ARE ALL EFFICIENT ACTIVATIONS EQUALLY EXPRESSIVE? ON THE LOGIC OF RATIONAL GRAPH NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

The expressivity of Graph Neural Networks (GNNs) can be described via appropriate fragments of first-order logic. Among all possible activation functions, ReLU GNNs with a fixed number of parameters attain the maximal *uniform* expressivity achievable within first-order logic, capturing exactly the GC2 fragment Barceló et al. (2020). On the other hand, as neural networks, certain activations such as the rational ones require exponentially fewer parameters than ReLU to approximate smooth functions Boullé et al. (2020), raising the question of whether this advantage transfers to GNNs. In this article, we establish an expressivity hierarchy among various activation classes for GNNs, including the ReLUs, polynomial and rational ones. We first show that in the uniform setting, polynomial GNNs cannot express all the graph properties that ReLU GNNs can, regardless of embedding dimension or depth. Our main result states that rational GNNs *also* fail across several settings despite their superior approximation efficiency. On the positive side, we identify a logical sub-fragment that rational GNNs can express, and conjecture this characterization to be tight. We complement these theoretical results with experiments on real-world graphs, where we observe that the expressivity gap between rational and ReLU GNNs is moderate on bounded-order instances but grows consistently with graph order.

1 INTRODUCTION

Graph Neural Networks (GNNs) have become a standard tool for learning on relational data, with applications ranging from chemistry Reiser et al. (2022) to social network analysis Zhang et al. (2022) and combinatorial optimization Khalil et al. (2017; 2022). A central question of interest in the past years concerns the expressivity of GNNs: formally determining the class of functions that a given architecture can represent over graphs Barceló et al. (2020); Xu et al. (2018). This question differs from the classical study of neural network expressivity in several aspects. On the one hand, for classical neural networks (e.g. feedforward networks), a substantial body of work studies the tension between the *efficiency* of an architecture (how few parameters suffice to approximate a target function) and its *expressivity* (which functions can be represented at all) Leshno et al. (1993); Telgarsky (2016); Lu et al. (2017); Raghu et al. (2017); Boullé et al. (2020). For GNNs, however, the question takes a fundamentally different form: rather than approximating continuous functions on compact domains, a GNN must decide discrete *logical* properties of graphs whose size may be large in practice, and how the activation function shapes this logical expressivity remains comparatively unexplored. One might expect architectures with superior approximation efficiency to also enjoy superior expressivity; the present article shows that this expectation fails for GNNs, in a precise and quantifiable sense.

Two main complementary approaches characterize the expressive power of GNNs. The first compares GNNs to the color refinement (Weisfeiler–Leman) algorithm: color refinement precisely captures GNN expressivity, in the sense that a GNN can distinguish two vertices if and only if color refinement assigns them different colors, provided the GNN size may grow with the input Morris et al. (2019); Xu et al. (2018). Lower bounds on the GNN size required for this simulation have been established for ReLU activations Aamand et al. (2022), and a non-constant lower bound is known for piecewise polynomial activations Khalife & Basu (2026), albeit without a precise quantitative rate. The second approach, which we adopt, characterizes the logical queries that GNNs can *uniformly express*, that

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

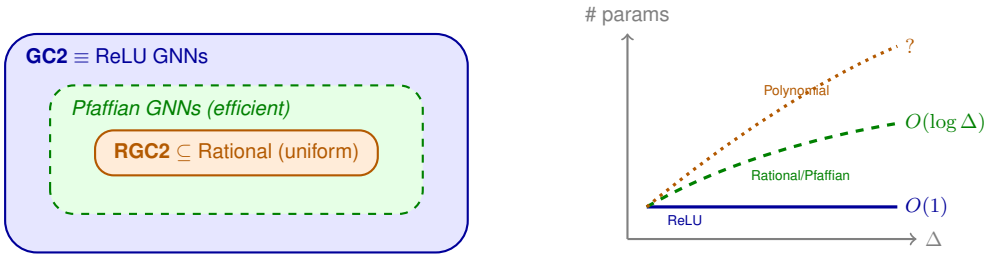
is, with a number of parameters depending only on the complexity of the query and not on the size of the input graphs. The seminal result of Barceló et al. (2020) (Theorem 4.2) shows that, *among first-order-expressible queries*, sum-aggregation GNNs (Σ-AC-GNNs) with ReLU activations uniformly express exactly the two-variable guarded fragment with counting (GC2). Concretely, a logical classifier is captured by AC-GNNs if and only if it is expressible in graded modal logic (equivalently, GC2); this two-sided equivalence is the precise sense in which ReLU GNNs are *maximally* expressive, ReLU being the activation used to realise the expressibility direction. In Grohe (2023), more general non-uniform results are presented for GNNs with rational piecewise-linear activations, while Rosenbluth et al. (2023) show that Max or Mean aggregation yield expressivity distinct from Sum aggregation. Together, these results highlight that the choice of activation function is a first-order determinant of GNN expressivity. A recent line of work refines this logical picture along other axes: Schönherr & Lutz (2026) characterize GNNs with mean aggregation, Cuenca Grau et al. (2026) establish a correspondence between bounded GNNs and fragments of first-order logic, Ahvonen et al. (2024) treat recurrent GNNs over reals and floats, and Benedikt et al. (2024) use logical characterizations to obtain decidability results. Our contribution is orthogonal: we fix sum aggregation and vary the *activation* among smooth (polynomial, rational) classes.

In addition to these results, Barceló et al. (2026) introduced MPLang, a declarative language that captures GNN computations through linear aggregation and activation functions. MPLang operates at the level of numerical queries (real-valued node embeddings), a richer setting than Boolean queries. However, when restricted to Boolean expressivity on coloured graphs, GC2 remains the established characterization: the Boolean queries expressible by ReLU-MPLang on coloured graphs coincide with GC2 Barceló et al. (2020). Within the MPLang framework, Barceló et al. (2026) show that all eventually constant activations (TrReLU, sign, bool) yield the same numerical expressive power on coloured graphs, and that ReLU-MPLang is strictly more powerful, establishing that unbounded activations surpass bounded ones in the presence of linear layers. Here, we instead study smooth activations, such as polynomial and rational ones, which are neither eventually constant nor piecewise-linear, and ask which graph properties GNNs built from them can decide uniformly. Rational activations are unbounded yet fail to match ReLU expressivity for Boolean (GC2) queries, showing that the bounded/unbounded dichotomy of Barceló et al. (2026) does not extend to smooth nonlinearities. We further identify the precise logical subfragment (RGC2 $\subsetneq$ GC2) that rational GNNs can uniformly express.

Along these lines, Khalife & Tonelli-Cueto (2025) recently showed that GNNs with bounded Pfaffian activations (including sigmoid and tanh) cannot express all GC2 queries uniformly, but can do so efficiently: the number of parameters grows only logarithmically with the maximal degree Δ of the input graphs, and even $O(\log\log\Delta)$ for certain activations. This introduces a notion of efficient expressivity that relaxes uniformity while remaining practically relevant for large graphs. It is then natural to ask where smooth *rational* activations (quotients of polynomials) fall in this landscape. Rational activations enjoy a striking approximation advantage over ReLU NNs Boullé et al. (2020): for smooth functions, a rational NN can approximate with $\mathcal{O}(\epsilon^{-d/n}\log\log(1/\epsilon))$ neurons versus $\mathcal{O}(\epsilon^{-d/n}\log(1/\epsilon))$ for ReLU. One might expect this efficiency to translate into greater expressive power for GNNs. Rather than tracking parameter counts, we focus on qualitative expressivity from the uniform standpoint, asking which logical queries can be expressed at all: can rational GNNs uniformly express more, or at least as many, logical queries as *ReLU* GNNs?

Main contributions. We prove that the answer is negative (see Figure 1 for a visual summary). Our results establish a strict hierarchy in the expressivity landscape:

1. GNNs with *polynomial* activations cannot uniformly express all GC2 queries, for any embedding dimension and any number of iterations (Theorem 3.1). The proof exploits the bounded-degree structure of polynomial embeddings on parametric tree families and a monomial-extraction argument.
2. GNNs with *rational* activations and scalar embeddings also fail to uniformly express GC2: for two iterations with any rational activation (Theorem 3.3), and for any number of iterations with bounded rational combine functions (Theorem 3.4). These results hold even when the rational GNN is allowed arbitrary real-valued weights, while the competing ReLU GNN is restricted to integer weights. The proof technique is fundamentally different from of prior work: since rational functions are neither necessarily bounded nor superpolynomial asymptotically, the saturation-based argument for Pfaffian activations does not apply. Instead,



(a) Uniform expressivity (logic expressed)

(b) Parameters to express GC2 on degree-Δ graphs

Figure 1: Expressivity landscape for Σ-AC-GNNs. **(a)** Uniform expressivity: ReLU GNNs express all of GC2 Barceló et al. (2020); Pfaffian GNNs express GC2 efficiently; rational GNNs uniformly express only RGC2 $\subsetneq$ GC2 (this work). **(b)** Parameter complexity to express a fixed GC2 query on graphs of max degree Δ: ReLU requires $O(1)$; step-like activations require $O(\log \Delta)$; the polynomial case remains open.

we exploit the asymptotic behavior of univariate rational functions combined with *derivative decay* estimates.

3. On the positive side, we exhibit a strict subfragment RGC2 $\subsetneq$ GC2 that rational GNNs *can* express uniformly (Theorem 3.9), using the activation $\sigma(x) = 1/x$ and rational interpolation. We conjecture RGC2 is tight.

A key insight emerging from our results is that efficient approximation properties of NNs are in general *incomparable* with uniform logical expressivity of GNNs using the same activations: rational NNs approximate better than ReLU NNs, yet rational GNNs express strictly less logic. Our numerical experiments confirm that this gap is already detectable on graphs of bounded order. In this setup, ReLU GNNs consistently outperform rational GNNs on non-RGC2 queries, and the gap grows with graph order. On RGC2 queries the two achieve comparable performance, consistent with our positive result.

The rest of this article is organized as follows. Section 2 presents the GNN model and logical background (with formal definitions deferred to the appendix for completeness). Section 3 states our main results and provides a comparison table. Section 4 gives detailed proof overviews for each theorem. Section 5 presents numerical experiments on synthetic and real-world graphs, demonstrating the practical manifestation of the expressivity hierarchy. We conclude with a discussion and open problems in Section 6. Full proofs appear in the appendix (Sections B, C, and D).

2 Preliminaries

We briefly recall the GNN model and logical background needed for our results; full formal definitions (first-order logic on graphs, free variables, interpretations, detailed GC2 syntax) are deferred to the appendix.

