

SCALE CONTINUITY IN GRAPH LEARNING: GOING BEYOND SPECTRAL METHODS

Anonymous authors

Paper under double-blind review

ABSTRACT

Standard message passing graph neural networks (MPNNs) assign vastly different latent embeddings to graphs that describe the same underlying object at different resolution scales. As a result, message passing graph neural networks generically fail to generalize across resolution scales. Previous work showed that this issue can be overcome if instead of MPNNs certain special types of spectral graph neural networks are used. In this tiny-paper, we show that spectral methods are not necessary for achieving scale continuity. We demonstrate that if message passing networks are suitably modified, they can be rendered scale-continuous as well. By identifying the structural requirements for continuity, we derive a class of message passing architectures that provably preserve embeddings across resolution scales. Empirically, we show that these models match the cross-scale generalization performance of spectral approaches while retaining the flexibility and scalability of local message passing methods.

1 INTRODUCTION

A key theoretical pillar underpinning the widespread adoption of graph neural networks is the commonly held belief that graph neural networks (GNNs) define *stable* and *continuous* mappings: Small changes in the input graph are expected to induce correspondingly small changes in learned representations (Ruiz et al., 2020; Maskey et al., 2021; Roddenberry et al., 2022; Le & Jegelka, 2023). Recently this view has been challenged in the setting of stability under variations in graph resolution scale: In Koke et al. (2024; 2025) it was established that common graph neural networks based on message passing layers are not continuous under graph coarsening operations. Latent embeddings F, \underline{F} generated by MPNNs for a graph G and its coarsified version \underline{G} are vastly different. The root cause behind this discontinuity across scales was identified to lie in a fundamental limitation of standard message passing neural networks. To overcome this limitation and build models that *are* able to generalize across scales, Koke et al. (2024; 2025) took a spectral approach and showed that if the layer-wise update rule of a node-feature matrix X inside a *spectral* GNN is implemented as

$$X \mapsto \sum_{k=1}^K \psi_k(L) X W_k, \quad (1)$$

the resulting network will be scale continuous; assigning similar latent embeddings ($\underline{F} \approx F$) to graphs G, \underline{G} describing the same object at different resolutions. Here L is the natural (“un-normalized”) graph Laplacian (c.f. also Appendix A.1), and the functions ψ_k applied to L have a specific form (arising as Laplace transforms; c.f. Section 3 below).

These works also showed (c.f. e.g. Koke et al. (2025, Appendix B.1)) that *standard* MPNNs *never* are scale continuous, generating vastly different latent embeddings F, \underline{F} for similar graphs G, \underline{G} describing the same object at different resolutions. This might lead one to assume that only spectral graph networks (with layer-wise update as defined in (1)) can be scale continuous. Here we however show that this is not the case: **Also MPNNs may be rendered scale continuous**, if message passing is implemented appropriately. Concretely, we show that MPNNs with layer-wise update

$$h_u^{(\ell+1)} = \sum_{v \in G} m_{uv}, \quad m_{uv} = [\psi(L)]_{vu} \cdot \phi(h_u^{(\ell)}, h_v^{(\ell)}), \quad h_0 = \psi(L)X, \quad (2)$$

are in fact scale continuous. Here $h_u^{(\ell)}$ is the latent representation of node u in layer ℓ , $\phi(\cdot, \cdot)$ is a learnable message function and $\psi(L)$ is a specific (type of; c.f. Section 3) function applied to L .

2 RECAP: DISCONTINUITY OF STANDARD MPNNs ACROSS SCALES

To demonstrate that conventional GNN architectures struggle to coherently handle data across different resolutions, we adopt the experimental setting of (Koke et al., 2024; 2025) and make use of the QM7 benchmark dataset (Rupp et al., 2012). This dataset consists of small organic molecules composed of hydrogen and heavier elements, with the task of predicting their atomization energies. Each molecule is encoded as a weighted graph whose adjacency matrix is defined by $A_{ij} = Z_i Z_j |\vec{x}_i - \vec{x}_j|^{-1}$, representing the Coulomb interaction energy between atoms i and j .

From a physical perspective, describing a molecule at the level of interacting atoms corresponds to a specific choice of resolution scale, where interactions of individual protons and neutrons inside individual atoms are discarded. To test the scale continuity of GNNs we also consider a version of QM7 where we further lower the resolution scale: Here we aggregate each heavy atomic core together with its surrounding (single-proton) hydrogen atoms into super-nodes.

To showcase the failure of GNNs to consistently incorporate multiple scales, we confront models during inference with a version of QM7 on a scale different from the one they were trained on. As summarized in Table 1, mean absolute errors rise sharply when models are transferred from a same-resolution- to a cross-resolution setting. No architectures—including pooling- or multiscale variants (SAG-M through PushNet)—are able to reliably process multiple scales. This can be traced back to the latent embeddings $\{F\}$ and $\{\underline{F}\}$ that are being generated for original- $\{G\}$ and coarsified graphs $\{\underline{G}\}$: For models of Table 1 on average $10 \lesssim \|F - \underline{F}\| \lesssim 10^3$ (c.f. also Table 2 below): Latent representations of graphs describing the same object at different resolutions differ significantly.

As argued in (Koke et al., 2024; 2025) the underlying reason for this difference in latent embeddings is that standard GNNs are not continuous. To see this, we interpolate between fine and coarse resolution: Original graphs $\{G\}$ of QM7 are modified ($\{G_\omega\}$) by moving hydrogen atoms towards their corresponding heavy atom by a factor of $\omega \geq 1$ (i.e. $\text{dist}_{\text{new}} = \text{dist}_{\text{equilib.}}/\omega$). For $\omega \rightarrow \infty$, they arrive at the respective heavy atom ($\{\underline{G}\}$). In Fig. 2 we compare the latent distance between the coarse embeddings \underline{F} and the embeddings for the intermediate graph F_ω . Embeddings F_ω do *not* converge to the coarse embeddings \underline{F} . Since the convergence of graph-sequence G_ω to the limit graph \underline{G} is not turned into a convergence of latent embeddings $F_\omega \rightarrow \underline{F}$ we conclude: GNNs are not continuous in this setting. This discontinuity explains why GNNs can map similar graphs (describing the same object at different resolutions) to different latent representations.

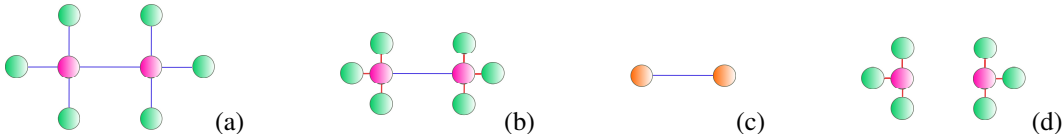


Figure 1: (a) G of QM7 (b) Modified G_ω (c) Coarsified \underline{G} (d) Effective propagation in GCN

Table 1: QM7 regression. Mean Absolute Error

Resolution	MAE (\downarrow) on QM7 [kcal/mol]			
	Fine		Coarse	
Training:	Fine		Coarse	
Inference:	Coarse	Fine	Coarse	Fine
GCN	136.7 \pm 6.6	(63.6 \pm 1.3)	(63.6 \pm 1.3)	138.1 \pm 2.4
GATv2	423.5 \pm 337.1	(67.4 \pm 8.2)	(59.7 \pm 2.7)	257.4 \pm 139.1
ChebNet	447.8 \pm 6.0	(66.7 \pm 1.4)	(71.5 \pm 2.1)	158.7 \pm 57.4
GIN	658.4 \pm 85.8	(17. \pm 2.8)	(38.3 \pm 21.6)	1835.4 \pm 925.8
SAG	589.7 \pm 44.9	(68.2 \pm 2.6)	(107.9 \pm 1.1)	283.6 \pm 39.3
SAG-M	194.4 \pm 29.9	(66.6 \pm 1.9)	(77.8 \pm 6.0)	219.5 \pm 11.7
UFGNet	131.5 \pm 6.1	(62.4 \pm 0.7)	(69.4 \pm 0.7)	148.1 \pm 6.3
Lanczos	938.4 \pm 2.5	(9.9 \pm 2.5)	(88.2 \pm 2.7)	658.6 \pm 199.2

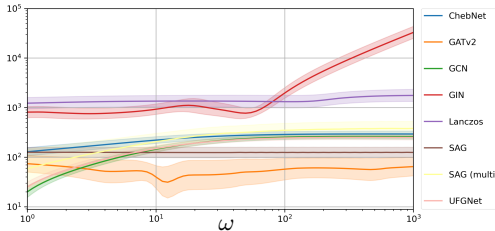


Figure 2: Latent distance $\|F_\omega - \underline{F}\|$.

To understand this discontinuity, one may exemplarily investigate GCN (Kipf & Welling, 2017). There the layer-wise update acts as $X \mapsto \hat{A}XW$, with the feature matrix $X \in \mathbb{R}^{N \times F}$ (N nodes; latent dimension F), the weight matrix $W \in \mathbb{R}^{F \times F}$ and the *renormalized* adjacency matrix $\hat{A} \in \mathbb{R}^{N \times N}$. As hydrogen atoms move closer to heavy atoms, the entries $\hat{A}_{\text{heavy,heavy}}$ in $\hat{A}_{ij} \sim A_{ij}/\sqrt{d_i d_j}$ tend to zero (degrees d_{heavy} tend to infinity). This disrupts information-flow over the graph (Fig 1 (d)).

3 GLOBAL LAPLACE PROPAGATION IN GNNs

To avoid disconnected limit propagation as in Fig. 1 (d), GCN’s propagation scheme is then modified:

3.1 SPECTRAL NETWORKS

To connect the information flows over G_ω and \underline{G} one may note that features in G_ω should equalize faster between nodes connected by large edge weights. When such a large weight tends to infinity, features between strongly connected nodes are then equalized immediately, so that entire strongly connected clusters exactly behave as the single nodes in \underline{G} . This is exactly the behavior that heat dissipating over a graph exhibits. Hence we may make use of the the heat diffusion equation $dX(t)/dt = -L \cdot X(t)$ (with Graph Laplacian L and time t) and the structure $X(t) = e^{-Lt} \cdot X(0)$ of its solutions, when designing scale continuous graph networks: Let $\hat{\psi}$ be a bounded (generalized) function defined on $[0, \infty)$. A **Global Laplacian Propagation Matrix** $\psi(L)$ is any matrix arising as $\psi(L) := \int_0^\infty e^{-tL} \hat{\psi}(t) dt$. Thus $\psi(L)$ represent a weighted sum of diffusion flows that have progressed to various times. Specifically, choosing the Dirac distribution $\hat{\psi}_{\delta_{t_k}}(t) := \delta(t - t_k)$ as the weighting function $\hat{\psi}_k$, we obtain **exponential** matrices $\psi_k(L) = \int_0^\infty \delta(t - t_k) e^{-tL} dt = e^{-t_k L}$ and $\hat{\psi}_k := (-t)^{k-1} e^{-\lambda t}$ to get powers of **resolvents** $\psi_k(L) = [(zId + L)^{-1}]^k$.

In (Koke et al., 2024; 2025) it was then proved that if the update rule $X \mapsto \hat{A}XW$ of GCN is replaced with the update $X \mapsto \psi(L)XW$ (with $\psi(L)$ a Laplacian propagation matrix), the resulting networks are scale continuous (e.g. $\|F_\omega - F\| \rightarrow 0$ in the setting of Fig. 2). Furthermore, it was noted that GCN may be thought of as the simplest spectral network, which allowed to extend scale continuity results to more general spectral networks, with layer-wise update rule $X \mapsto \sum_{k=1}^K \psi_k(L)XW_k$.

3.2 MESSAGE PASSING NETWORKS:

Here we note that GCN is not only the simplest spectral GNN, but also the simplest *message-passing* based GNN. Hence we aim to extend previous results on how to turn spectral GNNs scale continuous also to the realm of message passing networks.

To determine which modifications exactly are needed in order to turn *message passing* based networks scale continuous, we gain intuition from the the binary edge setting (adjacency entries $A_{uv} \in \{0, 1\}$): For networks with sum aggregation, the layer-wise update of a MPNN may then be written as $h_u^{(\ell+1)} = \sum_{v \in G} m_{uv}$, with message function $m_{uv} = A_{uv} \cdot \phi(h_v^{(\ell)}, h_u^{(\ell)})$. As we saw in Section 2, such adjacency-focused propagation schemes result in networks that are not continuous across scales. We hence make the replacement $A \mapsto \psi(L)$ in our propagation scheme:

$$h_u^{(\ell+1)} = \sum_{v \in G} m_{vu}, \quad m_{uv} = [\psi(L)]_{uv} \cdot \phi(h_u^{(\ell)}, h_v^{(\ell)}), \quad h_0 = \Psi(L)X, \quad (3)$$

For the choice $\phi(h_u^{(\ell)}, h_v^{(\ell)}) = W \cdot h_v^{(\ell)}$ this reduces to the modified GCN version of Section 3.1.

Scalability: At first glance, propagation along Laplace-transform matrices $\psi(L)$ may appear difficult to scale, since matrices such as $\psi(L) = e^{-tL}$ are typically dense. A naïve implementation would thus require sending N^2 messages over an N -node graph. This does however not pose a practical bottleneck: most entries are vanishingly small (Bauer et al., 2017). Hence, propagation remains effectively sparse, and negligible edges can be safely discarded in practice.

