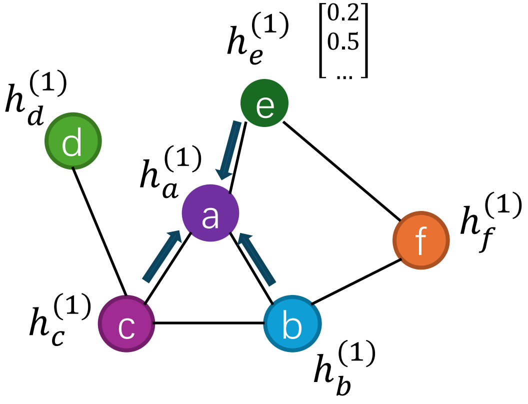
# On the Stability of Expressive Positional Encoding for Graphs

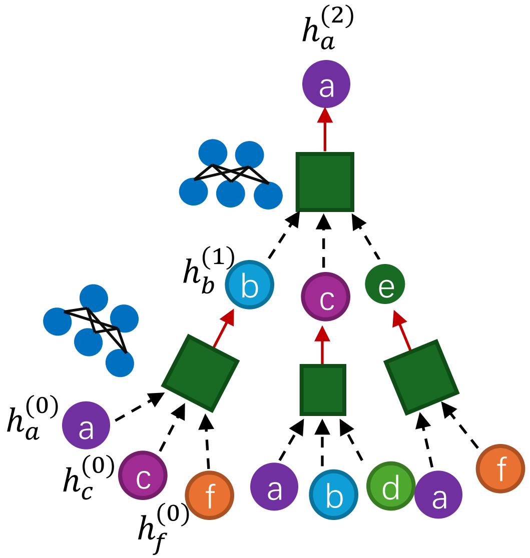
Yinan Huang[*,1], William Lu[*, 2], Joshua Robinson[3], Yu Yang[4], Muhan Zhang[5], Stefanie Jegelka[6], Pan Li[1]

[*]equal contribution, [1]Georgia Institute of Technology, [2]Purdue University, [3]Stanford University, [4]Tongji University, [5]Peking University, [6]MIT

# Graph Neural Networks

- Graph neural networks (GNNs) adopts message passing scheme:
  1. Aggregate features from neighbor into message
  2. Update self features based on message and previous self features



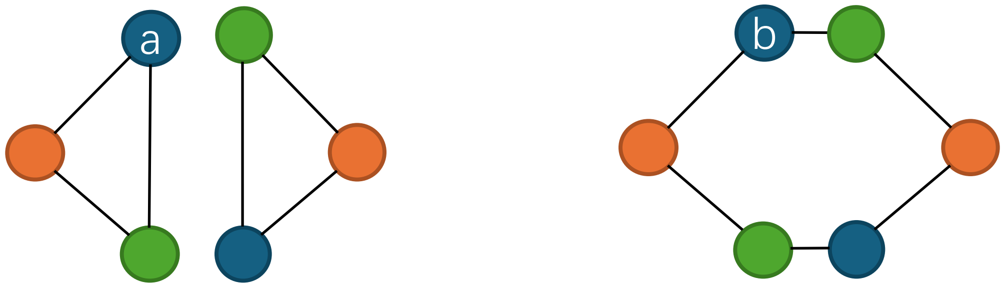

$$h_v^{(t+1)} = f_{update}\left(h_v^{(t)}, f_{agg}\left(\{h_u^{(t)} \middle| u \in N_v\}\right)\right),$$

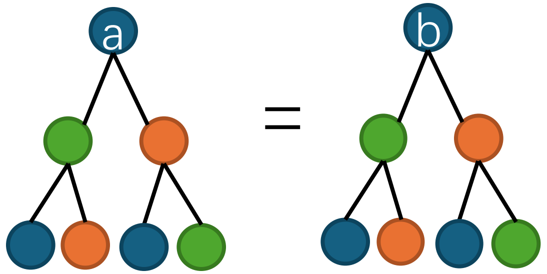
where $N_v$ denotes the set of the neighbors of node $v$.

