EXPONENTIAL SUCCINCTNESS OF FINITE-PRECISION GNNS OVER GRADED MODAL SUBSTITUTION CALCULUS

Anonymous authors
Paper under double-blind review

ABSTRACT

Expressive-equivalence results characterize which node properties different graph models can represent, but not how compactly they can represent them. We study this succinctness question for finite-precision recurrent graph neural networks (GNNs), graded modal substitution calculus (GMSC), its global-counting extension (GGMSC), and bounded finite counting message-passing automata (FCMPAs). Our main result gives an explicit six-dimensional recurrent GNN with $n + \mathcal{O}(1)$-bit precision and $\mathcal{O}(n)$ encoded size that accepts a node exactly when its out-degree is even and below 2^n, whereas the minimum equivalent GMSC/GGMSC size is $\Theta(n2^n)$ bits. The same family requires exponentially many graded modalities even for any fixed fidelity above $1/2$ under the uniform distribution over degrees below the cutoff, and the lower bounds hold on both directed and undirected graphs. The separation is not specific to recurrence: a depth-$\mathcal{O}(n)$ feedforward GNN of encoded size $m = \mathcal{O}(n^2)$ yields a $2^{\Omega(\sqrt{m})}$ gap, while a depth-one weighted-threshold GNN with $m = \Theta(n)$ yields a $2^{\Omega(m)}$ gap and extends a known graded-modal-logic separation to recursive GMSC. We also prove a 2^n state lower bound for counting message-passing automata on a property with an $\mathcal{O}(n)$-size GMSC program and a $\mathrm{poly}(n)$-size GNN. Together, these results establish exponential succinctness gaps between GNNs and expressively equivalent logical or automata-based representations, driven by arithmetic on neighborhood multiplicities and by the cost of explicit state.

1 INTRODUCTION

Graph neural networks are usually compared by expressive power: which node or graph properties can a model represent? This question has led to precise logical characterizations. Aggregate-combine GNNs correspond to fragments of graded modal logic (Barceló et al., 2020; Grohe, 2021), and recurrent GNNs over finite floating-point systems have the same node-classification power as graded modal substitution calculus (GMSC) and bounded finite counting message-passing automata (FCMPAs) (Ahvonen et al., 2024). Global-counting GMSC (GGMSC) extends GMSC with a modality that counts satisfying nodes across the whole graph (Ahvonen et al., 2025).

Expressive equivalence leaves representation size unresolved. A compact network may correspond to a much larger logical formula or automaton, even when the models define the same class of properties. Symbolic extraction, model checking, explanation, and automata-based verification all inherit the size of this representation. Succinctness therefore determines the intrinsic output cost of translating a learned model into symbolic form. Descriptional complexity (Holzer & Kutrib, 2011) studies precisely this gap between the smallest equivalent representations in two formalisms.

Succinctness has already proved informative for sequence models. Fixed-precision transformers can be exponentially more succinct than linear temporal logic (LTL) and recurrent networks, and doubly exponentially more succinct than finite automata (Bergsträßer et al., 2026). Graphs introduce a different source of compression, because a message-passing layer aggregates multiplicities, and learned weights can perform arithmetic on those counts. Graded modal logic sees the same neighborhood only through individual threshold tests. This difference between arithmetic and threshold access motivates our first exponential gap.

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.

We use bounded degree parity as the main test case. For $n \geq 1$, define the node property

$$E_n(v) \quad \Longleftrightarrow \quad \deg^+(v) < 2^n \text{ and } \deg^+(v) \text{ is even.} \tag{1}$$

In words, E_n accepts a node exactly when its out-degree is even and less than 2^n. On an undirected graph, this is its ordinary degree. A six-coordinate recurrent GNN computes E_n using $\mathcal{O}(n)$ encoded bits. The network stores the normalized degree on a 2^{-n} grid and advances a clock on that same grid. When the clock first passes the stored degree, an alternating bit reveals its parity. Every operation is exact in the chosen $\mathcal{O}(n)$-bit binary floating-point system.

