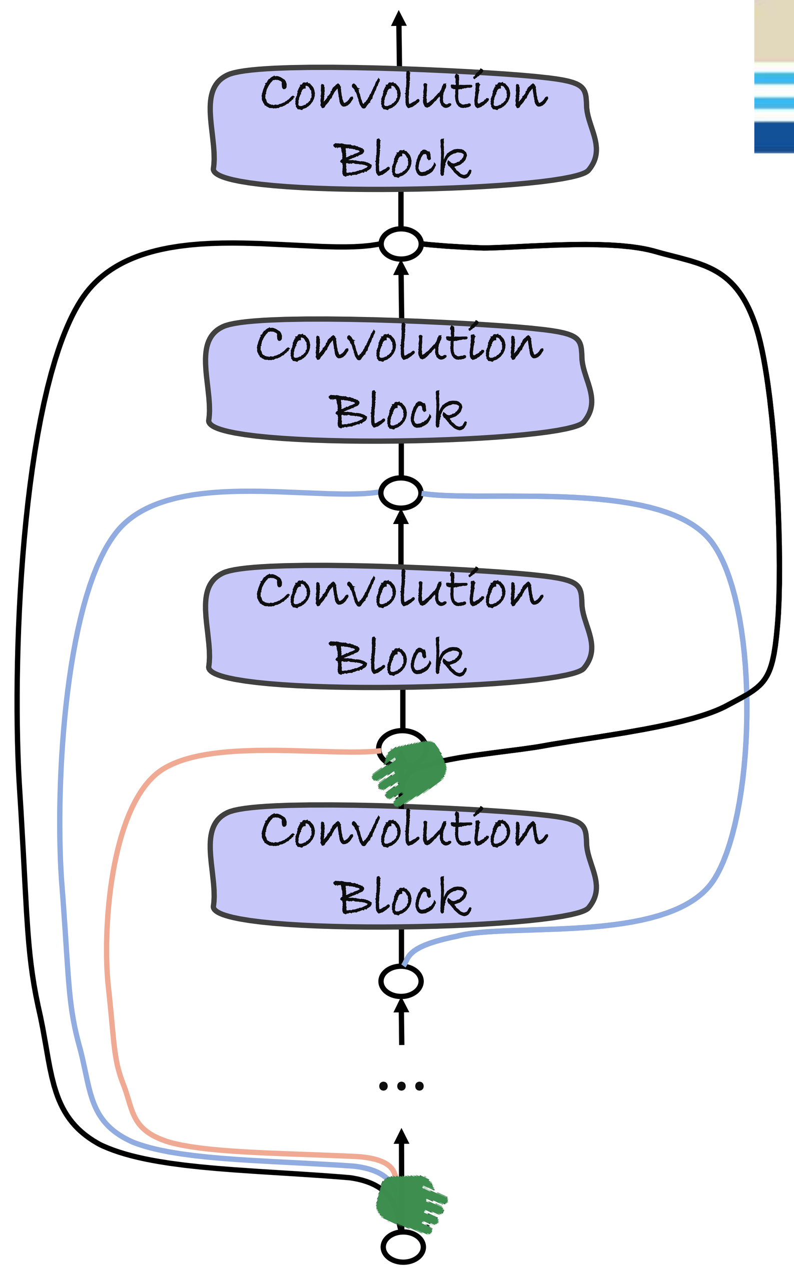


GitHub Repo

zhewei@ruc.edu.cn
Contact

Clenshaw Graph Neural Networks

Yuhe Guo and Zhewei Wei*

Background

Spatial GNNs: $H \mapsto \tilde{P}HW^{(\ell)}$

Residual connections equipped spatial GNNs
- ***Categorization***: Vanilla/initial/dense residuals.
- ***Motivations***: Tackling model degradation; enhance expressiveness.

Spectral GNNs: $x \mapsto \mathbf{U} \cdot \mathrm{diag}\{\theta_1, \ldots, \theta_N\} \cdot \mathbf{U}^T \cdot x$ See Footnote.

Polynomial Filters: $x \mapsto \mathbf{U} \cdot \mathrm{diag}\{h(\lambda_1), \ldots, h(\lambda_N)\} \cdot \mathbf{U}^T \cdot x$

$x \mapsto h(\tilde{P})x$ or $b(\tilde{P})x$ — Computed Localized !