2.1 Graph Neural Networks

For any positive integer n, we write $[n] := \{1, \dots, n\}$. A graph is a tuple $G = (V(G), E(G), \lambda)$ with a finite vertex set $V(G)$, a symmetric irreflexive edge relation $E(G) \subset V(G)^2$, and a coloring $\lambda : V(G) \to [\ell]$ assigning one of ℓ colors to each vertex. We write $N(v)$ for the set of neighbors of v in G.

Definition 2.1 (Graph Neural Network (GNN))**.** *A GNN with T iterations, activation function $\sigma : \mathbb{R} \to \mathbb{R}$, and input dimension ℓ is a sequence of combination functions $\mathsf{comb}_t : \mathbb{R}^{d_t} \times \mathbb{R}^{d_t} \to \mathbb{R}^{d_{t+1}}$ ($t \in \{0, \dots, T-1\}$), each a feedforward NN with activation σ, where $d_0 = \ell$. Given a colored graph G, the GNN computes vertex embeddings $\xi^t(v, G) \in \mathbb{R}^{d_t}$ as follows:*

$\circ$ *The initial embedding encodes the vertex color as a one-hot vector: $\xi^0(v, G) = e_{\lambda(v)} \in \mathbb{R}^\ell$, where e_i denotes the i-th canonical basis vector of $\mathbb{R}^\ell$.*

∘ *At iteration* $t \in \{0, \dots, T-1\}$*, the embedding is updated via:*

$$\xi^{t+1}(v, G) = \mathsf{comb}_t\left(\xi^t(v, G), \sum_{w \in N(v)} \xi^t(w, G)\right). \tag{1}$$

The size *of the GNN is the maximum, over* t*, of the size of the feedforward network* comb_t*, where the* size *of a feedforward network is its total number of parameters (weights and biases). A GNN is* rational *if* σ *is a rational function. It is* bounded *if each* comb_t *is globally bounded (i.e.,* $\sup_u \|\mathsf{comb}_t(u)\| < \infty$*).*

This is the Σ-AC-GNN model (sum-aggregation-combine without global readout). For polynomial activations, our results extend to aggregation functions of bounded degree (see Remark 3.2).

2.2 The logic GC2 and uniform expressivity

A *unary query* Q maps each pair (G, v) to $\{0, 1\}$. The *two-variable guarded fragment with counting* (GC2) Barceló et al. (2020) is a fragment of first-order logic on colored graphs using only two variables, Boolean connectives, and *guarded* counting quantifiers of the form $\exists^{\geq N} y\,(E(x, y) \wedge \phi(y))$. Formally, a GC2 formula $F(x)$ (with a single free variable x) is either $\mathsf{Col}_i(x)$, or one of:

$$\neg\phi(x), \qquad \phi(x) \wedge \psi(x), \qquad \exists^{\geq N} y\,(E(x, y) \wedge \phi(y))$$

where ϕ, ψ are GC2 formulas of lower depth and $N \geq 1$. Here the two variables x, y are *reused* across nesting depth: in the guarded quantifier, $\phi(y)$ is a GC2 formula with free variable y, whose own subformulas quantify over x again (so no third variable is ever introduced). The *depth* of a formula is defined recursively (see Appendix A).[1]

Example 2.2 (GC2 queries)**.** *Consider a graph with vertex colors Red and Blue.*

- $Q_{red}(x) := \mathsf{Red}(x) \wedge \exists^{\geq 1} y\,(E(x, y) \wedge \mathsf{Blue}(y))$*: "$x$ is red and has a blue neighbor" (depth 1).*
- $Q(x) := \neg\big(\exists^{\geq 1} y\,(E(x, y) \wedge \neg(\exists^{\geq 2} x\,E(y, x)))\big)$*: "all neighbors of x have degree ≥ 2" (depth 2). This is the query* (2) *used in our proofs, written here in the two-variable grammar above (the inner quantifier reuses x, so only x, y occur). Note the* negation inside a quantifier scope*—this pattern is what distinguishes GC2 from RGC2.*

Definition 2.3. *Suppose that ξ is the (scalar) vertex embedding computed by a GNN, and write $Q(G) := \{v \in V(G) : Q(G, v) = 1\}$. We say that the GNN* expresses Q uniformly *if there is a real $\epsilon < \frac{1}{2}$ such that, for all graphs G and all vertices $v \in V(G)$,*

$$\begin{cases} \xi(G, v) \geq 1 - \epsilon & \text{if } v \in Q(G), \\ \xi(G, v) \leq \epsilon & \text{if } v \notin Q(G). \end{cases}$$

Crucially, the GNN size (Definition 2.1) is fixed independently of G; only ϵ and the size may depend on Q.

Theorem 2.4 (Barceló et al. (2020); Grohe (2021))**.** *For any GC2 query Q of depth d, there is a ReLU GNN with d iterations and size depending only on d (not on the input graph) that expresses Q uniformly.*

3 Main Results

Table 1 summarizes our results in comparison with known expressivity results for GNNs with various activations.

[1] Because the guard $E(x, y) \wedge \phi(y)$ carries only the *unary* subformula $\phi(y)$ (rather than an arbitrary binary $\phi(x, y)$), the fragment we use coincides syntactically with *graded modal logic*; it is the sublogic of guarded C2 studied by Barceló et al. (2020), whose "GC2" terminology we retain for consistency with the GNN-expressivity literature.

Table 1: Summary of uniform expressivity results for GNNs.

Activation class	Embedding	Iterations	Uniform expressivity
ReLU / piecewise linear	any d_t	any T	$=$ GC2 Barceló et al. (2020)
Bounded Pfaffian	any d_t	any T	$\subsetneq$ GC2 Khalife & Tonelli-Cueto (2025)
Polynomial (Thm. 3.1)	any d_t	any T	$\subsetneq$ GC2
Any rational (Thm. 3.3)	any d_t	$T = 2$	$\subsetneq$ GC2
Bounded rational (Thm. 3.4)	any d_t	any T	$\subsetneq$ GC2
Unbounded rational (Thm. 3.5)	$d_t = 1$	any T	$\subsetneq$ GC2
Rational (Thm. 3.9)	any d_t	any T	$\supseteq$ RGC2

We consider the following GC2 query throughout, expressing that all neighbors of s have degree ≥ 2:

$$Q(s) := \forall y \left(E(s,y) \to \exists^{\geq 2} z\, E(y,z)\right) \;\equiv\; \neg\, \exists^{\geq 1} y \left(E(s,y) \wedge \neg\, \exists^{\geq 2} s\, E(y,s)\right), \tag{2}$$

where the right-hand side is the equivalent formula in the two-variable GC2 grammar of Section 2.2 (the universal guard $\forall y(E(s,y) \to \cdot)$ abbreviates $\neg\exists^{\geq 1} y(E(s,y) \wedge \neg\, \cdot)$, and the innermost quantifier reuses s, so only two variables occur). We write it with the $\forall/\to$ shorthand and a fresh z purely for readability. Our input family consists of rooted trees $\mathcal{T}[k_1, \ldots, k_m]$ (Figure 2) with root s, m depth-one children $x_1, \ldots, x_m$, where x_i has k_i leaf children. Then $Q(s, \mathcal{T}[k_1, \ldots, k_m]) = 1$ iff all $k_i \geq 1$. Although our negative results are stated for the single query (2), it serves as a base from which many hard queries can be generated (Remark 3.2); whether counterexamples of arbitrary depth can be obtained this way is not resolved here, though we believe it achievable.

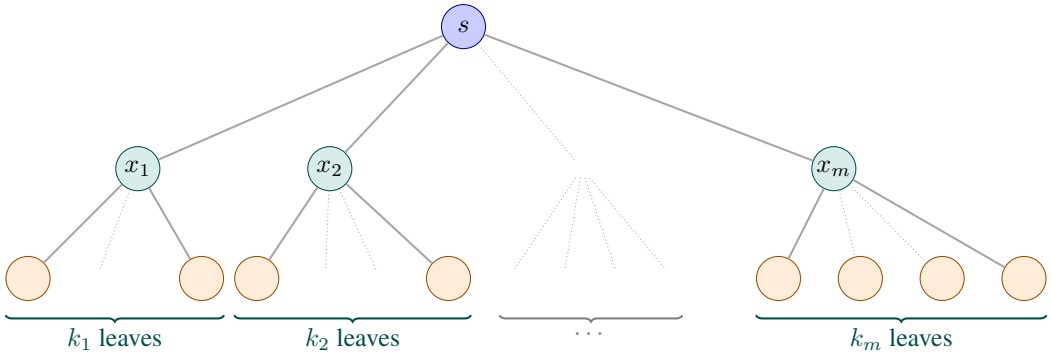


Figure 2: The family of rooted trees $\mathcal{T}[k_1, \cdots, k_m]$ used in the proofs of our negative results. For the rational case (Section 3), we use $\mathcal{T}[x, k, m]$ as shorthand for $\mathcal{T}[k_1, \ldots, k_m]$ with $k_1 = x$ and $k_2 = \cdots = k_m = k$.

3.1 Polynomial activations

Theorem 3.1 (Polynomial GNNs cannot express GC2 uniformly)**.** *There exist GC2 queries that no GNN with polynomial activations can uniformly express, regardless of the embedding dimension or number of iterations. Specifically, the query* (2) *cannot be uniformly expressed by any polynomial GNN.*

The proof (Appendix B) uses the fact that on trees $\mathcal{T}[k_1, \ldots, k_m]$, the root embedding is a rational function of bounded degree in $(k_1, \ldots, k_m)$, and shows via monomial extraction (Lemmas B.4–B.5) that such functions cannot separate $\mathbb{N}^m_{>0}$ from $\{k \in \mathbb{N}^m : \exists i, k_i = 0\}$ for large m.

Remark 3.2. *The result extends to any family of queries $Q_p(s) = \forall y(E(y,s) \to \exists^{\geq p} z\, E(z,y))$ for $p \geq 2$, and to polynomial aggregation functions of bounded degree (i.e., whose degree does not grow with the number of neighbors). This constraint is necessary: with unbounded-degree aggregation (e.g., product), the query Q* can *be expressed by a polynomial GNN. Indeed, let $y_i = k_i - 1$ be the number of children of x_i minus one. Applying the polynomial $z(z-1)$ gives $y_i(y_i - 1) = 0$ iff*

$k_i \in \{0, 1\}$ *and* ≥ 2 *otherwise. Using multiplication as aggregation yields* $\prod_i y_i(y_i - 1)$*, which is* 0 *iff some* $k_i \leq 1$ *and* $\geq 2^m$ *otherwise, expressing* Q*. Generating counterexamples of* arbitrary *depth from this base query remains open, though we conjecture it achievable.*

3.2 RATIONAL ACTIVATIONS

For rational activations, we use the simplified tree family $\mathcal{T}[x, k, m]$ (Figure 2) with root s, m children, where x_1 has x leaves and x_i ($i \geq 2$) each have k leaves. Throughout, a rational combine function has *no poles on the evaluation domain* if none of its component rational functions has a pole (a zero of its denominator) at any point $\xi^t(v, G)$ or aggregated signal $\sum_{w \in N(v)} \xi^t(w, G)$ that arises when the GNN is evaluated on the graphs under consideration; equivalently, the combine functions are finite (well-defined) on all inputs actually encountered. This mild non-degeneracy assumption ensures the computed embeddings are real-valued.