Theoretical Guarentees: Modifying message-passing based graph neural networks exactly as outlined in (3) then indeed yields scale continuous networks, as we show next. Our main Theorem (proved in Appendix C.2) establishes scale-continuity of the modified MPNNs:

Theorem 3.1. *In the setting of Figure 2 let F_ω and \underline{F} be the latent embeddings generated for graphs G_ω and \underline{G} by a modified message passing network. For such latent embeddings we have $\|F_\omega - \underline{F}\| \rightarrow 0$ as ω increases.*

Hence the modified MPNNs of (3) are indeed continuous in the setting of Section 2 and thus assign similar latent embeddings $F \approx \underline{F}$ to graphs G, \underline{G} describing the same object at different resolutions.

4 NUMERICAL INVESTIGATION OF SCALE CONTINUITY FOR MPNNs

Continuity across scales: Theorem 3.1 implies that MPNNs employing Laplace-transform based propagation schemes as in (3) are continuous as maps from the space of graphs into their latent spaces. To numerically verify this, we repeat the experiment of Figure 2 for models belonging to this category (using resolvent and exponential matrices; cf. Section 3 in a spectral and a spatial setting). As is evident from Fig. 3, latent embeddings generated by models employing Laplace transform propagation **including modified MPNN based methods** do indeed converge ($\|F_\omega - \underline{F}\| \rightarrow 0$). Hence for the message passing models constructed in Section 3.2, we have indeed verified the desired continuity across scales.

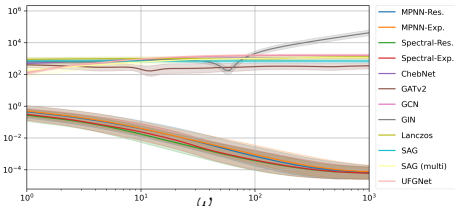


Figure 3: Latent distance $\|F_\omega - \underline{F}\|$. Latent embeddings generated by Laplace transform based GNNs converge; others do not.

Resulting Generalization Ability: In Section 2 we had identified lack of continuity as the obstruction to generalizing across scales. As verified above, MPNNs based on Laplace-transform propagation *are* continuous. Hence we expect them to map similar graphs to similar latent embeddings. To verify this, we here repeat the generalization experiment of Table 1 in Section 2.

Table 2: Embedding difference $\|F - \underline{F}\|$ across resolution scales averaged over 5 runs (mean \pm std). Lower is better (\downarrow).

Training \rightarrow Inference:	Fine \rightarrow Coarse	Coarse \rightarrow Fine
GCN	20.3 \pm 1.6	18.0 \pm 0.6
GATv2	76.8 \pm 52.4	42.5 \pm 24.3
ChebNet	123.0 \pm 3.8	158.5 \pm 57.40
GIN	881.2 \pm 268.7	2875.8 \pm 1527.7
SAG	113.5 \pm 10.3	43.3 \pm 5.4
SAG-M	75.0 \pm 15.1	61.7 \pm 4.6
UFGNet	24.5 \pm 0.3	26.8 \pm 2.1
Lanczos	1286.1 \pm 54.4	728.4 \pm 230.0
Spectral _{exp.}	0.3 \pm 0.1	0.3 \pm 0.1
Spectral _{res.}	0.3 \pm 0.1	0.3 \pm 0.3
MPNN _{exp.}	0.2 \pm 0.1	0.2 \pm 0.1
MPNN _{res.}	0.2 \pm 0.1	0.2 \pm 0.2

Table 3: QM7 regression. Mean Absolute Error (MAE \downarrow) in kcal/mol for training and inference at different resolutions scales.

Resolution	MAE (\downarrow) on QM7 [kcal/mol]			
	Fine		Coarse	
Training:	Fine		Coarse	
Inference:	Coarse	Fine	Coarse	Fine
GCN	136.7 \pm 6.6	(63.6 \pm 1.3)	(63.6 \pm 1.3)	138.1 \pm 2.4
GATv2	423.5 \pm 337.1	(67.4 \pm 8.2)	(59.7 \pm 2.7)	257.4 \pm 139.1
ChebNet	447.8 \pm 6.0	(66.7 \pm 1.4)	(71.5 \pm 2.1)	158.7 \pm 57.4
GIN	658.4 \pm 85.8	(17.2 \pm 2.8)	(38.3 \pm 21.6)	1835.4 \pm 925.8
SAG	589.7 \pm 44.9	(68.2 \pm 2.6)	(107.9 \pm 1.1)	283.6 \pm 39.3
SAG-M	194.4 \pm 29.9	(66.6 \pm 1.9)	(77.8 \pm 6.0)	219.5 \pm 11.7
UFGNet	131.5 \pm 6.1	(62.4 \pm 0.7)	(69.4 \pm 0.7)	148.1 \pm 6.3
Lanczos	938.4 \pm 2.5	(9.9 \pm 2.5)	(88.2 \pm 2.7)	658.6 \pm 199.2
Spectral _{exp.}	15.9 \pm 1.1	(15.9 \pm 1.1)	(16.0 \pm 1.5)	16.0 \pm 1.5
Spectral _{res.}	17.2 \pm 3.1	(17.2 \pm 3.1)	(15.8 \pm 1.4)	15.8 \pm 1.4
MPNN _{exp.}	17.4 \pm 3.4	(17.4 \pm 3.4)	(17.9 \pm 5.2)	17.9 \pm 5.2
MPNN _{res.}	18.0 \pm 4.3	(18.0 \pm 4.3)	(17.6 \pm 4.9)	17.6 \pm 4.9

Comparing with standard GNNs in Table 2 we see that in cross-resolution settings the difference $\|F - \underline{F}\|$ of latent embeddings generated by methods employing global Laplacian propagation schemes are *lower* than those of standard graph learning methods by factors of order 10^2 to 10^4 . Furthermore we note that indeed not only those of spectral methods, but **also embeddings generated MPNNs based on Laplace transform propagation are stable** ($\|F - \underline{F}\| \approx 0$).

The small-to-negligible variations in latent representations then translate into small-to-negligible variations in prediction performance: As we infer from Table 3 below, MAEs generated by Laplace-Transform based GNNs in the cross resolution setting are essentially the same as those corresponding to same-resolution settings. GNNs based on Laplace Transform propagation schemes are able to *generalize* across scales. This is indeed not limited to spectral methods: **Also modified MPNN-based methods are able to generalize across scales.**

5 DISCUSSION

We showed that scale continuity is not limited to spectral graph neural networks but can also be achieved with message passing models. By replacing adjacency-based propagation with Laplace-transform based information flow, MPNNs can be rendered provably continuous across graph resolutions. This unifies spectral and spatial approaches under a common principle: continuity depends on the propagation operator rather than on the network family. Empirically, the new resulting message passing based models match the cross-scale generalization ability of spectral methods.

216 REFERENCES

- 217 Frank Bauer, Bobo Hua, and Shing-Tung Yau. Sharp davies–gaffney–grigor’yan lemma on graphs.
218 *Mathematische Annalen*, 368(3–4):1429–1437, 2017. doi: 10.1007/s00208-017-1529-z.
- 219 L. C. Blum and J.-L. Reymond. 970 million druglike small molecules for virtual screening in the
220 chemical universe database GDB-13. *J. Am. Chem. Soc.*, 131:8732, 2009.
- 221 Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? In
222 *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event,*
223 *April 25-29, 2022*. OpenReview.net, 2022. URL [https://openreview.net/forum?id=](https://openreview.net/forum?id=F72ximsx7C1)
224 [F72ximsx7C1](https://openreview.net/forum?id=F72ximsx7C1).
- 225 Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning:
226 Grids, groups, graphs, geodesics, and gauges, 2021.
- 227 Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann Lecun. Spectral networks and locally
228 connected networks on graphs. In *International Conference on Learning Representations*
229 *(ICLR2014), CBLS, April 2014*, 2014.
- 230 F. R. K. Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.
- 231 Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on
232 graphs with fast localized spectral filtering. *Advances in neural information processing systems*,
233 29, 2016.
- 234 Matthias Fey, Jinu Sunil, Akihiro Nitta, Rishi Puri, Manan Shah, Blaž Stojanovič, Ramona Bendias,
235 Alexandria Barghi, Vid Kocijan, Zecheng Zhang, Xinwei He, Jan Eric Lenssen, and Jure Leskovec.
236 Pyg 2.0: Scalable learning on real world graphs. *arXiv preprint arXiv:2507.16991*, 2025. URL
237 <https://arxiv.org/abs/2507.16991>.
- 238 Johannes Gasteiger, Aleksandar Bojchevski, and Stephan Günnemann. Predict then propagate:
239 Graph neural networks meet personalized pagerank. In *7th International Conference on Learning*
240 *Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL
241 <https://openreview.net/forum?id=H1gL-2A9Ym>.
- 242 Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural
243 message passing for quantum chemistry. In *International conference on machine learning*, pp.
244 1263–1272. PMLR, 2017.
- 245 William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning
246 on large graphs. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M.
247 Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in*
248 *Neural Information Processing Systems 30: Annual Conference on Neural Information*
249 *Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 1024–
250 1034, 2017. URL [https://proceedings.neurips.cc/paper/2017/hash/](https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7e9-Abstract.html)
251 [5dd9db5e033da9c6fb5ba83c7a7e9-Abstract.html](https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7e9-Abstract.html).
- 252 Mingguo He, Zhewei Wei, Zengfeng Huang, and Hongteng Xu. Bernnet: Learning
253 arbitrary graph spectral filters via bernstein approximation. In Marc’Aurelio Ranzato,
254 Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.),
255 *Advances in Neural Information Processing Systems 34: Annual Conference on Neural*
256 *Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp.
257 14239–14251, 2021. URL [https://proceedings.neurips.cc/paper/2021/hash/](https://proceedings.neurips.cc/paper/2021/hash/76f1cfd7754a6e4fc3281bccb3d0902-Abstract.html)
258 [76f1cfd7754a6e4fc3281bccb3d0902-Abstract.html](https://proceedings.neurips.cc/paper/2021/hash/76f1cfd7754a6e4fc3281bccb3d0902-Abstract.html).
- 259 Mingguo He, Zhewei Wei, and Ji-Rong Wen. Convolutional neural networks
260 on graphs with chebyshev approximation, revisited. In *NeurIPS*, 2022.
261 URL [http://papers.nips.cc/paper_files/paper/2022/hash/](http://papers.nips.cc/paper_files/paper/2022/hash/2f9b3ee2bcea04b327c09d7e3145bd1e-Abstract-Conference.html)
262 [2f9b3ee2bcea04b327c09d7e3145bd1e-Abstract-Conference.html](http://papers.nips.cc/paper_files/paper/2022/hash/2f9b3ee2bcea04b327c09d7e3145bd1e-Abstract-Conference.html).
- 263 Matthias Hein, Jean-Yves Audibert, and Ulrike von Luxburg. Graph laplacians and their convergence
264 on random neighborhood graphs. *J. Mach. Learn. Res.*, 8:1325–1368, 2006. URL <https://api.semanticscholar.org/CorpusID:1355782>.

- 270 Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels:
271 First steps. *Social Networks*, 5(2):109–137, 1983. ISSN 0378-8733. doi: [https://doi.org/](https://doi.org/10.1016/0378-8733(83)90021-7)
272 10.1016/0378-8733(83)90021-7. URL [https://www.sciencedirect.com/science/](https://www.sciencedirect.com/science/article/pii/0378873383900217)
273 [article/pii/0378873383900217](https://www.sciencedirect.com/science/article/pii/0378873383900217).
274
- 275 Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012.
276
- 277 Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec.
278 Strategies for pre-training graph neural networks. In *International Conference on Learning*
279 *Representations (ICLR)*, 2020. doi: 10.48550/arXiv.1905.12265. URL [https://arxiv.org/](https://arxiv.org/abs/1905.12265)
280 [abs/1905.12265](https://arxiv.org/abs/1905.12265).
- 281 Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks.
282 In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-*
283 *26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL [https://openreview.](https://openreview.net/forum?id=SJU4ayYgl)
284 [net/forum?id=SJU4ayYgl](https://openreview.net/forum?id=SJU4ayYgl).
- 285 Christian Koke. Di-graphs with tightly connected clusters: effective graph laplacians and resolvent
286 convergence. *arXiv preprint arXiv:2601.18057*, 2026. URL [https://arxiv.org/abs/](https://arxiv.org/abs/2601.18057)
287 [2601.18057](https://arxiv.org/abs/2601.18057).
288
- 289 Christian Koke and Daniel Cremers. Holonets: Spectral convolutions do extend to directed graphs.
290 In *The Twelfth International Conference on Learning Representations*, 2024. URL [https://](https://openreview.net/forum?id=EhmEwfavOW)
291 openreview.net/forum?id=EhmEwfavOW.
292
- 293 Christian Koke, Abhishek Saroha, Yuesong Shen, Marvin Eisenberger, Michael M. Bronstein, and
294 Daniel Cremers. Transferability for graph convolutional networks. In *ICML 2024 Workshop on*
295 *Geometry-grounded Representation Learning and Generative Modeling*, 2024. URL [https://](https://openreview.net/forum?id=rKEdfcaqYX)
296 openreview.net/forum?id=rKEdfcaqYX.
- 297 Christian Koke, Yuesong Shen, Abhishek Saroha, Marvin Eisenberger, Bastian Rieck, Michael M.
298 Bronstein, and Daniel Cremers. On incorporating scale into graph networks. In *ICLR 2025*
299 *Workshop on Machine Learning for Multiscale Physics (MLMP)*. OpenReview, March 2025. URL
300 <https://openreview.net/forum?id=SRCsyJafgP>.
301
- 302 Thien Le and Stefanie Jegelka. Limits, approximation and size transferability for GNNs on sparse
303 graphs via graphops. In *Thirty-seventh Conference on Neural Information Processing Systems*,
304 2023. URL <https://openreview.net/forum?id=kDQwossJuI>.
305
- 306 Junhyun Lee, Inyeop Lee, and Jaewoo Kang. Self-attention graph pooling. In Kamalika Chaudhuri
307 and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine*
308 *Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3734–3743. PMLR,
309 09–15 Jun 2019. URL <https://proceedings.mlr.press/v97/lee19c.html>.
- 310 Ron Levie, Michael M. Bronstein, and Gitta Kutyniok. Transferability of spectral graph convolutional
311 neural networks. *CoRR*, abs/1907.12972, 2019a. URL [http://arxiv.org/abs/1907.](http://arxiv.org/abs/1907.12972)
312 [12972](http://arxiv.org/abs/1907.12972).
313
- 314 Ron Levie, Federico Monti, Xavier Bresson, and Michael M. Bronstein. Cayleynets: Graph
315 convolutional neural networks with complex rational spectral filters. *IEEE Trans. Signal Process.*,
316 67(1):97–109, 2019b. doi: 10.1109/TSP.2018.2879624. URL [https://doi.org/10.1109/](https://doi.org/10.1109/TSP.2018.2879624)
317 [TSP.2018.2879624](https://doi.org/10.1109/TSP.2018.2879624).
- 318 Renjie Liao, Zhizhen Zhao, Raquel Urtasun, and Richard S. Zemel. Lanczosnet: Multi-scale deep
319 graph convolutional networks. In *7th International Conference on Learning Representations,*
320 *ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL [https://](https://openreview.net/forum?id=BkedznAqKQ)
321 openreview.net/forum?id=BkedznAqKQ.
322
- 323 Andreas Loukas. Graph reduction with spectral and cut guarantees. *J. Mach. Learn. Res.*, 20:
116:1–116:42, 2019. URL <https://jmlr.org/papers/v20/18-680.html>.

