Towards Understanding the Expressive Power of GNNs with Global Readout (Extended Summary)

Maurice Funk[1,2], Daumantas Kojelis[1]

[1]Leipzig University, Germany
[2]ScaDS.AI Center Leipzig/Dresden

Abstract

We provide an extended summary of results in our preprint [1] in which we study the expressive power of message-passing aggregate-combine-readout graph neural networks (ACR-GNNs). Particularly, we focus on the first-order (FO) properties expressible by this formalism. While a tight logical characterisation remains a difficult open question, we make two contributions towards answering it. First, we show that sum aggregation and readout suffice for GNNs to capture FO properties that cannot be expressed in the logic C^2 on both directed and undirected graphs. This strengthens known results by Hauke and Wałęga [2] where aggregation and readout functions are specially crafted for the task. Second, we identify two natural ways of restoring characterisability (with regard to C^2) for ACR-GNNs. One option is to limit local aggregation (without imposing restrictions on global readout), whilst the second is to run ACR-GNNs over graphs of bounded degree (but unbounded size). In both cases, the FO properties captured by GNNs are exactly those definable by a formula in graded modal logic with global counting modalities. Our results thus establish an innate lower- and upper-bound in terms of how far (fragments of) C^2 can be taken to characterise GNNs, and imply that it is indeed the unbounded interaction of aggregation and readout that pushes the logical expressive power of GNNs above C^2.

Keywords

Graph Neural Networks, Expressive Power, Graded Modal Logic, Logical Characterisations, Global Counting Modalities

1. Introduction

Message-passing graph neural networks (GNNs) [3] are popular machine learning models for graph structured data and are by design permutation-invariant. They have found applications in diverse areas that involve graph data such as analysis of molecules [4], medicine [5], fraud detection [6], and traffic forecasting [7]. The many variants of GNNs in use all share central concepts. A GNN associates each vertex of a graph with an embedding vector and updates this embedding layer-by-layer. In each layer of a GNN, every vertex sends messages containing its current embedding to other vertices, and receives such messages themselves. The received messages are then aggregated and combined with the current embedding to form the new embedding.

The main differences between GNN variants lie in the choice of vertices that receive a message and the way the messages are aggregated and combined with the current embedding. Commonly, messages are passed along edges of the graph. Other variants are equipped with global readout functionality (or virtual vertices) – the ability to aggregate messages from *all* vertices in a graph regardless of their connection. Popular choices of functions that aggregate messages both along edges and from global readout are dimension-wise sum, max, or mean. The combination function, which forms the new vertex embeddings from the aggregated messages, is typically a trainable feed-forward neural network (FFNN) with a certain choice of activation function. To understand the limitations and relationships of all these possible choices, a growing body of literature is dedicated to characterising the expressive power of the GNN variants using logical formalisms.

One main direction of recent research is finding which vertex properties (or graph properties) can be expressed by GNNs. The central result in this area is by Barceló et al. [8], who show that GNNs with

LOGICNN 2026: Workshop on Logical Methods for Neural Network Analysis, July 25, 2026, Lisbon, Portugal
maurice.funk@uni-leipzig.de (M. Funk); daumantas.kojelis@uni-leipzig.de (D. Kojelis)
0000-0003-1823-9370 (M. Funk); 0000-0002-1632-9498 (D. Kojelis)

only local aggregation can express exactly the first-order (FO) vertex properties that are formulas of graded modal logic (GML), and that sum aggregation suffices for this. In the same paper, the authors also show that GNNs with both local aggregation and global readout can express all formulas of the logic C^2, that is, the two-variable fragment of first-order logic with counting quantifiers. Whether C^2 exactly corresponds to GNNs with aggregation and readout was left open.

Surprisingly, the question was later answered negatively by Hauke and Wałęga [2]. They show that there is an FO vertex property that can be expressed by certain aggregate-combine-readout GNN (ACR-GNN) architectures, but cannot be expressed by C^2. In light of this unexpected expressive power, it is an open question what logics can be used to characterise GNNs with readout.