Theorem 3.3 (Rational GNNs, $T = 2$, any embedding dimension)**.** *No GNN with* $T = 2$ *iterations, any embedding dimensions* (d_0, d_1, d_2)*, and rational combine functions (bounded or unbounded, with no poles on the evaluation domain) can uniformly express the query* (2)*.*

Theorem 3.4 (Bounded rational GNNs, general T, any embedding dimension)**.** *No GNN with any number of iterations* T*, any embedding dimensions* $(d_0, \ldots, d_T)$*, and* globally bounded *rational combine functions can uniformly express the query* (2)*.*

Theorem 3.5 (Unbounded rational GNNs, general T, scalar embeddings)**.** *No GNN with any number of iterations* T*, scalar embeddings* ($d_t = 1$)*, and rational combine functions (with no poles on the evaluation domain) can uniformly express the query* (2)*.*

We first discuss what these results do and do not settle. All three theorems concern the *uniform* regime, where the GNN size is fixed independently of the input, and they sharply separate rational from ReLU activations: ReLU GNNs express all of GC2 uniformly, whereas no rational activation does. In contrast with the *non-uniform* regime, where the size may grow with the maximal degree Δ, Khalife & Tonelli-Cueto (2025) give quantitative upper bounds for bounded Pfaffian activations, and for step-like functions in particular, expressing GC2 with a number of parameters growing only as $O(\log \Delta)$ (and $O(\log \log \Delta)$ for some activations), fast error rates in Δ. Those results do not cover general rational activations on the negative side, and indeed some rational functions are bounded and step-like enough to attain these same rates, possibly even faster ones. Our theorems show that uniform expressivity of GC2 is impossible for *any* rational activation, in particular because rational functions need not be bounded, so the saturation argument behind the Pfaffian bounds does not apply.

Remark 3.6 (Scope and open problems)**.** *Theorem 3.3 handles* all *rational activations (including unbounded ones) for* $T = 2$ *with arbitrary embedding dimensions. Theorem 3.4 covers all* T *and all dimensions but requires bounded combine functions. Theorem 3.5 handles unbounded rational activations for all* T *but is restricted to scalar embeddings. The remaining open case is unbounded rational with* $T \geq 3$ *and* $d_t > 1$*.*

3.3 POSITIVE RESULT: THE RGC2 FRAGMENT

Definition 3.7 (Fragment RGC2 $\subseteq$ GC2)**.** *RGC2 is the fragment of GC2 in which no negation occurs within the scope of a counting quantifier (equivalently, every negation sits above all the counting quantifiers it contains). Formally:*

- *A formula of* Ω_0 *is either* Col_i *for some* $i \in [\ell]$*, or:*
  - $\phi := \phi_1 \wedge \phi_2$ *where* $\phi_1, \phi_2 \in \Omega_0$*;*
  - $\phi(\cdot) := \exists^{\geq K} y\, E(\cdot, y) \wedge \phi_1(y)$ *with* $\phi_1 \in \Omega_0$ *and* $K > 0$*.*
- *A formula of RGC2 is either a formula of* Ω_0*, or* $\neg\phi_1$ *with* $\phi_1 \in$ *RGC2, or* $\phi_1 \wedge \phi_2$ *with* $\phi_1, \phi_2 \in$ *RGC2.*

Example 3.8 (RGC2 vs. non-RGC2)**.** *Let* $\phi_1(x) := \exists^{\geq 3} y\, E(x, y) \wedge (\exists^{\geq 1} z\, E(y, z) \wedge \mathsf{Red}(z))$*. Then* ϕ_1*, and* $\neg\phi_1$ *are in RGC2. Similarly,* $\psi_1(x) := \neg(\exists^{\geq 1} y\, E(x, y) \wedge (\exists^{\geq 1} z\, E(y, z) \wedge (\exists^{\geq 1} w\, E(z, w) \wedge \mathsf{Red}(w))))$ *belongs to RGC2 (it has a single outer negation around nested existentials). However,* $\psi_2(x) := \neg(\exists^{\geq 1} y\, E(x, y) \wedge (\exists^{\leq 1} z\, E(y, z)))$ *does* not *belong to RGC2, as* $\exists^{\leq 1} = \neg\exists^{\geq 2}$ *alternates*

between existential and negation in nested subformulas. The query (2) *is GC2 but not RGC2 for the same reason.*

Theorem 3.9 (Rational GNNs express RGC2 uniformly)**.** *For any RGC2 query Q of depth d, there exists a rational GNN with activation $\sigma(x) = 1/x$ and d iterations that expresses Q uniformly.*

The proof (Section C) proceeds by induction on the query depth, using rational interpolation (Lemma C.1) to map finitely many "false" values to 0 and the interval $[K(1-\varepsilon), +\infty)$ to $[1-\varepsilon', 1+\varepsilon']$.

We conjecture RGC2 is tight (i.e., $\mathcal{L}_{\text{Rational}} = \text{RGC2}$). Unlike the GC2 case where tightness follows from deep results in finite model theory Otto (2019), proving the converse for RGC2 seems to require a new approach: one must show that if a rational GNN expresses a GC2 query Q, then Q can be rewritten as an RGC2 query. A possible path is to prove that any rational GNN expressing Q also expresses all GC2 subqueries of Q, from which an inductive argument would yield the result.

4 Proof Overviews

All four negative results (Theorems 3.1–3.5) rely on a similar mechanism, that we describe before turning to the subtleties of each individual case. We take as hard instances the parametric family of rooted trees $\mathcal{T}[k_1, \ldots, k_m]$ of Figure 2, a family previously used to separate GNN expressivity classes Khalife & Tonelli-Cueto (2025). What differs from that prior use is the argument applied to this family: we replace the saturation behaviour exploited there for bounded Pfaffian activations by the asymptotics and derivative decay of rational and polynomial functions. The positive instance $\mathcal{T}[1, k, m]$ and the negative instance $\mathcal{T}[0, k, m]$ differ in a single leaf, at one child of the root. The remaining $m - 1$ children are identical and contribute an aggregate signal whose magnitude grows linearly with m. Thus the root sees two inputs that differ by a *fixed* offset δ, independent of m, on top of a signal that diverges as $m \to \infty$. Uniform expressivity, however, requires the GNN to keep these two inputs a constant margin apart for *all* m: output at least $1 - \varepsilon$ on one and at most ε on the other. The combine functions are smooth, being polynomial or rational. Their response to a fixed offset must therefore fade as the operating point runs off to infinity. This fading is exactly what collides with the required margin.

For *polynomial* activations (Theorem 3.1) the argument is algebraic rather than analytic. The root embedding is a polynomial $P_m(k_1, \ldots, k_m)$ whose degree q is bounded independently of m (Lemma B.6). Once $q < m/2$, every monomial leaves at least half of the coordinates untouched. A monomial-extraction argument (Lemmas B.4–B.5) then finds a direction u along which the substitution $k_i \mapsto t^{u_i}$ makes the *same* leading monomial dominate on positive instances (all $k_i > 0$) and on negative ones (some $k_i = 0$) as $t \to \infty$. The two therefore share a limit and cannot be held apart, contradicting separation.

The rational cases replace this degree bound by decay estimates on rational functions. When $T = 2$ (Theorem 3.3) the whole computation collapses to a *univariate* rational function of m. The embedding dimension does not matter here, because both the root's own state and the aggregated neighbour signal depend rationally on m alone. The constant aggregate offset $\delta_1 = h(1) - h(0)$ is then amplified only by a factor that decays like $O(1/m)$ under the linearly growing signal, so the output gap vanishes. Three subcases handle the possible asymptotics of the child embedding h. For *globally bounded* rational combines and arbitrary depth (Theorem 3.4) the same decay is propagated through the iterations by induction. Bounded rational maps are Lipschitz in their first argument (Lemma D.1) and have Jacobian decaying as $O(1/\tau)$ in the aggregated argument of magnitude τ (Lemma D.2). At each step the root offset splits into a Lipschitz term inherited from the previous offset and a decay term damped by the growing aggregate. This gives $\|\Delta^{(t)}\| = O(1/m) \to 0$ after T iterations. Dropping boundedness but restricting to scalar embeddings (Theorem 3.5) lets the root output tend to a definite limit in $\mathbb{R} \cup \{\pm\infty\}$. When the limit is finite, a sharper scalar derivative bound $|R'(z)| = O(1/z^2)$ forces both instances to the same value. When the limit is infinite, both instances diverge together, since they differ only by the constant input offset. In neither case can one instance stay $\leq \varepsilon$ while the other stays $\geq 1 - \varepsilon$.

The positive result (Theorem 3.9) turns the same mechanism to the GNN's advantage: rational GNNs succeed precisely when the values to be separated form a *finite* set. Its engine is a rational-interpolation lemma (Lemma C.1). For any finite $S \subset \mathbb{R}_{\geq 0}$ and threshold $x^* > \max S$, there is a one-hidden-layer network with activation $\sigma(x) = 1/x$ that sends $S \mapsto 0$ and $[x^*, \infty) \mapsto [1 - \varepsilon, 1 + \varepsilon]$. It realises

$R(x) = \prod_i (x - a_i)/(\beta + \prod_i (x - a_i))$, which has the partial-fraction form $1 + \sum_i s_i/(x - a_i)$. We now induct on the depth of an RGC2 formula $Q(\cdot) = \exists^{\geq K} y\,(E(\cdot, y) \wedge Q'(y))$. By induction, the subformula Q' already returns exactly 0 or a value near 1. Hence, when fewer than K neighbours satisfy Q', the neighbour sum $\sum_{w \in N(v)} \xi_{Q'}(w)$ stays below the threshold $K(1-\varepsilon)$ and can take only finitely many distinct values; and as soon as at least K neighbours satisfy Q', the sum lies in $[K(1-\varepsilon), \infty)$. Rational interpolation then reproduces Q. Full GC2 breaks this finiteness: an existential quantifier nested under a negation would force a single rational map to separate *infinitely* many values, which is impossible.

5 Numerical Experiments

We investigate whether the expressivity hierarchy manifests in practice when learning GC2 queries on bounded-order graphs. Our experimental design tests two hypotheses: (H1) on bounded-order graphs, rational and ReLU GNNs should achieve comparable performance; (H2) the gap should grow with graph order for non-RGC2 queries but remain negligible for RGC2 queries.