The matching lower bound is already visible on a star with c indistinguishable leaves. Each graded modality tests one threshold in c. Recursion can reuse that test in later rounds, but it cannot introduce a new threshold. The truth value of E_n changes between every pair of consecutive degrees below 2^n, so an equivalent GMSC program needs $2^n - 1$ threshold occurrences and $\Omega(n2^n)$ encoded bits. The same bit bound holds with global counting. In particular, the empty proposition alphabet isolates arithmetic on neighborhood multiplicities from the width of node labels.

Our contributions are:

- We prove that recurrent finite-precision GNNs are exponentially more succinct than the expressively equivalent logic GMSC. An explicit six-coordinate R-simple network uses $\mathcal{O}(n)$ encoded parameter bits, while the minimum equivalent logical description has size $\Theta(n2^n)$. The lower bound allows recursion, binary thresholds, directed acyclic graph (DAG) sharing, and global counting, and remains exponential at any fixed fidelity above $1/2$.
- Feedforward networks exhibit the same logical gap. A depth-$(2n+2)$ layered GNN extracts the binary digits of the degree and has encoded size $m = \mathcal{O}(n^2)$, hence a $2^{\Omega(\sqrt{m})}$ logical lower bound. At depth one, weighted aggregation yields a $2^{\Omega(m)}$ separation by a two-dimensional slope argument. This second construction recovers the nonrecursive counting-perceptron lower bound of Galliani et al. (2023) and extends it to recursive GMSC.
- A separate argument gives a 2^n state lower bound for counting message-passing automata and isolates initialization width: if the automaton maps two node labels to the same initial state, no later round can distinguish them. This result uses a proposition alphabet of size $2n$ and compares explicit automaton states with a compact symbolic initialization map.

These results separate three resources that are conflated by expressivity alone: arithmetic on multiplicities, initialization width, and temporal reuse. The main separation is driven by the first resource and therefore applies even when the vocabulary is empty and the architecture is feedforward.

Table 1 summarizes the four comparisons and gives the size measure used in each. The parity rows compare GNNs with an expressively equivalent logic, the weighted row compares one-layer arithmetic with threshold-only counting, and the automata row compares symbolic initialization with explicit state count.

Table 1: Summary of the four succinctness comparisons and their size measures.

Property	Compact representation (measure)	Lower-bound target (measure)	Separation mechanism
Even degree below 2^n	Recurrent GNN: $\mathcal{O}(n)$ encoded bits	Min. GMSC/GGMSC: $\Theta(n2^n)$ encoded bits	Exact clock and parity over the empty alphabet
Same parity property	Depth-$(2n+2)$ GNN: $\mathcal{O}(n^2)$ encoded bits	Min. GMSC/GGMSC: $\Theta(n2^n)$ encoded bits	Feedforward binary digit extraction
$c_p + 2c_q \geq 2^n$	Depth-one GNN: $\mathcal{O}(n)$ encoded bits	Min. GML/GMSC: $\Theta(n2^n)$ encoded bits	Weighted versus threshold-only counting
Matched neighbor labels	GMSC: $\mathcal{O}(n)$ syntax, GNN: $\mathrm{poly}(n)$ encoded bits	CMPA: 2^n states	Symbolic initialization width

Significance for symbolic extraction. The main theorem gives an output-size lower bound for symbolic compilation. Call a procedure an *exact GNN-to-GMSC extractor* if it takes an encoded recurrent finite-precision GNN and returns an equivalent explicit GMSC program. For the six-dimensional family computing E_n, the input has size $\Theta(n)$ but every valid output has size $2^{\Omega(n)}$. Any exact extractor therefore needs exponential space for its output and exponential time simply to write it. This unconditional output bound applies to every implementation of exact extraction, as well as to GMSC explanations or certificates that agree with the network on all finite graphs in the chosen directed or undirected domain.