- 324 Andreas Loukas and Pierre Vandergheynst. Spectrally approximating large graphs with smaller
325 graphs. In Jennifer G. Dy and Andreas Krause (eds.), *Proceedings of the 35th International*
326 *Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15,*
327 *2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 3243–3252. PMLR, 2018.
328 URL <http://proceedings.mlr.press/v80/loukas18a.html>.
- 329 Sohir Maskey, Ron Levie, and Gitta Kutyniok. Transferability of graph neural networks: an extended
330 graphon approach. *CoRR*, abs/2109.10096, 2021. URL [https://arxiv.org/abs/2109.](https://arxiv.org/abs/2109.10096)
331 [10096](https://arxiv.org/abs/2109.10096).
- 332 Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the construction
333 of internet portals with machine learning. *Inf. Retr.*, 3(2):127–163, 2000. doi: 10.1023/A:
334 1009953814988. URL <https://doi.org/10.1023/A:1009953814988>.
- 335 Olaf Post. *Spectral Analysis on Graph-like Spaces / by Olaf Post*. Lecture Notes in Mathematics, 2039.
336 Springer Berlin Heidelberg, Berlin, Heidelberg, 1st ed. 2012. edition, 2012. ISBN 3-642-23840-8.
337
- 338 T. Mitchell Roddenberry, Fernando Gama, Richard G. Baraniuk, and Santiago Segarra. On local
339 distributions in graph signal processing. *IEEE Trans. Signal Process.*, 70:5564–5577, 2022. doi:
340 10.1109/TSP.2022.3223217. URL <https://doi.org/10.1109/TSP.2022.3223217>.
- 341 Luana Ruiz, Luiz F. O. Chamon, and Alejandro Ribeiro. Graphon neural networks and the
342 transferability of graph neural networks. In Hugo Larochelle, Marc’Aurelio Ranzato, Raia Hadsell,
343 Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing*
344 *Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020,*
345 *December 6-12, 2020, virtual*, 2020. URL [https://proceedings.neurips.cc/paper/](https://proceedings.neurips.cc/paper/2020/hash/12bcd658ef0a540cab36cdf2b1046fd-Abstract.html)
346 [2020/hash/12bcd658ef0a540cab36cdf2b1046fd-Abstract.html](https://proceedings.neurips.cc/paper/2020/hash/12bcd658ef0a540cab36cdf2b1046fd-Abstract.html).
- 347 M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld. Fast and accurate modeling of
348 molecular atomization energies with machine learning. *Physical Review Letters*, 108:058301,
349 2012.
- 350 Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-
351 Rad. Collective classification in network data. *AI Magazine*, 29(3):93, Sep. 2008. doi:
352 10.1609/aimag.v29i3.2157. URL [https://ojs.aaai.org/index.php/aimagazine/](https://ojs.aaai.org/index.php/aimagazine/article/view/2157)
353 [article/view/2157](https://ojs.aaai.org/index.php/aimagazine/article/view/2157).
- 354 Gerald Teschl. *Mathematical Methods in Quantum Mechanics*. American Mathematical Society,
355 2014.
- 356 Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua
357 Bengio. Graph attention networks. In *6th International Conference on Learning Representations,*
358 *ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings.*
359 *OpenReview.net*, 2018. URL <https://openreview.net/forum?id=rJXmpikCZ>.
- 360 Zhiyang Wang, Luana Ruiz, and Alejandro Ribeiro. Stability of neural networks on riemannian
361 manifolds. *2021 29th European Signal Processing Conference (EUSIPCO)*, pp. 1845–1849, 2021.
362 URL <https://api.semanticscholar.org/CorpusID:232110514>.
- 363 Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural
364 networks? In *7th International Conference on Learning Representations, ICLR 2019, New Orleans,*
365 *LA, USA, May 6-9, 2019. OpenReview.net*, 2019. URL [https://openreview.net/forum?](https://openreview.net/forum?id=ryGs6iA5Km)
366 [id=ryGs6iA5Km](https://openreview.net/forum?id=ryGs6iA5Km).
- 367 Xuebin Zheng, Bingxin Zhou, Junbin Gao, Yuguang Wang, Pietro Lió, Ming Li, and Guido
368 Montúfar. How framelets enhance graph neural networks. In Marina Meila and Tong Zhang (eds.),
369 *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July*
370 *2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12761–12771.
371 PMLR, 2021. URL <http://proceedings.mlr.press/v139/zheng21c.html>.
- 372
- 373
- 374
- 375
- 376
- 377

A GRAPHS AND GRAPH LEARNING

We first briefly review graphs and graph neural networks:

A.1 GRAPHS AND THEIR DESCRIPTION

We consider finite, weighted graphs $G = (V, E)$ equipped with a *node weight matrix* $M = \text{diag}(\mu_v)_{v \in V}$. Each weight $\mu_v > 0$ reflects node-wise information. Examples are physical mass or charge of atom v in a molecular graph, number of users represented if v is a node in a social network, or a volume element if v is part of a point cloud. Often, all node masses are set to $\mu_v = 1$ so that $M = I$. Edges $(u, v) \in E$ of a graph carry symmetric non-negative (potentially binary) weights A_{uv} which we collect in the adjacency matrix A . The corresponding degree matrix is $D = \text{diag}(d_v)$ with $d_v = \sum_u A_{uv}$.

The natural self-adjoint Laplacian associated with a node-weighted graph is $L = M^{-1}(D - A)$. It is self-adjoint with respect to the M -weighted inner product $\langle f, g \rangle_M = f^\top M g$. When the node weights encode local volumes or sampling densities it may be thought of as the discrete counterpart of the Laplacian $L = -\Delta$ on a continuum domain (Chung, 1997). While the natural Laplacian is central to our work, more commonly used in machine learning applications is instead the normalized Laplacian $\mathcal{L} = I - D^{-1/2} A D^{-1/2}$. It is self-adjoint in the standard Euclidean inner product and its spectrum is contained in the interval $[0, 2]$.

A.2 GRAPH NEURAL NETWORKS

Graph neural networks are machine learning methods adapted to handle graph structured data. Corresponding network architectures broadly fall into two categories:

Message Passing Networks: Message Passing Neural Networks (MPNNs) Gilmer et al. (2017); Bronstein et al. (2021) are arguably the most widely used class of GNNs. At each layer k , node features $h_v^{(k)}$ are updated by aggregating information from neighbouring nodes:

$$m_{vu}^{(k)} = \phi(h_v^{(k)}, h_u^{(k)}, A_{uv}) \quad (4)$$

$$h_v^{(k+1)} = \text{AGG}_{u \in \mathcal{N}(v)} m_{vu} \quad (5)$$

Here A_{uv} denotes edge weights. Architectures differ in which message- and aggregation functions are being used (Xu et al., 2019; Velickovic et al., 2018; Kipf & Welling, 2017; Hamilton et al., 2017), with typical choices for the neighborhood aggregation function AGG being sum, avg, or (seldom) max (Fey et al., 2025).

Spectral Networks: Spectral graph convolutional layers (Bruna et al., 2014; Defferrard et al., 2016) apply learnable filters $g_\theta(\mathcal{L})$ to node-wise information via matrix-multiplication. The filters themselves are composed of learnable scalar functions $g_\theta(\cdot)$ which are applied to the normalized graph Laplacian \mathcal{L} . In practice, the filter g_θ is not implemented via eigendecomposition but approximated via polynomials (He et al., 2022; 2021; Koke & Cremers, 2024; Levie et al., 2019b). The prototypical example is ChebNet, where filters are parametrized as $g_\theta(\lambda) \approx \sum_{\ell=0}^K \theta_\ell T_\ell(\lambda)$, with T_ℓ the ℓ^{th} -Chebyshev polynomial. With node feature matrix X and learnable weight matrices $\{W_\ell\}_\ell$, a spectral layer is then efficiently implemented as

$$X^{(k+1)} = \sum_{\ell=0}^K T_\ell(\mathcal{L}) X^{(k)} W_\ell \quad (6)$$

B STABILITY OF LATENT REPRESENTATIONS GENERATED BY LAPLACE-TRANSFORM BASED GNNs FOR GRAPHS ON THE SAME NODE SET

B.1 NODE LEVEL STABILITY

In this Section, we are concerned with two graphs G_1 and G_2 that share a common node set. We assume that their heat kernels are similar in the sense of

$$\|e^{-tL_1} - e^{-tL_2}\| \approx 0. \quad (7)$$

We are then interested in bounding the variation in latent representations generated by a GNN $\Phi_{\mathcal{W}, \mathcal{B}, \Psi}$ with weights \mathcal{W} , (potential) biases, and Laplace transform propagation matrices Ψ .

B.1.1 SPECTRAL LAPLACE-TRANSFORM METHODS

We first consider spectral Laplace Transform based spectral GNNs:

Theorem B.1. Let $\Xi_{\mathcal{W}, \mathcal{B}, \Psi}$ be a K -layer deep spectral graph convolutional architecture. Assume in each layer $1 \leq \ell \leq K$ that $\sum_i \|W_i^\ell\| \leq W$ and $\|B^\ell\| \leq B$ with weight- and bias-matrices $\{W^\ell, B^\ell\}$. Assume the non-linearity ρ is 1-Lipschitz. Choose a constant C such that $C \geq \|\Psi_i(L_1)\|, \|\Psi_i(L_2)\|$ for all Laplacian propagation matrices $\Psi_i(L_1), \Psi_i(L_2)$. W.l.o.g. assume $CW > 1$. With this, we have with $\delta = \max_{i \in I} \{\|\Psi_i(L_1) - \Psi_i(L_2)\|\}$ and input node-feature matrix X , that

$$\|\Xi_{\mathcal{W}, \mathcal{B}, \Psi}(L, X) - \Xi_{\mathcal{W}, \mathcal{B}, \Psi}(\tilde{L}, X)\| \leq \left[K \cdot C^K W^{K-1} \cdot \left(\|X\| + \frac{1}{CW-1} B \right) \right] \cdot \delta. \quad (8)$$

Proof. For simplicity in notation, let us denote the hidden representations in the network corresponding to L_2 by X_i^ℓ . With this, we note:

$$\|X_1^K - X_2^K\| \leq \sum_{i \in I} \|\psi_i(L_1) - \psi_i(L_2)\| \cdot \|X_1^{K-1}\| \cdot \|W_i^K\| + \sum_{i \in I} \|\psi_i(L_2)\| \cdot \|X_2^{K-1} - X_1^{K-1}\| \cdot \|W_i^K\| \quad (9)$$

$$\leq \delta W \|X_1^{K-1}\| + CW \|X_1^{K-1} - X_2^{K-1}\| \quad (10)$$

$$\leq \delta W \|X_1^{K-1}\| + CW \delta \|X_1^{K-2}\| + (CW)^2 \|X_2^{K-1} - X_1^{K-1}\| \quad (11)$$

$$\leq \frac{\delta}{C} \cdot \left(\sum_{\ell=1}^K (CW)^\ell \|X_1^{K-\ell}\| \right) \quad (12)$$

$$= \frac{\delta}{C} \cdot \left(\sum_{j=0}^{K-1} (CW)^{K-j} \|X_1^j\| \right) \quad (13)$$

Hence we need to bound the quantity $\|X_1^j\|$ in terms of C, W, B and X .