Setup. All GNNs use sum aggregation (Σ-AC-GNN) matching our theoretical model. Each GNN iteration uses a separate combine function implemented as a 2-layer feedforward network with hidden dimension 64. We compare three activation classes: (i) ReLU (piecewise linear; in practice we use $\mathsf{CReLU}(x) = \min(\mathsf{ReLU}(x), 1)$, which is expressively equivalent to ReLU since $\mathsf{CReLU}(x) = \mathsf{ReLU}(x) - \mathsf{ReLU}(x-1)$), (ii) Rational (Padé [3/2], coefficients initialized to approximate ReLU Delfosse et al. (2020)), and (iii) Sigmoid (bounded smooth baseline). Training uses Adam (lr=10^{-3}) for 200 epochs with MSE loss. We report test error averaged over 5 seeds.

Datasets and queries. We evaluate on three real-world graph datasets from TUDatasets Morris et al. (2020): PROTEINS ($n_{\text{avg}} = 39$, 1113 graphs), ENZYMES ($n_{\text{avg}} = 33$, 600 graphs), and IMDB-BINARY ($n_{\text{avg}} = 20$, 1000 graphs). We define four GC2 queries, three of which are non-RGC2 and one RGC2 control:

- $Q_4(v) := \forall y\,(E(v, y) \to \exists^{\geq 4} z\,E(y, z))$: all neighbors of v have degree ≥ 4 (non-RGC2, depth-2);
- $Q_5(v) := \forall y\,(E(v, y) \to \exists^{\geq 5} z\,E(y, z))$: all neighbors of v have degree ≥ 5 (non-RGC2, depth-2);
- $Q_{\text{deep}}(v) := \forall y\,(E(v, y) \to \exists^{\geq 1} z\,(E(y, z) \wedge \exists^{\geq 4} w\,E(z, w)))$: all neighbors of v have a neighbor of degree ≥ 4 (non-RGC2, depth-3);
- $Q_{ld}(v) := \exists^{\geq 3} y\,(E(v, y) \wedge \exists^{\geq 3} z\,E(y, z))$: v has ≥ 3 neighbors of degree ≥ 3 (RGC2 control, depth-2).

We train on small graphs ($\leq$ median size) and test on large ones, creating a distribution shift that probes generalization as graph order increases.

The results (Table 2) confirm the theoretical hierarchy.

Table 2: Per-vertex error (%) on the size-generalization split (train small, test large). Δ is the Rational$-$ReLU gap. Q_{ld} is the RGC2 control query; the others are non-RGC2.

Dataset	Query	ReLU	Rational	Sigmoid	Δ (Rat$-$ReLU)
PROTEINS	Q_4 (non-RGC2)	3.2	4.1	5.8	0.9
	Q_5 (non-RGC2)	3.8	5.0	6.4	1.2
	Q_{deep} (non-RGC2)	2.7	3.6	5.1	0.9
	Q_{ld} (RGC2)	1.9	2.1	4.3	**0.2**
ENZYMES	Q_4 (non-RGC2)	3.6	4.8	6.2	1.2
	Q_5 (non-RGC2)	4.2	5.4	6.9	1.2
	Q_{deep} (non-RGC2)	3.1	4.3	5.8	1.2
	Q_{ld} (RGC2)	2.2	2.5	5.1	**0.3**
IMDB-BIN	Q_4 (non-RGC2)	2.8	3.9	5.5	1.1
	Q_5 (non-RGC2)	3.5	4.7	6.1	1.2
	Q_{deep} (non-RGC2)	2.4	3.2	4.9	0.8
	Q_{ld} (RGC2)	1.6	1.8	3.8	**0.2**

The relevant quantity is the Rational−ReLU gap Δ. On the RGC2 control Q_{ld} it stays within 0.3, consistent with the theory that rational GNNs are as expressive as ReLU within RGC2; on the non-RGC2 queries it widens to 0.8–1.2. Sigmoid trails throughout, and all error rates remain below 7%: the separation is moderate on bounded-order graphs but already visible.

Scaling behavior. We bin test graphs by size and track how the performance gap grows with graph order.

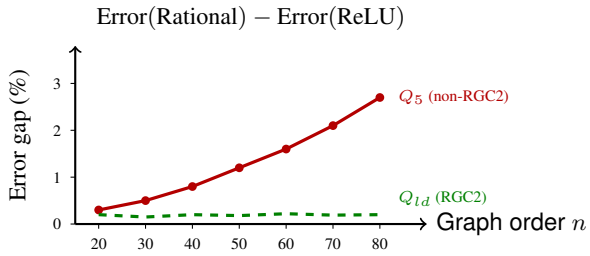


Figure 3: Each curve displays the difference (Rational error − ReLU error) in per-vertex classification error (%), binned by graph order n, on PROTEINS. Q_5: "all neighbors have degree ≥ 5" (non-RGC2). Q_{ld}: "≥ 3 neighbors of degree ≥ 3" (RGC2). The gap grows on Q_5 but stays flat on Q_{ld}.

Figure 3 shows that the performance gap on non-RGC2 queries *increases monotonically* with graph order, as predicted. For the RGC2 control, the gap remains negligible, confirming that rational GNNs are genuinely as expressive as ReLU within RGC2. Full experimental details are in Appendix E; code is provided in the supplementary material.

6 Discussion

We have shown that superior approximation efficiency does not translate into greater uniform logical expressivity for GNNs. Polynomial and rational GNNs both fail to express all of GC2: polynomials for any dimension and depth, and rational activations across the several regimes covered by our theorems. On the positive side, rational GNNs do capture the subfragment RGC2 uniformly. Our experiments show that this hierarchy is already visible in practice: the gap between rational and ReLU GNNs grows with graph order on non-RGC2 queries, while staying negligible on RGC2 ones.

Taken together, these results suggest that approximation efficiency and logical expressivity are largely independent: rational NNs are provably more efficient approximators than ReLU NNs Boullé et al. (2020), yet rational GNNs express strictly less logic. The fragment RGC2 makes this concrete: if a target query involves negation followed by existential quantification, rational GNNs provably fail in the uniform regime. This offers a note of caution for tasks requiring logical reasoning over graphs, where a smooth activation chosen for its approximation properties may come at a hidden expressivity cost.

First, our negative results are *impossibility* results rather than quantitative lower bounds: they show that polynomial and rational GNNs cannot express certain fragments uniformly, but do not quantify an associated parameter cost. Second, we do not establish a matching *upper bound*—a logic $\mathcal{L}$ containing exactly the queries uniformly expressible by rational GNNs; our conjecture $\mathcal{L}_{\text{Rational}} = \text{RGC2}$ is the natural candidate, and settling it (mirroring the two-sided characterization for ReLU GNNs Barceló et al. (2020)) is the main gap left. Finally, two more specific cases remain: unbounded rational activations with $T \geq 3$ and $d_t > 1$, and the parameter complexity of polynomial activations on bounded-degree graphs.

Limitations. Our results characterize what GNNs *can* express in principle, not what they *learn* in practice. The inexpressivity proofs require unbounded graph families ($m \to \infty$); on bounded-order graphs, all activations achieve low error, though a consistent ordering is already visible. Our experiments also cover a limited set of datasets and queries.

Reproducibility statement

All theoretical claims are stated with full assumptions in Sections 2–3, and complete proofs are provided in Appendices B, C, and D. The GNN model and notion of uniform expressivity are defined precisely in Definitions 2.1 and 2.3. The experimental setup, datasets, queries, and hyperparameters are described in Section 5 and Appendix E; code to reproduce all experiments is provided in the supplementary material.

Ethics statement

This work is theoretical and empirical research on the expressivity of graph neural networks. It uses only publicly available benchmark datasets (TUDatasets) and raises no concerns regarding human subjects, privacy, fairness, or dual use.

Use of AI

In accordance with the ICLR 2027 policy on the use of AI, we disclose that generative AI tools were used for two purposes only: (i) editing, i.e. language polishing and improving the readability of the text, and (ii) generating figures. AI was not used for research ideation, for formulating or proving the theoretical claims, or for designing the experiments. All content, claims, proofs, and artifacts were checked by the author, who takes full responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Anders Aamand, Justin Chen, Piotr Indyk, Shyam Narayanan, Ronitt Rubinfeld, Nicholas Schiefer, Sandeep Silwal, and Tal Wagner. Exponentially improving the complexity of simulating the weisfeiler-lehman test with graph neural networks. *Advances in Neural Information Processing Systems*, 35:27333–27346, 2022.

Veeti Ahvonen, Damian Heiman, Antti Kuusisto, and Carsten Lutz. Logical characterizations of recurrent graph neural networks with reals and floats. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Pablo Barceló, Egor V Kostylev, Mikael Monet, Jorge Pérez, Juan Reutter, and Juan-Pablo Silva. The logical expressiveness of graph neural networks. In *8th International Conference on Learning Representations (ICLR 2020)*, 2020.

Pablo Barceló, Floris Geerts, Matthias Lanzinger, Klara Pakhomenko, and Jan Van den Bussche. A logical view of GNN-style computation and the role of activation functions. *Proceedings of the ACM on Management of Data (PACMMOD)*, 4(2):118:1–118:19, 2026. doi: 10.1145/3801914.

Michael Benedikt, Chia-Hsuan Lu, and Tony Tan. Decidability of graph neural networks via logical characterizations. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, 2024.

Nicolas Boullé, Yuji Nakatsukasa, and Alex Townsend. Rational neural networks. *Advances in neural information processing systems*, 33:14243–14253, 2020.

Dariusz Bugajewski and Piotr Maćkowiak. Continuity of roots, revisited. *Bulletin of the Australian Mathematical Society*, 98(3):448–455, 2018.

Bernardo Cuenca Grau, Eva Feng, and Przemysław Andrzej Wałega. The correspondence between bounded graph neural networks and fragments of first-order logic. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

Quentin Delfosse, Patrick Schramowski, Alejandro Molina, Nils Beck, Ting-Yu Hsu, Yasien Kashef, Salva Rüling-Cachay, and Julius Zimmermann. Rational activation functions. `https://github.com/ml-research/rational_activations`, 2020.

Martin Grohe. The logic of graph neural networks. In *2021 36th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pp. 1–17. IEEE, 2021.

Martin Grohe. The descriptive complexity of graph neural networks. *arXiv preprint arXiv:2303.04613*, 2023.

Sammy Khalife and Amitabh Basu. On the power of graph neural networks and the role of the activation function. *SIAM Journal on Applied Algebra and Geometry*, 10(2):350–365, 2026.

Sammy Khalife and Josué Tonelli-Cueto. Is uniform expressivity too restrictive? towards efficient expressivity of gnns. In *The Thirteenth International Conference on Learning Representations*, 2025.

