Full-Spectrum Graph Neural Networks: Expressive and Scalable

Xiaohan Wang*, Deyu Bo*, Longlong Li, Kelin Xia

School of Physical and Mathematical Sciences, Nanyang Technological University

Motivation and Contributions

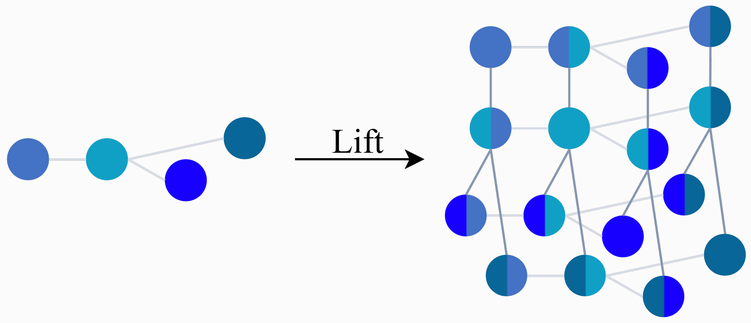
Classical spectral GNNs apply univariate filters over eigenvalues, which fundamentally restricts their expressive power to the 1-WL.

We propose **Full-Spectrum GNNs** (FSpecGNNs) that lift graph signals to the node-pair domain and extend spectral filtering from eigenvalues to eigenvalue pairs, enabling more expressive spectral GNNs.

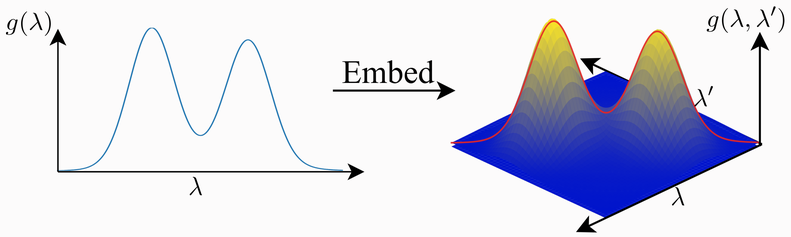
- **Expressivity**: Match the expressivity of Local 2-GNN.
- **Scalability**: Comparable to classical spectral GNNs.

Full-Spectrum Filtering

Signal Domain Lifted:



Filter From Univariate to Bivariate:



Mathematical Formulation:

$$g(L)x, \quad x \in \mathbb{R}^{|V|}$$

$$\Updownarrow$$

$$U\big(g_\lambda \odot (U^\top x)\big) = \sum_i g(\lambda_i)\, u_i u_i^\top x$$

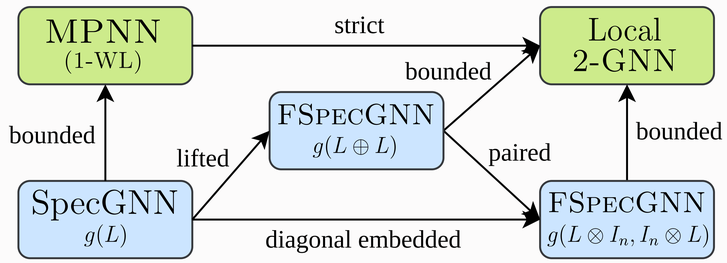
$$\Downarrow$$

$$U\big(G_\lambda \odot (U^\top \varepsilon U)\big) U^\top = \sum_{i,j} g(\lambda_i, \lambda_j)\, u_i u_i^\top \varepsilon\, u_j u_j^\top$$

$$\Updownarrow$$

$$g(L \otimes I, I \otimes L)\, \varepsilon, \quad \varepsilon \in \mathbb{R}^{|V|^2}$$