More is known when the aggregation and readout functions of ACR-GNNs are not arbitrary functions, but are bounded in the sense that they are only sensitive to messages up to a certain multiplicity. Cuenca Grau et al. [9] show that ACR-GNNs with bounded aggregation and bounded readout correspond exactly to GML extended with a global counting modality. This holds for all vertex properties, not just the ones expressible in FO. Unfortunately, common aggregation and readout functions such as sum and mean are not bounded in this sense. Likewise, it is not known whether these specific aggregation functions allow ACR-GNNs to express FO properties outside C^2. This leaves a gap in the understanding of the expressive power of ACR-GNNs, as the known upper and lower bounds are relatively far apart. We make two contribution towards closing this gap.

First, we show that *simple* ACR-GNNs which only use (unbounded) sum for aggregation and readout can express FO properties outside C^2. We show this for both directed (Theorem 1) and undirected graphs (Theorem 3). This strengthens the mentioned result by Hauke and Wałęga [2], which relies on specialized aggregation, readout, and combination functions. In fact, for directed graphs we use the same FO property as Hauke and Wałęga (strict linear orders), but use a novel characterisation in terms of *homomorphism counts* to show that this property is expressible using only sum. For undirected graphs, we also follow a similar approach as Hauke and Wałęga and simulate directed edges via undirected gadgets, but we use a different directed graph gadgetisation for technical reasons.

As our second contribution, we identify two relevant cases that each result in ACR-GNNs capturing exactly the FO vertex properties expressible in GML extended with a global counting modality on both directed and undirected graphs. The first case is the restriction to only graphs in which the degree of vertices is bounded by some constant (Theorem 4). This is a natural assumption for many graph datasets, for example, those that originate from molecules or road networks. The second, closely related, case is that of ACR-GNNs in which the aggregation functions are bounded, but the readout functions can be arbitrary (Proposition 5). This, for FO vertex properties, generalises the result by Cuenca Grau et al. [9], where *both* aggregation and readout functions are taken to be bounded. It is quite surprising that allowing unbounded readout functions does not increase the expressive power of ACR-GNNs relative to FO. This reveals that it is indeed the unbounded interaction of readout and aggregation that moves ACR-GNNs outside C^2 in terms of expressivity. Our characterisation result relies on the combination of first-order model theory and architectural properties of ACR-GNNs. The constructions used may be of independent interest in the study of finite model theory.

Full proofs are available in the preprint [1].

2. Preliminaries

We introduce some preliminaries to formally state our results. For a d-dimensional vector $x \in \mathbb{R}^d$, we denote its i-th component by $x[i]$. All vectors encountered are row vectors.

Graphs A graph $G = (V, E)$ consists of a non-empty set of vertices V and set of edges $E \subseteq V \times V$. Unless explicitly stated, graphs are *directed* and can contain reflexive edges. In *undirected* graphs $(v, u) \in E$ implicitly means that also $(u, v) \in E$. The *out-neighbourhood* of a vertex $v \in V$, denoted $N^G_{\text{out}}(v)$, is the set of all vertices reachable from v with an outgoing edge. Formally, $N^G_{\text{out}}(v) = \{u \in V \mid (v, u) \in E\}$. The *in-neighbourhood* $N^G_{\text{in}}(v)$ is defined similarly. When dealing with undirected graphs,

the in- and out-neighbourhoods coincide and are simply referenced as $N^G(v)$. A graph has *bounded degree* c if $|N^G_{\text{out}}(v)| \leq c$ for all $v \in V$; it being understood that $c \in \mathbb{N}$.

We say that a graph G is *d-featured* (for some $d \in \mathbb{N}$) if it is associated with an embedding $f\colon V \to \{0,1\}^d$. We then write $G = (V, E, f)$. A *pointed* graph is a tuple (G, v) where $G = (V, E)$ and $v \in V$. We also allow pointed graphs to be d-featured.