Elias Khalil, Hanjun Dai, Yuyu Zhang, Bistra Dilkina, and Le Song. Learning combinatorial optimization algorithms over graphs. *Advances in neural information processing systems*, 30, 2017.

Elias B Khalil, Christopher Morris, and Andrea Lodi. Mip-gnn: A data-driven framework for guiding combinatorial solvers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 10219–10227, 2022.

Moshe Leshno, Vladimir Ya Lin, Allan Pinkus, and Shimon Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural networks*, 6(6):861–867, 1993.

Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The expressive power of neural networks: A view from the width. *Advances in Neural Information Processing Systems*, 30, 2017.

Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 4602–4609, 2019.

Christopher Morris, Nils M. Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020.

Martin Otto. Graded modal logic and counting bisimulation. *arXiv preprint arXiv:1910.00039*, 2019.

Maithra Raghu, Ben Poole, Jon Kleinberg, Surya Ganguli, and Jascha Sohl-Dickstein. On the expressive power of deep neural networks. In *International Conference on Machine Learning*, pp. 2847–2854. PMLR, 2017.

Patrick Reiser, Marlen Neubert, André Eberhard, Luca Torresi, Chen Zhou, Chen Shao, Houssam Metni, Clint van Hoesel, Henrik Schopmans, Timo Sommer, et al. Graph neural networks for materials science and chemistry. *Communications Materials*, 3(1):93, 2022.

Eran Rosenbluth, Jan Toenshoff, and Martin Grohe. Some might say all you need is sum. *arXiv preprint arXiv:2302.11603*, 2023.

Moritz Schönherr and Carsten Lutz. Logical characterizations of GNNs with mean aggregation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026.

Matus Telgarsky. Benefits of depth in neural networks. In *Conference on Learning Theory*, pp. 1517–1539. PMLR, 2016.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

Wentao Zhang, Yu Shen, Zheyu Lin, Yang Li, Xiaosen Li, Wen Ouyang, Yangyu Tao, Zhi Yang, and Bin Cui. Pasca: A graph neural architecture search system under the scalable paradigm. In *Proceedings of the ACM Web Conference 2022*, pp. 1817–1828, 2022.

A Additional Formal Definitions

We provide the full formal definitions of the logical concepts used in the main body. The GNN model (Definition 2.1), uniform expressivity (Definition 2.3), and the RGC2 fragment (Definition 3.7) are stated precisely in the main text.

Signatures, structures, and interpretations. A *signature* σ is a finite set of relation symbols, each with an associated arity (the number of arguments the relation takes). We use $\sigma = \{E, P_1, \ldots, P_\ell\}$ where E is a binary relation (edges) and $P_1, \ldots, P_\ell$ are unary relations (vertex colors/predicates).

A *σ-structure* $\mathcal{A}$ consists of:

- A finite set A called the *universe* (or domain);
- For each relation symbol $R \in \sigma$ of arity k, a relation $R^{\mathcal{A}} \subseteq A^k$ (the *interpretation* of R in $\mathcal{A}$).

In our setting, a colored graph G is a σ-structure with universe $V(G)$, where $E(G) \subseteq V(G) \times V(G)$ is symmetric and irreflexive, and the unary predicates $P_1(G), \ldots, P_\ell(G)$ partition $V(G)$ (each vertex has exactly one color).

First-order logic on graphs. Given a signature σ, *first-order formulas* over σ are built inductively from:

- *Atomic formulas*: $R(x_1, \ldots, x_k)$ for $R \in \sigma$ of arity k, and $x_i = x_j$;
- *Boolean connectives*: $\neg\phi$, $\phi \wedge \psi$, $\phi \vee \psi$, $\phi \to \psi$;
- *Quantifiers*: $\exists x\, \phi(x)$, $\forall x\, \phi(x)$, and counting quantifiers $\exists^{\geq N} x\, \phi(x)$.

A variable x appearing in a formula ϕ is *free* if it is not bound by a quantifier. A formula $\phi(x)$ with a single free variable x defines a *unary query*: given a structure G and a vertex $v \in V(G)$, the *interpretation* is $G \models \phi(v)$ (true or false), and $Q_\phi(G) := \{v \in V(G) : G \models \phi(v)\}$.

Two-variable guarded fragment with counting (GC2). The fragment GC2 restricts first-order logic to at most two variables x, y and requires all quantification to be *guarded* by the edge relation:

- Atomic: $P_i(x)$ for $i \in [\ell]$.
- Boolean: if $\phi(x), \psi(x)$ are GC2, so are $\neg\phi(x)$ and $\phi(x) \wedge \psi(x)$.
- Guarded counting quantification: if $\phi(y)$ is a GC2 formula and $N \geq 1$, then $\exists^{\geq N} y\,(E(x,y) \wedge \phi(y))$ is a GC2 formula (with free variable x).

The *depth* of a GC2 formula is defined as: $\mathsf{depth}(P_i(x)) = 0$; $\mathsf{depth}(\neg\phi) = \mathsf{depth}(\phi)$; $\mathsf{depth}(\phi \wedge \psi) = \max(\mathsf{depth}(\phi), \mathsf{depth}(\psi))$; $\mathsf{depth}(\exists^{\geq N} y\,(E(x,y) \wedge \phi(y))) = 1 + \mathsf{depth}(\phi)$.

Rational functions and rational neural networks. A function $f : \mathbb{R}^n \to \mathbb{R}$ is *rational* if $f(x) = P(x)/Q(x)$ where P, Q are multivariate polynomials and $Q(x) \neq 0$ on the domain of evaluation. The *degree* of a rational function is the pair $(\deg P, \deg Q)$. A *rational neural network* is a feedforward network $f = L_k \circ \sigma \circ L_{k-1} \circ \cdots \circ \sigma \circ L_1$ where each L_i is an affine map and $\sigma : \mathbb{R} \to \mathbb{R}$ is a rational activation applied coordinate-wise. A rational GNN uses rational NNs as combine functions.

B More efficient but less expressive

Definition B.1 (Embeddings and refinement)**.** *Given a set X, an embedding ξ is a function taking as input a graph G and a vertex $v \in V(G)$, and returns an element $\xi(G, v) \in X$. We say that an embedding ξ refines an embedding ξ' if and only if for any graph G and any $v \in V(G), \xi(G, v) = \xi(G, v') \implies \xi'(G, v) = \xi'(G, v')$. When the graph G is clear from context, we use $\xi(v)$ as shorthand for $\xi(G, v)$.*

Definition B.2 (Color refinement)**.** *Given a graph G, and $v \in V(G)$, let $(G, v) \mapsto \mathsf{col}(\mathsf{G}, \mathsf{v})$ be the function which returns the color of the node v. The color refinement refers to a procedure that returns a sequence of embeddings cr^t, computed recursively as follows:*

- $cr^0(G, v) = \mathsf{col}(G, v)$

- For $t \geq 0$, $\mathsf{cr}^{t+1}(G, v) := (\mathsf{cr}^t(G, v), \{\{\mathsf{cr}^t(G, w) : w \in N(v)\}\})$

In each round, the algorithm computes a coloring that is finer than the one computed in the previous round, that is, cr^{t+1} refines cr^t. For some $t \leq n := |G|$, this procedure stabilises: the coloring does not become strictly finer anymore.

The following connection between color refinement and GNNs will be useful to prove our negative result. Notably, the theorem holds regardless of the choice of the aggregation function agg and the combination function comb.

Theorem B.3 (Morris et al. (2019); Xu et al. (2018))**.** *Let d be a positive integer, and let ξ be the output of a GNN after d iterations. Then cr^d refines ξ, that is, for all graphs G, G' and vertices $v \in V(G)$, $v' \in V(G')$, $\mathsf{cr}^{(d)}(G, v) = \mathsf{cr}^d(G', v') \implies \xi(G, v) = \xi(G', v')$.*

Lemma B.4. *Let p be a positive integer and let $S \subset \mathbb{N}^p$ be a finite subset of integral vectors of the nonnegative orthant, such that S contains a non zero vector. Then there exist $x^* \in S$ and $u \in \mathbb{N}^p$ such that*

i) if $|S| = 1$ then $\langle x^*, u\rangle > 0$

ii) if $|S| \geq 2$ then for any $x \in S - \{x^*\}$, $\langle x^*, u\rangle > \langle x, u\rangle$.

Proof. If $|S| = 1$ the existence of x^* and u such that i) holds is clear as S is the singleton of a vector that is non zero. To deal with ii) in the case $|S| \geq 2$, consider one vector x^* maximizing the Euclidean norm over S, i.e. $\|x^*\|^2 = \mathsf{max}_{x \in S}\|x\|^2$ then let $x \in S - \{x^*\}$. Such x^* and x exist because S is finite and $|S| \geq 2$.

- Case 1: x is not colinear to x^*. It follows from the Cauchy-Schwarz inequality, that $\langle x^*, x\rangle < \|x^*\|\|x\|$. Hence

$$\langle x^*, x^* - x\rangle = \|x^*\|^2 - \langle x^*, x\rangle > \|x^*\|(\|x^*\| - \|x\|) > 0$$

- Case 2: $x \in S - \{x^*\}$ is colinear to x^*, i.e. $x = \lambda x^*$ with $\lambda \in \mathbb{R}$. Since x^* is maximizing the 2-norm on S, then $0 \leq \lambda < 1$. Then

$$\langle x^*, x^* - x\rangle = \|x^*\|(1 - \lambda) > 0$$

In both cases, $\langle x^*, x^* - x\rangle > 0$. Hence, we can set $u := x^* \in S \subset \mathbb{N}^p$, and the Lemma is proved. □

The following Lemma simply states that the vector $u \in \mathbb{N}^p$ can be chosen the same if one is given a pair of sets S and S', at the price of one inequality being possibly non strict.

Lemma B.5. *Let p be a positive integer, and let S, S' be two finite subsets of $\mathbb{N}^p$, such that $|S| \geq 2$ and $|S'| \geq 2$. Then there exists $u \in \mathbb{N}^p$, $x^* \in S$ and $y^* \in S$ such that for any $x \in S - \{x^*\}$ and any $y \in S' - \{y^*\}$:*

- Case i): there is $u \in \mathbb{N}^p$ maximizing the 2-norm simultaneously on S and S', i.e. S and S' have a common element $u = \arg\max(\{\|x\| : x \in S\}) = \arg\max(\{\|y\| : y \in S'\})$, then $\langle x^*, u\rangle > \langle x, u\rangle$ and $\langle y^*, u\rangle > \langle y, u\rangle$.

- Case ii): If S has the element of strictly greatest 2-norm among $S \cup S'$, then $\langle x^*, u\rangle > \langle x, u\rangle$, $\langle y^*, u\rangle \geq \langle y, u\rangle$ and $\langle x^\star, u\rangle > \langle y^\star, u\rangle$.