# Graph Neural Networks

- GNNs suffer from limited expressive power:
  - GNNs cannot approximate certain functions over graphs

    e.g. count k-cycles, $k \geq 3$



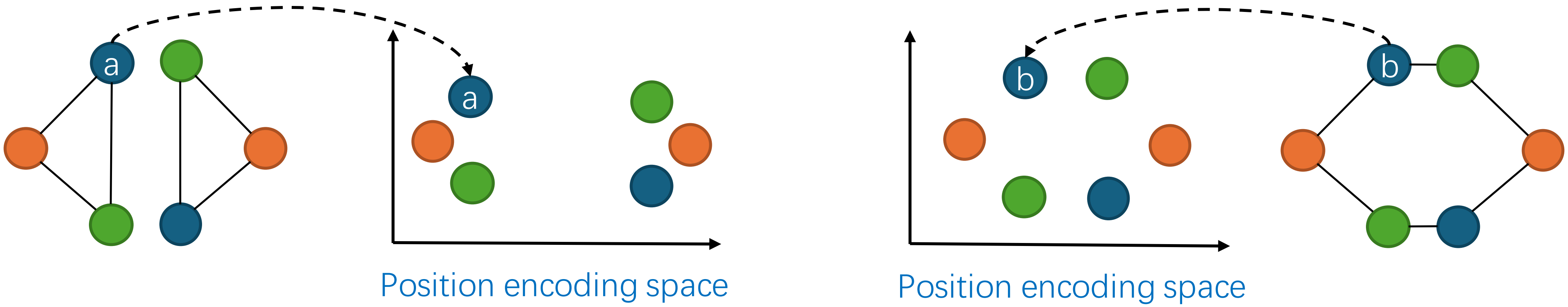

How powerful are graph neural networks? Xu et al., ICLR 2019

Weisfeiler and leman go neural: Higher-order graph neural networks, Morris et al., AAAI 2019

Can graph neural networks count substructures? Chen et al., NeurIPS 2020

# Positional Encodings for Graphs

- Positional encoding (PE)
  - Graph adjacency matrix $A \in \mathbb{R}^{n \times n}$, denote PE by
$$z(A) \in \mathbb{R}^{n \times p},$$
    where $[z(A)]_v$ is PE for node $v$
  - $[z(A)]_v$ characterizes the position of node $v$ in the graph
  - Helps GNN distinguish nodes and improve expressivity




**Key question: How to design and properly use positional encodings for graphs?**

# Laplacian Eigenmaps

- Graph (Normalized) Laplacian:

$$L = I - D^{-\frac{1}{2}} A D^{-\frac{1}{2}}$$

$A$: adjacency matrix, $D$: diagonal degree matrix

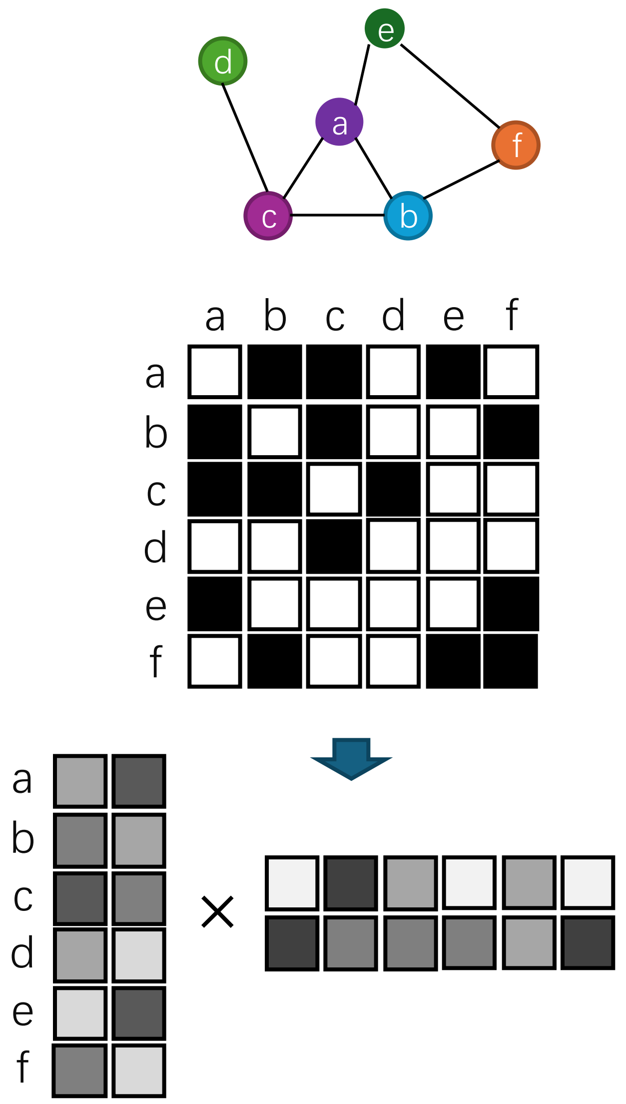
- Laplacian eigenmaps:

$$L = \mathrm{Vdiag}(\Lambda) V^T$$

where $\mathrm{V} = [v_1, v_2, \ldots, v_n]$,

$$\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n), \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n.$$

$V_{u,1:p}$ serves as the positional encoding for node $u$




Positional encoding



Laplacian Eigenmaps for D.R., Belkin & Niyogi, Neural Computation, 2003

# Motivation: the Ambiguity Issue of Laplacian Eigenmaps

- Basis ambiguity

The decomposition is not unique:

$$L = V\mathrm{diag}(\Lambda)V^T = (VQ)\mathrm{diag}(\Lambda)(VQ)^T$$

for any orthogonal and block-diagonal $\mathrm{Q} \in \bigoplus_i O(d_i)$, if the eigenvalues follow $\lambda_1 = \cdots = \lambda_{d_1} < \lambda_{d_1+1} = \cdots = \lambda_{d_1+d_2} < \cdots$, where $d_i's$ are eigenvalue multiplicities.

# Motivation: the Ambiguity Issue of Laplacian Eigenmaps

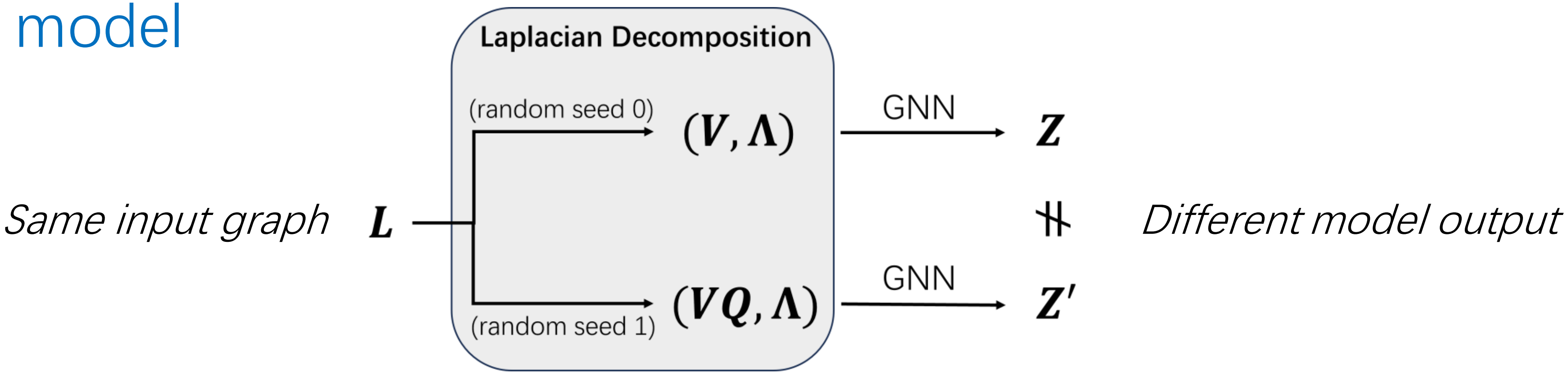
- Basis ambiguity

The decomposition is not unique:

$$L = V\mathrm{diag}(\Lambda)V^T = (VQ)\mathrm{diag}(\Lambda)(VQ)^T$$

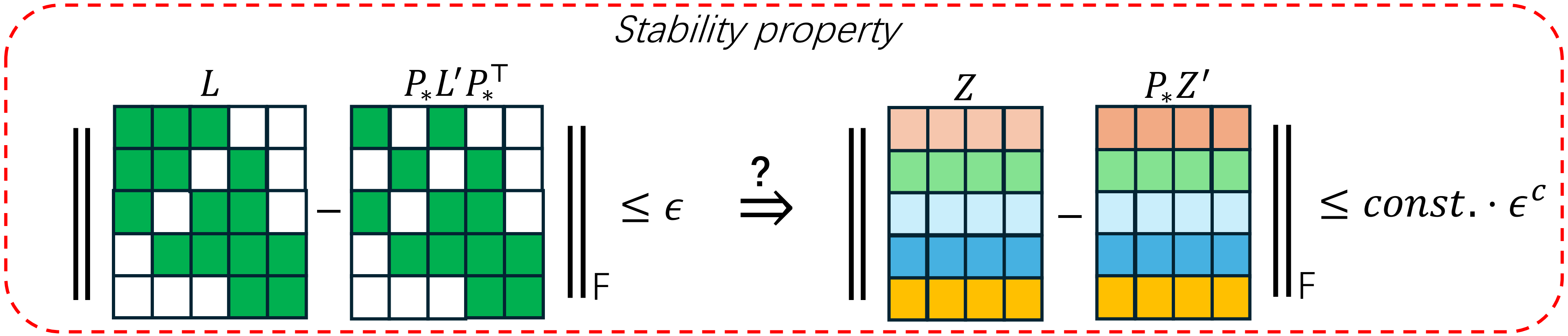
for any orthogonal and block-diagonal $Q \in \bigoplus_i O(d_i)$, if the eigenvalues follow $\lambda_1 = \cdots = \lambda_{d_1} < \lambda_{d_1+1} = \cdots = \lambda_{d_1+d_2} < \cdots$, where $d_i's$ are eigenvalue multiplicities.

- Such non-uniqueness -> the ambiguity issues of the model



# Motivation: the Stability Issue of Laplacian Eigenmaps

- Stability generalizes the idea "same input, same output", stating that "small perturbation to input, small change of output".
- Consider two closed Laplacian $L, L'$ and their PE $Z, Z'$




where $P_* = \text{argmin}_{P \in \Pi_n} ||L - PL'P^T||$ , $\Pi_n$ denotes the set of n-by-n permutation matrix

We generally denote $||\cdot||$ as L2 norm for vectors and F-norm for matrices

# Motivation: the Stability Issue of Laplacian Eigenmaps

- Stability generalizes the idea "same input, same output", stating that "small perturbation to input, small change of output".

**Definition (Stability)**
A positional encoding function $z(\cdot): \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times p}$ is called stable, if for some $c, C > 0$, for any $L$, $L'$, it satisfies

$$||z(L) - P_* z(L')|| \leq C\left||L - P_* L' P_*^T|\right|^c$$

where $P_* = \text{argmin}_{P\in\Pi_n}\left||L - PL'P^T|\right|$, $\Pi_n$ denotes the set of n-by-n permutation matrix

We generally denote $||\cdot||$ as L2 norm for vectors and F-norm for matrices

- Instead of the normal Lipschitz continuity as stability definition, we use Hölder continuity, a looser condition.

# Existing Ways to Use Laplacian Eigenmaps are Unstable

- randomly sign flipping [Dwivedi & Bresson, 2021; Kreuzer et al., 2021]
- Use sign invariant functions [Lim et al.]
- Use basis invariant functions [Lim et al.]

- Ambiguity gets addressed.
- Stability does not hold for all of them (see Appendix C of our paper for details).

