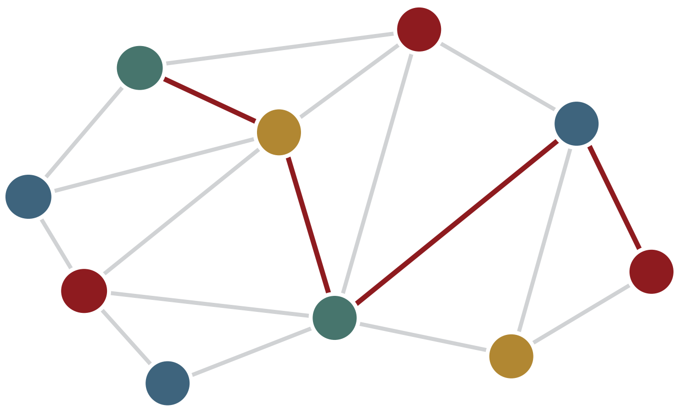
Stability of Graph Convolutional Filters and Graph Convolutional NNs



September 3

Zhiyang Wang · Electrical & Systems Engineering · WashU

Class 4 · Stability Analysis of Graph Filters and GNNs

Three questions organize the lecture

01 **WHAT HAS CHANGED?**

Separate arbitrary node indexing from genuine changes in topology or edge weights -- absolute and relative perturbations

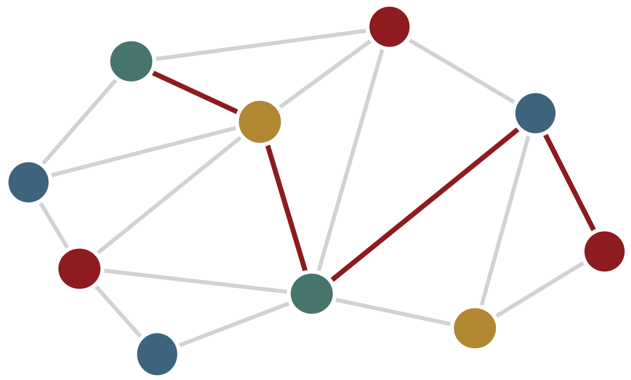
02 **WHEN IS A FILTER STABLE?**

Match the perturbation model with the correct smoothness condition on $h(\lambda)$ -- Lipschitz or integral Lipschitz.

03 **WHY CASCADE?**

Show that normalized nonlinear layers inherit stability while restoring discrimination.

Separate Graph Changes from Permutation Equivariance.



01 Equivariance is exact for any relabeling

02 Exact graph symmetries are rare and fragile

03 Align first: measure changes modulo permutation

04 Stability bounds the output error by ε

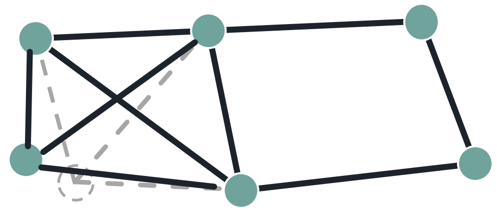
Equivariance handles relabelings; stability handles change.

Stability matters in deployment and learning

DEPLOYMENT · THE GRAPH DRIFTS

Sensors move or fail.

Edge weights are re-estimated.
Controller sees graphs not present at training.

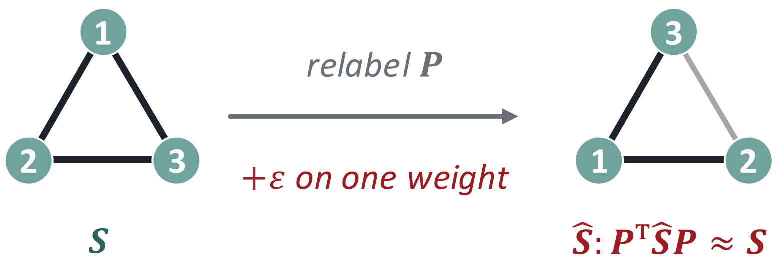


a sensor moves: $S \to \hat{S}$

LEARNING · SYMMETRIES ARE APPROXIMATE

Relabelings give exact equivariance.

Near-symmetries act like implicit data augmentation.
Similar graphs should produce similar representations.



Goal: graph distance $O(\varepsilon)$ $\Rightarrow$ output distance $O(\varepsilon)$.

Stability is approximate equivariance under structural change.

Equivariance become quasi-equivariance

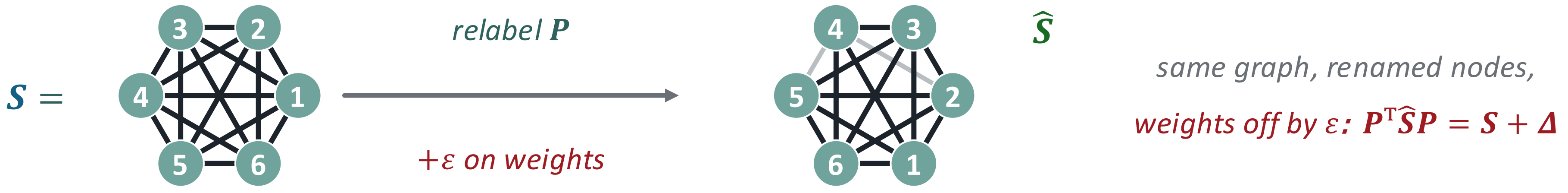
For permutation equivariance, the shift and filter commute with permutations:

$$\boldsymbol{P}^T\hat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} \quad \Rightarrow \quad h(\boldsymbol{P}^T\hat{\boldsymbol{S}}\boldsymbol{P})\boldsymbol{P}^T\boldsymbol{x} = \boldsymbol{P}^T h(\boldsymbol{S})\boldsymbol{x}$$

For a quasi-equivariance, the best relabeling leaves a small residual structural change:

$$\boldsymbol{P}^T\hat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} + \boldsymbol{\Delta}, \qquad \|\boldsymbol{\Delta}\| = \varepsilon$$

- Exact permutation equivariance gives zero output discrepancy.
- Stability makes the output discrepancy to be bounded with ε.



Equivariance is the zero-perturbation endpoint of stability.

A graph filter can be perturbed in three places

$$\boldsymbol{y} = h(\boldsymbol{S})\boldsymbol{x}$$

01 INPUT $\boldsymbol{x} \to \boldsymbol{x} + \boldsymbol{e}$

Easy: $h(\boldsymbol{S})$ is linear in $\boldsymbol{x}$, so the output change is controlled directly by $\|h(\boldsymbol{S})\|$.

