$\mathcal{N}$-WL: A New Hierarchy of Expressivity for Graph Neural Networks

Qing Wang, Dillon Chen, Asiri Wijesinghe,
Shouheng Li, Muhammad Farhan

Graph Research Lab
School of Computing
Australian National University

The work is partially funded by the Australian Research Council under Discovery Project DP210102273.

Classical k-WL Hierarchy

k-Weisfeiler-Lehman (k-WL) hierarchy is a theoretical framework for graph isormorphism tests

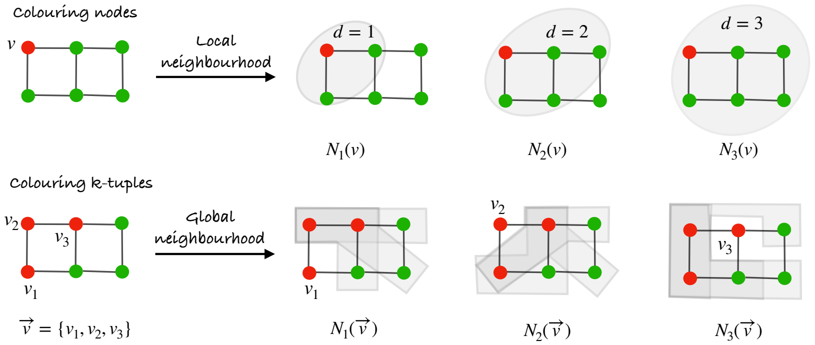
$\hookrightarrow$ but not practically useful when $k \geq 3$!

- GIN $\equiv$ 1-WL [Xu et al., 2019]
- Many expressive GNNs go beyond 1-WL

Question:

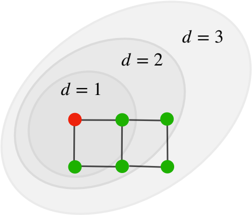
Is k-WL hierarchy a good yardstick for measuring expressivity of GNNs?

GNNs vs k-WL

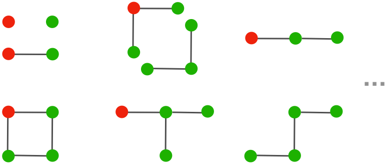


Colouring nodes
v
Local neighbourhood
$d = 1$
$d = 2$
$d = 3$
$N_1(v)$
$N_2(v)$
$N_3(v)$
Colouring k-tuples
v_2
v_3
v_1
Global neighbourhood
v_1
v_2
v_3
$\vec{v} = \{v_1, v_2, v_3\}$
$N_1(\vec{v})$
$N_2(\vec{v})$
$N_3(\vec{v})$

$\mathscr{N}$-WL hierarchy computes node coloring via t-order induced subgraphs within d-hop neighbourhoods.



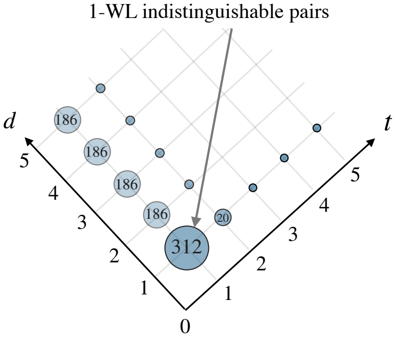
d-hop neighbourhoods



t-order induced subgraphs

A graph isomorphism test on 312 pairs of simple graphs of 8 vertices:

- *None-or-all*: none by 1-WL but all by 3-WL
- *Progressive*: varying with d and t by $\mathcal{N}$-WL



Increasing the order of induced subgraphs, the expressive power increases
– *Not surprising*

Theorem:
(Weak Hierarchy) $\mathscr{N}^{-}(t, d)$-WL $\subsetneq$ $\mathscr{N}^{-}(t+1, d)$-WL

Increasing the hops of neighbourhood, the expressive power may decrease
– *Surprising but can be fixed*

Theorem:
(Strong Hierarchy) $\mathscr{N}(t, d)$-WL $\subsetneq$ $\mathscr{N}(t+1, d)$-WL
$\mathscr{N}(t, d)$-WL $\subsetneq$ $\mathscr{N}(t, d+1)$-WL

Induced connected subgraphs remain the same expressive power
– *Surprising but can be proved*