We have

$$\|X_1^j\| \leq \sum_i \|\psi_i(L_1)\| \cdot \|X_1^{j-1}\| \cdot \|W_i^j\| + \|B^j\| \quad (14)$$

$$\leq CW \|X_1^{j-1}\| + B \quad (15)$$

$$\leq (CW)^2 \|X_1^{j-2}\| + CW B + B \quad (16)$$

$$\leq B \left(\sum_{k=0}^{j-1} (CW)^k \right) + (CW)^j \|X\| \quad (17)$$

$$= \begin{cases} B \frac{(CW)^j - 1}{CW - 1} + (CW)^j \|X\| & ; CW \neq 1 \\ jB + \|X\| & ; CW = 1 \end{cases}. \quad (18)$$

486 For the case $CW = 1$, we thus find

$$487 \quad \|X_1^K - X_2^K\| \leq \frac{\delta}{C} \cdot \left(\sum_{j=0}^{K-1} (jB + \|X\|) \right) \quad (19)$$

$$491 \quad = \frac{\delta}{C} \cdot \left(K\|X\| + B \frac{K(K-1)}{2} \right). \quad (20)$$

494 For the case $CW \neq 1$, we find

$$496 \quad \|X_1^K - X_2^K\| \leq \frac{\delta}{C} \cdot \left(\sum_{j=0}^{K-1} (CW)^{K-j} \left[B \frac{(CW)^j - 1}{CW - 1} + (CW)^j \|X\| \right] \right) \quad (21)$$

$$499 \quad (22)$$

500 For $CW > 1$, we may further estimate this as

$$502 \quad \|X_1^K - X_2^K\| \leq \frac{\delta}{C} \cdot \left(\sum_{j=0}^{K-1} (CW)^{K-j} \left[B \frac{(CW)^j - 1}{CW - 1} + (CW)^j \|X\| \right] \right) \quad (23)$$

$$506 \quad \leq \delta \cdot \frac{K(CW)^K}{C} \left[\frac{B}{CW - 1} + \|X\| \right]. \quad (24)$$

508 This proves the claim. \square

510 Next we note that we can estimate δ in terms of $\max_i \left\{ \int_0^\infty \hat{\psi}_i(t) \|e^{-tL_1} - e^{-tL_2}\| dt \right\}$.

511 **Lemma B.1.** *We have*

$$513 \quad \delta \leq \max_i \left\{ \int_0^\infty \hat{\psi}_i(t) \|e^{-tL_1} - e^{-tL_2}\| dt \right\}. \quad (25)$$

515 *Proof.* We note that by definition, we have

$$517 \quad \psi_i(L) := \int_0^\infty e^{-tL} \hat{\psi}_i(t) dt. \quad (26)$$

519 From this we may infer

$$521 \quad \|\psi_i(L_1) - \psi_i(L_2)\| = \left\| \int_0^\infty \hat{\psi}_i(t) [e^{-tL_1} - e^{-tL_2}] dt \right\| \leq \int_0^\infty \hat{\psi}_i(t) \|e^{-tL_1} - e^{-tL_2}\| dt. \quad (27)$$

523 Taking the maximum over i indexing all propagation matrices. yields the claim.

525 \square

526 Combining Lemma B.1 with Theorem B.1 then establishes a bound on node-wise latent embeddings in terms of the heat kernel distance $\|e^{-tL_1} - e^{-tL_2}\|$ for spectral Laplace transform based methods.

530 B.1.2 MESSAGE PASSING BASED LAPLACE TRANSFORM METHODS:

531 We next establish similar results for Message passing networks. To this end, we first introduce some notation. We continue to denote the node-feature matrix by X . We have as layer-wise update the implemented scheme

$$533 \quad m_{vu}^{(\ell)} = [\psi(L)]_{vu} \cdot \phi(h_v^{(k)}, h_u^{(k)}) \quad (28)$$

$$534 \quad h_v^{(\ell+1)} = \sum_{u \in G} m_{vu}. \quad (29)$$

538 Denote the feature dimension of layer ℓ by d_ℓ . Hence we have $h_u^{(\ell)} \in \mathbb{R}^{d_\ell}$ and $\phi(h_v^{(\ell)}, h_u^{(\ell)}) \in \mathbb{R}^{d_{\ell+1}}$.

As shorthand notation, we collect the latent representations in layer (ℓ) into a feature matrix X^ℓ whose columns are given as

$$[X^\ell]_u = h_u^{(\ell)}. \quad (30)$$

We then define the map $\Phi : \mathbb{R}^{N \times d_\ell} \rightarrow \mathbb{R}^{N \times N \times d_{\ell+1}}$ componentwise as

$$[\Phi(X^\ell)]_{vu} = \phi([X^\ell]_{v:}, [X^\ell]_{u:}) \equiv \phi(h_v^\ell, h_u^\ell). \quad (31)$$

With this we find

$$[X^{\ell+1}]_v = \sum_u [\psi(L)]_{vu} [\Phi(X^\ell)]_{vu}. \quad (32)$$

This now allows us to prove:

Theorem B.2. *Suppose we are given two graphs G_1, G_2 defined on a common node set. Denote the collection of node-wise latent representations generated by a K -layer deep Laplace transform based message passing network $\Xi_{\psi, \phi}$ by X_1^K, X_2^K respectively.*

Suppose there is a constant C so that $C \geq \|\psi(L_1)\|, \|\psi(L_2)\|$. Assume that the message function ϕ is Lipschitz continuous:

$$\|\phi([X_1]_{v:}, [X_1]_{u:}) - \phi([X_2]_{v:}, [X_1]_{u:})\| \leq \frac{L}{4} (\|[X_1]_{v:}, -[X_2]_{v:}\| + \|[X_1]_{u:}, -[X_2]_{u:}\|). \quad (33)$$

Further assume that Φ is bounded, with $\|\Phi(\cdot)\| \leq D$. Then there is a constant C_Ξ so that

$$\|X_1^K - X_2^K\| \leq C_\Xi \cdot \|\psi(L_1) - \psi(L_2)\|. \quad (34)$$

Proof. We note:

$$[X_1^K]_{v:} - [X_2^K]_{v:} = \sum_u [\psi(L_1)]_{vu} [\Phi(X_1^{K-1})]_{vu} - \sum_u [\psi(L_2)]_{vu} [\Phi(X_2^{K-1})]_{vu}. \quad (35)$$

$$= \left[\sum_u ([\psi(L_1)]_{vu} - \psi(L_2)_{vu}) [\Phi(X_1^{K-1})]_{vu} \right] + \left[\sum_u [\psi(L_2)]_{vu} ([\Phi(X_1^{K-1})]_{vu} - [\Phi(X_2^{K-1})]_{vu}) \right]. \quad (36)$$

With this, we find

$$\|X_1^K - X_2^K\| \quad (37)$$

$$\leq \left(\sum_v \left\| \sum_u ([\psi(L_1)]_{vu} - \psi(L_2)_{vu}) [\Phi(X_1^{K-1})]_{vu} \right\|_{\mu_v}^2 \right)^{\frac{1}{2}} \quad (38)$$

$$+ \left(\sum_v \left\| \sum_u [\psi(L_2)]_{vu} ([\Phi(X_1^{K-1})]_{vu} - [\Phi(X_2^{K-1})]_{vu}) \right\|_{\mu_v}^2 \right)^{\frac{1}{2}} \quad (39)$$

with the node weights (c.f. Section A.1) denoted by $\{\mu_v\}_v$.

To bound the first term, we note:

$$\left(\sum_v \left\| \sum_u ([\psi(L_1)]_{vu} - \psi(L_2)_{vu}) [\Phi(X_1^{K-1})]_{vu} \right\|_{\mu_v}^2 \right)^{\frac{1}{2}} \quad (40)$$

$$\leq D \cdot \left(\sum_{u,v} |[\psi(L_1)]_{vu} - \psi(L_2)_{vu}|^2 \mu_v \right)^{\frac{1}{2}} \quad (41)$$

$$\lesssim D \|\psi(L_1) - \psi(L_2)\| \cdot \left(\sum_v \mu_v \right)^{\frac{1}{2}} \quad (42)$$

$$= D \sqrt{\mu(G)} \|\psi(L_1) - \psi(L_2)\|, \quad (43)$$

Here we have bounded $|\psi(L_1)_{vu} - \psi(L_2)_{vu}| \lesssim \|\psi(L_1) - \psi(L_2)\|$ ¹

Let us next bound the second term. We have

$$\left(\sum_v \left\| \sum_u [\psi(L_2)]_{vu} (\Phi(X_1^{K-1})_{vu} - \Phi(X_2^{K-1})_{vu}) \right\|^2 \mu_v \right)^{\frac{1}{2}} \quad (44)$$

$$\leq C \cdot \left(\sum_{u,v} \|(\Phi(X_1^{K-1})_{vu} - \Phi(X_2^{K-1})_{vu})\|^2 \mu_v \mu_u \right)^{\frac{1}{2}} \quad (45)$$

$$\leq CL \left(\sum_u \mu_u \right)^{\frac{1}{2}} \cdot \left(\sum_u \|[X_1^{K-1}]_{u:} - [X_2^{K-1}]_{u:}\|^2 \mu_u \right)^{\frac{1}{2}}. \quad (46)$$

$$= CL \sqrt{\mu(G)} \cdot \|X_1^{K-1} - X_2^{K-1}\|. \quad (47)$$

Putting the two terms together yields the iteration step

$$\|X_1^K - X_2^K\| \leq C_A \|\psi(L_1) - \psi(L_2)\| + C_B \|X_1^{K-1} - X_2^{K-1}\|, \quad (48)$$

with constants C_A, C_B as determined above. Iterating this through to $K = 0$ yields the following:

Let $d_K := \|X_1^K - X_2^K\|$ and $\Delta := \|\psi(L_1) - \psi(L_2)\|$. Assume the recursive bound

$$d_K \leq C_A \Delta + C_B d_{K-1}, \quad K \geq 1.$$

Furthermore we have the initial condition $d_0 = \|X_1^0 - X_2^0\| = \|X - X\| = 0$.

Iterating the inequality yields

$$\begin{aligned} d_K &\leq C_A \Delta + C_B d_{K-1} \\ &\leq C_A \Delta + C_B (C_A \Delta + C_B d_{K-2}) \\ &= C_A \Delta (1 + C_B) + C_B^2 d_{K-2} \\ &\vdots \\ &\leq C_A \Delta \sum_{j=0}^{K-1} C_B^j + C_B^K d_0. \end{aligned}$$

Using $d_0 = 0$, we obtain

$$\|X_1^K - X_2^K\| \leq C_A \|\psi(L_1) - \psi(L_2)\| \sum_{j=0}^{K-1} C_B^j.$$

Evaluating the geometric series gives the closed-form bound

$$\|X_1^K - X_2^K\| \leq \begin{cases} C_A \|\psi(L_1) - \psi(L_2)\| \frac{1 - C_B^K}{1 - C_B}, & C_B \neq 1, \\ K C_A \|\psi(L_1) - \psi(L_2)\|, & C_B = 1. \end{cases}$$

In particular, if $0 \leq C_B < 1$, then

$$\|X_1^K - X_2^K\| \leq \frac{C_A}{1 - C_B} \|\psi(L_1) - \psi(L_2)\|,$$

which provides a uniform bound independent of K .

□

¹We are using the standard notation ' \lesssim ' to denote smaller than, up to a fixed 'universal' constant.

Combining the above Theorem with (27) then yields

Corollary B.3. *In the setting of Theorem B.2 we have*

$$\|X_1^K - X_2^K\| \leq C_{\Xi} \cdot \int_0^{\infty} \hat{\psi}(t) \|e^{-tL_1} - e^{-tL_2}\| dt = C_{\Xi} \max_i \left\{ \int_0^{\infty} \hat{\psi}_i(t) \|e^{-tL_1} - e^{-tL_2}\| dt \right\}. \quad (49)$$

Here the last equality follows since we are taking the maximum over all used propagation matrices, and only ever use a single one.

Hence we have established a bound on variations of node-wise latent embeddings in terms of the heat kernel distance $\|e^{-tL_1} - e^{-tL_2}\|$ for spectral Laplace transform based methods.

B.2 GRAPH LEVEL STABILITY

Next we promote the node-level stability results we derived in Subsection B.1 above to graph level stability results:

We may summarize Theorems B.1 & B.2 as: Given an input node-feature matrix X on graphs G_1, G_2 sharing a common node set, there exists constants C so that the output variation $\|X_1^K - X_2^K\|$ of K -layer deep Laplace transform based GNNs may be bounded as

$$\|X_1^K - X_2^K\| \leq C \max_i \left\{ \int_0^{\infty} \hat{\psi}_i(t) \|e^{-tL_1} - e^{-tL_2}\| dt \right\}. \quad (50)$$

We are now interested in how these stability results translate to graph level features. To this end we first specify our graph level aggregation method Ω :

Definition B.2. At the final (K^{th} -)layer We aggregate node-level embeddings $X^K \in \mathbb{R}^{N \times d_K}$ of individual nodes to graph-embeddings $\Omega(X) \in \mathbb{R}^{d_K}$ as $\Omega(X^K)_j = \sum_{i=1}^N |X_{ij}^K| \cdot \mu_i$. Here $\{\mu_i\}_i$ is the set of node-weights.