02 TAPS $h \to h + \Delta h$

Easy: the filter is linear in its coefficients, so difference controlled by $\|\boldsymbol{S}\|$ and $\|\boldsymbol{x}\|$.

03 SHIFT $\boldsymbol{S} \to \widehat{\boldsymbol{S}}$

Hard and essential: $\boldsymbol{S}$ enters through powers $\boldsymbol{S}^{k}$, so graph transfer is nonlinear in the operator.

Fix signal and taps, perturb only the graph

$$h(\boldsymbol{S})\boldsymbol{x} = \sum_k h_k \boldsymbol{S}^k \boldsymbol{x} \qquad h(\hat{\boldsymbol{S}})\boldsymbol{x} = \sum_k h_k \hat{\boldsymbol{S}}^k \boldsymbol{x}$$

The same input $\boldsymbol{x}$ and coefficients h_k are deployed on two nearby graph shifts.

THE GRAPH DEPENDENCE LIVES IN THE POWERS

$$\boldsymbol{\Delta}_h \boldsymbol{x} = \left[h(\boldsymbol{S}) - h(\hat{\boldsymbol{S}})\right]\boldsymbol{x}$$

$$= \Big[\underbrace{\textstyle\sum_k h_k \left(\boldsymbol{S}^k - \hat{\boldsymbol{S}}^k\right)}_{\text{Operator norm}}\Big]\boldsymbol{x}$$

A small shift change affects every power. Perturbations propagate by diffusions.

The stability problem is therefore an operator perturbation problem.

Compare induced filter operators after the best relabeling.

Compare graph operators modulo relabeling

Operator modulo permutation $\left\|\hat{\boldsymbol{S}} - \boldsymbol{S}\right\|_{\mathcal{P}} = \min_{\mathcal{P}}\left\|\hat{\boldsymbol{S}}\boldsymbol{P}^{\boldsymbol{T}} - \boldsymbol{P}^{\boldsymbol{T}}\boldsymbol{S}\right\| = \min_{\mathcal{P}}\left\|\boldsymbol{P}\hat{\boldsymbol{S}}\boldsymbol{P}^{\boldsymbol{T}} - \boldsymbol{S}\right\|$

Apply the same measure to the induced filter operators:

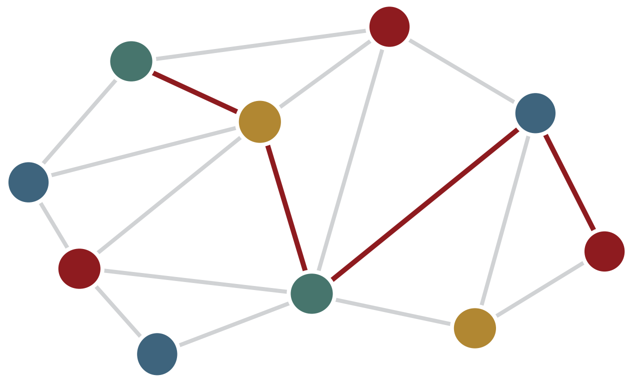
$$\left\|h(\hat{\boldsymbol{S}}) - h(\boldsymbol{S})\right\|_{\mathcal{P}} = \min_{\mathcal{P}}\left\|h(\hat{\boldsymbol{S}})\boldsymbol{P}^{\boldsymbol{T}} - \boldsymbol{P}^{\boldsymbol{T}}h(\boldsymbol{S})\right\|$$

- Distance zero recovers exact permutation equivariance.
- A small nonzero distance represents a quasi-symmetry after the best node correspondence.

Measure structural change within permutations.

PART II

Continuous Filters Ensure Stability under Absolute and Relative Perturbations.



01 Absolute model: fixed-size edge changes

02 Relative model: errors scale with the graph

03 Lipschitz filters are stable to absolute errors

04 Integral Lipschitz handles relative errors

**Two error models,
two filter conditions,
one bound in ε.**

Error and distance modulo permutation

Definition of the error matrix — start from the set of symmetric error matrices:

$$\mathcal{E}(\mathbf{S},\hat{\mathbf{S}}) = \left\{\tilde{\mathbf{E}} : \quad \mathbf{P}^T\hat{\mathbf{S}}\mathbf{P} = \mathbf{S} + \tilde{\mathbf{E}}, \qquad \mathbf{P} \in \mathcal{P}\right\}$$

- Each permutation $\mathbf{P} \in \mathcal{P}$ gives a different error matrix $\tilde{\mathbf{E}} = \mathbf{P}^T\hat{\mathbf{S}}\mathbf{P} - \mathbf{S}$ in the set.
- The error matrix modulo permutation is the one with the smallest norm: $\mathbf{E} = \arg \min_{\tilde{\mathbf{E}}\in\mathcal{E}(\mathbf{S},\hat{\mathbf{S}})} \|\tilde{\mathbf{E}}\|$.
- $\|\mathbf{E}\| = d(\mathbf{S},\hat{\mathbf{S}})$ measures how far $\mathbf{S}$ and $\hat{\mathbf{S}}$ are from being permutations of each other.

$$d(\mathbf{S},\hat{\mathbf{S}}) = \|\mathbf{E}\| = \min_{\tilde{\mathbf{E}}\in\mathcal{E}(\mathbf{S},\hat{\mathbf{S}})} \|\tilde{\mathbf{E}}\|$$

The distance modulo permutation is the norm of the best-aligned error matrix.

$\boldsymbol{\Delta}$ shifts eigenvalues and rotates eigenvectors

Add and subtract the mixed term — the filter difference splits exactly into two parts:

$$h(\hat{\boldsymbol{S}}) - h(\boldsymbol{S}) = \hat{\boldsymbol{V}} h(\hat{\boldsymbol{\Lambda}})\,\hat{\boldsymbol{V}}^{\mathrm{T}} - \hat{\boldsymbol{V}}\,h(\boldsymbol{\Lambda})\hat{\boldsymbol{V}}^{\mathrm{T}} + \hat{\boldsymbol{V}}\,h(\boldsymbol{\Lambda})\,\hat{\boldsymbol{V}}^{\mathrm{T}} - \boldsymbol{V} h(\boldsymbol{\Lambda})\boldsymbol{V}^{\mathrm{T}}$$

$$= \hat{\boldsymbol{V}}[h(\hat{\boldsymbol{\Lambda}}) - h(\boldsymbol{\Lambda})]\,\hat{\boldsymbol{V}}^{\mathrm{T}} + [\hat{\boldsymbol{V}}\,h(\boldsymbol{\Lambda})\,\hat{\boldsymbol{V}}^{\mathrm{T}} - \boldsymbol{V} h(\boldsymbol{\Lambda})\boldsymbol{V}^{\mathrm{T}}]$$

- **Eigenvalue channel —** the diagonal of $\boldsymbol{\Delta}$ in the eigenbasis. Weyl: every eigenvalue moves by at most ε
- **Eigenvector channel —**Davis–Kahan tells the distance between eigenvector spaces, or by defining eigenvector mismatches

Filter function controls the eigenvalue channel. Perturbation decides the eigenvector channel.

