On the Connections Between Graph Transformers and GNNs

Veeti Ahvonen

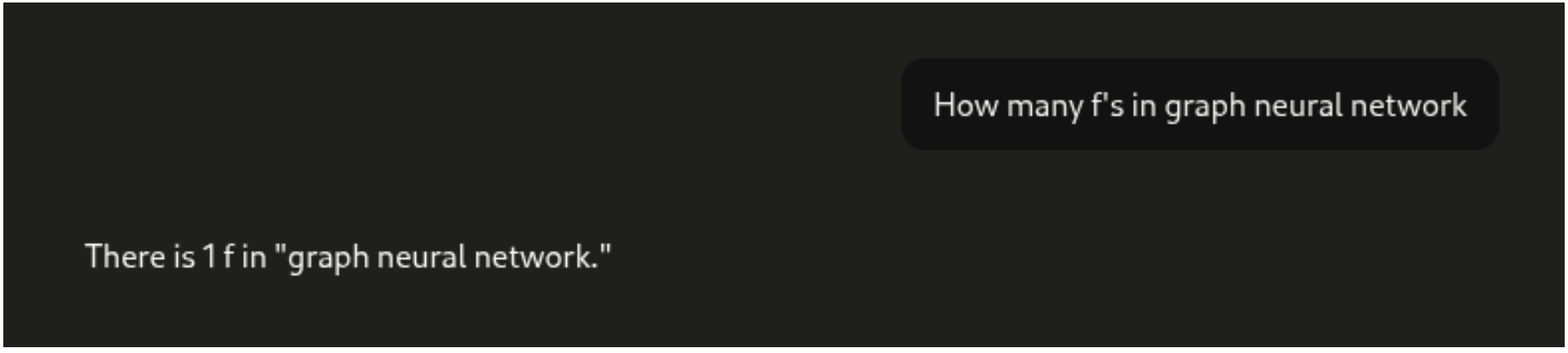
Mathematics Research Centre

25.6.2026

Based on joint work "Expressive power of graph transformers via logic" (AAAI 2026) with Maurice Funk, Damian Heiman, Antti Kuusisto and Carsten Lutz

Background and motivation

Transformers (Vaswani et al. NIPS 2017) form the basis for modern LLMs such as ChatGPT, Claude, Copilot, etc.



Little is known about their precise expressive power on graphs

Standard transformers have been studied via temporal logics by Yang et al. (NeurIPS 2024), Chiang et al. (ICML 2023), Li and Cotterel (NeurIPS 2025), Jerad et al. (ACL 2025).

Background and motivation

Graph transformers use a self-attention mechanism to aggregate global information from nodes

GNNs with global readout combine local and global message passing

This raises the question: *How are GNNs with global readout and graph transformers related to each other?*

Rosenbluth et al. (ICLR 2024) proved that GNNs with global readout and GPS-networks (a graph transformer architecture) using reals have incomparable expressive power w.r.t. function approximation.

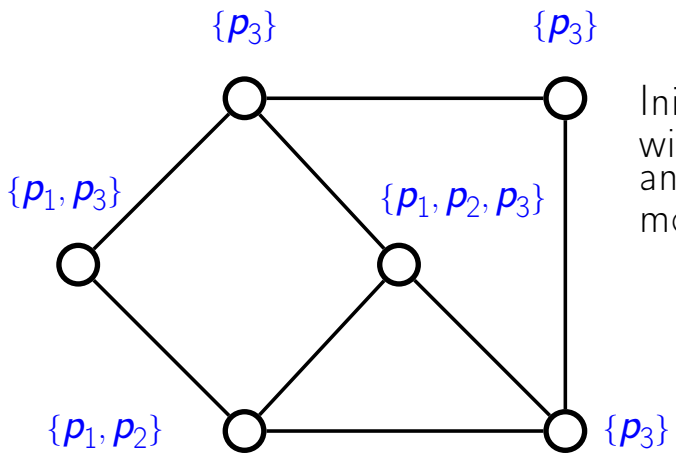
Our approach

We built logical characterizations for GNNs with global readout and graph transformers in two settings:

1. computations are carried by using real numbers and
2. computations are carried by using floating-point numbers

