INVARIANT AND EQUIVARIANT GRAPH NETWORKS

2/2/2025

Arun Cheriakara Joseph,
Faculty Of Math

UNIVERSITY OF WATERLOO

What Is the Problem?

- Graph Neural Networks (GNNs) must handle **permutation symmetry**
- Two key properties
  - **Invariant models** (graph-level tasks)
  - **Equivariant models** (node-level tasks)
- How can we design **neural network layers** that respect these symmetries?

Why Is It Important?

- Graphs represent **social networks, molecules, knowledge graphs, etc.**
- Traditional deep learning models **do not** handle graphs effectively.
- If models ignore graph symmetry, they may learn **arbitrary, non-generalizable functions.**

Why Don’t Previous Methods Work?

- **Message Passing GNNs** don’t explicitly enforce permutation symmetry.
- Previous work (e.g., Hartford et al. 2018) provides equivariant layers but lacks a **complete basis**.
- This limits **expressivity**—some important graph functions cannot be learned..

What Is the Proposed Solution?

- Defines **all possible linear invariant and equivariant layers**
- Uses **fixed-point equations** to enforce symmetry
- Provides a **universal basis** for equivariant functions
- Can approximate **any message-passing network**

Problem Formulation

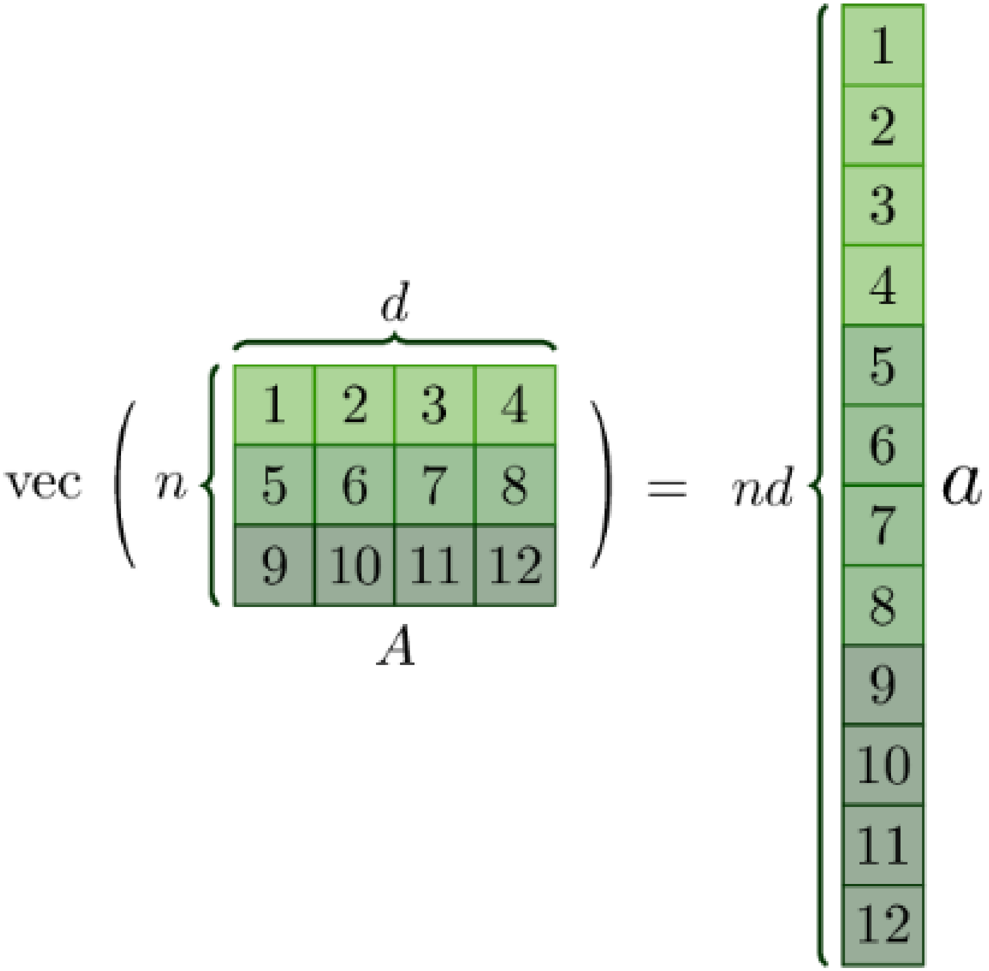
- Input: A graph **G = (V, A)** where A is an adjacency matrix.
- Task: Define neural networks that respect **graph symmetries**.
- **Two primary goals**:
  - Find all possible invariant linear layers.
  - Find all possible equivariant linear layers.