Expressivity Hierarchy

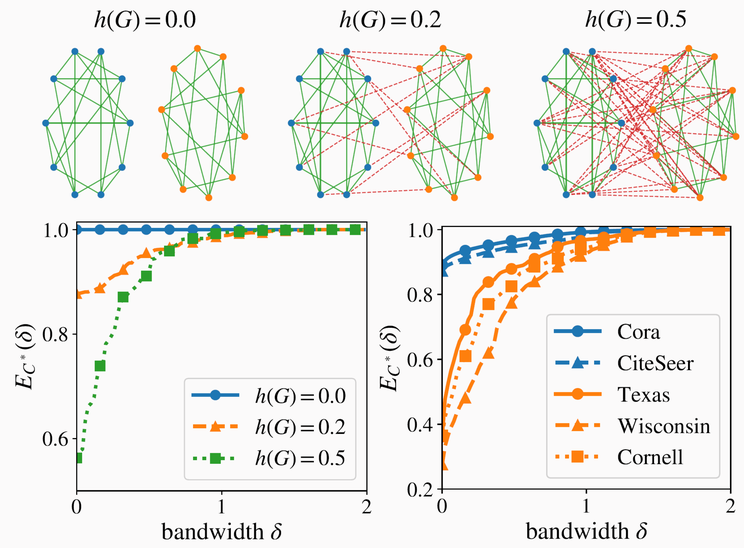


Experimental Validations:

Task / Model	Graph-level						Node-level						Edge-level					
MPNN	.358	.208	.188	.146	.261	.205	.600	.413	.300	.207	.318	.237	-	-	-	-	-	-
Spec. Inv. GNN	.072	.072	.089	.089	.060	.099	-	-	-	-	-	-	-	-	-	-	-	-
Subgraph GNN	.010	.020	.024	.046	.007	.027	.003	.005	.092	.082	.050	.073	.001	.003	.090	.096	.038	.065
Local 2-GNN	.008	.011	.017	.034	.007	.016	.002	.005	.010	.023	.004	.015	.001	.005	.010	.019	.005	.014
Local 2-FGNN	.003	.004	.010	.020	.003	.010	.004	.006	.012	.021	.004	.014	.003	.006	.012	.022	.005	.012
FSpecGNN	.005	.018	.024	.033	.006	.032	.003	.009	.020	.029	.009	.021	.003	.008	.026	.045	.009	.033

Expressive Heterophilic Graph Learning

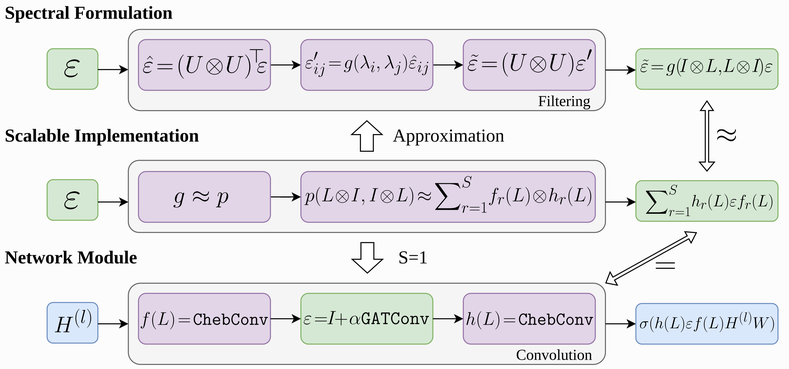
Heterophilic graph learning is inherently a second-order problem: we show that, for heterophilic graph learning, the optimal convolutions cannot be represented by classical spectral filters $g(L)$, but can be captured by full-spectrum filters $g(L \otimes I, I \otimes L)$.