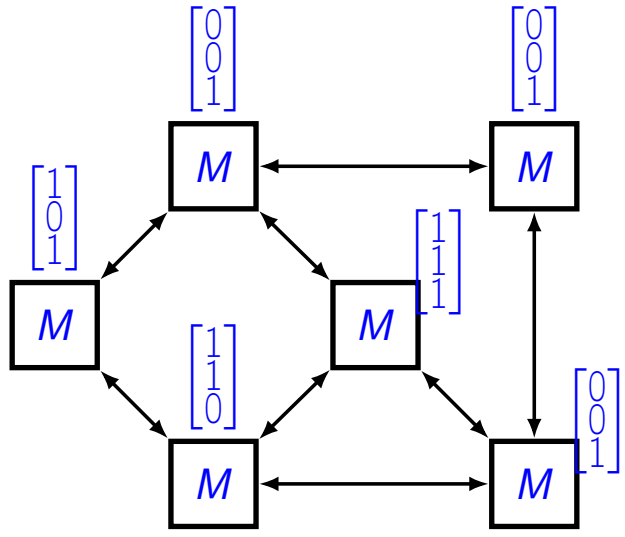
Ultimately, with these characterizations we find links between graph transformers and GNNs with global readout.

Graphs



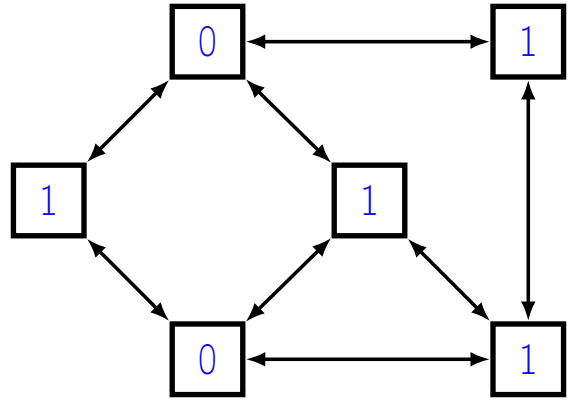
Labeled graph

Initializing each vertex with a feature vector and place same computing model on each node



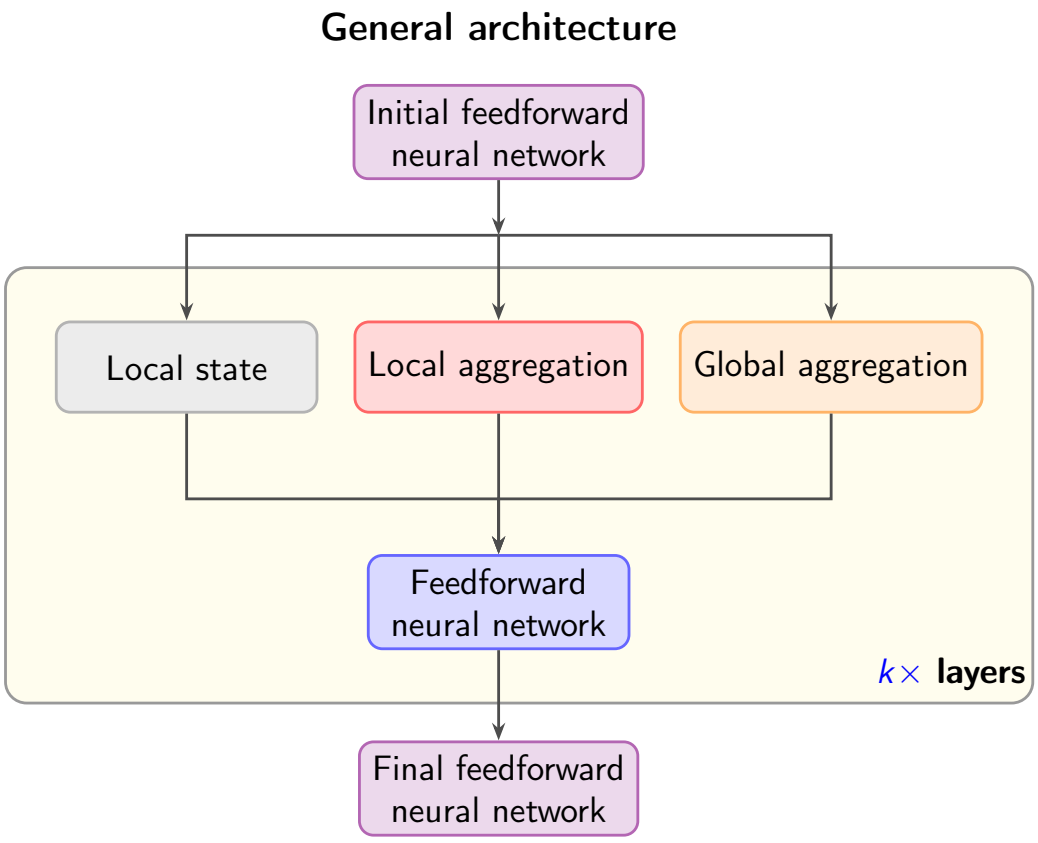
Initial configuration

Message passing between vertices



Outputs

Architectures



The initial FNN gives a state in $\mathbb{R}^d$ for each node based on its initial configuration