Eigenvector rotation appears in the stability bound

Why does the bound contain an alignment factor δ?

- Eigendecompose the shift and the error: $\boldsymbol{S} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}^{\boldsymbol{H}}$ and $\boldsymbol{E} = \boldsymbol{U}\boldsymbol{M}\boldsymbol{U}^{\boldsymbol{H}}$.
- Define the eigenvector misalignment between $\boldsymbol{S}$ and $\boldsymbol{E}$ as

$$\delta = (\|\boldsymbol{U} - \boldsymbol{V}\| + 1)^2 - 1$$

- $\boldsymbol{U}$ and $\boldsymbol{V}$ are unitary, $\|\boldsymbol{U}\| = \|\boldsymbol{V}\| = 1 \quad \Rightarrow \quad \delta \leq (2+1)^2 - 1 = 8.$ – worst case bound

- If $\boldsymbol{S}$ and $\boldsymbol{E}$ share eigenvectors ($\boldsymbol{U} \approx \boldsymbol{V}$), then $\delta \approx 0$ — the perturbation mainly moves eigenvalues.

The misalignment δ can be small, depending on the error model.

Eigenvalue sensitivity is shaped by h. Eigenvector sensitivity is shaped by the graph error.

Lipschitz filters are stable to additive perturbations

- Shift matrix related with $\boldsymbol{P}^T\hat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} + \boldsymbol{E}$
- Error matrix $\|\boldsymbol{E}\| = \varepsilon$
- Lipschitz frequency response

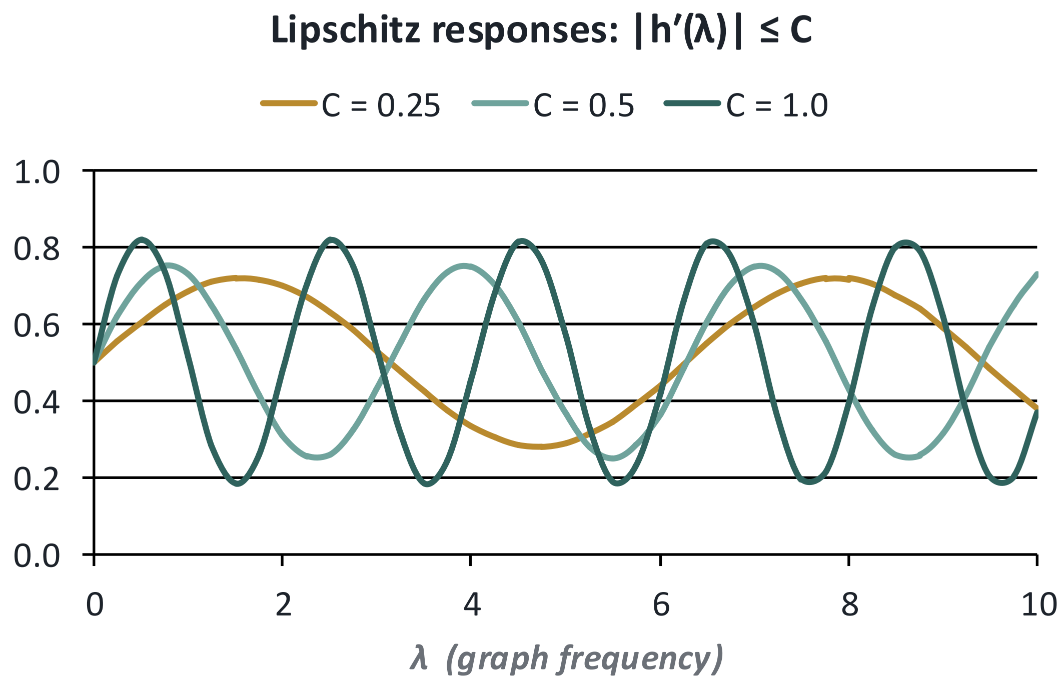
$$|h(\lambda_2) - h(\lambda_1)| \leq C|\lambda_2 - \lambda_1|.$$

$$\boxed{\left\|h(\hat{\boldsymbol{S}}) - h(\boldsymbol{S})\right\|_{\mathcal{P}} \leq C(1 + \delta\sqrt{n})\varepsilon + O(\varepsilon^2)}$$

FIRST-ORDER STRUCTURE

filter error = response slope × alignment × graph error

The leading term is linear in ε. The higher-order remainder vanishes faster than ε as the perturbation shrinks.



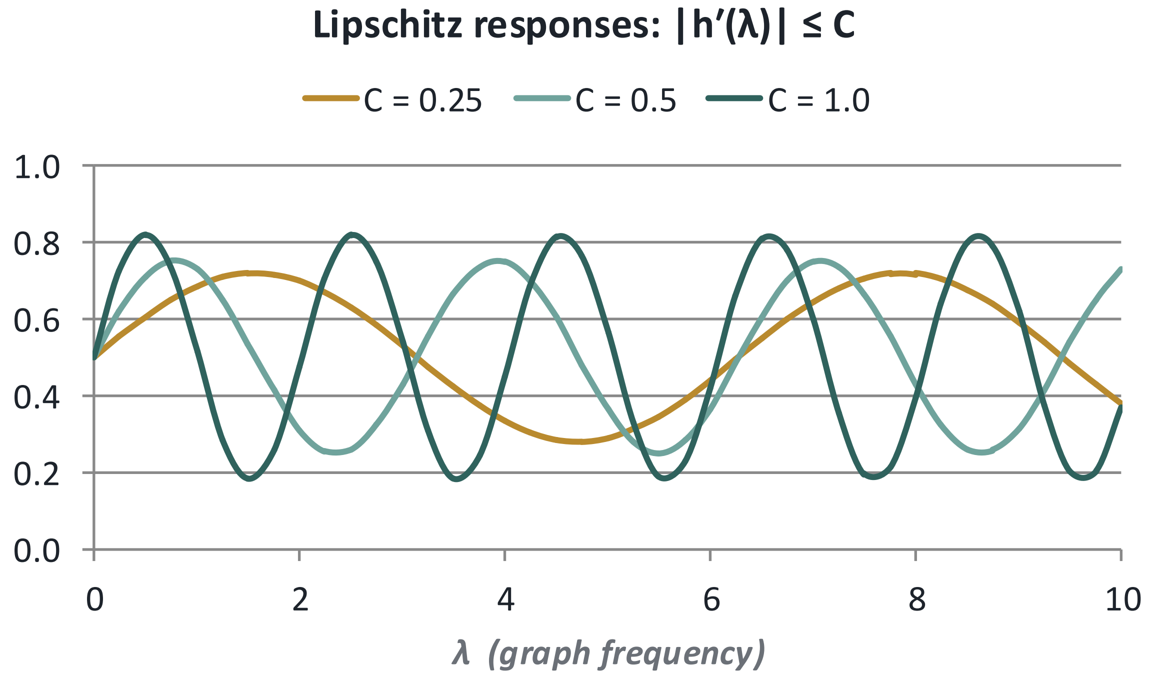
A controllable spectral slope turns graph closeness into output closeness.