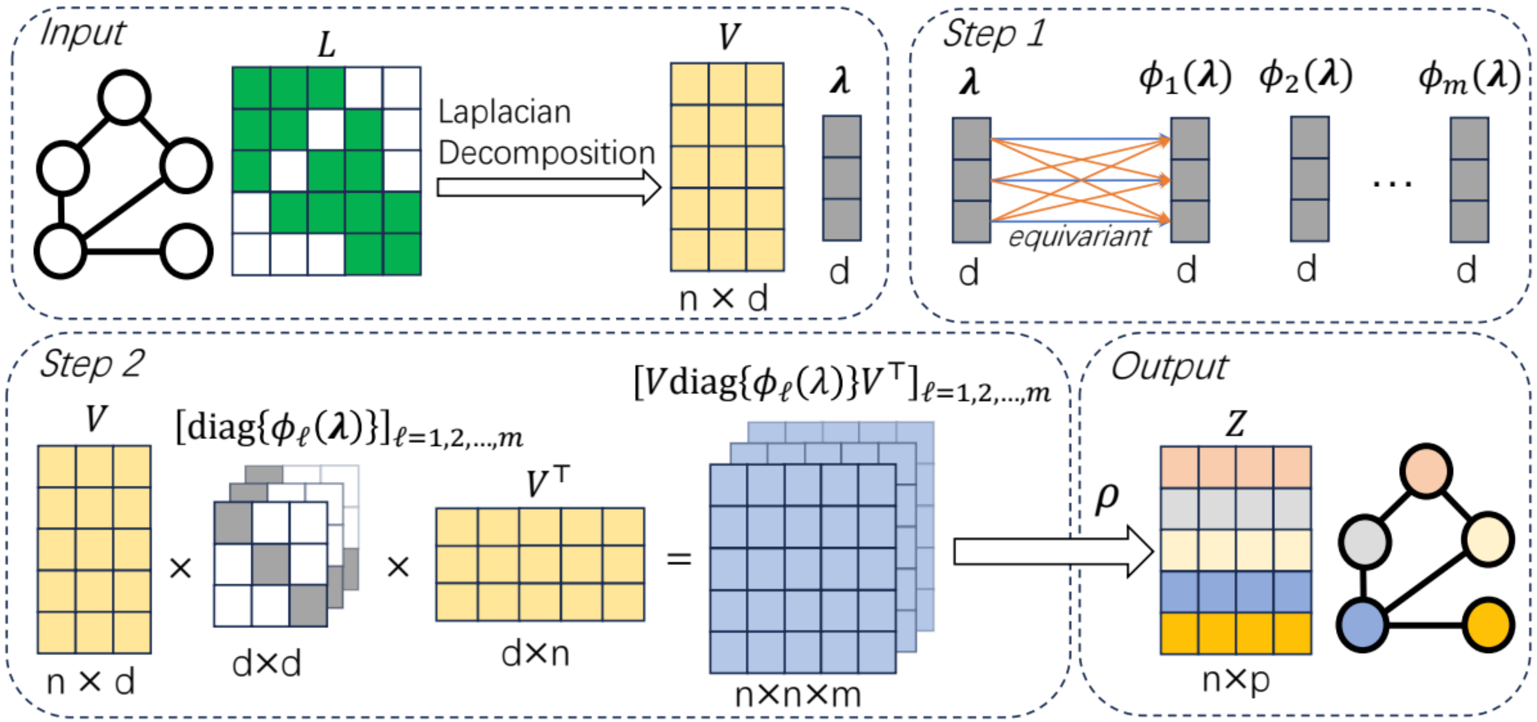
A generalization of transformer networks to graphs, Dwivedi & Bresson, 2021
Rethinking graph transformers with spectral attention, Kreuzer et al., NIPS 2021
Sign and Basis Invariant Networks for Spectral Graph Representation Learning, Lim et al., ICLR 2023

# Stable and Expressive Positional Encodings (SPE)

- Our choice

$$z(L) = z(V, \Lambda) = \rho(V\text{diag}(\phi_1(\Lambda))V^T, \dots, V\text{diag}(\phi_m(\Lambda))V^T)$$

Where $\phi_i: \mathbb{R}^d \to \mathbb{R}^d$ are permutation equivariant functions in $d$.
$\rho: \mathbb{R}^{n\times n\times m} \to \mathbb{R}^{n\times p}$ is another permutation equivariant function in n.



# Key results: stability and expressivity of SPE

SPE: $z(L) = z(V, \Lambda) = \rho(V\mathrm{diag}(\phi_1(\Lambda))V^T, \ldots, V\mathrm{diag}(\phi_m(\Lambda))V^T)$

Where $\phi_i: \mathbb{R}^d \to \mathbb{R}^d, \rho: \mathbb{R}^{n\times n\times l} \to \mathbb{R}^{n\times p}$ are permutation equivariant **Lipschitz continuous** functions.