In each layer an FNN gives a new state in $\mathbb{R}^d$ for each node based on

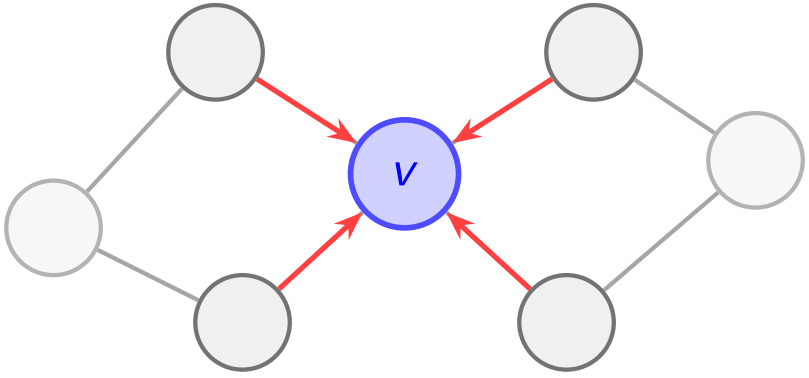
- the node's own local state in $\mathbb{R}^d$
- an aggregated state in $\mathbb{R}^d$ from the node's neighbours
- an aggregated state in $\mathbb{R}^d$ from all nodes in the underlying graph

The final FNN gives a boolean classification based on the state computed by the last layer

Local vs. global aggregation

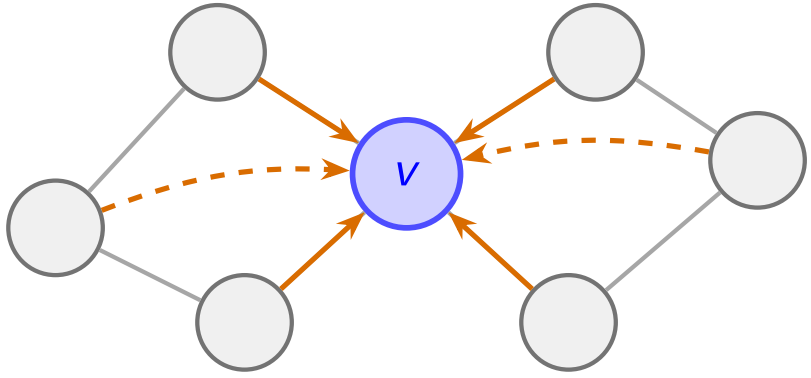
An aggregation function $multi(\mathbb{R}^d) \to \mathbb{R}^d$, where $multi(\mathbb{R}^d)$ is the set of multisets over $\mathbb{R}^d$

Local aggregation



v aggregates from its neighbors only

Global aggregation



v aggregates from all vertices

Typically an aggregation function is a sum, mean or max

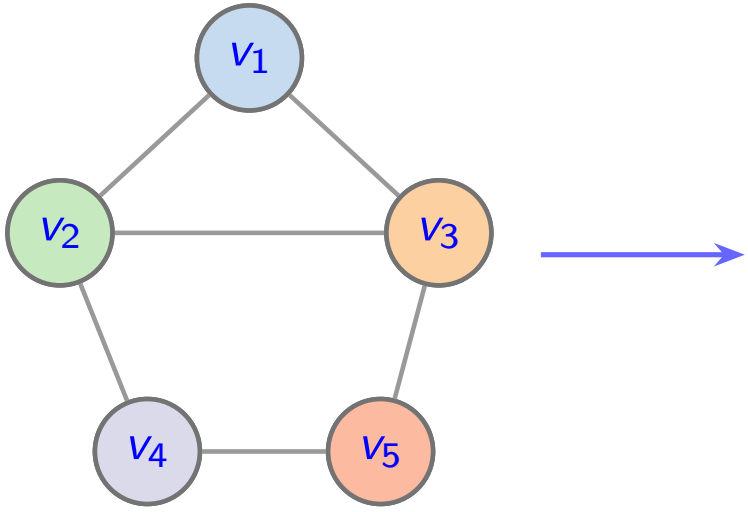
Self-attention

Self-attention: A special global aggregation used by transformers:

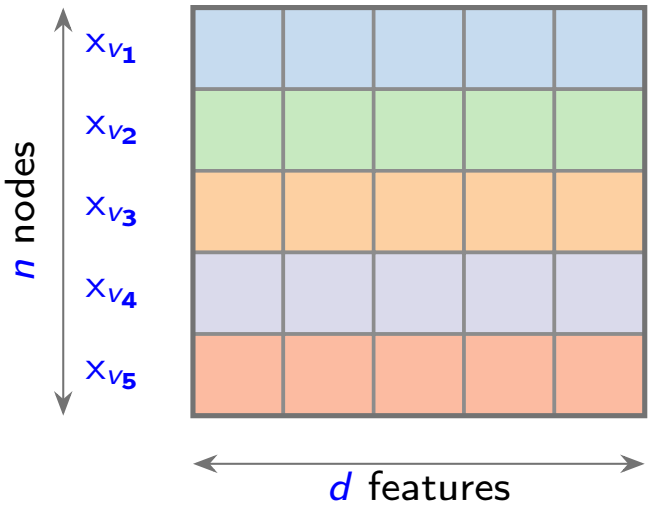
$$\mathrm{softmax}\Big(\frac{(XW_Q)(XW_K)^\top}{\sqrt{d}}\Big)XW_V,$$

where W_Q, W_K and W_V are real-valued $d \times d$-matrices and X is the feature matrix of the graph.

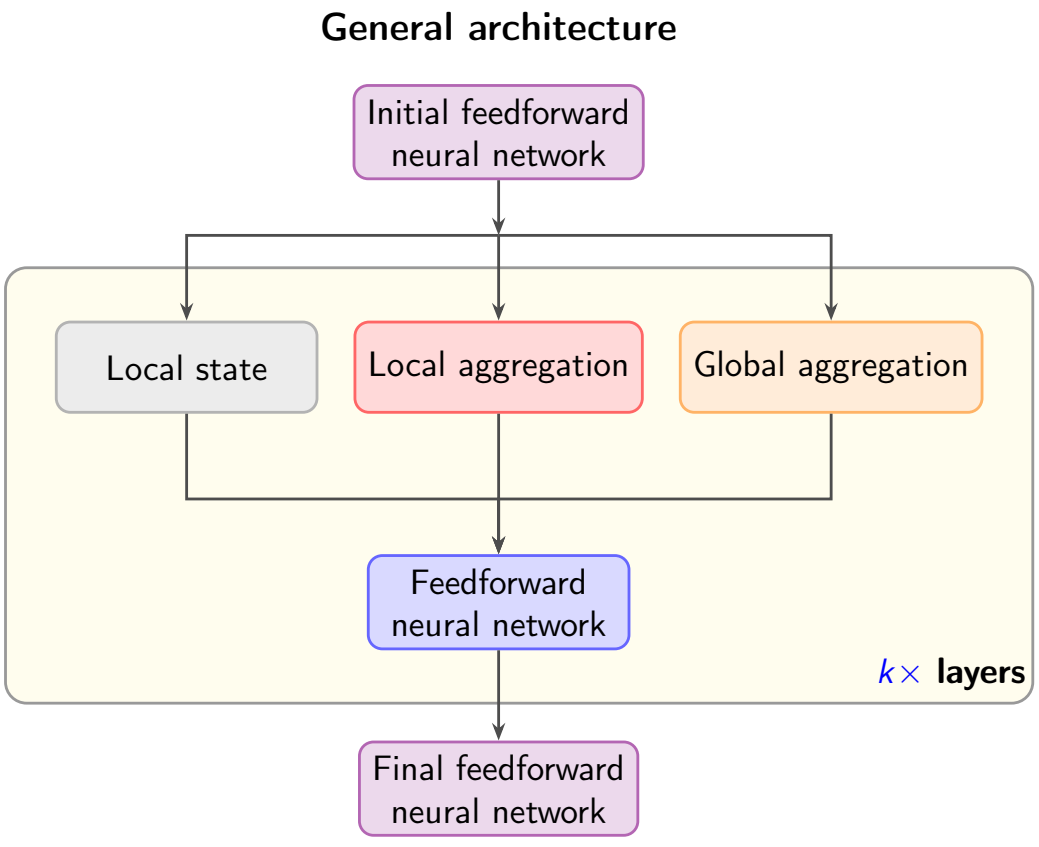
Graph



Feature matrix X



Architectures



GNNs with global readout (or GNNs+GR): Local and global aggregation is typically a sum, mean or max