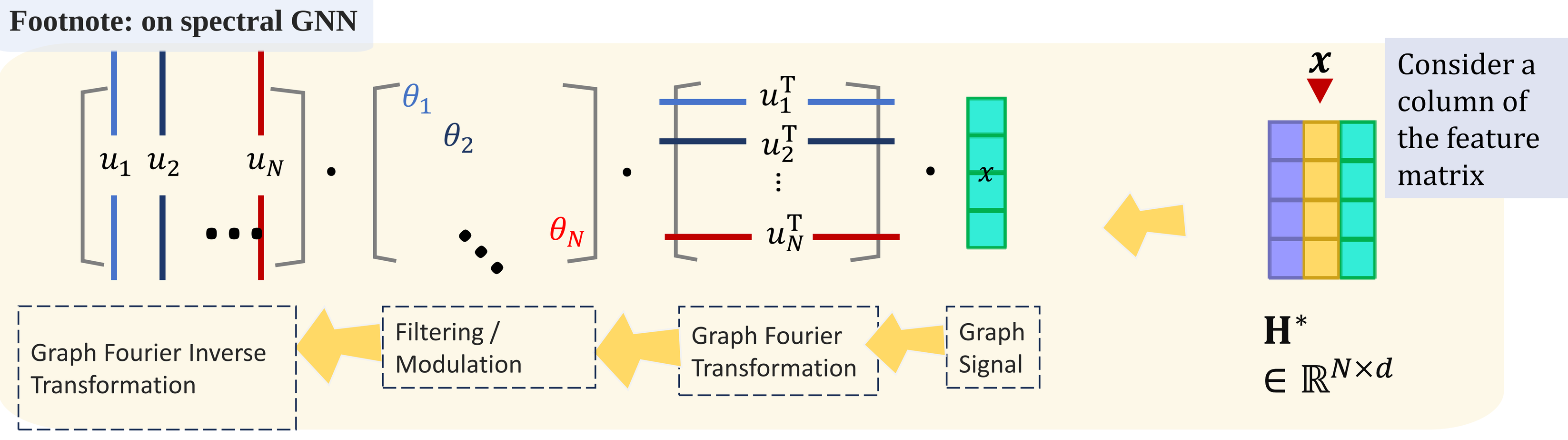
$$\sum_{k=0}^{K} \alpha_k g_k(\tilde{P})$$

Feature I: **Arbitrary** coefficients.

Feature II: Utilizing **polynomial basis**.

Our contribution
- Simple & novel residuals;
- Equip a spatial GCN to simulate a full-featured polynomial filter.
- Gap between spatial & spectral GNNs.

Footnote: on spectral GNN



TL;DR:
- A small modification on GCNII allows for *arbitrary* polynomial filter expressiveness.
- Underlying is a correspondence between *evaluating method* and *convolution stack*.

A Warm-up

GCNII: $\mathbf{H}^{(\ell+1)} = \sigma\left(\left((1-\alpha)\tilde{\mathbf{P}}\mathbf{H}^{(\ell)} + \alpha\mathbf{H}^*\right)\left((1-\theta_\ell)\mathbf{W}^{(\ell)} + \theta_\ell \mathbf{I}\right)\right)$

$\mathbf{H}^{(\ell+1)} = (1-\alpha)\tilde{\mathbf{P}}\mathbf{H}^{(\ell)} + \alpha\mathbf{H}^*$ — **Unfold it !** — $\mathbf{H}^{(K)} = \sum_{\ell=0}^{K} \hat{\alpha}_\ell \tilde{\mathbf{P}}^\ell \mathbf{H}^*$

$$\hat{\alpha}_\ell = \begin{cases} \alpha(1-\alpha)^\ell, & \ell < K \\ (1-\alpha)^K, & \ell = K \end{cases}$$

Free the coefficients!

$\mathbf{H}^{(\ell+1)} = \tilde{\mathbf{P}}\mathbf{H}^{(\ell)} + \alpha_\ell \mathbf{H}^*$ — **Unfold it !** — $\mathbf{H}^{(K)} = \sum_{\ell=0}^{K} \alpha_{K-\ell} \tilde{\mathbf{P}}^\ell \mathbf{H}^*$

Unfolding Process (1) corresponds to *Horner's Method*! An *iterative* method for numerical evaluating.

Horner's Method: Evaluates the value of $p(x)$ at x_0 where $p(x) = a_0 + a_1 x + \cdots + a_n x^n$.

$$\begin{aligned} b_n &:= a_n, \\ b_{n-1} &:= a_{n-1} + b_n x_0, \\ &\vdots \\ b_0 &:= a_0 + b_1 x_0, \\ \Rightarrow p(x_0) &:= b_0. \end{aligned}$$