Graph Neural Networks The GNNs concerned in this paper are of the aggregate-combine-readout (ACR) variety. For the purposes of this presentation, we confine ourselves to *simple* ACR-GNNs [8]. An L-layer *simple ACR-GNN* with input dimension $d \in \mathbb{N}$ is a $(4L{+}1)$-tuple $\mathcal{N} = (\{A_i\}_{i\leq L}, \{C_i\}_{i\leq L}, \{R_i\}_{i\leq L}, \{b_i\}_{i\leq L}, \mathsf{cls})$, such that there exist $d_0, \ldots, d_L \in \mathbb{N}$ with $d = d_0$ and, for all $i \in [1, L]$, the components A_i, C_i, R_i are matrices of dimension $d_{i-1} \times d_i$ and b_i is a row vector of length d_i. The component cls is a classification function which will be taken to be the threshold $\mathsf{cls} : \mathbb{R} \to \{0,1\}$ with $\mathsf{cls}(x) = 1$ if $x \leq 0$ and $\mathsf{cls}(x) = 0$ otherwise.

On a d-featured graph $G = (V, E, f)$ an L-layer simple ACR-GNN with input dimension d computes a series of functions $f^{(i)}\colon V \to \mathbb{R}^{d_i}$ for $i \leq L$, starting with $f^{(0)} = f$, and then computing

$$f^{(i)}(v) = \mathrm{ReLU}\Big(\big(f^{(i-1)}(v)\big)C_i + \Big(\sum_{u\in N^G_{\text{out}}(v)} f^{(i-1)}(u)\Big)A_i + \Big(\sum_{u\in V} f^{(i-1)}(u)\Big)R_i\Big)$$

for all $v \in V$ and $i \leq L$; it being understood that $\mathrm{ReLU}(x) = \max(0, x)$ is applied componentwise.

An L-layer ACR-GNN $\mathcal{N}$ with input dimension d *accepts* a pointed d featured graph (G, v) if $\mathsf{cls}(f^{(L)}(v)) = 1$. We write $\mathcal{N}(G, v) = 1$ when $\mathcal{N}$ accepts (G, v), and $\mathcal{N}(G, v) = 0$ when it does not.

Graded Modal Logic Fix a $d \in \mathbb{N}$. To express vertex properties over d-featured graphs, we will make heavy use of graded modal logic with graded global modality ($\mathrm{GML}^{\exists}$) with d propositional letters. Formally, $\mathrm{GML}^{\exists}$ with d propositional letters is the set of all formulas φ over the grammar:

$$\varphi ::= \top \mid p_i \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond_{\geq k}\varphi \mid \exists_{\geq k},$$

it being understood that $k \in \mathbb{N}$ and p_i is a proposition letter indexed by integers $i \in [1, d]$.

We evaluate formulas of $\mathrm{GML}^{\exists}$ with d propositional letters over pointed d-featured graphs (G, v). Let $G = (V, E, f)$. We define satisfaction of a formula φ at vertex v in G, written $G, v \models \varphi$, inductively as follows:

$G, v \models \top$	always	$G, v \models \varphi_1 \wedge \varphi_2$	if $G, v \models \varphi_i$ for $i \in \{1, 2\}$
$G, v \models p_i$	if $f(v)[i] = 1$	$G, v \models \Diamond_{\geq k}\varphi$	if $\lvert\{u \in N^G_{\text{out}}(v) \mid G, u \models \varphi\}\rvert \geq k$
$G, v \models \neg\varphi$	if $G, v \not\models \varphi$	$G, v \models \exists_{\geq k}\varphi$	if $\lvert\{u \in V \mid G, u \models \varphi\}\rvert \geq k$.

In the sequel we suppress reference to d if it can be inferred from context.

First-order Logic We write FO for first-order logic and C^2 for its two-variable fragment with counting quantifiers. In the sequel we will be concerned with the capabilities of formal languages in terms of classifying featured pointed graphs. Thus, we call FO, C^2, and $\mathrm{GML}^{\exists}$ formulas over the language of d-featured pointed graphs FO, C^2, and, respectively, $\mathrm{GML}^{\exists}$ *classifiers*. We say that a classifier φ of the above sort (or, respectively, an ACR-GNN $\mathcal{N}$) *captures* a property $\mathcal{P}$ if, for all featured pointed graphs (G, v), we have $G, v \models \varphi$ (respectively, $\mathcal{N}(G, v) = 1$) if and only if (G, v) has the property $\mathcal{P}$. Note that the property $\mathcal{P}$ can itself be thought of as a classifier.