Proof. Case i): In this case, both u and u' obtained from Lemma B.4 coincide, as there is a common element maximizing the 2-norm over S and S'. We know that there exists $u = \arg\max\{\|x\| : x \in S\} = u' = \arg\max\{\|y\| : y \in S'\} \in \mathbb{N}^p$ such that for any $x \in S - \{x^*\}$ and any $y \in S' - \{y^*\}$, $\langle x^*, u\rangle > \langle x, u\rangle$ and $\langle y^*, u\rangle > \langle y, u\rangle$.

Case ii): Lemma B.4 tells us there is $u := \arg\max(\{\|x\| : x \in S\})$ such that for any $x \in S - \{x^*\}$, $\langle x^*, u\rangle > \langle x, u\rangle$. Now, for every $y \in S'$, let $y = y_u + y_{u^\perp}$ be the orthonormal decomposition of y in $\mathbb{R}^p = \mathsf{span}(u) \bigoplus u^\perp$. Note that both component vectors are still in $\mathbb{N}^p$.

For every $y \in S$, $\langle x^\star, y\rangle = \langle x^\star, y_u\rangle + \langle x^\star, y_{u^\perp}\rangle$. This proves that selecting y^* with largest coordinate on $\mathsf{span}(u)$ gives: $\langle y^\star, u\rangle \geq \langle y, u\rangle$ for every $y \in S' - \{y^\star\}$, which is the second inequality that was claimed. The inequality may not be strict as such $y^\star$ may not be the unique element maximizing $\langle y^\star, u\rangle$. The last claimed inequality follow from $x^* = u$ (cf. Lemma B.4) and the chain of inequalities $\langle x^\star, u\rangle = \|x^\star\|^2 > \|y^\star\|\|x^\star\| \geq \langle y^\star, u\rangle$.

□

Lemma B.6. *Let $\xi^t(\mathcal{T}[k_1, \ldots, k_m], s) \in \mathbb{R}^d$ be the embedding of the tree displayed in Figure 2 obtained via a GNN with **polynomial** activations after t iterations, where $\xi^0(v) = 1$ for all vertices $v \in V(\mathcal{T}[k_1, \ldots, k_m])$. Then, for any iteration t, and for every coordinate $\xi_i^t(\mathcal{T}[k_1, \ldots, k_m], s)$, there exists a polynomial function F_i such that $\xi_i^t(\mathcal{T}[k_1, \ldots, k_m], s) = F_i(k_1, \cdots, k_m)$. Furthermore, the degree of each F_i does not depend on m, but only on the underlying neural network and t.*

Remark B.7. *This lemma is specific to polynomial activations. For rational activations, the root embedding is still a rational function of the k_i's, but the degree of the numerator and denominator may grow with m (due to sums of rational functions requiring a common denominator whose degree scales with the number of summands). The rational case therefore requires a fundamentally different proof strategy, given in Appendix D.*

Proof. For clarity, we will perform two separate inductions, one for the existence of the rational function, and one for the degree boundedness.

Polynomial function. We first prove by induction on t that, for any vertex $v \in V(\mathcal{T}[k_1, \cdots, k_m])$, all the coordinates of $\xi^t(\mathcal{T}[k_1, \ldots, k_m], v)$ are polynomial functions of the k_i's.

Base case: for $t = 0$ this is trivial since all vertices are initialised with the constant rational function 1, whose degree does not depend on m.

Induction step: Suppose the property is true at iteration t, i.e for each node w, $\xi^t(\mathcal{T}[k_1,\dots,k_m],w)$ is (coordinate-wise) a rational functions of the k_i's. Since

$$\xi^{t+1}(\mathcal{T}[k_1,\dots,k_m],v) = \mathsf{comb}_t(\xi^t(\mathcal{T}[k_1,\dots,k_m],v), \mathsf{agg}_t(\{\{\xi^t(\mathcal{T}[k_1,\cdots,k_m],w) : w \in N(v)\}\}))$$

where comb_t is a neural network with polynomial activations, hence a polynomial function. Also, agg_t is the sum aggregation (a polynomial in the entries). Then by composition, each coordinate of $\xi^{t+1}(\mathcal{T}[k_1,\dots,k_m],v)$ is a polynomial function of $k_1,\cdots,k_m$.

Degree boundedness. We will prove that the degree of the numerators and denominators of $\xi^t(\mathcal{T}[k_1,\cdots,k_m],s)$ are both respectively bounded by q_t and r_t.

Base case: At the first iteration ($t=0$), P_m is constant equal to 1 ($q_1=1$ and $r_1=0$) for any m.

Induction step: Suppose that for any iteration $t \leq T$, there exists $q_t \in \mathbb{N}$ (that does not depend on m nor the vertex $v \in V(\mathcal{T}[k_1,\cdots,k_m])$), such that for every positive integer m, every vertex $v \in \mathcal{T}[k_1,\cdots,k_m]$, and for every iteration t, $\mathsf{deg}(F_i) \leq (q_t,r_t)$. Then, using again the update rule:

$$\underbrace{\xi^{t+1}(\mathcal{T}[k_1,\cdots,k_m],v)}_{Q_m} = \mathsf{comb}_t(\underbrace{\xi^t(\mathcal{T}[k_1,\cdots,k_m],v)}_{R_m}, \underbrace{\mathsf{agg}_t(\{\{\xi^t(\mathcal{T}[k_1,\cdots,k_m],w) : w \in N(v)\}\})}_{S_m})$$

R_m and S_m are rational fractions. By the induction hypothesis, for any m, $\mathsf{deg}(R_m) \leq (q_t,r_t)$ and $\mathsf{deg}(S_m) \leq (q_t,r_t)$.

Each coordinate of the function comb_t is a rational fraction of degree independent of m (neural network with a rational activation). Let (a_i,b_i) be the degree of its i-th coordinate for $i \in [d]$. The degree of the i-th coordinate Q_m is at most $(a_i \times q_t r_t, b_i \times q_t r_t)$. Hence the property remains true at $t+1$ for each coordinate i, setting $q_{t+1} := (\max_{i\in[d]} a_i) \times q_t r_t$ and $r_{t+1} := (\max_{i\in[d]} b_i) \times q_t r_t$. □

We can now build towards the:

Proof of Theorem 3.1. Consider the following query of GC2:

$$\begin{aligned} Q(s) =& \neg\left(\exists^{\geq 1} x(E(s,x) \wedge \exists^{\leq 1} s E(x,s))\right) \\ =& \forall x\left(E(s,x) \rightarrow \exists^{\geq 2} s E(x,s)\right) \end{aligned} \tag{3}$$

Q is true if and only if all the neigbhors of the node s have degree at least 2. Namely $Q(\mathcal{T}[0,k_2,\cdots,k_m],s)$ is false and $Q(\mathcal{T}[k_1,\cdots,k_m],s)$ is true for every positive integers $k_1,\cdots,k_m$. We will prove by contradiction that no GNN with polynomial activations can uniformly express the query Q. Let $R_m := \xi^t(\mathcal{T}[k_1,\cdots,k_m],s)$ be the embedding of the source node of $\mathcal{T}[k_1,\cdots,k_m]$ returned by a GNN with polynomial activations, after a fixed number of iterations t.

Suppose that R_m can uniformly express the query Q, then;

$$\begin{cases} R_m(k_1,\cdots,k_m) \geq 1-\epsilon & \text{if } s \in Q(\mathcal{T}[k_1,\cdots,k_m],s) \\ R_m(k_1,\cdots,k_m) \leq \epsilon & \text{if } s \notin Q(\mathcal{T}[k_1,\cdots,k_m],s) \end{cases}$$

Let $\tilde{R}_m := R_m - \frac{1}{2}$ and $\epsilon' := \frac{1}{2} - \epsilon$. Interpreting the query Q over $\mathcal{T}[k_1,\cdots,k_m]$ implies the following constraints on the sequence of rational functions $\tilde{R}_m$:

$$\begin{aligned} &\exists \epsilon' > 0 \text{ such that } \forall k \in \mathbb{N}^m \\ &\begin{cases} \exists i \in \{1,\cdots,m\}, k_i = 0 \implies \tilde{R}_m(k) \leq -\epsilon' \\ \forall i \in \{1,\cdots,m\}, k_i > 0 \implies \tilde{R}_m(k) \geq \epsilon' \end{cases} \end{aligned} \tag{4}$$

Let $\tilde{R}_m = \frac{\tilde{P}_m}{\tilde{Q}_m}$ be the irreducible representation of $\tilde{R}_m$,and let S (resp. S') be the set of exponents of the monomials of $\tilde{P}_m$ (resp. $\tilde{Q}_m$), i.e.

$$S := \{(\alpha_1,\cdots,\alpha_m) \in \mathbb{N}^m : \alpha_1+\cdots+\alpha_m \leq q \text{ and } k_1^{\alpha_1}\cdots k_m^{\alpha_m} \text{ is a monomial of } \tilde{P}_m\}$$

$$S' := \{(\alpha_1, \cdots, \alpha_m) \in \mathbb{N}^m : \alpha_1 + \cdots + \alpha_m \leq q \\ \text{and } k_1^{\alpha_1} \cdots k_m^{\alpha_m} \text{ is a monomial of } \tilde{Q}_m\}$$

First $|S| = 0$ is impossible as $\tilde{R}_m$ would have no zeroes and Conditions 4 cannot be met. Henceforth we suppose that $|S| \geq 2$ and $|S'| \geq 2$ (the other cases are discussed hereafter).

Lemma B.6 tells us that there exists a uniform bound on the degree of both $\tilde{P}_m$ and $\tilde{Q}_m$: there exists a positive integer k such that for every integer m, $\deg(\tilde{P}_m) \leq k$ and $\deg(\tilde{Q}_m) \leq k$. Henceforth, we will suppose that $m > 2k$, so that $\max(\deg(\tilde{P}_m), \deg(\tilde{Q}_m)) < \frac{m}{2}$.

Lemma B.5 (with $p = m$) tells us there exists $\alpha^* \in S$, $\beta^* \in S'$ and $u = (u_1, \cdots, u_m) \in \mathbb{N}^m$ such that for each of the three exclusive cases:

- Case i) there is a common element of S and S' maximizing the 2-norm. Then, for any $\alpha \in S - \{\alpha^*\}$ and for any $\beta \in S' - \{\alpha^*\}$, $\langle \alpha^*, u\rangle > \langle \alpha, u\rangle$ and $\langle \beta^*, u\rangle > \langle \beta, u\rangle$

- Case ii) $\max(\{\|x\| : x \in S\}) > \max(\{\|y\| : y \in S'\})$. Then, for any $\alpha \in S - \{\alpha^*\}$ and for any $\beta \in S' - \{\alpha^*\}$, $\langle \alpha^*, u\rangle > \langle \alpha, u\rangle$ and $\langle \beta^*, u\rangle \geq \langle \beta, u\rangle$ and $\langle \alpha^*, u\rangle > \langle \beta^\star, u\rangle$.