Setup & Notation

- Order-2 tensor $A \in R^{n\ x\ n}$ = adjacency matrix
- $\boldsymbol{vec}(\boldsymbol{X})$ = stack columns of X into one vector
- Permutation matrix P → re-label nodes
- Two types of linear maps:
  - $L: R^{n\ x\ n} \rightarrow R$ (invariant case)
  - $L: R^{n\ x\ n} \rightarrow R^{n\ x\ n}$(equivariant case)

Vectorization & vec(L)

- $\boldsymbol{vec}(\boldsymbol{X})$ = stack columns of X into one vector
- A linear operator L can be "vectorized" too
  - $L: R^{n\,x\,n} \rightarrow R$ (invariant case)
  - $L: R^{n\,x\,n} \rightarrow R^{n\,x\,n}$(equivariant case)
- Why? → Helps rewrite ($P^T A\ P$) via **Kronecker** product



Kronecker Product & Permutations

- **Kronecker product** $X \otimes Y$ stacks blocks of $X \cdot y_{ij}$
- **Identity**:

$$vec(XYZ) = (Z^T \otimes X)vec(Y).$$

  - For **permutation** P, we get $vec(P^T AP) = (P^T \otimes P^T)vec(A)$.

Invariance Condition (Equation 1)

- **Invariant**: $\mathrm{L}(P^T AP) = L(A)$. for all permutations P
- Rewrite using **vec**: $vec(P^T AP) = (P^T \otimes P^T)vec(A)$.
- We want $L(vec(P^T AP)) = L\big(vec(A)\big)$ for all A
- Leads to fixed-point eq.

$$(P \otimes P)vec(L) = vec(L)$$

Equivariance Condition (Equation 2)

- **Equivariant**: $[\mathrm{L}\ \mathrm{vec}(P^T AP)] = P^T[L\ vec(A)]P$
- Using **Kronecker** property

$$LP^T \otimes P^T vec(A) \ = P^T \otimes P^T Lvec(A)$$

- Leads to fixed-point eq.

$$P \otimes P \otimes P \otimes Pvec(L) = vec(L)$$

Extending to Order-k

- For a **k-dim** tensor $A \in R$, each index can be permuted
- In ***vec*** form, that's $(P)^{\otimes k}$
  - Where $(P)^{\otimes k}$ is $P \otimes P \ldots \otimes P$ *upto k times*
- **Invariance** $\Longrightarrow (P)^{\otimes k} vec(L) = vec(L)$ -> Equation 3
- **Equivariance** $\Longrightarrow (P)^{\otimes 2k} vec(L) = vec(L)$ -> Equation 4

UNIVERSITY OF WATERLOO

Proposition 1 (Fixed-Point Characterization)

Proposition 1. *A linear layer is invariant (equivariant) if and only if its coefficient matrix satisfies the fixed-point equations, namely equation 3 (equation 4).*

- **Invariance** $\Rightarrow (P)^{\otimes k} vec(L) = vec(L)$ -> Equation 3
- **Equivariance** $\Rightarrow (P)^{\otimes 2k} vec(L) = vec(L)$ -> Equation 4

SOLVING THE FIXED-POINT EQUATIONS

- **Equations 3 & 4**: $(P)^{\otimes l} vec(X) = vec(X)$(invariance) or $(P)^{\otimes 2l} vec(X) = vec(X)$(equivariance)
- So many permutations, but solution space has **constant dimension** (independent of n) Preserves permutation invariance in expectation
- Rewritten as: $Q \star X = X$ for all Q

Group Action & Equivalence Relation

- **Permutation group** acts on $\mathbb{R}^{n^l} by\ X \to Q \star X$
- **Fixed points**: Tensors X with $Q \star X = X$ for all Q
- **Equivalence relation** on $[n]^l : a \sim b\ if\ a_i = a_j \iff b_i = b_j$ (same "equality pattern")

UNIVERSITY OF WATERLOO

Partitions & Bell Numbers

- Each “equality pattern” $\leftrightarrow$ a **partition** of $\{1, \ldots, \ell\}$
- **# of partitions** = Bell number $b(\ell)$
- Example: ℓ=2 → 2 patterns ;
  - Keep both elements together: **{1,2}**
  - Separate them: **{1}, {2}**
  - So, b(2)=2
- ℓ=3→ 5 patterns; etc.