The same exponential barrier holds for approximate extraction. Under the uniform distribution on stars of degrees $0, \dots, 2^n - 1$, any GMSC explanation with fidelity $1 - \varepsilon > 1/2$ still needs $(1 - 2\varepsilon)2^n - 1$ graded modalities. Thus, any fixed error below one half preserves the exponential output cost on this family.

The theorem identifies both a sharp worst-case barrier and the architectural feature responsible for it. It also points to four routes toward efficient extraction: structured source families, favorable data distributions, a target logic with arithmetic counting, and bounded input domains. Each route adds structure that avoids the alternating-threshold family used here.

Related work. Logical characterizations connect message-passing GNNs to graded modal logic, Weisfeiler–Leman refinement, and finite-state graph computation (Barceló et al., 2020; Grohe, 2021; Ahvonen et al., 2024; Xu et al., 2019; Morris et al., 2019). We ask the complementary quantitative question of how large an equivalent representation must be. Our size conventions follow Bergsträßer et al. (2026), and our breakpoint arguments complement general modal formula-size methods. The closest succinctness precursor is the nonrecursive weighted-counting separation of Galliani et al. (2023); our bounded-parity result instead treats exact finite-precision recurrence and proves lower bounds for recursive GMSC and GGMSC. The extraction interpretation connects these bounds to global symbolic and rule-based GNN explanations. Appendix B gives a fuller comparison with arithmetic counting logics, GNN expressivity and complexity, explanation methods, and verification.

2 Models and size measures

We compare three ways to represent the same node classifiers. GMSC provides the threshold-counting target language, finite-precision GNNs provide the arithmetic source model, and counting message-passing automata expose a separate state-complexity question. Their common synchronous-round semantics makes the representation-size differences directly comparable.

Graphs and counting logic. A pointed graph over a finite proposition alphabet Π is a finite directed graph $G = (V, E, \lambda)$ with a distinguished node v and labeling $\lambda : V \to \mathcal{P}(\Pi)$. Graded modal logic (GML) extends Boolean logic with modalities $\Diamond_{\geq k}\varphi$, which are true at a node when at least k out-neighbors satisfy φ; thresholds are binary encoded. Undirected graphs form the subclass with symmetric E. There, out-degree is the ordinary degree. We take $\mathbb{N} = \{0, 1, \dots\}$. The parity and weighted-counting lower bounds below remain valid when equivalence is required only on the undirected subclass.

GMSC adds synchronous recursion (Ahvonen et al., 2024). A program contains finitely many head predicates X, a base rule $X(0) : -\varphi_X$, and an induction rule $X : -\psi_X$. Base rules determine round zero. Induction rules are evaluated from the preceding round. A pointed graph is accepted when an accepting head predicate holds at the distinguished node in some round. Program size counts proposition and head symbols and all rule-body occurrences, using the binary length of grading indices. For a DAG representation, a shared modal subformula is counted once. This also covers arbitrary Boolean circuits over modal atoms. Global-counting GMSC (GGMSC) additionally has a modality $\langle E \rangle_{\geq k}\varphi$ that counts satisfying nodes in the whole graph (Ahvonen et al., 2025).

These logics observe multiplicities through threshold modalities. The GNN model below receives the same neighborhood information as a numeric aggregate and can therefore combine counts arithmetically.

Finite-precision recurrent GNNs. Following Ahvonen et al. (2024), a recurrent $\mathrm{GNN}[F]$ has states in S^d for a finite floating-point system S, an initialization $\pi : \mathcal{P}(\Pi) \to S^d$, and a shared

aggregate-combine transition. We use the R-simple form

$$x_v^{t+1} = \sigma\big(Cx_v^t + A\,\mathrm{SUM}_S\{\{x_u^t : (v,u) \in E\}\} + \beta\big)\,, \tag{2}$$

where $C, A \in S^{d\times d}$, $\beta \in S^d$, and $\sigma(z) = \min(\max(0,z),1)$ coordinatewise. Here C acts on the current state, A acts on the coordinatewise neighbor sum SUM_S, and β is the bias. All arithmetic is performed in S. In each affine coordinate, products are summed in increasing coordinate order, and the resulting self, neighbor, and bias terms are added in the displayed order. Every primitive operation is rounded in S. Acceptance is existential over rounds – the affine readout threshold must hold at the distinguished node in some round. A layered aggregate-combine GNN (AC-GNN) instead reads out after its finite sequence of transitions.