Unfolding Process

$$\begin{aligned} \mathrm{H}^{(0)} &= \alpha_0 \mathrm{H}^*, \\ \mathrm{H}^{(1)} &= \tilde{\mathrm{P}}\left(\alpha_0 \mathrm{H}^*\right) + \alpha_1 \mathrm{H}^*, \\ \mathrm{H}^{(2)} &= \tilde{\mathrm{P}}\left(\tilde{\mathrm{P}}\left(\alpha_0 \mathrm{H}^*\right) + \alpha_1 \mathrm{H}^*\right) + \alpha_2 \mathrm{H}^*, \\ &\vdots \\ \mathrm{H}^{(K)} &= \tilde{\mathrm{P}}\left(\cdots\left(\tilde{\mathrm{P}}\left(\tilde{\mathrm{P}}\left(\alpha_0 \mathrm{H}^*\right) + \alpha_1 \mathrm{H}^*\right) + \alpha_2 \mathrm{H}^*\right)\cdots\right) + \alpha_K \mathrm{H}^* \\ \Rightarrow \mathrm{H}^{(K)} &= \alpha_K \mathrm{H}^* + \alpha_{K-1}\tilde{\mathrm{P}}\mathrm{H}^* + \cdots + \alpha_0 \tilde{\mathrm{P}}^K \\ &= \sum_{\ell=0}^{K} \alpha_{K-\ell} \tilde{\mathrm{P}}^\ell \mathrm{H}^*. \end{aligned}$$

Model
ClenshawGCN

A new **negative** second-order residual.

$$\mathbf{H}^{(\ell+1)} = \sigma\left(\left(2\tilde{\mathbf{P}}\mathbf{H}^{(\ell)} - \mathbf{H}^{(\ell-1)} + \alpha_\ell \mathbf{H}^*\right)\left((1-\theta_\ell)\mathbf{W}^{(\ell)} + \theta_\ell \mathbf{I}\right)\right)$$

- Two shortcuts.
- Allows for simulating ***arbitrary*** polynomial filter upon **Chebyshev basis** (the second kind).
- The **convolution process** mimics **Clenshaw's algorithm**.
- Inherits strengths from both spatial GNNs and spectral GNNs.

Clenshaw's Algorithm: Evaluates the value of $p(x)$ at x_0 where $p(x) = a_0 + a_1 U_1(x) + \cdots + a_n U_n(x)$, and U_k is the k-th Chebyshev polynomial (the 2nd kind).

$$\begin{aligned} b_{n+2}(x_0) &:= 0 \\ b_{n+1}(x_0) &:= 0 \\ b_k(x_0) &:= a_k + 2x_0 b_{k+1}(x_0) - b_{k+2}(x_0) \\ &(k = n, n-1, \ldots, 0) \end{aligned}$$

Unfolding Process of ClenshawGCN

$$\begin{aligned} \mathbf{H}^{(-2)} &= \mathbf{0} \\ \mathbf{H}^{(-1)} &= \mathbf{0} \\ \mathbf{H}^{(\ell+1)} &= 2\tilde{\mathbf{P}}\mathbf{H}^{(\ell)} - \mathbf{H}^{(\ell-1)} + \alpha_\ell \mathbf{H}^{(0)} \\ &(i = 0, 1, \ldots, K) \end{aligned}$$

(a) Comparison with other models equipped with different kinds of residual connections. Mean classification accuracies *(± standard derivations)* of twenty random splits are displayed. Besides the ClenshawGCN, all the results are taken directly from Luan et al. [28] and Lim et al. [26].

Datasets	Chameleon	Squirrel	Actor	Texas	Cornell	Cora	Citeseer	PubMed
$\lvert\mathcal{V}\rvert$	2,277	5,201	7,600	183	183	2,708	3,327	19,717
MLP	46.59±1.84	31.01±1.18	40.18±0.55	86.81±2.24	84.15±3.05	76.89±0.97	76.52±0.89	86.14±0.25
GCN	60.81±2.95	45.87±0.88	33.26±1.15	76.97±3.97	65.78±4.16	87.18±1.12	79.85±0.78	86.79±0.31
GCNII	63.44±0.85	41.96±1.02	36.89±0.95	80.46±5.91	84.26±2.13	88.46±0.82	79.97±0.65	89.94±0.31
H_2GCN	52.30±0.48	30.39±1.22	38.85±1.17	85.90±3.53	86.23±4.71	87.52±0.61	79.97±0.69	87.78±0.28
MixHop	36.28±10.22	24.55±2.60	33.13±2.40	76.39±7.66	60.33±28.53	65.65±11.31	49.52±13.35	87.04±4.10
GCN+JK	64.68±2.85	53.40±1.90	32.72±2.62	80.66±1.91	66.56±13.82	86.90±1.51	73.77±1.85	90.09±0.68
ClenshawGCN	**69.44±2.06**	**62.14±1.65**	**42.08±1.99**	**93.36±2.35**	**92.46±3.72**	**88.90±1.26**	**80.34±1.26**	**91.99±0.41**

v.s. Representative residual-enhanced spatial GNNs.