Design controls C. Deployment controls ε, δ, n

CONTROLLED · FILTER SMOOTHNESS C

Choose smoother $h(\lambda)$.

Regularize or constrain the taps.
Trade sharp spectral selectivity for robustness.



UNCONTROLLED · ε, δ, n

ε measures graph drift.

δ measures eigenspace misalignment.

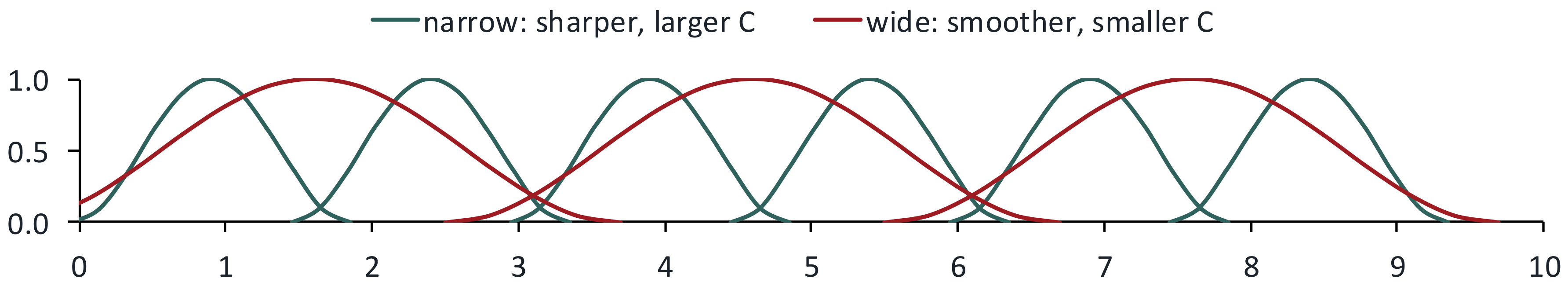
n records graph size in the universal worst-case estimate.

The theorem exposes an architectural knob and an environmental cost.

Stability trades off discriminability

- Bound is universal for all graphs with nodes n
  - Continuity of filter's frequency response
  - Properties of perturbation $\boldsymbol{E}$
- Discriminability and stability tradeoff – Larger C, sharper discriminability, worse stability
- The theorem factors learned-filter sensitivity from deployed-graph geometry.

Kernel frequency responses over λ



Continuity says the map behaves locally. Stability quantifies how well it transfers.

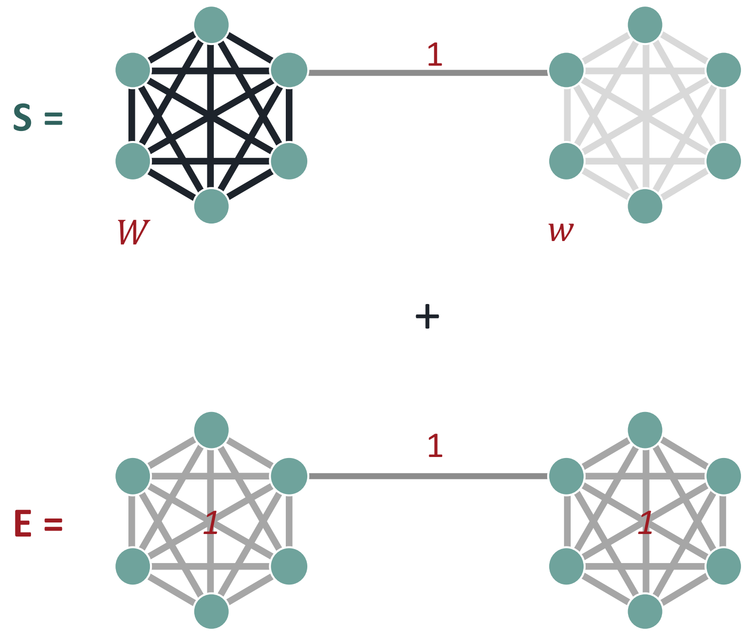
Absolute edge errors can be physically misleading

Suppose $w \ll 1 \ll W.$

$$\boldsymbol{P}^T\widehat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} + \boldsymbol{E}$$

- Can we claim the perturbation is large or small?
- For edges with weights w– changes are larger

Many physical graphs are governed by proportional calibration, distance, or gain errors.

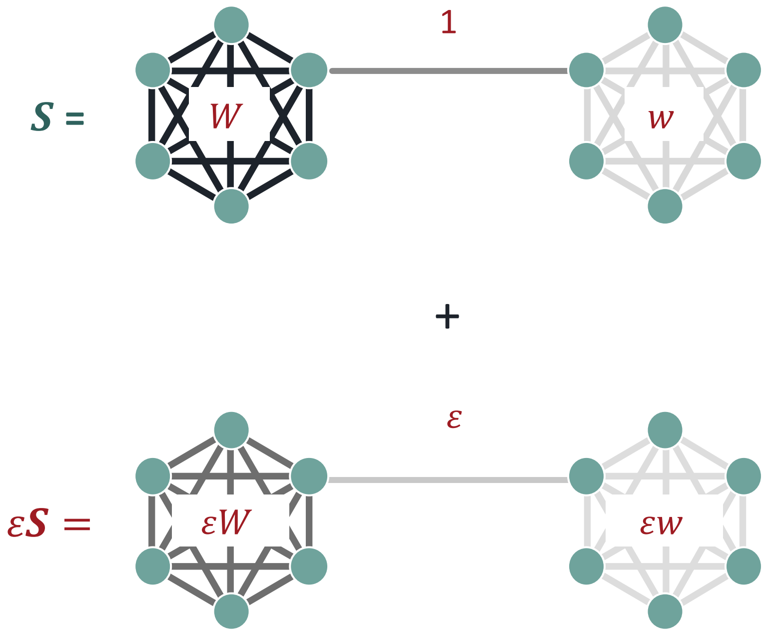


Weighted graphs often require a perturbation measured relative to local connectivity.