Graph level features F_1, F_2 are then generated as $F_1 = \Omega(X_1^K)$ and $F_2 = \Omega(X_2^K)$.

With this we find:

Theorem B.4. *In the setting of (50), with graph level aggregation as in Definition B.2, we find*

$$\|F_1 - F_2\| \leq C \max_i \left\{ \int_0^{\infty} \hat{\psi}_i(t) \|e^{-tL_1} - e^{-tL_2}\| dt \right\}. \quad (51)$$

Proof. We only have to note that the aggregation method Ω is 1-Lipschitz (as a consequence of the reverse triangle inequality). Hence

$$\|F_1 - F_2\| = \|\Omega(X_1^K) - \Omega(X_2^K)\| \leq \|X_1^K - X_2^K\| \quad (52)$$

and the claim follows from 50. \square

C STABILITY OF LATENT REPRESENTATIONS GENERATED BY LAPLACE-TRANSFORM BASED GNNs FOR GRAPHS ON DIFFERING RESOLUTION SCALES

In this section, we establish bounds on differences in latent representations generated for Laplace-transform based GNNs when confronted with graphs describing the same underlying object at multiple resolutions. We briefly recall the setting:

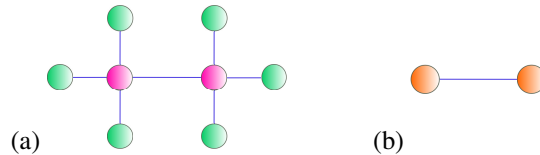


Figure 4: (a) Original graph G (b) Coarse grained \underline{G}

We have a high resolution graph G with associated Laplacian L and node-feature matrix X . We also have a lower resolution graph \underline{G} , with associated Laplacian \underline{L} and node-feature matrix $\underline{X} := J^\downarrow X$ arising from the original node feature matrix X on G via projection to \underline{G} . Here we have made use of the projection operator J^\downarrow which averages node information over clusters that are condensed into supernodes. Interpolation J^\uparrow instead assigns information at each supernode in \underline{G} to every node making up the corresponding cluster in G .

We have (Koke, 2026) (c.f. also Appendix E) that $J^\downarrow J^\uparrow = I_{\underline{G}}$ (i.e. equal to the identity matrix on \underline{G}).

C.1 NODE LEVEL RESULTS

We are then initially interested in the following question: Suppose we have a node feature matrix X on the graph G . We can generate node-level latent embeddings $\Xi(X)$ by feeding this node-wise information into a (Laplace-transform based) node-level GNN Ξ . How different is this outcome from projecting X onto the coarser graph \underline{G} , running the GNN there to generate node-level embeddings $\Xi(J^\downarrow X)$ and then interpolating the generated embeddings back up to G via the interpolation operator J^\uparrow . That is to say, we are interested in the difference $\|\Xi(X) - J^\uparrow \Xi(J^\downarrow X)\|$.

C.1.1 SPECTRAL LAPLACE-TRANSFORM METHODS

We begin by establishing $\|\Xi(X) - J^\uparrow \Xi(J^\downarrow X)\| \leq C \cdot \max_i \left\{ \int_0^\infty |\hat{\psi}_i(t)| \cdot \|e^{-tL} - J^\uparrow e^{-t\underline{L}} J^\downarrow\| \right\}$ for spectral methods:

Theorem C.1. Let $\Xi_{\mathcal{W}, \mathcal{B}, \Psi}$ be a K -layer deep Laplace transform based network. Assume that $\sum_{i \in I} \|W_i^\ell\| \leq W$ and $\|B^k\| \leq B$. Choose $C \geq \|\Psi_i(L)\|, \|\Psi_i(\underline{L})\|$ ($i \in I$) and w.l.o.g. assume $CW > 1$. Set $\max_{i \in I} \{\|\psi_i(L) - J^\uparrow \psi_i(\underline{L}) J^\downarrow\|\} = \delta$. With this, we have that

$$\|\Xi_{\mathcal{W}, \mathcal{B}, \Psi}(X) - J^\uparrow \Xi_{\mathcal{W}, \mathcal{B}, \Psi}(J^\downarrow X)\| \leq \left[K \cdot C^K W^{K-1} \cdot \left(\|X\| + \frac{1}{CW-1} B \right) \right] \cdot \delta. \quad (53)$$

Proof. Let us define

$$\underline{X} := J^\downarrow X. \quad (54)$$

Let us further use the notation $\underline{\psi}_i := \psi_i(\underline{L})$ and $\psi_i := \psi_i(L)$.

Denote by X^ℓ and \underline{X}^ℓ the (hidden) feature matrices generated in layer ℓ for networks based on ψ_i and $\underline{\psi}_i$ respectively: I.e. we have

$$X^\ell = \rho \left(\sum_{i \in I} \psi_i X^{\ell-1} W_i^\ell + B^\ell \right) \quad (55)$$

and

$$\underline{X}^\ell = \rho \left(\sum_{i \in I} \underline{\psi}_i \underline{X}^{\ell-1} W_i^\ell + \underline{B}^\ell \right). \quad (56)$$

Since bias terms are proportional to constant vectors on the graphs, we have

$$J^\downarrow B = \underline{B} \quad (57)$$

and

$$J^\uparrow \underline{B} = B \quad (58)$$

for bias matrices B and \underline{B} in networks deployed on G and \underline{G} respectively.

We then have

756

757

758

$$\|\Xi_{\mathcal{W}, \mathcal{B}, \Psi}(X) - J^\uparrow \Xi_{\mathcal{W}, \mathcal{B}, \Psi}(J^\downarrow X)\| \quad (59)$$

759

$$= \|X^K - J^\uparrow \underline{X}^K\| \quad (60)$$

760

761

$$= \left\| \rho \left(\sum_{i \in I} \psi_i X^{K-1} W_i^K + B^K \right) - J^\uparrow \rho \left(\sum_{i \in I} \underline{\psi}_i \underline{X}^{K-1} W_i^K + \underline{B}^L \right) \right\| \quad (61)$$

762

763

764

$$= \left\| \rho \left(\sum_{i \in I} \psi_i X^{K-1} W_i^K + B^K \right) - \rho \left(J^\uparrow \sum_{i \in I} \tilde{\psi}_i \underline{X}^{K-1} W_i^K + B^L \right) \right\| \quad (62)$$

765

766

767

Here we used the fact that ρ and J^\uparrow commute (since J^\uparrow is an assignment matrix and has all non-zero entries equal to one). Using the fact that the non-linearity $\rho(\cdot)$ is 1-Lipschitz-continuous we can establish

771

$$\|\Xi_{\mathcal{W}, \mathcal{B}, \Psi}(X) - J^\uparrow \Xi_{\mathcal{W}, \mathcal{B}, \Psi}(J^\downarrow X)\| \quad (64)$$

772

773

$$\leq \left\| \rho \left(\sum_{i \in I} \psi_i X^{K-1} W_i^K + B^K \right) - \rho \left(J^\uparrow \sum_{i \in I} \underline{\psi}_i \underline{X}^{K-1} W_i^K + B^K \right) \right\| \quad (65)$$

774

775

776

$$\leq \left\| \sum_{i \in I} \psi_i X^{K-1} W_i^K + B^K - J^\uparrow \sum_{i \in I} \underline{\psi}_i \underline{X}^{K-1} W_i^K + B^K \right\|. \quad (66)$$

777

Since $J^\downarrow J^\uparrow = Id_G$, we have

779

780

$$\|\Xi_{\mathcal{W}, \mathcal{B}, \Psi}(X) - J^\uparrow \Xi_{\mathcal{W}, \mathcal{B}, \Psi}(J^\downarrow X)\| \quad (67)$$

781

782

$$\leq \left\| \sum_{i \in I} \psi_i X^{K-1} W_i^K - \sum_{i \in I} (J^\uparrow \underline{\psi}_i J^\downarrow) J^\uparrow \underline{X}^{K-1} W_i^K \right\| \quad (68)$$

783

From this, we find (using $\|J^\uparrow\|, \|J^\downarrow\| \leq 1$), that

784

785

786

$$\|\Xi_{\mathcal{W}, \mathcal{B}, \Psi}(X) - J^\uparrow \Xi_{\mathcal{W}, \mathcal{B}, \Psi}(J^\downarrow X)\| \quad (69)$$

787

788

$$\leq \left\| \sum_{i \in I} \psi_i X^{K-1} W_i^K - \sum_{i \in I} (J^\uparrow \underline{\psi}_i J^\downarrow) J^\uparrow \underline{X}^{K-1} W_i^K \right\| \quad (70)$$

789

790

791

$$\leq \left\| \sum_{i \in I} (\psi_i - J^\uparrow \underline{\psi}_i J^\downarrow) X^{K-1} W_i^K \right\| + \sum_{i \in I} \|J^\uparrow \underline{\psi}_i J^\downarrow\| \cdot \|J^\uparrow \underline{X}^{K-1} - X^{K-1}\| \cdot \|W_i^K\| \quad (71)$$

792

793

794

$$\leq \left\| \sum_{i \in I} (\psi_i - J^\uparrow \underline{\psi}_i J^\downarrow) X^{K-1} W_i^K \right\| + CW \cdot \|J^\uparrow \underline{X}^{K-1} - X^{K-1}\| \quad (72)$$

795

796

797

$$\leq \sum_{i \in I} \left\| (\psi_i - J^\uparrow \underline{\psi}_i J^\downarrow) \right\| \cdot \|X^{K-1}\| \cdot \|W_i^K\| + CW \cdot \|\tilde{J}^\downarrow \underline{X}^{K-1} - X^{K-1}\| \quad (73)$$

798

799

800

$$\leq \delta \cdot \|X^{K-1}\| W + CW \cdot \|\tilde{J}^\uparrow \underline{X}^{K-1} - X^{K-1}\| \quad (74)$$

801

802

Arguing as in the proof of Theorem B.1 then yields the claim.

803

804

805

C.1.2 MESSAGE PASSING BASED LAPLACE TRANSFORM METHODS:

806

807

808

809

We next establish similar results for message passing networks. Sticking with the notation of the preceding Subsection C.1.1, we are hence interested in bounding $X^K - J^\uparrow \underline{X}^K$, with \underline{X} being the node feature matrix generated after a K -layer Laplace transform based MPNN.

Theorem C.1. Assume there is a constant C so that $C \geq \|\psi(L)\|, \|\psi(\underline{L})\|$. Assume that the message function ϕ is Lipschitz continuous:

$$\|\phi([X_1]_{v:}, [X_1]_{u:}) - \phi([X_2]_{v:}, [X_1]_{u:})\| \leq \frac{L}{4} (\|[X_1]_{v:}, -[X_2]_{v:}\| + \|[X_1]_{u:}, -[X_2]_{u:}\|). \quad (76)$$

Further assume that Φ is bounded, with $\|\Phi(\cdot)\| \leq D$. Then there is a constant C_{Ξ} so that

$$\|X^K - J^\dagger \underline{X}^K\| \leq C_{\Xi} \cdot \int_0^\infty \hat{\psi}(t) \|e^{-tL} - J^\dagger e^{-t\underline{L}} J^\dagger\|. \quad (77)$$

Proof. We note:

$$X_{v:}^K - [J^\dagger \underline{X}^K]_{v:} = X_{v:}^K - \sum_{a \in \underline{G}} J_{va}^\dagger \underline{X}_{a:}^K = \sum_{u \in G} [\psi(L)]_{vu} [\Phi(X^{K-1})]_{vu:} - \sum_{a, b \in \underline{G}} J_{va}^\dagger [\psi(\underline{L})]_{ab} [\Phi(\underline{X}^{K-1})]_{ab:}. \quad (78)$$

Creatively adding zero yields

$$X_{v:}^K - [J^\dagger \underline{X}^K]_{v:} \quad (79)$$

$$= \sum_{u \in G} [\psi(L)]_{vu} [\Phi(X^{K-1})]_{vu:} - \sum_{u \in G} [\psi(L)]_{vu} [\Phi(J^\dagger \underline{X}^{K-1})]_{vu:} \quad (80)$$

$$+ \sum_{u \in G} [\psi(L)]_{vu} [\Phi(J^\dagger \underline{X}^{K-1})]_{vu:} - \sum_{a, b \in \underline{G}} J_{va}^\dagger [\psi(\underline{L})]_{ab} [\Phi(\underline{X}^{K-1})]_{ab:}. \quad (81)$$

For the first term, we note

$$\left(\sum_{v \in G} \left\| \sum_{u \in G} [\psi(L)]_{vu} [\Phi(X^{K-1})]_{vu:} - \sum_{u \in G} [\psi(L)]_{vu} [\Phi(J^\dagger \underline{X}^{K-1})]_{vu:} \right\|^2 \mu_v \right)^{\frac{1}{2}} \quad (82)$$

$$\leq C \cdot \left(\sum_{v, u \in G} \sum_{u \in G} \left\| [\Phi(X^{K-1})]_{vu:} - [\Phi(J^\dagger \underline{X}^{K-1})]_{vu:} \right\|^2 \mu_v \right)^{\frac{1}{2}} \quad (83)$$

$$\leq CL \left(\sum_u \mu_u \right)^{\frac{1}{2}} \cdot \left(\sum_u \|[X^{K-1}]_{u:}, -[J^\dagger \underline{X}]_{u:}\|^2 \mu_u \right)^{\frac{1}{2}}. \quad (84)$$

$$= CL \sqrt{\mu(G)} \cdot \|X^{K-1} - J^\dagger \underline{X}^{K-1}\|. \quad (85)$$