Building B^γ Tensors

For each equivalence class $\gamma \in [n]^\ell/_\sim$ we define an order-ℓ tensor $\mathbf{B}^\gamma \in \mathbb{R}^{n^\ell}$ by setting

$$\mathbf{B}^\gamma_{\boldsymbol{a}} = \begin{cases} 1 & \boldsymbol{a} \in \gamma \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

- The capital B^γ represents a **basis tensor** for the space of **invariant or equivariant functions**.
- The **γ** in the superscript represents an **equivalence class of multi-indices** that belong to the same **partition pattern** (or "symmetry class")
- Each equivalence class corresponds to a **basis element** B^γ. So the total number of **basis tensors** B^γ is exactly $b(\ell)$

Example for ℓ=2

- We have two elements: i_1, i_2
- There are **two possible partitions** (ways to group them):
  - $\boldsymbol{\gamma 1} = \{\{\mathbf{1}\}, \{\mathbf{2}\}\} \rightarrow$ **Distinct Elements**
    - Means $i_1 \neq i_2$
  - $\boldsymbol{\gamma 2} = \{\mathbf{1}, \mathbf{2}\} \rightarrow$ **Same Elements**
    - Means $i_1 = i_2$
  - So, b(2)=2

Result: $Dimension = b(\ell)$

- Each B^{γ} is invariant under permutations
- The entire solution space is spanned by $\{B^{\gamma}\}$
- **Dimension** of solutions = $b(\ell)$ → independent of n

Proposition 2 – Statement

- *"The tensors B^{γ} form an orthonormal basis in the standard inner product."*
- **Hence** solution space dimension = $b(\ell)$

Proof Idea of Proposition 2

- **If** X is constant on each equivalence class $\rightarrow$ it satisfies $Q \star X = X$.
- **If not**, we find a permutation q causing contradiction
- B^{γ} have **disjoint supports**, so $B^{\gamma} \cdot B^{\gamma'} = \mathbf{0}$ ***if*** γ not equal to γ'

Combining with Proposition 1

- **Proposition 1**: “Invariance/equivariance iff fixed-point under $(P)^{\otimes k}$ or $(P)^{\otimes 2k}$
- **Proposition 2**: B^{γ} are basis solutions
- Together ⇒ **Dimension** = $b(\ell)$ or $b(2\ell)$

UNIVERSITY OF WATERLOO

Theorem 1 – Final Dimension & Basis

Theorem 1. *The space of invariant (equivariant) linear layers* $\mathbb{R}^{n^k} \to \mathbb{R}$ $(\mathbb{R}^{n^k} \to \mathbb{R}^{n^k})$ *is of dimension* $\mathrm{b}(k)$ $(\mathrm{b}(2k))$ *with basis elements* $\mathbf{B}^\gamma$ *defined in equation 8, where* γ *are equivalence classes in* $[n]^k/_\sim$ $([n]^{2k}/_\sim)$.

- Each B^γ matches an equivalence class γ

Adding Bias & Features (Theorem 2)

- **Bias**: A constant output → also invariant
- If input has dimension d, or output dimension d': multiply the dimension accordingly

> - **Theorem 2**: "Space of invariant (equivariant) layers $\mathbb{R}^{n^k \times d} \to \mathbb{R}^{d'}$ is dimension . . . scaled by $b(k)$ or $b(2k)$."

Implementation

- Based on Bell numbers & fixed-point equations
- Uses a basis of 15 linear operations for equivariance
- Implemented using TensorFlow
- Composed of:
  - Equivariant Linear Layers
  - ReLU Activation
  - Max-pooling or Fully Connected Layers

Appendix A

For fast execution of order-2 layers we implemented the following 15 operations which can be easily shown to span the basis discussed in the paper. We denote by $\mathbf{1} \in \mathbb{R}^n$ the vector of all ones.

1. The identity and transpose operations: $L(\boldsymbol{A}) = \boldsymbol{A}$, $L(\boldsymbol{A}) = \boldsymbol{A}^T$.
2. The diag operation: $L(\boldsymbol{A}) = \mathrm{diag}(\mathrm{diag}(\boldsymbol{A}))$.
3. Sum of rows replicated on rows/ columns/ diagonal: $L(\boldsymbol{A}) = \boldsymbol{A}\mathbf{1}\mathbf{1}^T$, $L(\boldsymbol{A}) = \mathbf{1}(\boldsymbol{A}\mathbf{1})^T$, $L(\boldsymbol{A}) = \mathrm{diag}(\boldsymbol{A}\mathbf{1})$.
4. Sum of columns replicated on rows/ columns/ diagonal: $L(\boldsymbol{A}) = \boldsymbol{A}^T\mathbf{1}\mathbf{1}^T$, $L(\boldsymbol{A}) = \mathbf{1}(\boldsymbol{A}^T\mathbf{1})^T$, $L(\boldsymbol{A}) = \mathrm{diag}(\boldsymbol{A}^T\mathbf{1})$.
5. Sum of all elements replicated on all matrix/ diagonal: $L(\boldsymbol{A}) = (\mathbf{1}^T\boldsymbol{A}\mathbf{1}) \cdot \mathbf{1}\mathbf{1}^T$, $L(\boldsymbol{A}) = (\mathbf{1}^T\boldsymbol{A}\mathbf{1}) \cdot \mathrm{diag}(\mathbf{1})$.
6. Sum of diagonal elements replicated on all matrix/diagonal: $L(\boldsymbol{A}) = (\mathbf{1}^T\mathrm{diag}(\boldsymbol{A})) \cdot \mathbf{1}\mathbf{1}^T$, $L(\boldsymbol{A}) = (\mathbf{1}^T\mathrm{diag}(\boldsymbol{A})) \cdot \mathrm{diag}(\mathbf{1})$.
7. Replicate diagonal elements on rows/columns: $L(\boldsymbol{A}) = \mathrm{diag}(\boldsymbol{A})\mathbf{1}^T$, $L(\boldsymbol{A}) = \mathbf{1}\mathrm{diag}(\boldsymbol{A})^T$.