Relative perturbations are meaningful

Suppose $w \ll 1 \ll W.$

$$\boldsymbol{P}^T\hat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} + \boldsymbol{\varepsilon S}$$



- Can we claim the perturbation is large or small?
- It’s small with larger edge weights change more and smaller change less

Weighted graphs often require a perturbation measured relative to local connectivity.

Error and distance modulo permutation

Definition of the error matrix — start from the set of symmetric error matrices:

$$\mathcal{E}(\boldsymbol{S},\widehat{\boldsymbol{S}}) = \left\{\widetilde{\boldsymbol{E}} : \; \boldsymbol{P}^T\widehat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} + \widetilde{\boldsymbol{E}}\boldsymbol{S} + \boldsymbol{S}\widetilde{\boldsymbol{E}}, \qquad \boldsymbol{P} \in \mathcal{P}\right\} \qquad (\boldsymbol{I}+\boldsymbol{E})\boldsymbol{S}(\boldsymbol{I}+\boldsymbol{E})$$

- The error matrix modulo permutation is the one with the smallest norm: $\boldsymbol{E} = \arg \min_{\tilde{\boldsymbol{E}} \in \mathcal{E}(\boldsymbol{S},\hat{\boldsymbol{S}})} \|\tilde{\boldsymbol{E}}\|$.
- $\|\boldsymbol{E}\| = d(\boldsymbol{S},\widehat{\boldsymbol{S}})$ is a relative measure of how far $\boldsymbol{S}$ and $\widehat{\boldsymbol{S}}$ are from being permutations of each other.

$$d(\boldsymbol{S},\widehat{\boldsymbol{S}}) = \|\boldsymbol{E}\| = \min_{\widetilde{\boldsymbol{E}} \in \mathcal{E}(\boldsymbol{S},\hat{\boldsymbol{S}})} \|\widetilde{\boldsymbol{E}}\|$$

The distance modulo permutation is the norm of the best-aligned error matrix.

Relative errors follow local connectivity

$$[\boldsymbol{P}^T\widehat{\boldsymbol{S}}\boldsymbol{P}]_{ij} = [\boldsymbol{S} + \boldsymbol{ES} + \boldsymbol{SE}]_{ij} = \boldsymbol{S}_{ij} + \sum_{m\in\mathcal{N}_j} \boldsymbol{E}_{im}\boldsymbol{S}_{mj} + \sum_{m\in\mathcal{N}_i} \boldsymbol{S}_{im}\boldsymbol{E}_{mj}$$

Edge changes are proportional to original weights and perturbations at two endpoints.

DIAGONAL ERROR OPERATORS MAKE THE SCALING EXPLICIT

$$\boldsymbol{E} = diag(e_1, \ldots, e_n) \Rightarrow [\boldsymbol{ES} + \boldsymbol{SE}]_{ij} = (e_i + e_j)\boldsymbol{S}_{ij}$$

Parts of the graph with lower weighted degree see smaller changes

In additive perturbations weights can change the same regardless of the connectivity.

The model connects matrix perturbations to local physical calibration.

Relative errors need integral-Lipschitz smoothness

LIPSCHITZ · ABSOLUTE SCALE

Controls response change w.r.t. absolute frequency separation.
The same slope limit applies everywhere.

$$|h(\lambda) - h(\lambda')| \leq C|\lambda - \lambda'|$$

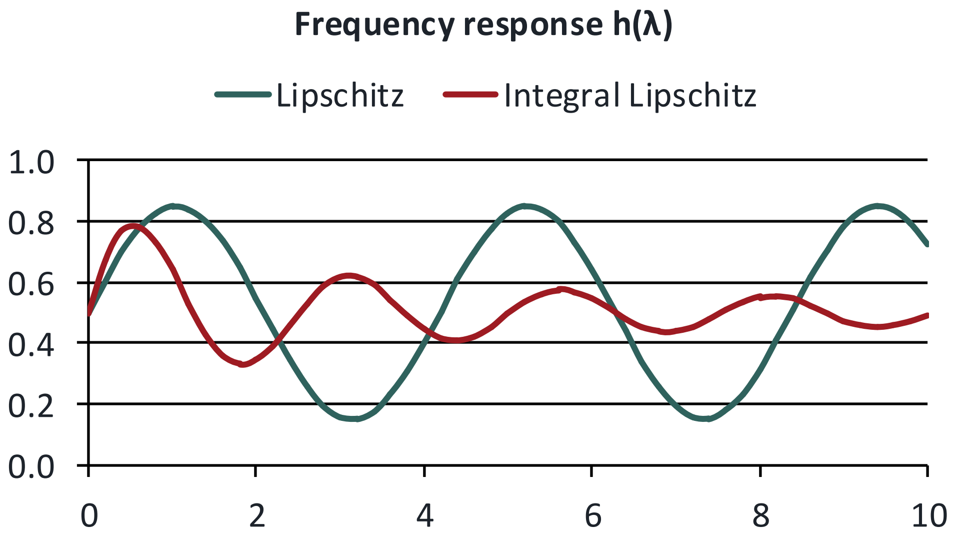
$$|h'(\lambda)| \leq C$$

INTEGRAL-LIPSCHITZ · RELATIVE SCALE

Controls response change relative to frequency changes.
The response becomes flatter at large $|\lambda|$.

$$|h(\lambda) - h(\lambda')| \leq \frac{C|\lambda - \lambda'|}{|\lambda + \lambda'|/2}$$

$$|\lambda h'(\lambda)| \leq C$$



Choose smoothness in the geometry induced by the perturbation.

Additive errors match ordinary Lipschitz. Relative errors match integral-Lipschitz.