Hence let us consider the second term. By Lemma C.2 below, we then have

864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917

$$\sum_{v \in G} \mu_g \left\| \sum_{u \in G} [\psi(L)]_{vu} [\Phi(J^\uparrow \underline{X}^{K-1})]_{vu} - \sum_{a, b \in \underline{G}} J_{va}^\uparrow [\psi(\underline{L})]_{ab} [\Phi(\underline{X}^{K-1})]_{ab} \right\|^2 \quad (86)$$

$$= \sum_{v \in G} \mu_g \left\| \sum_{u \in G} [\psi(L)]_{vu} \sum_{a, b \in \underline{G}} J_{va}^\uparrow J_{ub}^\uparrow [\Phi(\underline{X}^{K-1})]_{ab} - \sum_{a, b \in \underline{G}} J_{va}^\uparrow [\psi(\underline{L})]_{ab} [\Phi(\underline{X}^{K-1})]_{ab} \right\|^2 \quad (87)$$

$$= \sum_{v \in G} \mu_g \left\| \sum_{a, b \in \underline{G}} J_{va}^\uparrow [\psi(L) J^\uparrow]_{vb} [\Phi(\underline{X}^{K-1})]_{ab} - \sum_{a, b \in \underline{G}} J_{va}^\uparrow [\psi(\underline{L})]_{ab} [\Phi(\underline{X}^{K-1})]_{ab} \right\|^2 \quad (88)$$

$$= \sum_{v \in G} \mu_g \left\| \sum_{a, b \in \underline{G}} J_{va}^\uparrow [\psi(L) J^\uparrow]_{vb} [\Phi(\underline{X}^{K-1})]_{ab} - \sum_{a, b \in \underline{G}} J_{va}^\uparrow [J^\uparrow \psi(\underline{L})]_{vb} [\Phi(\underline{X}^{K-1})]_{ab} \right\|^2 \quad (89)$$

$$\leq D^2 \sum_{v \in G} \mu_g \left(\sum_{a, b \in \underline{G}} J_{va}^\uparrow |[\psi(L) J^\uparrow]_{vb} - [J^\uparrow \psi(\underline{L})]_{vb}| \right)^2 \quad (90)$$

$$\lesssim D^2 \sum_{v \in G} \mu_g \left(\sum_{a \in \underline{G}} J_{va}^\uparrow \|\psi(L) J^\uparrow - J^\uparrow \psi(\underline{L})\| \right)^2 \quad (91)$$

$$\leq D^2 \mu(G) \|\psi(L) J^\uparrow - J^\uparrow \psi(\underline{L})\|^2 \quad (92)$$

$$\lesssim D^2 \mu(G) \|\psi(L) - J^\uparrow \psi(\underline{L}) J^\downarrow\|^2. \quad (93)$$

Here the third step follows from the fact that $J_{va}^\uparrow [J^\uparrow \psi(\underline{L})]_{vb} = J_{va}^\uparrow [\psi(\underline{L})]_{ab}$. The last step follows from the fact that J^\downarrow is surjective.

Recursively, we thus have established that

$$\|X^K - J^\uparrow \underline{X}^K\| \leq C_A \|\psi(L) - J^\uparrow \psi(\underline{L}) J^\downarrow\| + C_B \|X^{K-1} - J^\uparrow \underline{X}^{K-1}\|. \quad (94)$$

We may unroll the recursion to obtain

$$\|X^K - J^\uparrow \underline{X}^K\| \leq C_A \sum_{i=0}^{K-1} C_B^i \|\psi(L) - J^\uparrow \psi(\underline{L}) J^\downarrow\| + C_B^K \|X^0 - J^\uparrow \underline{X}^0\|. \quad (95)$$

If $C_B \neq 1$, the geometric sum yields

$$\|X^K - J^\uparrow \underline{X}^K\| \leq C_A \frac{1 - C_B^K}{1 - C_B} \|\psi(L) - J^\uparrow \psi(\underline{L}) J^\downarrow\| + C_B^K \|X^0 - J^\uparrow \underline{X}^0\|. \quad (96)$$

Finally we recall from (3) that the initial input X^0 into the Laplace-transform message passing layers arises from the input features X via an initial Laplace-transform propagation step as

$$X^0 = \psi(L) X. \quad (97)$$

Thus we also find

$$J^\uparrow \underline{X}^0 = J^\uparrow \psi(\underline{L}) [J^\downarrow X]. \quad (98)$$

From this we may infer that

$$\|X^0 - J^\uparrow \underline{X}^0\| = \|\psi(L) X - J^\uparrow \psi(\underline{L}) J^\downarrow X\| \leq \|\psi(L) - J^\uparrow \psi(\underline{L})\| \cdot \|X\|. \quad (99)$$

Hence both summands in (96) may be bounded in terms of $\|\psi(L) - J^\uparrow \psi(\underline{L})\|$. Using

$$\|\psi(L) - J^\uparrow \psi(\underline{L})\| \leq \int_0^\infty \hat{\psi}(t) \|e^{-tL} - J^\uparrow e^{-t\underline{L}} J^\downarrow\| dt, \quad (100)$$

we have thus established the claim. \square

918 Finally we provide the Lemma used in the proof above:

919 **Lemma C.2.** *We have*

$$920 \quad [\Phi(J^\uparrow \underline{X}^\ell)]_{vu} = \sum_{a,b \in \underline{G}} J_{va}^\uparrow J_{ub}^\uparrow [\Phi(\underline{X}^\ell)]_{ab}. \quad (101)$$

921 *Proof.* Indeed, this follows because entries of J^\uparrow satisfy $J_{ua}^\uparrow \in \{0, 1\}$ and every node in G gets
 922 assigned information exactly from only one node in \underline{G} . Thus the sum $\sum_{a,b \in \underline{G}} J_{va}^\uparrow J_{ub}^\uparrow [[\Phi(\underline{X}^\ell)]_{ab}]$
 923 contains precisely *one* non-zero summand. This summand in turn, corresponds to those indices
 924 $a, b \in \underline{G}$ which are mapped to v and u respectively under J^\uparrow . Since entries in J^\uparrow are binary (and
 925 hence do not diminish or blow up signal if they are non-zero) the claim follows. \square

930 C.2 GRAPH LEVEL RESULTS AND PROOF OF THEOREM 3.1

931 It remains to translate the results of Theorems C.1 & C.1 to the graph level. To this end, we first note
 932 the following:

933 **Lemma C.2.** *With the graph level aggregation function Ω as in Definition B.2, we have*

$$934 \quad \Omega(\underline{X}) = \Omega(J^\uparrow \underline{X}). \quad (102)$$

935 *Proof.* This fact follows immediately from the fact that J^\uparrow assigns the information present at the
 936 supernode $a \in \underline{G}$ to all nodes u in the cluster within G that correspond to the supernode a , together
 937 with the fact that the node-weight μ_a of the node $a \in \underline{G}$ is the sum over all of the node weights μ_u of
 938 nodes u in the cluster within G that corresponds to the supernode a under coarse graining (c.f. also
 939 (117) in Appendix E). \square

940 Using the above lemma, we find

$$941 \quad \|\underline{F} - F\| = \|\Omega(\underline{X}^K) - \Omega(X^K)\| = \|\Omega(J^\uparrow \underline{X}^K) - \Omega(X^K)\|. \quad (103)$$

942 Using the 1-Lipschitzness of the aggregation function Ω then yields

$$943 \quad \|\Omega(J^\uparrow \underline{X}^K) - \Omega(X^K)\| \leq \|J^\uparrow \underline{X}^K - X^K\|. \quad (104)$$

944 Together with Theorems C.1 & C.1 we have thus proved:

945 **Theorem C.3.** *Let F and \underline{F} be the latent embeddings generated for a graph G its coarsified version
 946 \underline{G} by a (spectral or message passing) network employing Laplace transform propagation. With
 947 $\{\Psi_i(L) = \int_0^\infty \hat{\psi}_i(t) e^{-tL} dt\}_{i \in I}$ the collection of all Laplace-transform propagation matrices used
 948 within the network, we have $\|F - \underline{F}\| \leq C \cdot \max_i \left\{ \int_0^\infty |\hat{\psi}_i(t)| \cdot \|e^{-tL_\omega} - J^\uparrow e^{-tL} J^\downarrow\| \right\}$.*

949 Together with (112) we have here proved Theorem 3.1.

950 D ADDITIONAL EXPERIMENTAL CONSIDERATIONS

951 D.1 ADDITIONAL DETAILS ON INITIAL COARSE GRAINING EXPERIMENT

952 **Dataset:** The dataset we consider is the **QM7** dataset, introduced in Blum & Reymond (2009);
 953 Rupp et al. (2012). This dataset contains descriptions of 7165 organic molecules, each with up to
 954 seven heavy atoms, with all non-hydrogen atoms being considered heavy. A molecule is represented
 955 by its Coulomb matrix C^{Cmb} , whose off-diagonal elements

$$956 \quad C_{ij}^{\text{Cmb}} = \frac{Z_i Z_j}{|R_i - R_j|} \quad (105)$$

957 correspond to the Coulomb-repulsion between atoms i and j . We discard diagonal entries of Coulomb
 958 matrices; which would encode a polynomial fit of atomic energies to nuclear charge Rupp et al.
 959 (2012).

For each atom in any given molecular graph, the individual Cartesian coordinates R_i and the atomic charge Z_i are (in principle) also accessible individually. To each molecule an atomization energy - calculated via density functional theory - is associated. The objective is to predict this quantity. The performance metric is mean absolute error. Numerically, atomization energies are negative numbers in the range -600 to -2200 . The associated unit is $[kcal/mol]$.

Details on collapsing procedure: Again, we make use of the QM7 dataset Rupp et al. (2012) and its Coulomb matrix description

$$C_{ij}^{Cmb} = \frac{Z_i Z_j}{|R_i - R_j|} \quad (106)$$

of molecules. We modify (all) molecular graphs in QM7 by deflecting hydrogen atoms (H) out of their equilibrium positions towards the respective nearest heavy atom. This is possible since the QM7 dataset also contains the Cartesian coordinates of individual atoms. Edge weights between heavy atoms then remain the same, while Coulomb repulsions between H-atoms and respective nearest heavy atom increasingly diverge; as is evident from (106).

Given an original molecular graph G with node weights $\mu_i = Z_i$, the corresponding limit graph \underline{G} corresponds to a coarse grained description, where heavy atoms and surrounding H-atoms are aggregated into single super-nodes.

Mathematically, \underline{G} is obtained by removing all nodes corresponding to H-atoms from G , while adding the corresponding charges $Z_H = 1$ to the node-weights of the respective nearest heavy atom. Charges in (106) are modified similarly to generate the weight matrix \underline{W} .

On original molecular graphs, atomic charges are provided via one-hot encodings. For the graph of methane - consisting of one carbon atom with charge $Z_C = 6$ and four hydrogen atoms of charges $Z_H = 1$ - the corresponding node-feature-matrix is e.g. given as

$$X = \begin{pmatrix} 0 & 0 & \dots & 0 & 1 & 0 \dots \\ 1 & 0 & \dots & 0 & 0 & 0 \dots \\ 1 & 0 & \dots & 0 & 0 & 0 \dots \\ 1 & 0 & \dots & 0 & 0 & 0 \dots \\ 1 & 0 & \dots & 0 & 0 & 0 \dots \end{pmatrix} \quad (107)$$

with the non-zero entry in the first row being in the 6th column, in order to encode the charge $Z_C = 6$ for carbon.

The feature vector of an aggregated node represents charges of the heavy atom and its neighbouring H-atoms jointly.

Node feature matrices are translated as $\underline{X} = J^\downarrow X$. Applying J^\downarrow to one-hot encoded atomic charges yields (normalized) bag-of-word embeddings on \underline{G} : Individual entries of feature vectors encode how much of the total charge of the super-node is contributed by individual atom-types. In the example of methane, the limit graph \underline{G} consists of a single node with node-weight

$$\mu = 6 + 1 + 1 + 1 + 1 = 10. \quad (108)$$

The feature matrix

$$\underline{X} = J^\downarrow X \quad (109)$$

is a single row-vector given as

$$\underline{X} = \left(\frac{4}{10}, 0, \dots, 0, \frac{6}{10}, 0, \dots \right). \quad (110)$$

Experimental Setup: We randomly select 1500 molecules for testing and train on the remaining graphs. On QM7 we run experiments for 5 different random seeds and report mean and standard deviation. All experiments were performed on a single NVIDIA Quadro RTX 8000 graphics card.

Additional details on training and models: Typical GNN models are divided into **standard** architectures (GCN (Kipf & Welling, 2017), ChebNet (Defferrard et al., 2016), GATv2 (Brody et al., 2022)) GIN ((Xu et al., 2019) (together with its edge-weight enabled version (Hu et al., 2020) where

applicable)) and **multi-scale** architectures (UFGNet (Zheng et al., 2021), Lanczos (Liao et al., 2019)). Apart from UFGNet (already acting as a **pooling** layer) we also consider self-attention-pooling (Lee et al., 2019); both acting on the final layer (SAG) and as acting on the output of each individual layer, with resulting layer-wise features concatenated to produce the final embedding (SAG-M). All considered convolutional layers are incorporated into a two layer deep and fully connected graph convolutional architecture. In each hidden layer, we set the width (i.e. the hidden feature dimension) to $F_1 = F_2 = 64$. For ChebNet, we set the polynomial order to $K = 2$ as is customary. Lanczos uses 20 Lanczos iterations, as proposed in the original paper (Liao et al., 2019). UFGNet uses Haar wavelets. For all baselines, the standard mean-aggregation scheme is employed after the graph-convolutional layers to generate graph level features. Finally, predictions are generated via an MLP.