3. Expressiveness of ACR-GNNs

It is known that, in the context of undirected graphs, simple ACR-GNNs are able to express every property definable in C^2 [8]. Barceló et al. [8] left the converse direction (relativised to FO) as an open

problem. A negative answer was achieved by Hauke and Wałęga [2], albeit, for *non-simple* ACR-GNNs. Our first result continues in this direction and shows that even simple ACR-GNNs can express FO properties over directed graphs that remain outside C^2.

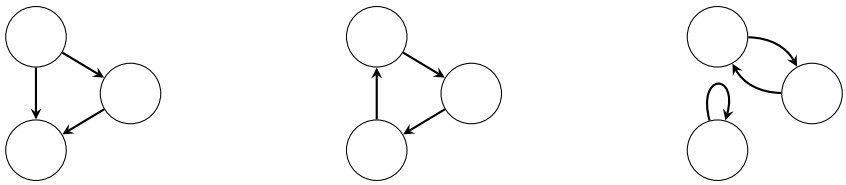
Theorem 1. *There is an FO vertex property over directed graphs that is not equivalent to any C^2 formula but is captured by a simple ACR-GNN.*

Our approach is similar to that of Hauke and Wałęga [2] in the sense that we construct an ACR-GNN that accepts graphs in which the edge relation is a strict linear order. In our setting, however, there is no clear way of checking (via a simple ACR-GNN) that the standard conditions of strict linear orders are met. We thus introduce a novel characterisation of strict linear orders in terms of homomorphism counts.

Let us recall that a homomorphism from a graph $H = (U, F)$ to a graph $G = (V, E)$ is a function $h\colon U \to V$ such that $(u, v) \in F$ implies $(h(u), h(v)) \in E$ for all $u, v \in U$. The set of all homomorphisms from H to G is denoted by $\hom(H, G)$. Now, let us write P_2 for the graph with vertices $\{v, u, w\}$ and edges $\{(v, u), (u, w)\}$. Intuitively, P_2 is simply a path of length 2. The following is then the promised characterisation which might be of independent interest in the study of graph theory.

Lemma 2. *A graph $G = (V, E)$ is a strict linear order iff $|E| = \binom{|V|}{2}$ and $|\hom(P_2, G)| = \binom{|V|}{3}$.*

The intuition behind this characterisation is that every deviation from a strict linear order that maintains $|E| = \binom{|V|}{2}$ increases the number of homomorphic matches of P_2. As an example, consider the following three graphs G_1, G_2, G_3 with three vertices each.



The graph G_1 is a strict total order and has the minimal number $\binom{3}{3} = 1$ of homomorphic matches of P_2. Inverting an edge, as in G_2, increases the number of matches to 3. The same happens when reflexive loops or cycles of length 2 are introduced: G_3 has 3 matches of P_2.

Notice that the values $|E|$ and $|\hom(P_2, G)|$ can be easily counted by an ACR-GNN. Indeed, sum aggregation can count the degree of each vertex thus allowing us to obtain $|E|$ by a single sum readout. The value $|\hom(P_2, G)|$ is obtained similarly only now two aggregation runs are needed before applying the readout (as the number of paths of length 2 starting from $v \in V$ is simply $\sum_{u \in N^G_{\text{out}}(v)} \sum_{w \in N^G_{\text{out}}(u)} 1$). The values $\binom{|V|}{2} = \frac{1}{2}|V|^2 + \frac{-1}{2}|V|$ and $\binom{|V|}{3} = \frac{1}{6}|V|^3 + \frac{-3}{6}|V|^2 + \frac{2}{6}|V|$ can be obtained by applying the sum readout function 2 and 3 times respectively (as $|V|^2 = \sum_{v \in V} \sum_{u \in V} 1$ and similarly for $|V|^3$). By computing the absolute values (via ReLU) of $|E| - \binom{|V|}{2}$ and $|\hom(P_2, G)| - \binom{|V|}{3}$ we classify G as a linear order if and only if their sum is non-positive.

We mention that our results also hold in the more typical setting of undirected graphs.