Integral-Lipschitz filters stabilize relative errors

- Shift matrix related with $\boldsymbol{P}^T\hat{\boldsymbol{S}}\boldsymbol{P} = \boldsymbol{S} + \boldsymbol{E}\boldsymbol{S} + \boldsymbol{S}\boldsymbol{E}$
- Error matrix $\|\boldsymbol{E}\| \leq \varepsilon$
- Lipschitz frequency response

$$|\lambda h'(\lambda)| \leq C.$$

$$\boxed{\left\|h(\hat{\boldsymbol{S}}) - h(\boldsymbol{S})\right\|_{\mathcal{P}} \leq 2C(1 + \delta\sqrt{n})\varepsilon + O(\varepsilon^2)}$$

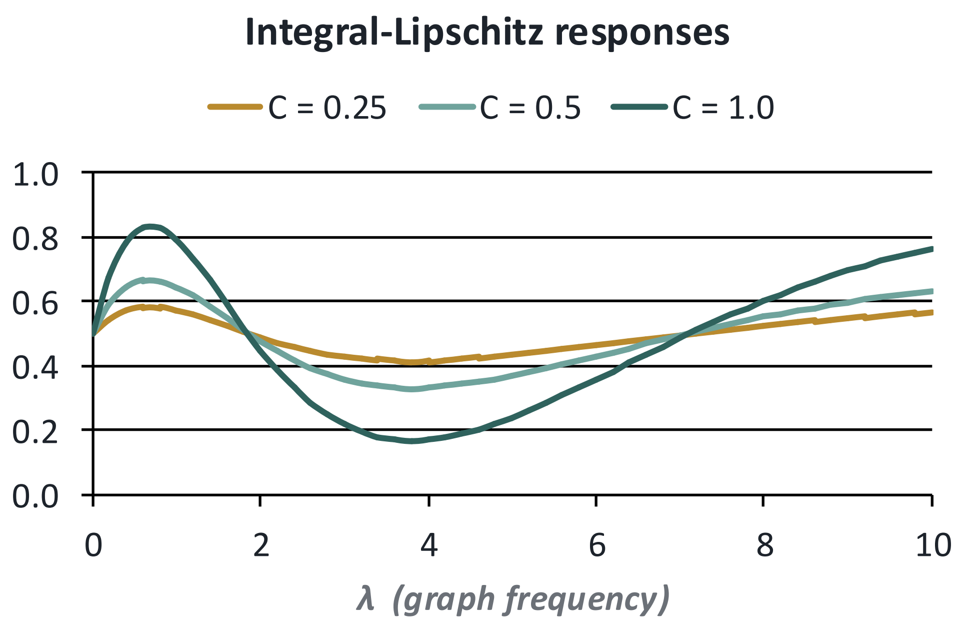
WHY THE FACTOR TWO?

relative error $= \boldsymbol{E}\boldsymbol{S} + \boldsymbol{S}\boldsymbol{E}$

FIRST-ORDER STRUCTURE

filter error = response slope × alignment × graph error

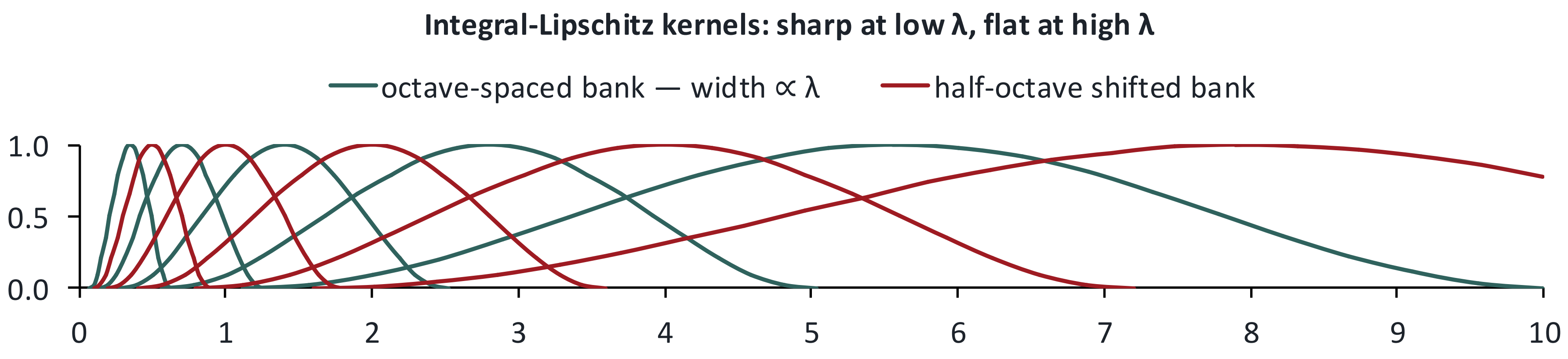
Apart from the factor two, the first-order interpretation matches the additive theorem.



Structure-aware errors still produce a linear first-order guarantee.

Stability tradeoff discriminability

- Bound is universal for all graphs with nodes n
  - Continuity of filter's frequency response – integral Lipschitz filters
  - Properties of perturbation $\boldsymbol{E}$
- Discriminability and stability tradeoff – Larger C, better discriminability, worse stability
- The theorem factors learned-filter sensitivity from deployed-graph geometry



Continuity says the map behaves locally. Stability quantifies how well it transfers.

Linear filters trade stability for discrimination

STABILITY

Small C. Smooth spectral transitions.

Smooth response at nearby frequencies.
Reliable transfer across perturbed graphs.
Limited sensitivity to spectral detail.

DISCRIMINATION

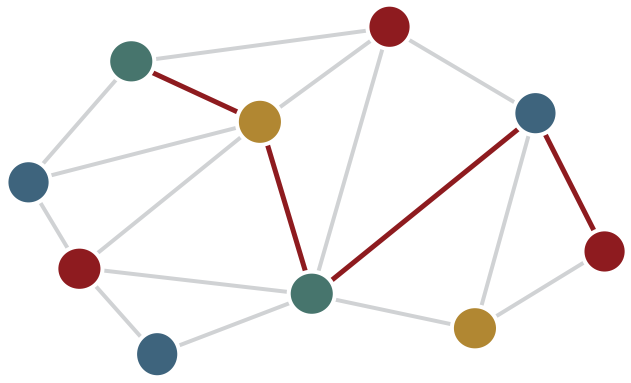
Large C. Sharp spectral transitions.

Resolve nearby eigenvalues.
Capture task-specific high-frequency structure.
Larger perturbation sensitivity.