A binary float system is determined by its mantissa precision and exponent range. We use round-to-nearest with ties to even, saturating overflow, and the increasing-magnitude summation convention of Ahvonen et al. (2024). Its domain S is finite, and it has a saturation threshold κ_S: replacing any input multiplicity above κ_S by κ_S does not change SUM_S. Finiteness gives the recurrent model a finite state space, while saturation underlies the bounded-depth witness argument in Appendix G.

Our *encoded parameter size* includes the dimension and proposition alphabet, the binary description of the float format, and all matrix and bias entries at the format's word width. Initialization is represented by a straight-line circuit over the inputs $[p \in \lambda(v)]$, using affine and clipped-ReLU gates over S. Its gates, wires, and constants count toward the size. The affine-threshold readout is treated the same way. The measure therefore includes both precision and initialization. The parity networks use $n + \mathcal{O}(1)$ mantissa bits and have size $\mathcal{O}(n)$, while the automata witness uses an $\mathcal{O}(n)$ characteristic-vector initialization circuit.

We report every rate both in the family parameter n and, when different, in source encoded size m. The recurrent parity and depth-one weighted families have $m = \Theta(n)$ and target size $2^{\Omega(m)}$. The layered parity network has $m = \mathcal{O}(n^2)$ and target size $2^{\Omega(\sqrt{m})}$. The displayed modality lower bounds hold for every threshold coding and also cover shared DAG syntax. Thus, the comparison does not depend on whether repeated logical subexpressions are written out or shared.

This intensional measure counts a shared transition once. Under it, the arithmetic transition has an $\mathcal{O}(n)$ encoding, whereas every explicit threshold-only GMSC description of the same classifier is exponential. Unrolling all 2^n rounds, enumerating the reachable states, or using the lookup-table representation of Ahvonen et al. (2024) produces larger source descriptions. The layered and depth-one constructions establish succinctness under explicit feedforward encodings as well.

Counting message-passing automata. A counting message-passing automaton (CMPA) is a tuple (Q, π, δ, F), where Q is a nonempty set of states, $\pi : \mathcal{P}(\Pi) \to Q$ is the initialization function, $\delta : Q \times \mathcal{M}(Q) \to Q$ is the transition function over finite multisets of states, and $F \subseteq Q$ is the set of accepting states. Node v starts in $\pi(\lambda(v))$ and updates by applying δ to its current state and the multiset of out-neighbor states. Acceptance occurs when the distinguished node reaches F. An FCMPA has finite Q. It is bounded if its transition depends only on multiplicities capped at a finite threshold. We measure state cardinality $|Q|$. The bound excludes small state spaces under every encoding, not compact circuits for π, δ, or F. The expressive equivalence uses bounded FCMPAs, while the lower bound holds for the larger class of all CMPAs. Observe that the model reads labels only through π at round zero. Subsequent rounds are deterministic functions of current states and out-neighbor-state multisets. This irreversible first step will drive the automata lower bound. We first turn to the central separation, where the relevant resource is arithmetic on multiplicities rather than initialization.

3 Exponential succinctness from degree arithmetic

For $c \in \mathbb{N}$, let $\mathrm{Star}_c^{\to}$ be a root with c unlabeled out-neighbors and no return edges, and let $\mathrm{Star}_c^{\leftrightarrow}$ be the corresponding undirected star. We write Star_c for either family when the statement holds for both. The first lemma captures the limitation of threshold-only counting on these families. It reduces every later logical lower bound to counting how often the target property changes as c increases.