- **Stability:** SPE is provably stable (Theorem 3.1)
- Out-of-distribution guarantee (Proposition 3.1)
- **Expressivity**: SPE can count at least 5-cycles with $\rho$ being 2-IGN (Proposition 3.4)

Invariant and equivariant graph networks, Maron et al., ICLR 2019

# Numerical Evaluation I

- Prediction performance over ZINC and Alchemy (molecular property prediction)

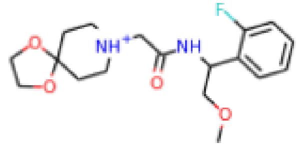


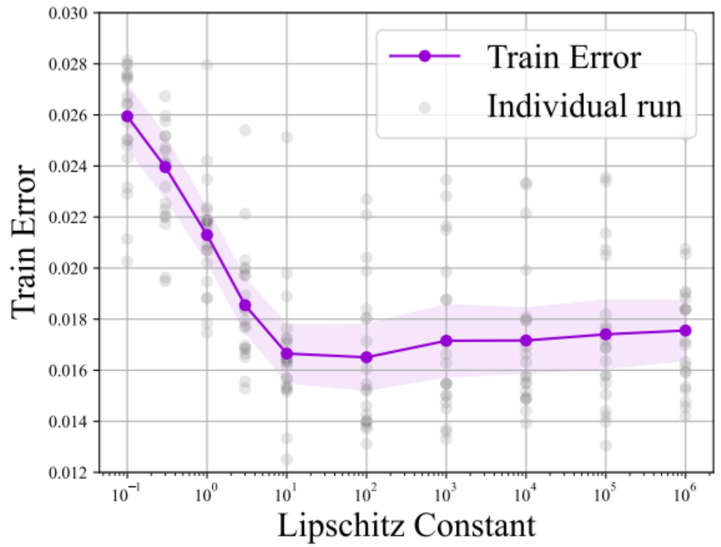

Table 1: Test MAE results (4 random seeds) on ZINC and Alchemy.

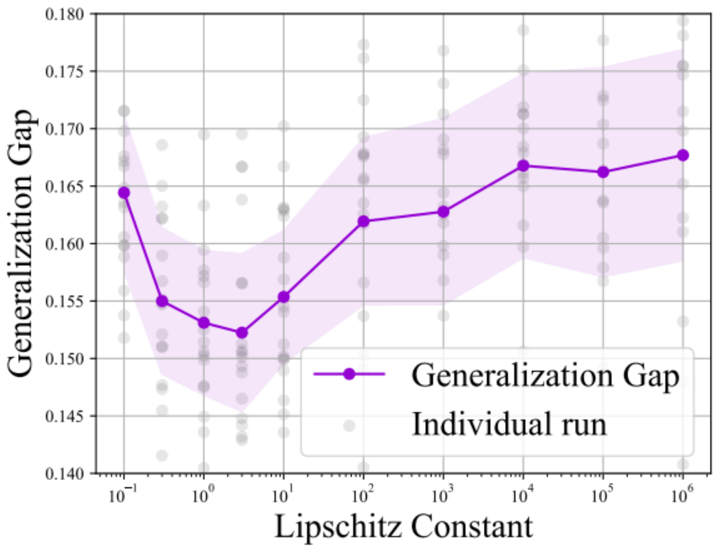
| Dataset | PE method | #PEs | #param | Test MAE |
|---|---|---|---|---|
| ZINC | No PE | N/A | 575k | $0.1772_{\pm 0.0040}$ |
| | PEG | 8 | 512k | $0.1444_{\pm 0.0076}$ |
| | PEG | Full | 512k | $0.1878_{\pm 0.0127}$ |
| | SignNet | 8 | 631k | $0.1034_{\pm 0.0056}$ |
| | SignNet | Full | 662k | $0.0853_{\pm 0.0026}$ |
| | BasisNet | 8 | 442k | $0.1554_{\pm 0.0068}$ |
| | BasisNet | Full | 513k | $0.1555_{\pm 0.0124}$ |
| | SPE | 8 | 635k | $0.0736_{\pm 0.0007}$ |
| | SPE | Full | 650k | $\mathbf{0.0693_{\pm 0.0040}}$ |
| Alchemy | No PE | N/A | 1387k | $0.112_{\pm 0.001}$ |
| | PEG | 8 | 1388k | $0.114_{\pm 0.001}$ |
| | SignNet | Full | 1668k | $0.113_{\pm 0.002}$ |
| | BasisNet | Full | 1469k | $0.110_{\pm 0.001}$ |
| | SPE | Full | 1785k | $\mathbf{0.108_{\pm 0.001}}$ |

# Numerical Evaluation II

- Idea: control the stability by regulating the complexity (Lipschitz constant) of PE functions. What is the trade-off between stability and:
  - ➢ the model generalization gap (test loss - training loss)?
  - ➢ the expressive power (final training loss after convergence)

- Setup: Use ZINC
  - ➢ study the training error and the generalization gap to the test set.



