    Cauchy-Schwarz
                                                         |
                                                    [black box: refined ineq]
                                                         |
                                                    sqrt(2 lambda_2)
```

12. Application: Grain Boundary Detection

We apply the full theory to a real physical problem: automatically locating grain boundaries in polycrystalline materials.

12.1 The Physical Problem

Inside a polycrystalline metal there are multiple grains, each with an ordered but differently oriented atomic lattice. The interfaces between grains are **grain boundaries**, where the atomic arrangement is disordered.

Question: can we locate grain boundaries automatically, without manual labeling or structural analysis?

12.2 Graph Construction

Model the material as a weighted graph:

- **Nodes**: atoms
- **Edges**: pairs of atoms within a cutoff distance
- **Weight** W_{ij}: bond stiffness (second derivative of the interatomic potential)

  Remark: Under the harmonic approximation, this weighted Laplacian $L = D - W$ is exactly the dynamical matrix of condensed-matter physics, and shares the same mathematical structure as phonon theory.

12.3 Heating the Material: Heat Diffusion Equation

Apply an initial temperature $\mathbf{T}(0)$ to some atoms. Internal heat diffusion obeys (Section 3):

$$\frac{d\mathbf{T}}{dt} = -L\,\mathbf{T}$$

By the derivation in Section 6, the solution is:

$$\mathbf{T}(t) = \sum_k \alpha_k\, e^{-\lambda_k t}\, u_k, \qquad \alpha_k = u_k^T \mathbf{T}(0)$$

12.4 Long-time Convergence to u_2

The heat kernel acts as a low-pass filter (Section 6.7): high-frequency modes decay faster. Choose t^* such that:

- $e^{-\lambda_2 t^*} \sim O(1)$ (the λ_2 mode survives)
- $e^{-\lambda_k t^*} \approx 0$ for $k \geq 3$ (higher modes have decayed)

Then:

$$\mathbf{T}(t^*) \approx \underbrace{\alpha_1 u_1}_{\text{equilibrium}} + \alpha_2 \, e^{-\lambda_2 t^*} \, u_2$$

After subtracting the equilibrium temperature, the residual distribution is proportional to u_2 (the Fiedler vector).

12.5 Why u_2 Corresponds to Grain Boundaries

From the Easy Half

Suppose the two grains contain a and b atoms respectively $(a + b = n)$. Construct the test vector:

$$v_i = \begin{cases} +b & i \in \text{Grain A} \\ -a & i \in \text{Grain B} \end{cases}$$

Check $v \perp \mathbf{1}$: $\sum_i v_i = a \cdot b + b \cdot (-a) = 0$ ✓

Substituting into the Rayleigh Quotient (only bonds crossing the grain boundary contribute):

$$v^T L v = \sum_{(i,j) \in \partial S} W_{ij} \, (b - (-a))^2 = \bar{k} \, |\partial S| \, n^2$$

$$v^T v = ab^2 + ba^2 = abn$$

$$\lambda_2 \le R(v) = \frac{\bar{k} \, |\partial S| \, n}{ab}$$

Fewer ($|\partial S|$ small) or softer ($\bar{k}$ small) grain-boundary bonds imply smaller λ_2 and stronger thermal resistance.

From the Cheeger Inequality

The grain boundary corresponds to a bottleneck cut; $h(G) = |\partial S|/|S|$ directly quantifies how "thin" the boundary is.