Lemma 3.1 (Breakpoint lemma)**.** *Let A be the set of grading indices appearing in a GMSC program Λ. On either star family, the verdict of Λ as a function of c has at most $|A|$ breakpoints. This remains*

true for tree or DAG syntax and for arbitrarily many recursion rounds. A GGMSC program has at most $2|A|$ breakpoints.

Proof. For the directed star, all leaves have the same label and empty out-neighborhood, so a direct induction over rule bodies and rounds applies. For the undirected star, simultaneously track the root and the uniform leaves. A local modality at the root is either false or $[c \geq a]$. At a leaf it sees only the root. It is false for $a > 1$ and, for $a = 1$, equals the inductively known truth of its argument at the root. Induction over rule bodies and rounds shows in either case that the joint root/leaf trajectory depends only on $([c \geq a])_{a \in A}$. Recursion can reuse these bits but cannot introduce a new threshold.

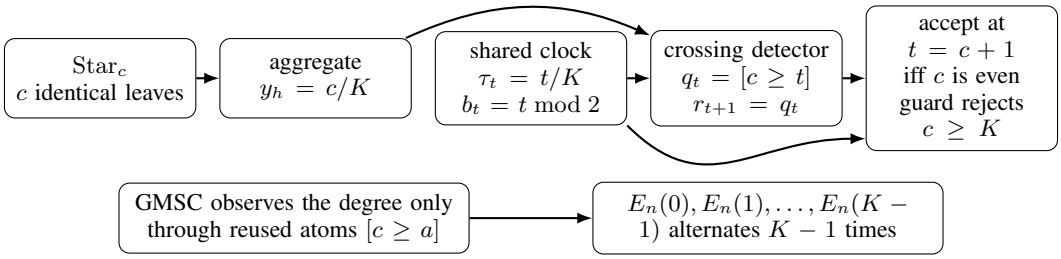
For GGMSC, simultaneously track the root and the uniform leaves. A global modality counts either ρ or $c + \rho$ satisfying nodes, where $\rho \in \{0, 1\}$ records the root. Its value is determined by $[c \geq a]$ and $[c \geq a - 1]$. Hence each global threshold contributes at most two breakpoints. □

Theorem 3.2 (Bounded parity)**.** *For $n \geq 1$, write $K = 2^n$ and let E_n be the property that the distinguished node has even out-degree smaller than K. Then:*

1. *E_n is GML-definable, and a recurrent R-simple GNN of dimension 6 and encoded parameter size $\mathcal{O}(n)$ expresses E_n.*
2. *a depth-$(2n + 2)$ layered R-simple GNN of encoded parameter size $\mathcal{O}(n^2)$ expresses E_n.*
3. *every equivalent GMSC program contains at least $2^n - 1$ graded modalities, and every equivalent GGMSC program contains at least 2^{n-1} graded modalities. Under binary grading indices, both have size $\Omega(n2^n)$.*

The lower bounds hold both over all finite directed graphs and when the domain is restricted to finite undirected graphs.

Figure 1 summarizes the two sides of the theorem. The upper path follows the recurrent computation, while the lower path shows why the alternating target requires a separate logical threshold at every degree change.



One fixed six-coordinate transition versus $K - 1 = 2^n - 1$ logical modalities.

Figure 1: Mechanism behind the bounded-parity separation. The GNN reuses one transition and accepts at the unique round when its clock passes the degree. GMSC must represent every change of the alternating degree predicate with an explicit threshold.

Proof. The proof has three parts. We first turn alternation into the logical lower bound, then construct the recurrent clock, and finally replace temporal reuse by a feedforward bit-extraction circuit.