How to resolve this tradeoff?

Use nonlinear depth to recover expressivity without an unstable linear response.

Stable filters become expressive when cascaded.



01 Add a pointwise nonlinearity after each filter

02 Carry multiple feature channels at every node

03 Normalize filters and activations

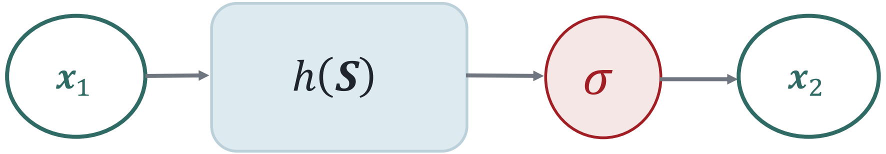
04 Propagate layer-wise error through depth

The cascade inherits stability and gains nonlinear discrimination.

A graph perceptron is a filter followed by σ

First apply a polynomial graph filter, then activate every node independently:

$$\boldsymbol{z} = h(\boldsymbol{S})\boldsymbol{x}_1 = \sum_k h_k \boldsymbol{S}^k \boldsymbol{x}_1 \qquad \boldsymbol{x}_2 = \sigma(\boldsymbol{z})$$



- $h(\boldsymbol{S})$ aggregates information -- coefficients determine the spectral response and the local mixing rule.
- σ is pointwise - ReLU, tanh, or sigmoid--adds nonlinear expressivity without adding analysis complexity if it's normalized Lipschitz

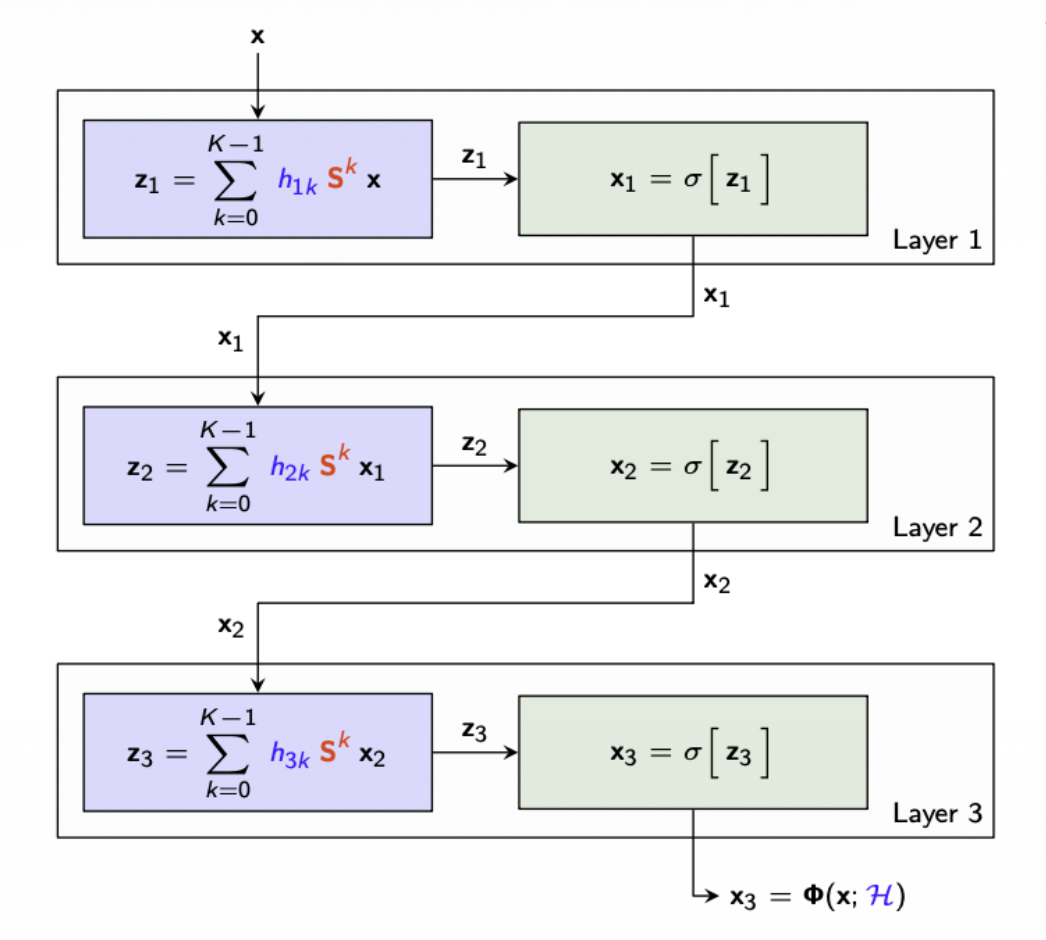
Filter locally, activate nodewise.

Filter, activate, repeat: the GNN cascade

$$\boldsymbol{x}_\ell = \sigma(h_\ell(\boldsymbol{S})\boldsymbol{x}_{\ell-1}), \qquad \boldsymbol{x}_0 = \boldsymbol{x},$$

$$\boldsymbol{x}_L = \boldsymbol{\Phi}(\boldsymbol{S}, \boldsymbol{x}; \boldsymbol{H})$$

Every stage reuses the graph geometry but learns its own filter and feature representation.



Depth expands the receptive field and composes nonlinear representations.

Normalization keeps the cascade nonexpansive

FILTER NONAMPLIFYING

$\|h_\ell(\boldsymbol{S})\| \leq 1$

A layer cannot magnify an incoming representation before the graph perturbation is considered.

ACTIVATION NORMALIZATION

$\|\sigma(a) - \sigma(b)\| \leq \|a - b\|$

ReLU and tanh are 1-Lipschitz
σ is applied independently at every node.

Each layer is non-expansive except for the mismatch between $h_l(\hat{\mathrm{S}})$ and $h_l(\boldsymbol{S})$.

Normalization converts a local filter bound into a depth-wise recurrence.

Each layer adds graph error to inherited error

Non-expansiveness removes the activation from the upper bound.