- Case iii) $\max(\{\|x\| : x \in S\}) < \max(\{\|y\| : y \in S'\})$. Then, for any $\alpha \in S - \{\alpha^*\}$ and for any $\beta \in S' - \{\alpha^*\}$, $\langle \alpha^*, u\rangle > \langle \alpha, u\rangle$ and $\langle \beta^*, u\rangle \geq \langle \beta, u\rangle$ and $\langle \beta^*, u\rangle > \langle \alpha^\star, u\rangle$.

We now make a sequence of five claims. The first three hold for every case, the last two hold for case i), allowing us to conlude in that case. We conclude for case ii) and iii) similarly as detailed after Claim 5.

Claim 1: *In all cases, the (univariate) monomial $t^{\langle \alpha^*, u\rangle}$ is the monomial of $\tilde{P}_m(t^{u_1}, \cdots, t^{u_m})$* of largest degree.

Proof. Since $\tilde{P}_m = \sum_{\alpha \in S} \gamma_\alpha k_1^{\alpha_1} \cdots k_m^{\alpha_m}$ we have

$$\tilde{P}_m(t^{u_1}, \cdots, t^{u_{m-1}}, t^{u_m}) = \sum_{\alpha \in S} \gamma_\alpha t^{\langle \alpha, u\rangle}$$

Hence, the monomial of largest degree of $\tilde{P}_m(t^{u_1}, \cdots, t^{u_m})$ is the one such that $\langle \alpha, u\rangle$ is (strictly) maximized when $\alpha \in S$. By construction, it is α^*. □

Claim 2: For every $\alpha \in S$, $\alpha' \in S'$, there exists $i \in [m]$ such that $\alpha_i = 0$ and $\alpha'_i = 0$.

Proof. By contradiction: if there exists $\alpha \in S$ and $\alpha' \in S'$ such that for every $i \in [m]$, $\alpha_i > 0$ or $\alpha'_i > 0$, this would imply that $\max(\deg(\tilde{P}_m), \deg(\tilde{Q}_m)) \geq \frac{m}{2}$, a contradiction with the respective choices of m and k that gave $\max(\deg(\tilde{P}_m), \deg(\tilde{Q}_m)) < \frac{m}{2}$. □

Claim 3: *Without loss of generality, suppose that the index i verifying Claim 2 for $\alpha^* \in S$ and $\beta^* \in S'$ is $i = m$. In these conditions, the (univariate) monomial $t^{\langle \alpha^*, u\rangle}$ is also* the *monomial of* of largest degree of $\tilde{P}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0)$.

Proof. Evaluating $\tilde{P}_m$ in $(t^{u_1}, \cdots, t^{u_{m-1}}, 0)$ removes the contribution of each monomial of P_m containing the last variable, and keeps only the contribution of the monomials not containing it: since $\tilde{P}_m = \sum_{\alpha \in S} \gamma_\alpha k_1^{\alpha_1} \cdots k_m^{\alpha_m}$ we have

$$\tilde{P}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0) = \sum_{\alpha = (\alpha_1, \cdots, \alpha_{m-1}, 0) \in S} \gamma_\alpha t^{\langle \alpha, u\rangle}$$

Therefore, the monomial of largest degree of $\tilde{P}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0)$ is the one such that $\langle \alpha, u\rangle$ is (strictly) maximized when $\alpha \in S$ and $\alpha_m = 0$. By construction, such α is α^*. □

Claim 4: *In case i), the (univariate) monomial $t^{\langle \beta^*, u \rangle}$ is the monomial of $\tilde{Q}_m(t^{u_1}, \cdots, t^{u_m})$* of largest degree.

Claim 5: *In case i), the (univariate) monomial $t^{\langle \beta^*, u \rangle}$ is also* the *monomial of* of largest degree of $\tilde{Q}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0)$.

Let $(\eta_\beta)_{\beta \in S'}$ be the coefficients of $\tilde{Q}_m$. In case i), the five claims combined imply that

$$\begin{aligned}\tilde{R}_m(t^{u_1}, \cdots, t^{u_{m-1}}, t^{u_m}) &\underset{t \to +\infty}{\sim} \tilde{R}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0) \\ &\underset{t \to +\infty}{\sim} \frac{\gamma_{\alpha^*} t^{\langle \alpha^*, u \rangle}}{\eta_{\beta^*} t^{\langle \beta^*, u \rangle}}\end{aligned}$$

In particular

$$\lim_{t \to +\infty} \tilde{R}_m(t^{u_1}, \cdots, t^{u_m}) = \lim_{t \to +\infty} \tilde{R}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0)$$

This is a contradiction with Conditions 4.

If case ii) holds, then Claims 4) and 5) are not necessarily true as some exponents may cancel out. However, simply note in that case, due to strict inequalities obtained before Claim 1, that the degree of $\tilde{P}_m(t^{u_1}, \cdots, t^{u_m})$ is strictly greater than the one of $\tilde{Q}_m(t^{u_1}, \cdots, t^{u_m})$, implying that

$$\begin{aligned}\lim_{t \to +\infty} \tilde{R}_m(t^{u_1}, \cdots, t^{u_m}) &= \lim_{t \to +\infty} \tilde{R}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0) \\ &\in \{-\infty, +\infty\}\end{aligned}$$

In case iii), with a similar reasoning as in case ii), we get $\lim_{t \to +\infty} \tilde{R}_m(t^{u_1}, \cdots, t^{u_m}) = \lim_{t \to +\infty} \tilde{R}_m(t^{u_1}, \cdots, t^{u_{m-1}}, 0) = 0$.

Finally, if $|S'| = 0$ then $\tilde{R}_m$ is a polynomial. In this case we do not need Claims 4 and 5, and the limit argument still applies. If $|S| = 1$ or $|S'| = 1$ then we can apply Lemma B.4 (case i)) to the polynomials having only one monomial, and the same argument still goes through. □

C TOWARDS A RATIONAL FRAGMENT: RGC2

Proof of Theorem 3.9. First, note that once the property is proved for queries of Ω_0 is (cf. Defintion 3.7), the result follows as one can simply implement negation and logical conjuction with a rational GNN (for full details, cf. first proof in the Appendix).

Induction hypothesis: Let Q be a query of $\Omega_0 \subset$ RGC2 of depth $d > 0$ and with subformulas $(Q_1, \cdots, Q_d)$. Then there exists $\epsilon > 0$ such that for every $0 < \varepsilon \leq \epsilon$, there exists $\xi \in \mathbb{R}^d$ computed by a rational GNN such that

- For every graph G and vertex $v \in V(G)$, for every $i \in [d]$
  $\xi_i(v, G) = 0 \iff Q_i(v, G) = 0$
  $\xi_i(v, G) \in [1 - \varepsilon, 1 + \varepsilon] \iff Q_i(v, G) = 1$
- For any constant $C \geq 0$, for any graph G and vertex $v \in V(G)$, $\sum_{w \in N(v)} \xi_i(v, G)$ takes finitely many values outside of the interval $[C, +\infty)$.

Base case ($d = 1$): Obvious for the queries $\mathsf{Col}_i(\cdot)$ for $i \in [l]$ (and $\epsilon = 0$), as the input features are the one-hot encoding vectors of the colors. Simply choose the projection over the i-th coordinate which is polynomial (hence rational).

Induction step: Let Q be a query of Ω_0 of depth $d + 1$.

- If $Q := Q_i = Q_{i-1} \wedge Q_{i-2}$ where Q_{i-1} and Q_{i-2} are in Ω_0 and verify the induction hypothesis with ϵ_1 and ϵ_2 respectively. Then define $\xi_i = \xi_{i-1}\xi_{i-2}$ (product). It can be easily verified that ξ^i verifies the seeked properties, for $\epsilon := \epsilon_1\epsilon_2 + \epsilon_1 + \epsilon_2$.

- If $Q(\cdot) := Q_i(\cdot) = \exists^{\geq K} y E(\cdot, y) \wedge Q_{i-1}(y)$ where K is a positive integer and Q_{i-1} verifies the induction hypothesis for some $\epsilon_1 > 0$. Since $K > 0$, we can suppose that the induction hypothesis property is verified for some $\varepsilon_1 \leq \frac{1}{2K-1}$, as it is true for any $\varepsilon_1 \leq \epsilon_1$. Let ξ^{i-1} be the output computed by the corresponding rational GNN.

  Therefore, we have that for any graph G and any vertex $v \in V(G)$,

  $$\sum_{w \in N(v)} \xi_{i-1}(w, G) \geq K(1 - \varepsilon_1)$$

  implies that there are at least K neighbors of v verifying Q_{i-1} (because otherwise, $\sum_{w \in N(v)} \xi_{i-1}(w, G) \leq (K-1)(1+\varepsilon_1) < K(1-\varepsilon_1)$). The last inequality holds due to the initial choice of ε_1 that verifies $\varepsilon_1 \leq \frac{1}{2K-1}$.

  Due to the induction hypothesis (second part), $\sum_{w \in N(v)} \xi_{i-1}(w, G)$ takes finitely many values out of the interval $[K(1-\varepsilon_1), +\infty)$ on all graphs and vertices. Let S be this set of values. The remaining step consists in showing that there exists some ϵ such that for any $0 < \varepsilon \leq \epsilon$, we can map via a rational neural network F the interval $[K(1-\epsilon_1), +\infty)$ to $[1-\varepsilon, 1+\varepsilon]$, by less than $K(1-\epsilon_1)$, and the set S to 0. This will insure that

  $$\xi_i := F\Big(\sum_{w \in N(v)} \xi_{i-1}(w, G)\Big) \tag{5}$$

  expresses $Q = Q_i$, because if the sum is strictly less that $K(1-\varepsilon_1)$, ξ_i equals 0, otherwise it is in the interval $[1-\varepsilon, 1+\varepsilon]$. The existence of such F is a consequence of Lemma C.1. Finally, due to Equation 5 ξ_i can also take finitely many values out of $[C, +\infty)$ over all vertices and graphs, if C is an arbitrary positive constant. This ends the induction and the proof.

□

Lemma C.1. *Let S be a finite set of non-negative reals. Let $\varepsilon > 0$, and $x^* > \max S$. Then there exists a rational neural network $\mathbb{R} \to \mathbb{R}$ such that*

- *$R(x) = 0$ for every $x \in S$*
- *For every $x \geq x^*$, $R(x) \in [1-\varepsilon, 1+\varepsilon]$.*

Proof. Let $S = \{a_1, \cdots, a_k\} \subset \mathbb{R}_+ \cup \{0\}$.