For item (3), the breakpoint set is $\{1, \ldots, K - 1\}$. In GMSC, it is contained in the grading-index set A, giving index cost $\sum_{j=1}^{K-1} \lceil \log_2(j + 1) \rceil = \Omega(nK)$. For GGMSC, it lies in $A \cup \{a - 1 : a \in A\}$. Covering its upper half needs $\Omega(K)$ indices of $\Omega(n)$ bits. The modality bounds follow from Lemma 3.1.

We now prove item (1). Put $\epsilon = K^{-1}$ and use binary floats with precision and exponent bound $n + 3$. Give every node the initial state $(\epsilon, 0, 0, 1, 0, 0)$ in coordinates $x = (h, \tau, b, q, r, g)$, and write

y_h for the aggregated h coordinate. The shared transition is

$$\begin{aligned} h' &= \sigma(h), & \tau' &= \sigma(\tau + \epsilon), & b' &= \sigma(1 - b), \\ q' &= \sigma(Ky_h - K\tau), & r' &= \sigma(q), & g' &= \sigma(Ky_h - (K-1)). \end{aligned} \tag{3}$$

These equations have exactly the R-simple matrix form in Eq. (2): the nonzero entries are

$$\begin{aligned} C_{hh} = C_{\tau\tau} = C_{rq} = 1, \quad C_{bb} = -1, \quad C_{q\tau} = -K, \\ A_{qh} = A_{gh} = K, \quad \beta_\tau = \epsilon, \quad \beta_b = 1, \quad \beta_g = -(K-1), \end{aligned} \tag{4}$$

with every unspecified entry zero. The readout accepts when

$$r - q + b - g > 3/2. \tag{5}$$

Fix a node of degree $c < K$. Every node preserves $h = \epsilon$. The partial sums $j\epsilon$ for $0 \le j \le c$ have at most $n+1$ significant binary digits, so aggregation gives $y_h = c/K$ exactly. Induction on the round t gives

$$\tau_t = \min(t/K, 1), \quad b_t = t \bmod 2, \quad q_t = [c \ge t], \quad g_t = 0, \tag{6}$$

and $r_0 = 0$, while $r_t = q_{t-1}$ for $t \ge 1$. Thus, $(r, q) = (1, 0)$ in the unique round $t = c + 1$. The readout exceeds $3/2$ in that round exactly when $b_{c+1} = 1$, equivalently when c is even. If $c \ge K$, positive summation gives $y_h \ge 1$ and activates the guard g, so the node never accepts. Importantly, the guard is essential. Without it, at the boundary $c = K$ one has $q_{K+1} = 0$, $r_{K+1} = 1$, and $b_{K+1} = 1$, which would cause a false acceptance. An accepted node therefore fires in at most K rounds, reusing the same encoded transition throughout.

It remains to account for precision and representation size. Every below-cutoff operation is an exact difference of integers of magnitude at most K or a multiple of K^{-1}. The selected float system represents these values, K, and $K-1$ exactly. The network has constantly many nonzero parameters, each stored in an $\mathcal{O}(n)$-bit word. Together, this gives an explicit six-dimensional realization.

The recurrent construction discovers parity when its clock reaches the degree. For item (2), a layered network instead extracts the degree bits directly. Use coordinates (h, z, b, a), initialized with $h = \epsilon$ and the other coordinates zero. A first layer preserves h and defines $z_0 = \sigma(y_h)$. For $i = 1, \ldots, n$, apply a threshold layer followed by a remainder layer:

$$b_i = \sigma(Kz_{i-1} - (K/2 - 1)), \qquad z_i = \sigma(2z_{i-1} - b_i). \tag{7}$$

Writing $c/K = 0.b_1 b_2 \cdots b_n$ on the K^{-1} grid, induction gives $z_{i-1} = 0.b_i b_{i+1} \cdots b_n$: the first equation tests whether $z_{i-1} \ge 1/2$, and the second deletes that leading bit. Thus, b_n is the parity bit. If $c \ge K$, then $z_0 = 1$ and every extracted bit is one, so no separate guard is needed. A final layer outputs $\sigma(1 - b_n)$. All unspecified coefficients preserve h, z, b as needed. The only nontrivial weights are K, 2, and -1, with bias $-(K/2 - 1)$. There are $2n + 2$ layers, each with constant dimension and $\mathcal{O}(n)$-bit parameters. Appendix C gives the complete matrices and layer-wise invariants for both constructions. □