The Cheeger Inequality guarantees:

$$\frac{\lambda_2}{2} \le h(G) \le \sqrt{2 \, \lambda_2}$$

Furthermore, the sweep-cut algorithm of Section 10 guarantees: **cutting by the sign of u_2 yields a solution within a $\sqrt{\ }$ factor of optimal.**

12.6 Complete Pipeline

Step	Operation	Tool
1	Build graph (atoms + bonds)	Section 0
2	Heat some atoms	Set $\mathbf{T}(0)$
3	Solve heat equation to t^*	Section 6 heat kernel
4	Measure each atom's temperature	$\mathbf{T}(t^*) \propto u_2$
5	Read the sign of the temperature	Sign of $u_2 \to$ grain assignment
6	Examine temperature differences	Large $\|u_{2,i} - u_{2,j}\| \to$ grain boundary bond

12.7 Conclusion

Heat the material; wait until $t^* \sim 1/\lambda_2$; measure each atom's temperature:

- Atoms with high temperature belong to one grain
- Atoms with low temperature belong to the other grain
- Where the temperature changes sharply is the grain boundary

The validity of the entire approach is guaranteed by the Cheeger Inequality. The mechanism is the heat kernel acting as a spectral low-pass filter, automatically discarding local fluctuations and retaining only the large-scale grain structure. **No prior knowledge of grain locations is needed.**

13. A Spectral View of GNNs

The final step: understand GNNs (specifically GCNs) within the spectral filter framework. We will find that **GCNs are mathematically the same as heat diffusion**, approximated with a different filter function.

13.1 What Does One GCN Layer Do?

Each node has a feature vector; the system state is the matrix $H^{(0)} \in \mathbb{R}^{n\times d}$ (d features).

One GCN layer update rule:

$$H^{(k+1)} = \sigma\Big(A\,H^{(k)}\,W^{(k)}\Big)$$

Three independent components:

Component	Role	Graph-dependent?
$A = D^{-1/2}WD^{-1/2}$	Aggregation: weighted neighbor average	Yes — fixed by graph structure
$W^{(k)}$	Linear transformation: learn feature combinations	No — learnable
σ	Non-linearity (ReLU, etc.)	No

We focus on the aggregation A.

13.2 Spectral Form of One-Layer GCN Aggregation

From Section 6.5, Example 2:

$$A = I - L_{sys}, \qquad h_{\text{1-layer}}(\lambda) = 1 - \lambda$$

One layer of GCN aggregation is one application of a filter with filter function $h(\lambda) = 1 - \lambda$.

Effect on each mode:

- $\lambda_1 = 0$: $h = 1$, fully preserved
- λ_2 small: $h \approx 1$, nearly preserved
- λ large: h small, suppressed

Conclusion: one GCN aggregation layer = one low-pass filter.

13.3 Multi-layer GCN

Ignoring σ and $W^{(k)}$, after k layers:

$$H^{(k)} = A^k \, H^{(0)} = U \, (I - \Lambda)^k \, U^T \, H^{(0)}$$

Corresponding filter:

$$h_{\text{k-layer}}(\lambda) = (1 - \lambda)^k$$

Each additional layer multiplies every mode by another factor of $(1 - \lambda)$.

13.4 Mathematical Correspondence with the Heat Kernel

For small λ ($\lambda \ll 1$), Taylor expansion:

$$(1 - \lambda)^k = e^{k \ln(1-\lambda)} \approx e^{-k\lambda}$$

So for low-frequency modes:

$$\boxed{h_{\text{GCN}}(\lambda) = (1 - \lambda)^k \approx e^{-\lambda k} = h_{\text{heat}}(\lambda)\Big|_{t=k}}$$

k layers of GCN $\approx$ heat flow running for time $t = k$. Each additional layer = one more diffusion step.

Differences at high frequencies:

λ	Heat $e^{-\lambda t}$	GCN $(1 - \lambda)^k$
$\lambda \to 0$	1	1
λ moderate	> 0, smooth decay	> 0, smooth decay
$\lambda = 1$	$e^{-t} > 0$	0 (exactly zero)
$\lambda > 1$	Still positive	Oscillates ($(1 - \lambda)^k$ changes sign)

GCN exhibits oscillatory behavior for $\lambda > 1$ (since L_{sys} has eigenvalues in $[0, 2]$), unlike the heat kernel, but the two are essentially equivalent for low-frequency modes.