FSpecGNNs outperform spectral GNNs on heterophilic datasets:

Model	Texas	Wisconsin	Chameleon	Squirrel	Roman	Minesweeper	Tolokers	Questions
ChebNet	36.76±6.76	33.17±7.83	25.65±3.56	22.58±2.56	45.32±0.65	86.34±0.19	69.88±0.26	48.55±1.04
ChebNetII	48.44±10.87	41.33±9.25	33.48±4.59	30.80±2.65	55.06±0.32	74.49±3.76	69.37±0.60	63.99±0.32
GPRGNN	48.55±7.00	40.79±3.62	30.44±4.29	24.33±2.68	55.48±1.30	86.68±0.32	67.05±0.97	53.76±0.41
JacobiConv	50.17±7.87	46.08±9.08	26.57±3.60	23.15±4.47	52.92±0.70	87.40±0.13	70.24±0.18	55.68±0.54
BernNet	52.14±8.09	49.33±8.21	29.45±3.22	25.94±3.35	55.30±0.41	76.64±0.29	69.31±0.69	65.41±0.33
FSpecGNN(Cheb)	55.32±4.57	49.87±8.29	33.09±0.92	**39.57**±0.67	54.35±0.73	**88.30**±0.82	**76.89**±0.91	75.87±0.37
FSpecGNN(ChebII)	**57.05**±2.20	50.00±5.42	**39.60**±2.77	37.70±0.63	**56.26**±1.08	84.25±0.65	76.37±0.67	77.00±0.37
FSpecGNN(Bern)	56.13±0.46	**54.58**±9.47	37.91±3.94	37.59±1.32	56.16±1.19	84.17±1.01	74.50±0.69	**77.11**±0.26

Scalable Architecture



FSpecGNN is a second-order architecture and thus naturally incurs higher computational cost. We improve scalability via a tensorial decomposition, where the filter is implemented as a sum of tensor products of classical graph convolution modules. The rank (S) controls the expressivity-scalability trade-off.

Scalability Validations

FSpecGNNs achieve lower time and memory overhead than spatial second-order models in substructure-counting tasks.

Model	Avg. Time (s/it)	Max GPU Mem (MB)
MPNN	6.162	9688
Subgraph NN	6.145	11228
Local 2-GNN	7.569	11532
Local 2-FGNN	4.856	19886
FSpecGNN	1.111	9532

FSpecGNNs achieve time and memory costs comparable to classical spectral models on heterophilic graph benchmarks.

Average runtime per run in seconds on heterophilic datasets. "–" indicates out-of-memory.

Model	Texas	Wisconsin	Chameleon	Squirrel	Roman	Minesweeper	Tolokers	Questions
GPRGNN	1.06	0.90	1.05	0.95	0.99	1.19	1.30	1.55
JacobiConv	1.10	1.13	1.08	1.09	1.24	1.98	1.23	1.65
Subgraph GNN	15.70	33.16	–	–	–	–	–	–
Local 2-GNN	23.20	51.24	–	–	–	–	–	–
ChebNet	0.94	0.97	1.51	1.93	1.66	1.90	2.32	2.45
FSpecGNN(Cheb)	2.04	2.10	2.55	2.51	3.88	3.83	3.96	3.40
ChebNetII	2.07	1.90	2.55	2.73	2.96	3.07	3.63	3.49
FSpecGNN(ChebII)	2.41	2.78	3.17	3.27	4.31	4.34	4.80	5.16
BernNet	2.21	2.15	2.21	2.24	2.53	3.25	3.85	4.13
FSpecGNN(Bern)	2.17	2.48	2.69	3.35	3.05	3.52	4.66	6.01

GPU memory usage (MiB) on heterophilic datasets. "–" indicates out-of-memory.

Model	Texas	Wisconsin	Chameleon	Squirrel	Roman	Minesweeper	Tolokers	Questions
GPRGNN	646	648	652	688	954	750	876	1160
JacobiConv	648	648	652	672	982	752	876	1172
Subgraph GNN	1724	2896	–	–	–	–	–	–
Local 2-GNN	1830	3128	–	–	–	–	–	–
ChebNet	534	534	558	594	864	654	742	1150
FSpecGNN(Cheb)	646	648	692	760	1130	754	976	1242
ChebNetII	646	648	660	716	954	750	914	1230
FSpecGNN(ChebII)	650	658	698	752	1062	756	998	1244
BernNet	648	648	658	698	1058	756	954	1254
FSpecGNN(Bern)	652	658	694	758	1102	774	1040	1276

xiaohan005@e.ntu.edu.sg