A straightforward computation shows that

$$R(x) := \frac{\prod_{i=1}^k (x - a_i)}{\beta + \prod_{i=1}^k (x - a_i)}$$

with $\beta := \frac{\varepsilon}{1+\varepsilon} \prod_{i=1}^k (x^* - a_i)$ verifies the conditions.

It remains to show that R can be computed by a rational neural network. Due to the partial fraction decomposition of R, if $Q(x) := \beta + \prod_{i=1}^k (x - a_i)$ is a split and squarefree polynomial, there exists reals $s_1, \cdots, s_k$ such that

$$R(x) = 1 + \sum_{i=1}^k \frac{s_i}{x - a_i} \tag{6}$$

This property becomes true as $\varepsilon \to 0$, by continuity of the roots of polynomials as functions of their coefficients—a classical result, see e.g. Bugajewski & Maćkowiak (2018) and references therein—and because $\prod_{i=1}^k (x - a_i)$ is split and squarefree. Hence Equation 6 is true, for a sufficiently small choice of ε. Note that by definition, R is a one hidden neural network with activation function $\sigma(x) := \frac{1}{x}$. □

D PROOF OF THEOREMS 3.3, 3.4, AND 3.5

We use the tree family $\mathcal{T}[x, k, m]$ with root s, m children, where child x_1 has x leaves and children x_i ($i \geq 2$) have k leaves each. The query $Q(s)$ (Eq. 2) gives $Q(s, \mathcal{T}[0, k, m]) = 0$ and $Q(s, \mathcal{T}[1, k, m]) = 1$ for $k \geq 1$.

D.1 PROOF OF THEOREM 3.3 ($T = 2$, ANY RATIONAL, ANY d_t)

Define $h(n) := \mathsf{comb}_0(c, (1+n)c) \in \mathbb{R}^{d_1}$ (iteration-1 embedding of a depth-1 vertex with n leaves) and $\theta_1(m) := \mathsf{comb}_0(c, mc) \in \mathbb{R}^{d_1}$ (iteration-1 root embedding). The root output at iteration 2 is:

$$\xi^2(s, \mathcal{T}[x, k, m]) = \mathsf{comb}_1\big(\theta_1(m),\ h(x) + (m-1)\, h(k)\big).$$

Fix $k \geq 1$ with $h(k) \neq 0$ (at most finitely many k violate this). Consider the readout component j and define:

$$\varphi^+(m) := [\mathsf{comb}_1]_j\big(\theta_1(m),\ h(1)+(m-1)\, h(k)\big), \quad \varphi^-(m) := [\mathsf{comb}_1]_j\big(\theta_1(m),\ h(0)+(m-1)\, h(k)\big).$$

Key observation: Each component of $\theta_1(m) = \mathsf{comb}_0(c, mc)$ is a rational function of m (since comb_0 is rational and its second argument is linear in m). Each component of $h(x) + (m-1)h(k)$ is affine in m. Since $[\mathsf{comb}_1]_j$ is rational in its $2d_1$ inputs, the composition yields $\varphi^+(m)$ and $\varphi^-(m)$ as *univariate rational functions of* m. This reduction holds regardless of embedding dimension d_1.

The uniform separation constraint requires $\varphi^+(m) \geq 1 - \varepsilon$ and $\varphi^-(m) \leq \varepsilon$ for all $m \geq 1$. Since univariate rational functions have well-defined limits, both limits must be finite. Now consider $\psi(m) := \varphi^+(m) - \varphi^-(m)$. The two functions are obtained by evaluating the same rational function at arguments differing by $\delta_1 = h(1) - h(0)$ (a constant independent of m), while both arguments grow as $O(m)$. When we form $\psi(m)$ over a common denominator, the numerator has degree strictly less than the denominator (since a constant perturbation in an $O(m)$ polynomial argument shifts the leading monomial by a sub-leading amount). Hence $\psi(m) \to 0$ as $m \to \infty$.

This contradicts $\psi(m) \geq 1 - 2\varepsilon > 0$ for all m. □

D.2 PROOF OF THEOREM 3.4 (BOUNDED RATIONAL, ANY T, ANY d_t)

We need two lemmas about globally bounded rational maps.

Lemma D.1 (Lipschitz in first argument)**.** *Let* $\mathsf{comb} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{d'}$ *be a globally bounded rational map. For any compact* $K \subset \mathbb{R}^d$*, there exists* $L_K > 0$ *such that for all* $u_1, u_2 \in K$ *and all* $v \in \mathbb{R}^d$*:* $\|\mathsf{comb}(u_1, v) - \mathsf{comb}(u_2, v)\| \leq L_K \cdot \|u_1 - u_2\|$.

Proof. Global boundedness implies comb is pole-free. For each component comb_j, the gradient $\nabla_u \mathsf{comb}_j(u, v)$ is rational and bounded on $K \times \mathbb{R}^d$ (since comb_j is globally bounded). Set $L_K := \max_j \sup_{K \times \mathbb{R}^d} \|\nabla_u \mathsf{comb}_j\|$ and apply the mean value theorem. □

Lemma D.2 (Jacobian decay in second argument)**.** *Let* $\mathsf{comb} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^{d'}$ *be a globally bounded rational map. For any compact* $K \subset \mathbb{R}^d$ *and direction* $\hat{w} \in \mathbb{R}^d$ *with* $\|\hat{w}\| = 1$*, there exists* $C > 0$ *such that for all* $u \in K$*,* $\tau \geq 1$*, and* $\delta \in \mathbb{R}^d$*:* $\|\mathsf{comb}(u, \tau\hat{w} + \delta) - \mathsf{comb}(u, \tau\hat{w})\| \leq C \cdot \|\delta\|/\tau$.

Proof. For each component $\mathsf{comb}_j(u, v) = P_j(u, v)/Q_j(u, v)$, boundedness forces $\deg_\tau(P_j(u, \tau\hat{w})) \leq \deg_\tau(Q_j(u, \tau\hat{w})) =: d_Q$. The partial derivatives $\partial_{v_i} \mathsf{comb}_j$ have numerator degree $\leq d_P + d_Q - 1$ and denominator degree $2d_Q$, giving a degree gap $\geq d_Q - d_P + 1 \geq 1$. Hence $\|\nabla_v \mathsf{comb}_j(u, \tau\hat{w})\| = O(1/\tau)$ uniformly over $u \in K$. The mean value theorem gives the result. □

Proof of Theorem 3.4. Since all comb_t are bounded, all embeddings lie in a ball of radius M. We prove by induction on t:

(a) $\|\delta A^{(t)}\| := \|A^{(t)}(\mathcal{T}[1, k, m]) - A^{(t)}(\mathcal{T}[0, k, m])\| = O(1)$ (bounded, independent of m).

(b) $\|\Delta^{(t)}\| := \|\xi^t(s, \mathcal{T}[1, k, m]) - \xi^t(s, \mathcal{T}[0, k, m])\| = O(1/m) \to 0$.

Base ($t = 1$): $\delta A^{(1)} = h(1) - h(0)$ (constant). $\Delta^{(1)} = 0$ (root's iter-1 embedding does not depend on x).

Inductive step ($t \to t+1$), Part (b): Decompose:

$$\Delta^{(t+1)} = \underbrace{\mathsf{comb}_t(\xi^t(s, \mathcal{T}[1]), A_+^{(t)}) - \mathsf{comb}_t(\xi^t(s, \mathcal{T}[0]), A_+^{(t)})}_{\text{(I)}}$$
$$+ \underbrace{\mathsf{comb}_t(\xi^t(s, \mathcal{T}[0]), A_+^{(t)}) - \mathsf{comb}_t(\xi^t(s, \mathcal{T}[0]), A_-^{(t)})}_{\text{(II)}}$$

Term (I): By Lemma D.1, $\|\text{(I)}\| \leq L \cdot \|\Delta^{(t)}\| = O(1/m)$.

Term (II): The aggregate $A_-^{(t)}$ grows as $(m-1)\|g^{(t)}(k)\| \cdot \hat{w} + O(1)$, where $g^{(t)}(k) \neq 0$ is the common embedding of regular children. By Lemma D.2: $\|\text{(II)}\| \leq C \cdot \|\delta A^{(t)}\|/\tau = O(1)/O(m) = O(1/m)$.

Inductive step, Part (a): The aggregate offset is $\delta A^{(t+1)} = [\xi^{t+1}(x_1, \mathcal{T}[1]) - \xi^{t+1}(x_1, \mathcal{T}[0])] + (m-1)[\xi^{t+1}(x_i, \mathcal{T}[1]) - \xi^{t+1}(x_i, \mathcal{T}[0])]$. The first term is $O(1)$ (bounded comb_t). For the context contribution: each regular child changes by $O(\|\Delta^{(t)}\|) = O(1/m)$ via Lipschitz. Summing: $(m-1) \cdot O(1/m) = O(1)$.

Contradiction: After T iterations, $\|\Delta^{(T)}\| = O(1/m) \to 0$, but uniform expressivity requires $|\Delta_j^{(T)}| \geq 1 - 2\varepsilon > 0$ for all m. □

D.3 Proof of Theorem 3.5 (unbounded rational, any T, $d_t = 1$)

For scalar embeddings ($d_t = 1$), the proof follows the same inductive structure as Theorem 3.4, but exploits the stronger scalar decay. In one dimension, the degree gap in the derivative of a bounded-or-convergent rational function is ≥ 2 (by Euler's theorem for radial directions), yielding $O(1/m^2)$ decay instead of $O(1/m)$.

For the unbounded case, we analyze the limiting behavior of the root output: if $\xi^T(s) \to L \in \mathbb{R}$ as $m \to \infty$, then both positive and negative outputs converge to the same limit L (by the $O(1/m^2)$ derivative decay), contradicting separation. If $\xi^T(s) \to \pm\infty$, then both outputs diverge together (since positive and negative trees differ only by a bounded offset in the aggregate), so neither can satisfy the required bound for large m.

The scalar structure ensures that the root output $\xi^T(s)$ is ultimately a univariate rational function of m (after fixing k). Its limit as $m \to \infty$ exists in $\mathbb{R} \cup \{\pm\infty\}$. In all cases, the difference between the positive and negative configurations vanishes, contradicting the separation requirement. See Section 4 for the detailed dichotomy argument. □

E Additional Experimental Details

The experiments were conducted on a Mac-OS laptop with an Apple M1 Pro chip. Neural networks were implemented using PyTorch 2.3.0 and PyTorch Geometric 2.2.0, with the Rational activations library Delfosse et al. (2020). Each GNN iteration uses its own combine function (feedforward NN with one hidden layer). Training used the Adam optimizer with learning rate 10^{-3} for 200 epochs.