Theorem 3. *There is an FO vertex property over undirected graphs that is not equivalent to any C^2 formula, but is captured by a simple ACR-GNN.*

Of course there is no proper notion of a strict linear order in the undirected setting. We thus use *gadgetisations* of linear orders in which each vertex v is split into two nodes s_v, t_v and a directed edge such as (u, v) is represented as an undirected edge (s_u, t_v). Note that the gadgetisation is different from that of Hauke and Wałęga [2] in order to be compatible with Lemma 2.

Our final contribution concerns the scenarios where a characterisation of ACR-GNNs (modulo FO) in terms of fragments of C^2 is possible. Turning to the setting of graphs of bounded degree (but not necessarily bounded size) we show the following.

Theorem 4. *Over graphs of bounded degree, an FO classifier φ is captured by an ACR-GNN if and only if φ is equivalent to a $\mathrm{GML}^{\exists}$ classifier.*

This contrasts Theorem 1 and Theorem 3, as over graphs of unbounded degree, ACR-GNNs can express properties outside $\mathrm{GML}^{\exists}$ and C^2. It is easy to see that every $\mathrm{GML}^{\exists}$ classifier can be captured by a simple ACR-GNN, even over all graphs, using the standard construction from [8].

To show the other direction of Theorem 4, we take a detour and first consider again graphs of unbounded degree. Say that an aggregation function is bounded if there is some $c \in \mathbb{N}$ such that the result of applying the function on any multiset of vectors $M : \mathbb{R}^d \to \mathbb{N}$ is the same as applying it on the multiset M', where $M'(x) = \min(c, M(x))$ for all $x \in \mathbb{R}^d$. We show the following characterisation for ACR-GNNs with bounded aggregation (but unbounded readout).

Proposition 5. *An FO classifier φ is captured by an ACR-GNN with bounded aggregation functions if and only if φ is equivalent to a $\mathrm{GML}^{\exists}$ classifier.*

Notice that the above generalises, for FO classifiers, the results by Cuenca Grau et al. [9], where the readout function is also taken to be bounded.

We argue that Proposition 5 implies Theorem 4.

Proof sketch of Theorem 4. Let c be the bound on the degree of vertices. We make use of a $\mathrm{GML}^{\exists}$-formula $\psi_c = \neg\exists_{\geq 1}\Diamond_{\geq c+1}\top$ that captures graphs of degree at most c.

If an FO classifier φ is equivalent to a $\mathrm{GML}^{\exists}$ classifier over graphs of bounded degree c, then $\varphi \wedge \psi_c$ is equivalent to a $\mathrm{GML}^{\exists}$ classifier over all graphs. By Proposition 5, $\varphi \wedge \psi_c$ is therefore captured by an ACR-GNN over all graphs. Hence, φ is captured by an ACR-GNN over all graphs of bounded degree c.

If φ is captured by an ACR-GNN over graphs of bounded degree c, then there is also an ACR-GNN with bounded aggregation that captures φ over graphs of bounded degree c. Furthermore, there is also an ACR-GNN with bounded aggregation that captures $\varphi \wedge \psi_c$ over all graphs. By Proposition 5, $\varphi \wedge \psi_c$ is therefore equivalent to a $\mathrm{GML}^{\exists}$ formula over all graphs, which means that in turn, φ is equivalent to a $\mathrm{GML}^{\exists}$ formula over graphs of bounded degree. □

The proof of Proposition 5 is then model theoretic in nature. In short, we introduce an appropriate notion of *bisimilarity* under which ACR-GNNs with bounded aggregation are invariant. We show that every FO formula that is invariant under this bisimilarity is then equivalent to a $\mathrm{GML}^{\exists}$ formula. Our notion of bisimilarity extends c-graded ℓ-bisimilarity [10] with a condition which demands that all c-graded ℓ-bisimulation types occur the same number of times in both graphs. To show our van-Benthem-Rosen style result, we then use an upgrading argument similar to those found in [11, 10]. An important difference is that standard operations such as unravelling and product do not preserve this kind of bisimilarity. In the upgrading argument we thus opt to, as it were, transform the graphs in place so that they meet our requirements. Perhaps the most intriguing part of our argument is the interplay between properties of ACR-GNN layers, bisimulations, and first-order logic, without which a van-Benthem-Rosen style result for $\mathrm{GML}^{\exists}$ (and, thus, a characterisation of ACR-GNNs with bounded aggregation in terms of a fragment of C^2) would not be possible.