For the $\text{Spectral}_{\text{Res.}}$ -architecture, we set $\lambda = 1$ and build filters using the $k = 1$ and $k = 2$ atoms in $\Psi^{\text{Res}} = \{(z + \lambda)^{-k}\}_{k \in \mathbb{N}}$. For $\text{MPNN}_{\text{Res.}}$ we use the $k = 1$ atom as ψ .

For the $\text{Spectral}_{\text{Exp.}}$ -, we set $t = 1$ and build filters using the $k = 1$ and $k = 2$ atoms in $\Psi^{\text{Exp}} = \{e^{-(kt_0)z}\}_{k \in \mathbb{N}}$. For $\text{MPNN}_{\text{Res.}}$ we use the $k = 1$ atom as ψ .

As aggregation, we employ the graph level feature aggregation scheme discussed in Definition B.2, with node weights set to atomic charges of individual atoms. Predictions are then generated via a final MLP with the same specifications as the one used for baselines.

D.2 FURTHER DISCUSSIONS ON STABILITY RESULTS OF TABLE 2

We can further understand the small difference of latent embeddings using Theorem C.3. Let us use the shorthand notation

$$\eta(t) = \|e^{-Lt} - J^\dagger e^{-tL} J^\downarrow\|. \quad (111)$$

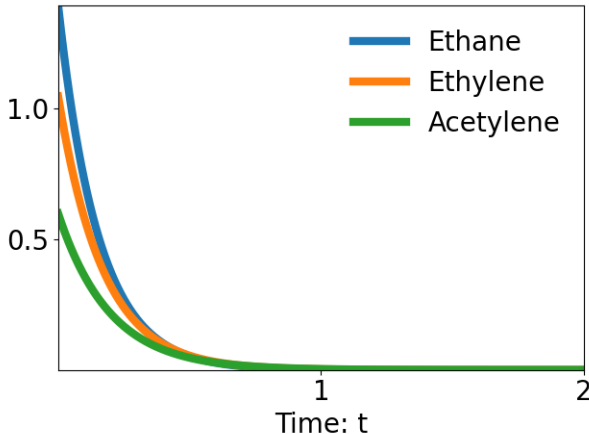


Figure 5: $\|e^{-Lt} - J^\dagger e^{-tL} J^\downarrow\|$ for molecules in QM7

Exemplarily considering exponential propagation matrices we have (with $t_k = k$) that $\int_0^\infty |\hat{\psi}_k(t)| \|e^{-tL} - J^\dagger e^{-tL} J^\downarrow\| dt = \|e^{-kL} - J^\dagger e^{-kL} J^\downarrow\|$, we thus have $\|F - \underline{F}\| \lesssim \max_{k \geq 1} |\eta(k)|$. Investigating the differences $\eta(t) = \|e^{-tL} - J^\dagger e^{-tL} J^\downarrow\|$ of diffusion flows, we find that $\eta(t)$ drops to zero fast, as exemplarily plotted in Fig. 5 for the first few molecules of QM7. In particular $|\eta(k)|_{k \geq 1} \approx 0$.

Using this as an upper bound in Theorem C.3 shows that embeddings F, \underline{F} of graphs describing the same molecule at different resolution scales are similar. This explains the ability to generalize between scales.

D.3 TRANSFERABILITY ON GRAPHS GENERATED VIA STOCHASTIC BLOCK MODELS

Stochastic Block Models: Stochastic block models (Holland et al., 1983) are generative models for random graphs that produce graphs containing strongly connected communities. In our experiments in this section, we consider a stochastic block model whose distributions is characterized by four parameters: The number of communities c_{number} determine how many (strongly connected) communities are present in the graph that is to be generated. The community size c_{size} determines the number of nodes belonging to each (strongly connected) community. The probability p_{connect} determines the probability that two nodes within the same community are connected by an edge. The probability p_{inter} determines the probabilities that two nodes in *different* communities are connected by an edge.

Experimental Setup: Since stochastic block models do not generate node-features, we equip each node with a randomly-generated unit-norm feature vector. Given such a graph G drawn from a stochastic block model, we then compute a version \underline{G} of this graph, where all communities are collapsed to single nodes. We then compare the feature vectors generated for G and \underline{G} . All experiments were performed on a single NVIDIA Quadro RTX 8000 graphics card.

Experiment: Varying the Connectivity within the Communities: We fix the parameters c_{number} , c_{size} and p_{inter} in our stochastic block model. We then vary the probability p_{connect} that two nodes within the same community are connected by an edge from $p_{\text{connect}} = 0$ to $p_{\text{connect}} = 1$. This corresponds to varying the connectivity within the communities from very sparse (or in fact no connectivity) to full connectivity (i.e. the community being a clique). In Figure 6 below, we then plot the difference of feature vectors generated by models for G and \underline{G} respectively. For each $p_{\text{connect}} \in [0, 1]$, results are averaged over 100 graphs randomly drawn from the same stochastic block model.

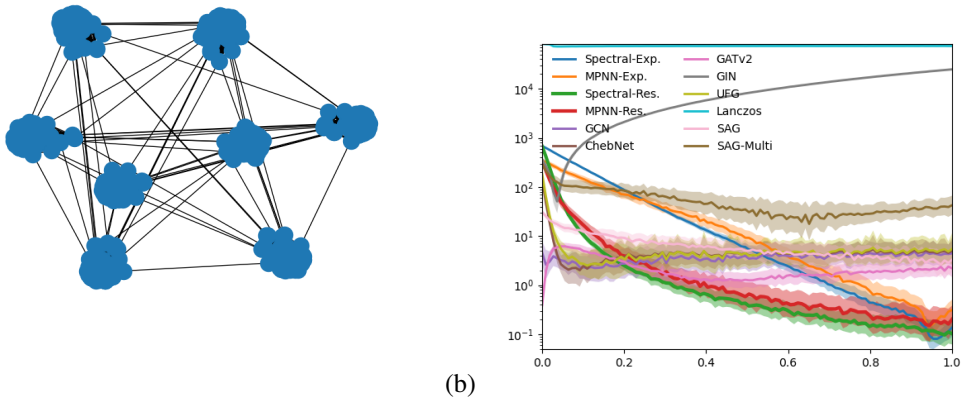
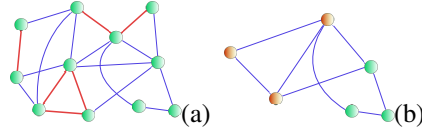


Figure 6: (a) Example Graph (b) Varying the parameter $p_{\text{connect}} \in [0, 1]$ for fixed $c_{\text{size}} = 60$, $p_{\text{inter}} = 2/c_{\text{size}}^2$ and $c_{\text{number}} = 12$.

We have chosen $p_{\text{inter}} = 2/c_{\text{size}}^2$ so that – on average – *clusters* are connected by two edges. The choice of two edges (as opposed to 1, 3, 4, 5, ...) between clusters is not important.

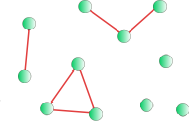
E FURTHER DISCUSSION OF THE SETTING OF COARSE-GRAINING GRAPHS

Let us consider graphs that contain clusters of nodes which are connected by significantly larger edge weights than those of edges outside of these clusters. From a diffusion perspective, information in a graph equalizes faster along edges with large weights.

Figure 7: (a) G (strongly connected) clusters in red (b) Coarse grained \underline{G}

In the limit where edge-weights within certain sub-graphs tend to infinity, information within these clusters equalizes immediately. Such clusters thus effectively behave as single nodes. We might thus consider a coarse grained graph \underline{G} where strongly connected clusters are fused together and represented only via single nodes. This naturally leads to the notion of graph coarsification, as first formalized and studied in (Loukas & Vandergheynst, 2018; Loukas, 2019).

In our case at hand the node set $\underline{\mathcal{G}}$ of the coarse grained graph \underline{G} is then given by the set of connected components in G_{cluster} (c.f. Fig 8). Edges $\underline{\mathcal{E}}$ are given by elements $(R, P) \in \underline{\mathcal{G}} \times \underline{\mathcal{G}}$ with non-zero accumulated edge weight $\underline{W}_{RP} = \sum_{r \in R} \sum_{p \in P} W_{rp}$. Node weights in \underline{G} are defined accordingly by aggregating as $\underline{\mu}_R = \sum_{r \in R} \mu_r$. To compare signals on these two graphs, we define intertwining operators J^\downarrow, J^\uparrow transferring information between G and \underline{G} : Let x be a scalar graph signal and let $\mathbf{1}_R$ be the vector that has 1 as entry for nodes $r \in R$ and is zero otherwise. Denote by u_R the entry of u at node $R \in \underline{\mathcal{G}}$. Projection J^\downarrow is then defined component-wise by evaluation at node $R \in \underline{\mathcal{G}}$ as the average of x over R : $(J^\downarrow x)_R = \langle \mathbf{1}_R, x \rangle / \underline{\mu}_R$. Going in the opposite direction, interpolation is defined as $J^\uparrow u = \sum_{R \in \underline{\mathcal{G}}} u_R \cdot \mathbf{1}_R$. In this setting, we have (c.f. the discussion below) that

Figure 8: G_{cluster}

$$\|e^{-tL} - J^\uparrow e^{-t\underline{L}} J^\downarrow\| \lesssim 1/w_{\text{high}}^{\min} \rightarrow 0 \quad \text{for any } t > 0. \quad (112)$$

Here $w_{\text{high}}^{\min} \gg 1$ denotes the minimal edge weight inside the strongly connected clusters in G . As the strength of the edge-weights in G_{cluster} tends to infinity, we have by (112) that also $\eta(t) = \|e^{-Lt} - J^\uparrow e^{-\underline{L}t} J^\downarrow\| \rightarrow 0$ for any $t > 0$. Thus (for $t > 0$) the diffusion process e^{-Lt} on G acts essentially as first projecting the input-signal to \underline{G} via J^\downarrow , then diffusing information on the coarse grained graph \underline{G} via $e^{-\underline{L}t}$ and finally interpolating back to the original graph G via J^\uparrow .

Following the proof in Koke (2026), we here illustrate

$$\|(L + Id)^{-1} - J^\uparrow (\underline{L} + Id)^{-1} J^\downarrow\| \lesssim 1/\lambda_1(L_{\text{high}}). \quad (113)$$

in this Appendix. Using (Koke, 2026, Corollary 3.18), this then also implies

$$\|e^{-tL} - J^\uparrow e^{-t\underline{L}} J^\downarrow\| \lesssim 1/w_{\text{high}}^{\min} \quad \text{for any } t > 0, \quad (114)$$

after noting the linear relation in scaling behaviour $\lambda_1(L_{\text{cluster}}) \sim w_{\text{high}}^{\min}$.

For convenience, we here explicitly restate the definitions leading up to this setting in full clarity and generality:

Definition E.1. Denote by $\underline{\mathcal{G}}$ the set of connected components in G_{high} . We give this set a graph structure as follows: Let R and P be elements of $\underline{\mathcal{G}}$ (i.e. connected components in G_{high}). We define the real number

$$\underline{W}_{RP} = \sum_{r \in R} \sum_{p \in P} W_{rp}, \quad (115)$$

with r and p nodes in the original graph G . We define the set of edges $\underline{\mathcal{E}}$ on \underline{G} as

$$\underline{\mathcal{E}} = \{(R, P) \in \underline{\mathcal{G}} \times \underline{\mathcal{G}} : \underline{W}_{RP} > 0\} \quad (116)$$

and assign \underline{W}_{RP} as weight to such edges. Node weights of limit nodes are defined similarly as aggregated weights of all nodes r (in G) contained in the component R as

$$\underline{\mu}_R = \sum_{r \in R} \mu_r. \quad (117)$$

In order to translate signals between the original graph G and the limit description \underline{G} , we need translation operators mapping signals from one graph to the other:

Definition E.2. Denote by $\mathbb{1}_R$ the vector that has 1 as entries on nodes r belonging to the connected (in G_{high}) component R and has entry zero for all nodes not in R . We define the down-projection operator J^\downarrow component-wise via evaluating at node R in \underline{G} as

$$(J^\downarrow x)_R = \langle \mathbb{1}_R, x \rangle / \mu_R. \quad (118)$$

The upsampling operator J^\uparrow is defined as

$$J^\uparrow u = \sum_R u_R \cdot \mathbb{1}_R; \quad (119)$$

where u_R is a scalar value (the component entry of u at $R \in \underline{G}$) and the sum is taken over all connected components in G_{high} .

Slightly extending established results in Koke (2026), we then have the following result:

Theorem E.3. We have

$$\|R_z(L) - J^\uparrow R_z(\underline{L}) J^\downarrow\| = \mathcal{O}\left(\frac{\|L_{\text{reg.}}\|}{\lambda_1(L_{\text{high}})}\right) \quad (120)$$

holds; with $\lambda_1(L_{\text{high}})$ denoting the first non-zero eigenvalue of L_{high} .

$$\lambda_{\max}(L_{\text{reg.}}) = \|L_{\text{reg.}}\|. \quad (121)$$

We here include the proof of this fact for convenience and self-containedness.