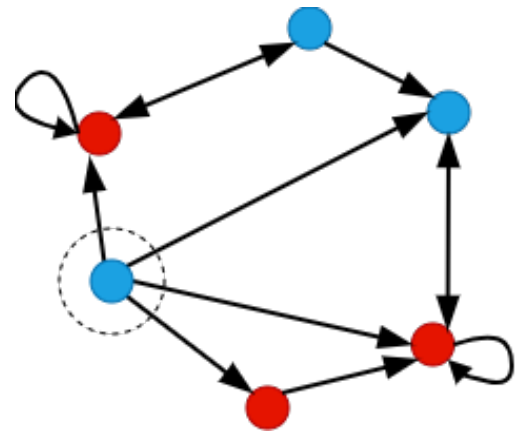
GPS-networks:

- Local aggregation is typically a sum, mean or max
- Global aggregation is a self-attention module

Logics

Graded modal logic with counting global modality (or $\mathrm{GML} + \mathrm{GC}$): Propositional logic + diamonds $\Diamond_{\geq k}$ and $\langle G \rangle_{\geq k}$.

- A vertex satisfies $\Diamond_{\geq k} \varphi$ iff at least k out-neighbours satisfy φ.
- A vertex satisfies $\langle G \rangle_{\geq k} \varphi$ iff at least k vertices in the graph satisfy φ.



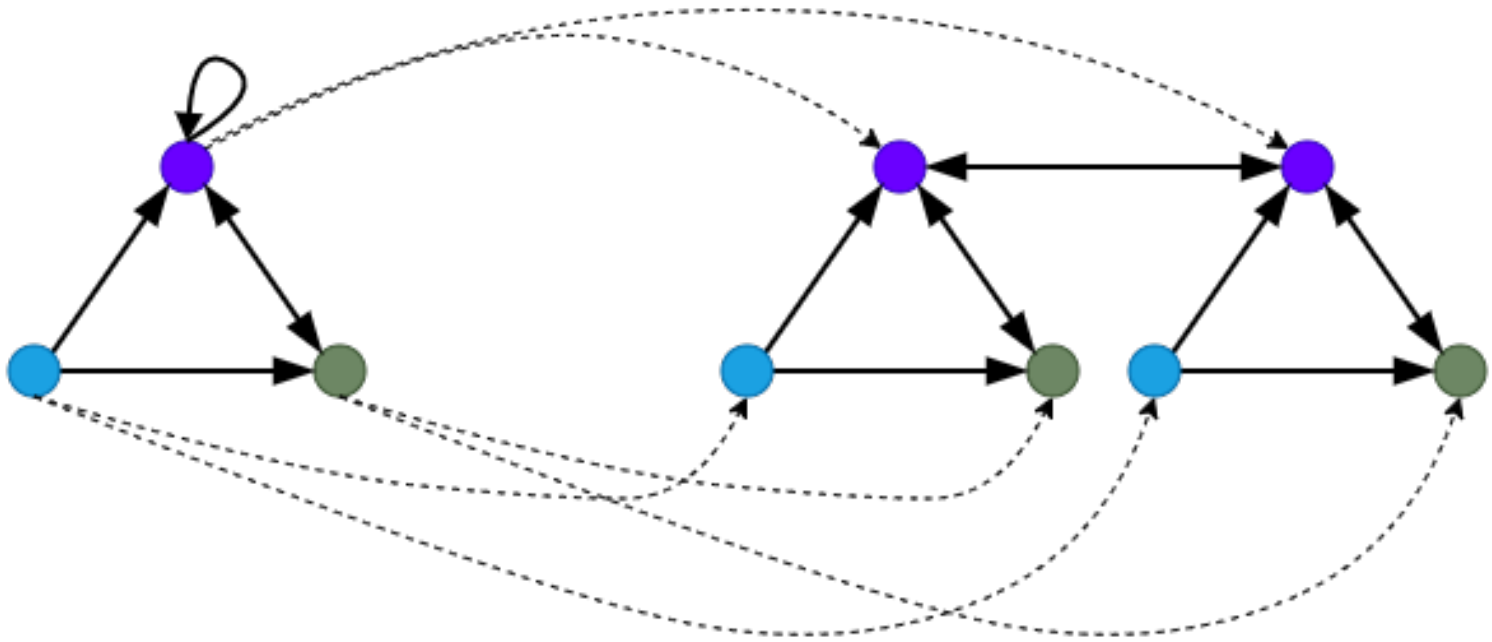
$\mathrm{GML} + \mathrm{G}$: Propositional logic + diamonds $\Diamond_{\geq k}$ and $\langle G \rangle_{\geq 1}$.