13.5 What Does $W^{(k)}$ Do?

A^k determines **how to propagate** (fixed by graph structure).

$W^{(k)}$ determines **which linear combinations of features are useful** (determined by training).

Division of labor:

- Graph structure $\to$ spectral filter $h(\lambda) = (1-\lambda)^k$
- Training $\to$ selection in feature space

13.6 Conclusion

Comparison	Heat Flow	GCN
Spectral filter	$h(\lambda) = e^{-\lambda t}$ (exact)	$h(\lambda) = (1-\lambda)^k$ (approximate)
"Time" scale	Physical time t	Number of layers k
Node signal	Temperature (external)	Features (learnable)
Goal	Reach equilibrium	Learn useful representations

Core conclusion:

$$\text{Heat flow knows the optimal filter} \xrightarrow{\text{approximate}} \text{GCN approximates it with } (1-\lambda)^k$$

$$\text{Heat flow: fixed node signals (temperature)} \xrightarrow{\text{generalize}} \text{GCN adds learnable feature transforms}$$

Without knowing u_2, GCNs automatically approximate the heat kernel through stacked layers, rediscovering the Fiedler vector. The Cheeger Inequality provides the scale of how many layers are needed, tying all of spectral graph theory into a coherent engineering principle.

Appendix A: Comparison of the Two Laplacian Versions

Item	Unnormalized L	Normalized L_{sys}
Matrix	$L = D - W$	$I - D^{-1/2} W D^{-1/2}$
Rayleigh Quotient	$\dfrac{\sum (v_i - v_j)^2}{\sum v_i^2}$	$\dfrac{\sum (v_i - v_j)^2}{\sum d_i v_i^2}$
Cheeger denominator	$\min(\lvert S\rvert, \lvert \bar{S}\rvert)$	$\min(\mathrm{vol}(S), \mathrm{vol}(\bar{S}))$
Cheeger Inequality	Less clean	$\dfrac{\lambda_2}{2} \leq h \leq \sqrt{2\lambda_2}$

Appendix B: Complete Rigorous Proof of the Hard Half (Technical Details)

The following is the complete rigorous proof of $h_{\text{norm}}(G) \le \sqrt{2\lambda_2}$ for the normalized Laplacian. This appendix is for reference; no pedagogical explanation is given.

Setup. Let f be the eigenvector of L_{sys} corresponding to λ_2. After the transformation $D^{1/2}$, it can be viewed as satisfying the generalized eigenvalue problem $Lf = \lambda_2 Df$, with:

$$\sum_i d_i f_i = 0, \qquad \sum_i d_i f_i^2 = 1, \qquad \sum_{(i,j)\in E} (f_i - f_j)^2 = \lambda_2$$

Step 1 (Positive/negative splitting). Sort nodes by f-value and choose the d-weighted median m satisfying:

$$\mathrm{vol}(\{i : f_i > m\}) \le \tfrac{1}{2}\,\mathrm{vol}(V), \quad \mathrm{vol}(\{i : f_i < m\}) \le \tfrac{1}{2}\,\mathrm{vol}(V)$$

Define $g_i = f_i - m$, and let $g_i^+ = \max(g_i, 0)$, $g_i^- = \max(-g_i, 0)$. Then $g_i^+ g_i^- = 0$ and $g_i = g_i^+ - g_i^-$.

Step 2 (Rayleigh quotient for each half). From the eigenvector property of f and $g = f - m\mathbf{1}$, at least one of the two halves (say g^+) satisfies:

$$\frac{\sum_{(i,j)\in E} (g_i^+ - g_j^+)^2}{\sum_i d_i (g_i^+)^2} \le \lambda_2 \tag{$\star$}$$

and $\mathrm{vol}(\mathrm{supp}(g^+)) \le \frac{1}{2}\,\mathrm{vol}(V)$. Write $h = g^+$.