# Numerical Evaluation III

- Evaluation over DrugOOD: Molecular graphs with three different types (Assay, Scaffold, Size) of distribution shifts between the training and test datasets

Table 2: AUROC results (5 random seeds) on DrugOOD.

| Domain | PE Method | ID-Val (AUC) | ID-Test (AUC) | OOD-Val (AUC) | OOD-Test (AUC) |
|---|---|---|---|---|---|
| Assay | No PE | $92.92_{\pm 0.14}$ | $92.89_{\pm 0.14}$ | $71.02_{\pm 0.79}$ | $71.68_{\pm 1.10}$ |
| | PEG | $92.51_{\pm 0.17}$ | $92.57_{\pm 0.22}$ | $70.86_{\pm 0.44}$ | $71.98_{\pm 0.65}$ |
| | SignNet | $92.26_{\pm 0.21}$ | $92.43_{\pm 0.27}$ | $70.16_{\pm 0.56}$ | $\mathbf{72.27_{\pm 0.97}}$ |
| | BasisNet | $88.96_{\pm 1.35}$ | $89.42_{\pm 1.18}$ | $71.19_{\pm 0.72}$ | $71.66_{\pm 0.05}$ |
| | SPE | $92.84_{\pm 0.20}$ | $92.94_{\pm 0.15}$ | $71.26_{\pm 0.62}$ | $\mathbf{72.53_{\pm 0.66}}$ |
| Scaffold | No PE | $96.56_{\pm 0.10}$ | $87.95_{\pm 0.20}$ | $79.07_{\pm 0.97}$ | $68.00_{\pm 0.60}$ |
| | PEG | $95.65_{\pm 0.29}$ | $86.20_{\pm 0.14}$ | $79.17_{\pm 0.29}$ | $\mathbf{69.15_{\pm 0.75}}$ |
| | SignNet | $95.48_{\pm 0.34}$ | $86.73_{\pm 0.56}$ | $77.81_{\pm 0.70}$ | $66.43_{\pm 1.06}$ |
| | BasisNet | $85.80_{\pm 3.75}$ | $78.44_{\pm 2.45}$ | $73.36_{\pm 1.44}$ | $66.32_{\pm 5.68}$ |
| | SPE | $96.32_{\pm 0.28}$ | $88.12_{\pm 0.41}$ | $80.03_{\pm 0.58}$ | $\mathbf{69.64_{\pm 0.49}}$ |
| Size | No PE | $93.78_{\pm 0.12}$ | $93.60_{\pm 0.27}$ | $82.76_{\pm 0.04}$ | $\mathbf{66.04_{\pm 0.70}}$ |
| | PEG | $92.46_{\pm 0.35}$ | $92.67_{\pm 0.23}$ | $82.12_{\pm 0.49}$ | $\mathbf{66.01_{\pm 0.10}}$ |
| | SignNet | $93.30_{\pm 0.43}$ | $93.20_{\pm 0.39}$ | $80.67_{\pm 0.50}$ | $64.03_{\pm 0.70}$ |
| | BasisNet | $86.04_{\pm 4.01}$ | $85.51_{\pm 4.04}$ | $75.97_{\pm 1.71}$ | $60.79_{\pm 3.19}$ |
| | SPE | $92.46_{\pm 0.35}$ | $92.67_{\pm 0.23}$ | $82.12_{\pm 0.49}$ | $\mathbf{66.02_{\pm 1.00}}$ |

# Thank you!

Paper: On the Stability of Expressive Positional Encodings for Graphs. Huang et al., ICLR 2024

Paper link: https://openreview.net/pdf?id=xAqcJ9XoTf

Code: https://github.com/Graph-COM/SPE