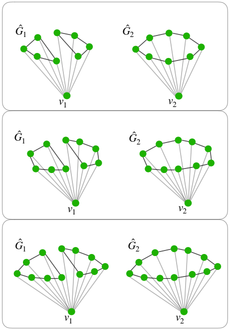
Theorem:
(Equivalence) $\mathscr{N}^{c}(t, d)$-WL $\equiv$ $\mathscr{N}(t, d)$-WL

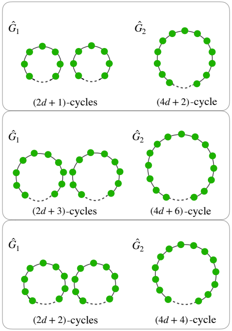
Main Ideas in Proofs (1)

Theorem:
(Strong Hierarchy)

$$\mathcal{N}(t,d)\text{-WL} \subsetneq \mathcal{N}(t{+}1,d)\text{-WL}$$
$$\mathcal{N}(t,d)\text{-WL} \subsetneq \mathcal{N}(t,d{+}1)\text{-WL}$$

We prove strictness of hierarchies by constructing counterexample graphs.

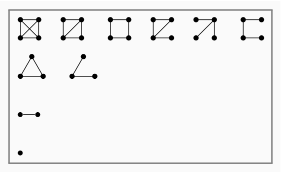




Theorem:
(Equivalence)

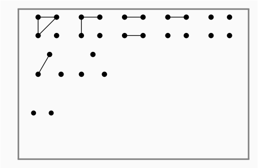
$$\mathscr{N}^c(t,d)\text{-WL} \equiv \mathscr{N}(t,d)\text{-WL}$$

Our proof is based on Kocay's Vertex Theorem [Kocay, 1982].



subgraph counts

implies
$\longrightarrow$



subgraph counts

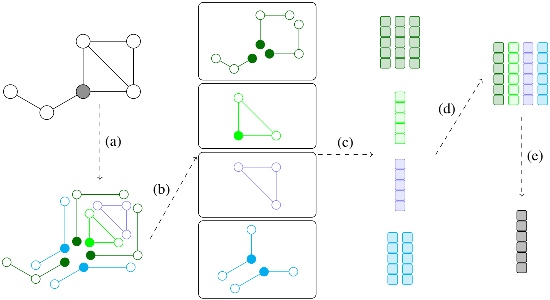
k-WL Hierarchy vs $\mathcal{N}$-WL Hierarchy

	k-WL	δ-k-LWL	(k,s)-LWL	$(k,c)(\leq)$-SETWL
#Coloured objects	n^k	n^k	$\text{subset}(n^k, s)$	$\text{subset}(\sum_{q=1}^{k}\binom{n}{q}, c)$
#Neighbour objects	$n \times k$	$a \times k$	$a \times k$	$n \times q$
ΔColoured objects	k-tuples	k-tuples	k-tuples	$\leq k$-sets
ΔNeighbour objects	k-tuples	k-tuples	k-tuples	[illegible]
Sparsity -awareness	✗	✓	✓	

$\mathcal{N}(t,d)$-WL	$\mathcal{N}^c(t,d)$-WL
n	n
$\binom{a^d}{t}$	$\text{subset}(\sum_{q=1}^{t}\binom{a^d}{q}, 1)$
nodes	nodes
t-sets	$\leq t$-sets
✗	✓

Theorem: 1-WL $\equiv \mathcal{N}(1,1)$-WL $\equiv \mathcal{N}^c(1,1)$-WL

Graph Neighbourhood Neural Network (G3N) instantiates the ideas of $\mathcal{N}$-WL algorithms for graph learning.



$$h_u^{(l+1)} = \textsc{Combine}\Big(h_u^{(l)}, \textsc{Agg}^{N}_{(i,j)\in I_t \times J_d}\Big(\textsc{Agg}^{T}_{S \in \mathcal{S}_u^{(l)}(i,j)}\Big(\textsc{Pool}(S)\Big)\Big)\Big)$$

Kocay, W. L. (1982).
Some new methods in reconstruction theory.
In *Combinatorial Mathematics IX*, pages 89–114. Springer.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. (2019).
How powerful are graph neural networks?
In *International Conference on Learning Representations (ICLR)*.

Thank You