Step 3 (Core lemma — Cheeger's lemma). For any non-negative function h with $\mathrm{vol}(\mathrm{supp}(h)) \le \frac{1}{2}\,\mathrm{vol}(V)$:

$$\min_t \phi_{\text{norm}}(\{i : h_i > t\}) \le \sqrt{2 \cdot \frac{\sum_{(i,j)\in E} (h_i - h_j)^2}{\sum_i d_i h_i^2}}$$

Proof (key technique: sweep h^2 rather than h).

For $T_t = \{i : h_i^2 > t\} = \{i : h_i > \sqrt{t}\}$ ($t \ge 0$):

$$\int_0^\infty |\partial T_t|\, dt = \sum_{(i,j)\in E} |h_i^2 - h_j^2|, \qquad \int_0^\infty \mathrm{vol}(T_t)\, dt = \sum_i d_i h_i^2$$

By the averaging argument:

$$\min_t \phi_{\text{norm}}(T_t) \le \frac{\sum_{(i,j)\in E} |h_i^2 - h_j^2|}{\sum_i d_i h_i^2}$$

Apply Cauchy–Schwarz to the numerator (**this is the key step that produces a clean constant**):

$$\sum_{(i,j)\in E} |h_i^2 - h_j^2| = \sum_{(i,j)\in E} |h_i - h_j| \cdot |h_i + h_j| \le \sqrt{\sum_{(i,j)\in E} (h_i - h_j)^2} \cdot \sqrt{\sum_{(i,j)\in E} (h_i + h_j)^2}$$

For the second factor, $(h_i + h_j)^2 \le 2(h_i^2 + h_j^2)$, so:

$$\sum_{(i,j)\in E} (h_i + h_j)^2 \le 2 \sum_{(i,j)\in E} (h_i^2 + h_j^2) = 2 \sum_i d_i h_i^2$$

Substituting:

$$\sum_{(i,j)\in E} |h_i^2 - h_j^2| \le \sqrt{2} \cdot \sqrt{\sum_{(i,j)\in E} (h_i - h_j)^2} \cdot \sqrt{\sum_i d_i h_i^2}$$

Dividing by $\sum_i d_i h_i^2$:

$$\min_t \phi_{\text{norm}}(T_t) \le \frac{\sqrt{2}\sqrt{\sum (h_i - h_j)^2}\sqrt{\sum d_i h_i^2}}{\sum d_i h_i^2} = \sqrt{\frac{2\sum_{(i,j)\in E}(h_i - h_j)^2}{\sum_i d_i h_i^2}}$$

□

Step 4 (Combining Steps 2 and 3). Substituting $h = g^+$ and using ($\star$):

$$\min_t \phi_{\text{norm}}(\{i : g_i^+ > t\}) \le \sqrt{2\lambda_2}$$

Since $\{i : g_i^+ > t\} = \{i : f_i > m + t\}$ are level sets of f with volume at most $\frac{1}{2}\,\text{vol}(V)$:

$$h_{\text{norm}}(G) \le \min_t \phi_{\text{norm}}(\{i : f_i > m + t\}) \le \sqrt{2\lambda_2}$$

■

Remark: Sweeping h^2 is what makes the Cauchy–Schwarz numerator and denominator align naturally — one side yields $(h_i - h_j)^2$, the other $(h_i + h_j)^2$ paired with $\sum d_i h_i^2$ — and is what produces the clean constant $\sqrt{2}$. Sweeping h directly (as in Sections 10.4–10.7) gets stuck at the denominator; this is exactly what the "black box" of Section 10.8 refers to.

References:

- F. R. K. Chung, *Spectral Graph Theory*, AMS, 1997.
- D. Spielman, *Spectral and Algebraic Graph Theory*, Yale lecture notes.
- L. Trevisan, *Lecture Notes on Expansion, Sparsest Cut, and Spectral Graph Theory.*