Proof. We will split the proof of this result into multiple steps. For $z < 0$ Let us denote by

$$R_z(L) = (L - zId)^{-1}, \quad (122)$$

$$R_z(L_{\text{high}}) = (L_{\text{high}} - zId)^{-1} \quad (123)$$

$$R_z(L_{\text{reg.}}) = (L_{\text{reg.}} - zId)^{-1} \quad (124)$$

the resolvents corresponding to L , L_{high} and $L_{\text{reg.}}$ respectively.

Our first goal is establishing that we may write

$$R_z(L) = [Id + R_z(L_{\text{high}})L_{\text{reg.}}]^{-1} \cdot R_z(L_{\text{high}}) \quad (125)$$

This will follow as a consequence of what is called the second resolvent formula Teschl (2014):

”Given self-adjoint operators A, B , we may write

$$R_z(A + B) - R_z(A) = -R_z(A)BR_z(A + B).” \quad (126)$$

In our case, this translates to

$$R_z(L) - R_z(L_{\text{high}}) = -R_z(L_{\text{high}})L_{\text{reg.}}R_z(L) \quad (127)$$

or equivalently

$$[Id + R_z(L_{\text{high}})L_{\text{reg.}}]R_z(L) = R_z(L_{\text{high}}). \quad (128)$$

Multiplying with $[Id + R_z(L_{\text{high}})L_{\text{reg.}}]^{-1}$ from the left then yields

$$R_z(L) = [Id + R_z(L_{\text{high}})L_{\text{reg.}}]^{-1} \cdot R_z(L_{\text{high}}) \quad (129)$$

as desired.

Hence we need to establish that $[Id + R_z(L_{\text{high}})L_{\text{reg.}}]$ is invertible for $z < 0$.

To establish a contradiction, assume it is not invertible. Then there is a signal x such that

$$[Id + R_z(L_{\text{high}})L_{\text{reg.}}]x = 0. \quad (130)$$

1242 Multiplying with $(L_{\text{high}} - zId)$ from the left yields

$$1243 \quad (L_{\text{high}} + L_{\text{reg.}} - zId)x = 0 \quad (131)$$

1245 which is precisely to say that

$$1246 \quad (L - zId)x = 0 \quad (132)$$

1247 But since L is a graph Laplacian, it only has non-negative eigenvalues. Hence we have reached our
1248 contradiction and established

$$1249 \quad R_z(L) = [Id + R_z(L_{\text{high}})L_{\text{reg.}}]^{-1} R_z(L_{\text{high}}). \quad (133)$$

1252 Our next step is to establish that

$$1253 \quad R_z(L_{\text{high}}) \rightarrow \frac{P_0^{\text{high}}}{-z}, \quad (134)$$

1256 where P_0^{high} is the spectral projection onto the eigenspace corresponding to the lowest lying eigenvalue
1257 $\lambda_0(L_{\text{high}}) = 0$ of L_{high} . Indeed, by the spectral theorem for finite dimensional operators (c.f. e.g.
1258 Teschl (2014)), we may write

$$1259 \quad R_z(\Delta_{\text{high}}) \equiv (L_{\text{high}} - zId)^{-1} = \sum_{\lambda \in \sigma(L_{\text{high}})} \frac{1}{\lambda - z} \cdot P_{\lambda}^{\text{high}}. \quad (135)$$

1262 Here $\sigma(L_{\text{high}})$ denotes the spectrum (i.e. the collection of eigenvalues) of L_{high} and the
1263 $\{P_{\lambda}^{\text{high}}\}_{\lambda \in \sigma(L_{\text{high}})}$ are the corresponding (orthogonal) eigenprojections onto the eigenspaces of the
1264 respective eigenvalues. Thus we find

$$1266 \quad \left\| R_z(L_{\text{high}}) - \frac{P_0^{\text{high}}}{-z} \right\| = \left\| \sum_{0 < \lambda \in \sigma(L_{\text{high}})} \frac{1}{\lambda - z} \cdot P_{\lambda}^{\text{high}} \right\|; \quad (136)$$

1269 where the sum on the right hand side now excludes the eigenvalue $\lambda = 0$.

1271 Using orthonormality of the spectral projections, the fact that $z < 0$ and monotonicity of $1/(\cdot + |z|)$
1272 we find

$$1273 \quad \left\| R_z(L_{\text{high}}) - \frac{P_0^{\text{high}}}{-z} \right\| = \frac{1}{\lambda_1(\Delta_{\text{high}}) + |z|}. \quad (137)$$

1275 Here $\lambda_1(L_{\text{high}})$ is the first non-zero eigenvalue of (L_{high}) .

1276 Non-zero eigenvalues scale linearly with the weight scale since we have

$$1278 \quad \lambda(S \cdot L) = S \cdot \lambda(L) \quad (138)$$

1279 for any graph Laplacian (in fact any matrix) L with eigenvalue λ . Thus we have

$$1281 \quad \left\| R_z(L_{\text{high}}) - \frac{P_0^{\text{high}}}{-z} \right\| = \frac{1}{\lambda_1(L_{\text{high}}) + |z|} \leq \frac{1}{\lambda_1(L_{\text{high}})} \rightarrow 0 \quad (139)$$

1284 as $\lambda_1(L_{\text{high}}) \rightarrow \infty$.

1286 Our next task is to use this result in order to bound the difference

$$1287 \quad I := \left\| \left[Id + \frac{P_0^{\text{high}}}{-z} L_{\text{reg.}} \right]^{-1} \frac{P_0^{\text{high}}}{-z} - [Id + R_z(L_{\text{high}})L_{\text{reg.}}]^{-1} R_z(L_{\text{high}}) \right\|. \quad (140)$$

1291 To this end we first note that the relation

$$1292 \quad [A + B - zId]^{-1} = [Id + R_z(A)B]^{-1} R_z(A) \quad (141)$$

1294 provided to us by the second resolvent formula, implies

$$1295 \quad [Id + R_z(A)B]^{-1} = Id - B[A + B - zId]^{-1}. \quad (142)$$

1296 Thus we have

$$1297 \quad \left\| [Id + R_z(L_{high})L_{reg.}]^{-1} \right\| \leq 1 + \|L_{reg.}\| \cdot \|R_z(L)\| \quad (143)$$

$$1299 \quad \leq 1 + \frac{\|L_{reg.}\|}{|z|}. \quad (144)$$

1301 With this, we have

$$1302 \quad \left\| \left[Id + \frac{P_0^{high}}{-z} L_{reg.} \right]^{-1} \cdot \frac{P_0^{high}}{-z} - R_z(L) \right\| \quad (145)$$

$$1303 \quad = \left\| \left[Id + \frac{P_0^{high}}{-z} L_{reg.} \right]^{-1} \cdot \frac{P_0^{high}}{-z} - [Id + R_z(L_{high})L_{reg.}]^{-1} \cdot R_z(L_{high}) \right\| \quad (146)$$

$$1307 \quad \leq \left\| \frac{P_0^{high}}{-z} \right\| \cdot \left\| \left[Id + \frac{P_0^{high}}{-z} L_{reg.} \right]^{-1} - [Id + R_z(L_{high})L_{reg.}]^{-1} \right\| + \left\| \frac{P_0^{high}}{-z} - R_z(L_{high}) \right\| \cdot \left\| [Id + R_z(L_{high})L_{reg.}]^{-1} \right\| \quad (147)$$

$$1311 \quad \leq \frac{1}{|z|} \left\| \left[Id + \frac{P_0^{high}}{-z} L_{reg.} \right]^{-1} - [Id + R_z(L_{high})L_{reg.}]^{-1} \right\| + \left(1 + \frac{\|L_{reg.}\|}{|z|} \right) \cdot \frac{1}{\lambda_1(L_{high})}. \quad (148)$$

1312 Hence it remains to bound the left hand summand. For this we use the following fact (c.f. Horn & Johnson (2012), Section 5.8. "Condition numbers: inverses and linear systems"):

1313 Given square matrices A, B, C with $C = B - A$ and $\|A^{-1}C\| < 1$, we have

$$1314 \quad \|A^{-1} - B^{-1}\| \leq \frac{\|A^{-1}\| \cdot \|A^{-1}C\|}{1 - \|A^{-1}C\|}. \quad (149)$$

1315 In our case, this yields (together with $\|P_0^{high}\| = 1$) that

$$1316 \quad \left\| \left[Id + P_0^{high}/(-z) \cdot L_{reg.} \right]^{-1} - [Id + R_z(L_{high})L_{reg.}]^{-1} \right\| \quad (150)$$

$$1317 \quad \leq \frac{(1 + \|L_{reg.}\|/|z|)^2 \cdot \|L_{reg.}\| \cdot \left\| \frac{P_0^{high}}{-z} - R_z(L_{high}) \right\|}{1 - (1 + \|L_{reg.}\|/|z|) \cdot \|L_{reg.}\| \cdot \left\| \frac{P_0^{high}}{-z} - R_z(L_{high}) \right\|} \quad (151)$$

1318 For S_{high} sufficiently large, we have

$$1319 \quad \left\| -P_0^{high}/z - R_z(L_{high}) \right\| \leq \frac{1}{2(1 + \|L_{reg.}\|/|z|)} \quad (152)$$

1320 so that we may estimate

$$1321 \quad \left\| \left[Id + L_{reg.} \frac{P_0^{high}}{-z} \right]^{-1} - [Id + L_{reg.} R_z(L_{high})]^{-1} \right\| \quad (153)$$

$$1322 \quad \leq 2 \cdot (1 + \|L_{reg.}\|) \cdot \left\| \frac{P_0^{high}}{-z} - R_z(L_{high}) \right\| \quad (154)$$

$$1323 \quad = 2 \frac{1 + \|L_{reg.}\|/|z|}{\lambda_1(L_{high})} \quad (155)$$

1324 Thus we have now established

$$1325 \quad \left\| \left[Id + \frac{P_0^{high}}{-z} L_{reg.} \right]^{-1} \cdot \frac{P_0^{high}}{-z} - R_z(L) \right\| = \mathcal{O} \left(\frac{\|L_{reg.}\|}{\lambda_1(L_{high})} \right). \quad (156)$$

1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403

Hence we are done with the proof, as soon as we can establish

$$\left[-zId + P_0^{high} L_{reg.}\right]^{-1} P_0^{high} = J^\uparrow R_z(\underline{L}) J^\downarrow, \quad (157)$$

with $J^\uparrow, \underline{L}, J^\downarrow$ as defined above. To this end, we first note that

$$J^\uparrow \cdot J^\downarrow = P_0^{high} \quad (158)$$

and

$$J^\downarrow \cdot J^\uparrow = Id_{\underline{G}}. \quad (159)$$

Indeed, the relation (158) follows from the fact that the eigenspace corresponding to the eigenvalue zero is spanned by the vectors $\{\mathbb{1}_R\}_R$, with $\{R\}$ the connected components of G_{high} . Equation (159) follows from the fact that

$$\langle \mathbb{1}_R, \mathbb{1}_R \rangle = \mu_R. \quad (160)$$

With this we have

$$\left[Id + P_0^{high} L_{reg.}\right]^{-1} P_0^{high} = \left[Id + J^\uparrow J^\downarrow L_{reg.}\right]^{-1} J^\uparrow J^\downarrow. \quad (161)$$

To proceed, set

$$\underline{x} := F^\downarrow x \quad (162)$$

and

$$\mathcal{X} = \left[P_0^{high} L_{reg.} - zId\right]^{-1} P_0^{high} x. \quad (163)$$

Then

$$\left[P_0^{high} L_{reg.} - zId\right] \mathcal{X} = P_0^{high} x \quad (164)$$

and hence $\mathcal{X} \in \text{Ran}(P_0^{high})$. Thus we have

$$J^\uparrow J^\downarrow (L_{reg.} - zId) J^\uparrow J^\downarrow \mathcal{X} = J^\uparrow J^\downarrow x. \quad (165)$$

Multiplying with J^\downarrow from the left yields

$$J^\downarrow (L_{reg.} - zId) J^\uparrow J^\downarrow \mathcal{X} = J^\downarrow x. \quad (166)$$

Thus we have

$$(J^\downarrow L_{reg.} J^\uparrow - zId) J^\uparrow J^\downarrow \mathcal{X} = J^\downarrow x. \quad (167)$$

This – in turn – implies

$$J^\uparrow J^\downarrow \mathcal{X} = \left[J^\downarrow L_{reg.} J^\uparrow - zId\right]^{-1} J^\downarrow x. \quad (168)$$

Using

$$P_0^{high} \mathcal{X} = \mathcal{X}, \quad (169)$$

we then have

$$\mathcal{X} = J^\uparrow \left[J^\downarrow L_{reg.} J^\uparrow - zId\right]^{-1} J^\downarrow x. \quad (170)$$

We have thus concluded the proof if we can prove that $J^\downarrow L_{reg.} J^\uparrow$ is the Laplacian corresponding to the graph \underline{G} defined in Definition E.1. But this is a straightforward calculation. \square

As a corollary, one finds

Corollary E.4. *We have*

$$R_z(L)^k \rightarrow J^\uparrow R^k(\underline{L}) J^\downarrow \quad (171)$$

Proof. This follows directly from the fact that

$$J^\downarrow J^\uparrow = Id_{\underline{G}}. \quad (172)$$

\square