Large margin on hete-graphs

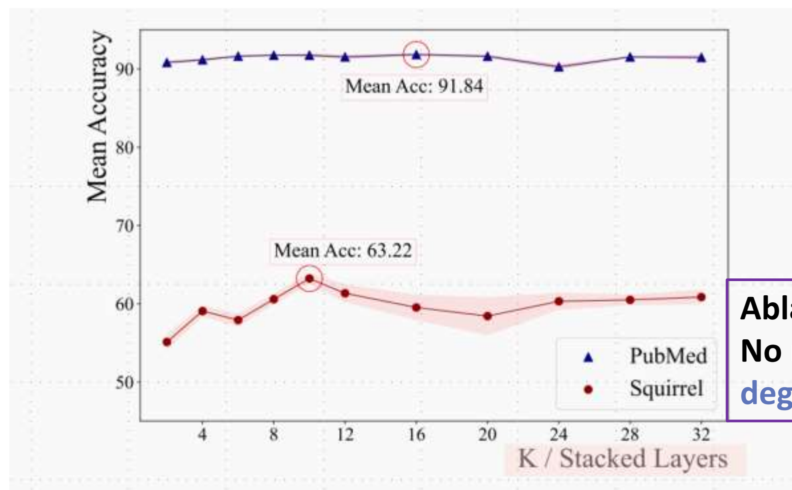
60%/20%/20%

Datasets	Chameleon	Squirrel	Actor	Texas	Cornell	Cora	Citeseer	PubMed
$\lvert\mathcal{V}\rvert$	2,277	5,201	7,600	183	183	2,708	3,327	19,717
ChebNet	59.51±1.25	40.81±0.42	37.42±0.58	86.28±2.62	83.91±2.17	87.32±0.92	79.33±0.57	87.82±0.24
ARMA	60.21±1.00	36.27±0.62	37.67±0.54	83.97±3.77	85.62±2.13	87.13±0.80	80.04±0.55	86.93±0.24
APPNP	52.15±1.79	35.71±0.78	39.76±0.49	90.64±1.70	91.52±1.81	88.16±0.74	80.47±0.73	88.13±0.33
GPRGNN	67.49±1.38	50.43±1.89	39.91±0.62	92.91±1.32	91.57±1.96	88.54±0.67	80.13±0.84	88.46±0.31
BernNet	68.53±1.68	51.39±0.92	41.71±1.12	92.62±1.37	92.13±1.64	88.51±0.92	80.08±0.75	88.51±0.39
ChebNetII	**71.37±1.01**	57.72±0.59	41.75±1.07	93.28±1.47	92.30±1.48	88.71±0.93	**80.53±0.79**	88.93±0.29
ClenshawGCN	69.44±0.92	**62.14±0.70**	**42.08±0.86**	**93.36±0.99**	**92.46±1.64**	**88.90±0.59**	80.34±0.57	**91.99±0.17**

v.s. SoTA spectral GNNs

Competitive !

Experiments
Node classification & Ablations

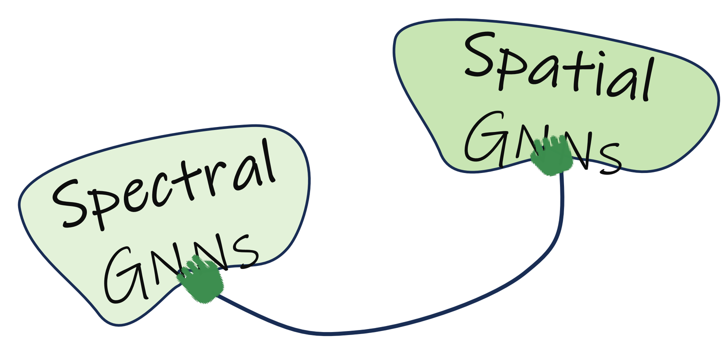


Ablation: No model degradation

Ablation: Effectiveness of interwining nonlinear tranformations into polynomial filters.

Models	ClenshawGCN	ClenshawGCN(-Act)	Clenshaw(-Act-W)
Squirrel	**62.14 ± 1.65**	61.55 ± 1.42	56.92 ± 2.13
Chameleon	69.45 ± 2.12	67.29 ± 2.35	**70.08 ± 2.43**
PubMed	**91.99 ± 0.41**	91.56 ± 0.46	91.27 ± 0.53
Penn94	**85.38 ± 0.25**	84.68 ± 0.56	84.39 ± 0.26

Take-Home Messages



- **Spatial GNNs** and **spectral GNNs** adopt different perspectives in utilizing graphs.
  - *Spectral GNNs* are dominated by **polynomial filters** so far. SoTA polynomial filters can approximate **arbitrary polynomial functions** using **polynomial basis**.
  - *Spatial GNNs* benefit from entangled non-linear transformations.
- We use **simple residual connections** to rewire the information flow, injecting ***spectral characteristics*** into a ***message passing (spatial) backbone,*** keeping the entangled transformations.
  - The *stack of convolution blocks* aligns with the *iteration of Clenshaw's algorithm*.
  - There is a special **negative residual**. The role of it is to use Chebyshev Basis.