Network Architecture

The network is designed with **1-4 equivariant layers**, with **ReLU activation functions** between them.

- **For Equivariant Tasks**:
  - The network is built with **equivariant linear layers** followed by **ReLU**.
- For Invariant Tasks
  - A **max operation** is applied to the outputs of the invariant basis.
- The **final architecture** consists of:
  - Equivariant Linear Layers (constructed from the 15 basis functions).
  - ReLU Activation.
  - Max-pooling or Fully Connected Layers (for invariance tasks).

Experiments

Table 1: Comparison to baseline methods on synthetic experiments.

	Symmetric projection			Diagonal extraction			Max singular vector				Trace		
# Layers	1	2	3	1	2	3	1	2	3	4	1	2	3
Trivial predictor	4.17	4.17	4.17	0.21	0.21	0.21	0.025	0.025	0.025	0.025	333.33	333.33	333.33
Hartford et al.	2.09	2.09	2.09	0.81	0.81	0.81	0.043	0.044	0.043	0.043	316.22	311.55	307.97
Ours	**1E-05**	**7E-06**	**2E-05**	**8E-06**	**7E-06**	**1E-04**	**0.015**	**0.0084**	**0.0054**	**0.0016**	**0.005**	**0.001**	**0.003**

Experiments

Table 3: Graph Classification Results.

dataset	MUTAG	PTC	PROTEINS	NCI1	NCI109	COLLAB	IMDB-B	IMDB-M
size	188	344	1113	4110	4127	5000	1000	1500
classes	2	2	2	2	2	3	2	3
avg node #	17.9	25.5	39.1	29.8	29.6	74.4	19.7	13
Results								
DGCNN	85.83±1.7	58.59±2.5	75.54±0.9	74.44±0.5	NA	73.76±0.5	70.03±0.9	47.83±0.9
PSCN (k=10)	88.95±4.4	62.29±5.7	75±2.5	76.34±1.7	NA	72.6±2.2	71±2.3	45.23±2.8
DCNN	NA	NA	61.29±1.6	56.61± 1.0	NA	52.11±0.7	49.06±1.4	33.49±1.4
ECC	76.11	NA	NA	76.82	75.03	NA	NA	NA
DGK	87.44±2.7	60.08±2.6	75.68±0.5	80.31±0.5	80.32±0.3	73.09±0.3	66.96±0.6	44.55±0.5
DiffPool	NA	NA	78.1	NA	NA	75.5	NA	NA
CCN	91.64±7.2	70.62±7.0	NA	76.27±4.1	75.54±3.4	NA	NA	NA
GK	81.39±1.7	55.65±0.5	71.39±0.3	62.49±0.3	62.35±0.3	NA	NA	NA
RW	79.17±2.1	55.91±0.3	59.57±0.1	> 3 days	NA	NA	NA	NA
PK	76±2.7	59.5±2.4	73.68±0.7	82.54±0.5	NA	NA	NA	NA
WL	84.11±1.9	57.97±2.5	74.68±0.5	84.46±0.5	85.12±0.3	NA	NA	NA
FGSD	92.12	62.80	73.42	79.80	78.84	80.02	73.62	52.41
AWE-DD	NA	NA	NA	NA	NA	73.93±1.9	74.45 ±5.8	51.54 ± 3.6
AWE-FB	87.87±9.7	NA	NA	NA	NA	70.99 ± 1.4	73.13 ±3.2	51.58 ± 4.6
ours	84.61±10	59.47±7.3	75.19±4.3	73.71±2.6	72.48±2.5	77.92±1.7	71.27±4.5	48.55±3.9

What Interesting Research Questions Remain?

- **Explored**
  - Characterized **all possible** linear invariant & equivariant layers.
  - Provided **explicit basis functions** for constructing such layers.
- **Future Research**:
  - Extending to **non-linear models**.
  - Exploring applications in **real-world graphs**.

UNIVERSITY OF WATERLOO

Thank You!