Proof technique -- Add and subtract $h_l(\boldsymbol{S})\hat{\boldsymbol{x}}_{l-1}$

Output difference: $\boldsymbol{\Delta}_\ell = \|\hat{\boldsymbol{x}}_\ell - \boldsymbol{x}_\ell\| \le \left\|h_l(\hat{\boldsymbol{S}})\hat{\boldsymbol{x}}_{l-1} - h_l(\boldsymbol{S})\boldsymbol{x}_{l-1}\right\|$

TRIANGLE INEQUALITY SPLITS THE TWO ERROR SOURCES

$$\boldsymbol{\Delta}_\ell \le \left\|\underbrace{\left[h_\ell(\hat{\boldsymbol{S}}) - h_\ell(\boldsymbol{S})\right]\hat{\boldsymbol{x}}_{\ell-1}}_{\text{Filter stability bound}}\right\| + \left\|\underbrace{h_\ell(\boldsymbol{S})}_{\text{Non-amplifying assumption}}\underbrace{(\hat{\boldsymbol{x}}_{\ell-1} - \boldsymbol{x}_{\ell-1})}_{\text{Iterative term}}\right\| \quad \le c\varepsilon + \boldsymbol{\Delta}_{\boldsymbol{l-1}}$$

Filter stability bounds the first term by $c\varepsilon + O(\varepsilon^2)$. Unit operator norm bounds the second by $\boldsymbol{\Delta}_{l-1}$.

GNNs inherit the additive and relative filter bounds

Apply the same inequality at layers $\ell - 1, \ell - 2$, and so on.

AFTER L LAYERS $\boldsymbol{\Delta}_L \leq c\varepsilon + c\varepsilon + \cdots + c\varepsilon = Lc\varepsilon$

ADDITIVE PERTURBATIONS

$$c = C(1 + \delta\sqrt{n})$$

$$\left\|\Phi(\cdot;\hat{\boldsymbol{S}}) - \Phi(\cdot;\boldsymbol{S})\right\|_{\mathcal{P}} \leq C(1 + \delta\sqrt{n})L\varepsilon + O(\varepsilon^2)$$

RELATIVE PERTURBATIONS

$$c = 2C(1 + \delta\sqrt{n})$$

$$\left\|\Phi(\cdot;\hat{\boldsymbol{S}}) - \Phi(\cdot;\boldsymbol{S})\right\|_{\mathcal{P}} \leq 2C(1 + \delta\sqrt{n})L\varepsilon + O(\varepsilon^2)$$

Under the normalizations, errors do not multiply across layers. The worst-case first-order cost grows linearly, not exponentially, with depth.

Depth accumulates stable layer errors additively.
The architecture preserves first-order robustness to graph perturbations.

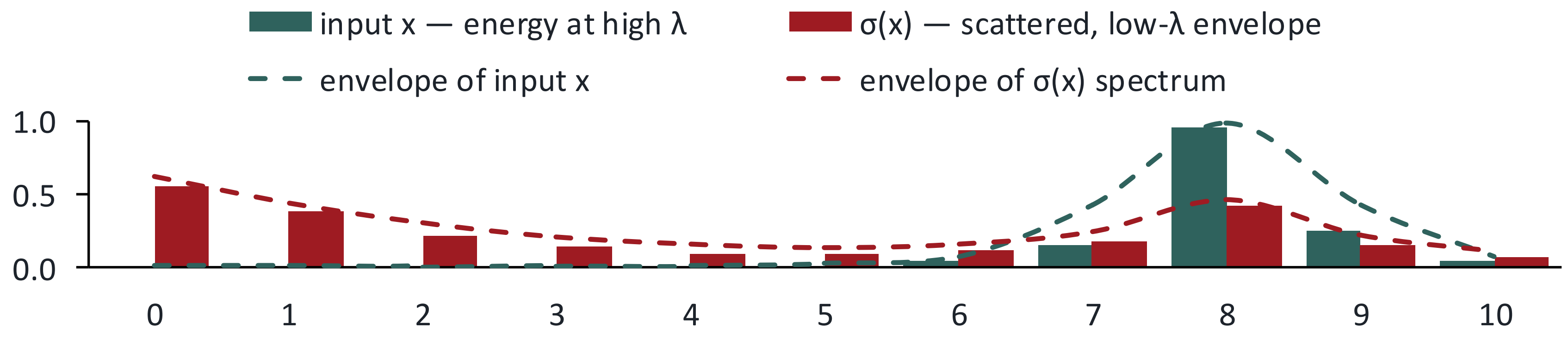
Pointwise Nonlinearity restores discriminability

Nonlinearity seems affect little in vertex domain, in spectral domain it is key

- The pointwise nonlinearity scatters spectral components across the spectrum.
- The next stable filter can extract the low frequency parts robustly.

Repeating this mechanism separates signals that one smooth linear filter cannot distinguish, thus improves the discriminability while maintain stability.

Spectral scattering by the pointwise nonlinearity σ



Nonlinearity moves discriminative information into frequency regions where stable filters can use it.

Checkpoint: derive the guarantee

01 **Before comparing two graph shifts, which optimization removes arbitrary node names?**

Think: align the graphs with a permutation matrix.

minimize over $\mathrm{P} \in \mathcal{P}$

02 **Which response condition matches the relative model $\boldsymbol{S} + \boldsymbol{E}\boldsymbol{S} + \boldsymbol{S}\boldsymbol{E}$?**

Think: multiplicative frequency changes.

integral-Lipschitz, $|\lambda h'(\lambda)| \leq C$

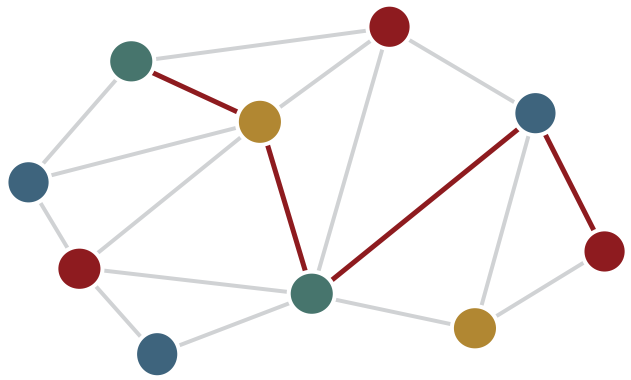
03 **Why does the normalized GNN bound grow as $L\varepsilon$ instead of exponentially?**

Think: nonexpansive layers and a telescoping recurrence.

each layer adds at most $c\varepsilon$ under unit norms

TAKEAWAYS

Stable layers.
Expressive cascade.



01 Align graph shifts before measuring perturbations

02 Lipschitz for additive · integral-Lipschitz for relative

03 Normalize filters and pointwise nonlinearities

04 Propagate layer errors to obtain the $O(L\varepsilon)$ bound

Nonlinear depth recovers discrimination without sacrificing first-order stability.