We remark that while we have only introduced simple ACR-GNNs here, Theorem 4 and Proposition 5 apply also to ACR-GNNs with other aggregation, readout and combination functions.

4. Conclusion

In a sense, Theorems 1, 3 and 4 give us a lower and upper bounds on how far (fragments of) C^2 can be taken to capture the expressive power of ACR-GNNs relative to FO. Interesting questions remain for future research: Which logic exactly characterises the expressive power of ACR-GNNs? Here a good starting point could be the logic FO+C as used by Grohe [12].

Acknowledgments

Daumantas Kojelis was supported by DFG project LU 1417/4-1.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] M. Funk, D. Kojelis, Towards understanding the expressive power of GNNs with global readout, 2026. URL: https://arxiv.org/abs/2604.22870. arXiv:2604.22870.

[2] S. P. Hauke, P. A. Wałęga, Aggregate-combine-readout GNNs can express logical classifiers beyond the logic C2, Proceedings of the AAAI Conference on Artificial Intelligence 40 (2026) 21594–21601. URL: https://ojs.aaai.org/index.php/AAAI/article/view/39308. doi:10.1609/aaai.v40i26.39308.

[3] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, G. Monfardini, The graph neural network model, IEEE Trans. Neural Networks 20 (2009) 61–80. doi:10.1109/TNN.2008.2005605.

[4] D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik, R. P. Adams, Convolutional networks on graphs for learning molecular fingerprints, in: Advances in Neural Information Processing Systems 28, 2015, pp. 2224–2232. URL: https://proceedings.neurips.cc/paper/2015/hash/f9be311e65d81a9ad8150a60844bb94c-Abstract.html.

[5] M. Zitnik, M. Agrawal, J. Leskovec, Modeling polypharmacy side effects with graph convolutional networks, Bioinform. 34 (2018) i457–i466. URL: https://doi.org/10.1093/bioinformatics/bty294. doi:10.1093/BIOINFORMATICS/BTY294.

[6] A. Deng, B. Hooi, Graph neural network-based anomaly detection in multivariate time series, in: Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI Press, 2021, pp. 4027–4035. URL: https://doi.org/10.1609/aaai.v35i5.16523. doi:10.1609/AAAI.V35I5.16523.

[7] W. Jiang, J. Luo, Graph neural network for traffic forecasting: A survey, Expert Syst. Appl. 207 (2022) 117921. URL: https://doi.org/10.1016/j.eswa.2022.117921. doi:10.1016/J.ESWA.2022.117921.

[8] P. Barceló, E. V. Kostylev, M. Monet, J. Pérez, J. Reutter, J. P. Silva, The logical expressiveness of graph neural networks, in: International Conference on Learning Representations, 2020. URL: https://openreview.net/forum?id=r1lZ7AEKvB.

[9] B. Cuenca Grau, E. Feng, P. A. Walega, The correspondence between bounded graph neural networks and fragments of first-order logic, in: Proccedings of the Fortieth AAAI Conference on Artificial Intelligence, 2026, pp. 19135–19142. doi:10.1609/AAAI.V40I23.38987.

[10] M. Otto, Graded modal logic and counting bisimulation, 2023. URL: https://arxiv.org/abs/1910.00039. arXiv:1910.00039.

[11] M. Otto, Modal and guarded characterisation theorems over finite transition systems, Annals of Pure and Applied Logic 130 (2004) 173–205. URL: https://www.sciencedirect.com/science/article/pii/S0168007204000685. doi:https://doi.org/10.1016/j.apal.2004.04.003, papers presented at the 2002 IEEE Symposium on Logic in Computer Science (LICS).

[12] M. Grohe, The descriptive complexity of graph neural networks, TheoretiCS Volume 3 (2024) 12238. doi:10.46298/theoretics.24.25.