Results on reals

We introduce **global-ratio graded bisimilarity** $\sim_{G\%}$ that takes into account the ratios by which the graded bisimilarity types appear in a graph.

Lemma 1

Every FO*-formula invariant under* $\sim_{G\%}$ *is equivalent to a formula of* $\mathrm{GML} + \mathrm{G}$.



Results on reals

Below, we let $\mathrm{GNN} + \mathrm{GRM}$ refer to GNNs with global readout that use mean as the global aggregation function.

Moreover, let $\mathrm{GNN} + \mathrm{GR1}$ refer to GNNs with global readout where the global aggregation function is of the form $\mathit{Pow}(\mathbb{R}^d) \to \mathbb{R}^d$, where $\mathit{Pow}(\mathbb{R}^d)$ denotes the power of set over $\mathbb{R}^d$.

Theorem 2

Relative to FO*, the following classes have the same expressive power:*

$$\mathrm{GML} + \mathrm{G} \equiv \textit{GPS-networks} \equiv \mathrm{GNN} + \mathrm{GRM} \equiv \mathrm{GNN} + \mathrm{GR1}$$

Proof.

It is easy to show by induction on the structure of a formula that computing models can be used to simulate GML+G formulae.
We show that each computing model is invariant under $\sim_{G\%}$ and then apply Lemma 1. □

Float-based architectures

Float-based sums are bounded, i.e., there exists some k such that whether a float appears k or ℓ times in the sum makes no difference for all $\ell > k$.

This is because:

- Float sum is not associative due to rounding errors, so the order of the sum has to be fixed.
- Summing in a random vertex order violates isomorphism invariance.
- Instead, summing in increasing order of floats is reasonable w.r.t. numerical accuracy, but leads to boundedness due to rounding.

Thus, floating-point aggregation functions are also bounded.

Float-based results

Below let GNN+GR refer to GNNs with global readout.

Theorem 3

The following classes have the same expressive power:

$$\mathrm{GML} + \mathrm{GC} \equiv \textit{GPS-networks with floats} = \textit{GNN+GR with floats}$$

This also applies when local aggregation functions are the sum and FNNs have two layers and are ReLU-activated.

Proof.

We show by induction on the structure of a formula that there is an equivalent computing model for each formula. To simulate diamonds $\langle G \rangle_{\geq k}$ by using self-attention modules we use underflow phenomenon of float arithmetic.

To simulate each computing model by a formula of GML+GC, we proceed as follows. All local steps (e.g., FNNs) can be simulated by using PL. Local (resp. global) aggregation functions can be handled by using diamonds $\Diamond_{\geq k}$ (resp. diamonds $\langle G \rangle_{\geq k}$) since aggregation functions are bounded. $\square$

Corollaries

Interestingly, we obtain the following

Theorem 4

1. *Relative to* FO*: GPS-networks with reals* $\subsetneq$ *GPS-networks with floats*
2. *GPS-networks with reals* $\not\equiv$ *GPS-networks with floats*
3. *Relative to* FO *and reals: GNN* $\subsetneq$ *GPS-networks* $\subsetneq$ *GNNs + GR*

Proof.

The first claim follows directly from the obtained characterizations.

The second one follows from the first claim and the fact that GPS-networks are invariant under global-ratio graded bisimilarity but GPS-networks with floats are not.

The third claim follows from our characterization and the results by Barceló et al. (ICLR 2020). □

Future work

- Characterize common positional encodings (e.g. graph Laplacian).
- Generalize all of our results for graph classifications (our float results already generalize for graph classification tasks).

See also our paper for many other results that I did not have time to go through.

Expressive power of graph transformers via logic, A., Funk, Heiman, Kuusisto and Lutz, AAAI 2026