Under the explicit-feedforward convention, each of the $\mathcal{O}(n)$ layer instances is represented at $\mathcal{O}(n)$-bit word width, giving size $\mathcal{O}(n^2)$. A source language that encoded the repeated two-layer block once, together with its repetition count, would give the same computation an $\mathcal{O}(n)$-bit description.

Corollary 3.3. *For the recurrent family, source size $m = \Theta(n)$ forces GMSC size $2^{\Omega(m)}$. For the layered family, $m = \mathcal{O}(n^2)$ forces $2^{\Omega(\sqrt{m})}$. Both represent E_n, whose minimum binary-encoded GMSC and GGMSC size is $\Theta(n2^n)$ by the lower bound above and the explicit formula in Appendix C. The general GNN-to-GMSC translation rate is a separate open question.*

The corollary concerns exact representations, while practical distillation usually optimizes fidelity. The same breakpoint mechanism also controls this relaxation. Fix either the directed- or undirected-star convention, and let $\mathcal{D}_n$ be uniform over the $K = 2^n$ pointed stars $\mathrm{Star}_0, \ldots, \mathrm{Star}_{K-1}$.

Theorem 3.4 (Approximate extraction)**.** *If a GMSC program with r graded-modality occurrences agrees with E_n with probability at least $1 - \varepsilon$ under $\mathcal{D}_n$, where $0 \le \varepsilon < 1/2$, then*

$$r \ge (1 - 2\varepsilon)2^n - 1. \tag{8}$$

For GGMSC, $r \ge ((1 - 2\varepsilon)2^n - 1)/2$. Hence every fixed fidelity strictly above $1/2$ requires exponential size.

Proof. By Lemma 3.1, the GMSC verdict is constant on at most $r+1$ consecutive degree intervals. On an interval of length ℓ, a constant verdict matches the alternating parity labels at most $\lceil \ell/2 \rceil \leq \ell/2 + 1/2$ times. Summing gives fidelity at most $1/2 + (r+1)/(2K)$ and rearranging proves the claim. GGMSC has at most $2r+1$ intervals, giving $1/2 + (2r+1)/(2K)$. A complete statement, including the matching optimum for threshold surrogates, is provided in Theorem C.4. □

Figure 2 compares the theorem's uniform bound with the exact optima under uniform and triangular degree weights. The marked budgets show where each optimum first reaches fidelity 0.9.

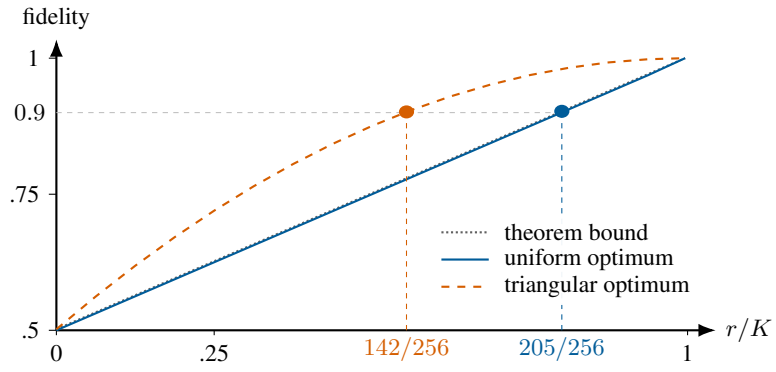


Figure 2: Optimal threshold-surrogate fidelity as a function of the normalized breakpoint budget for $n = 8$ ($K = 256$). The triangular distribution assigns degree c weight proportional to $K - c$. Fidelity 0.9 first occurs at 205 breakpoints under the uniform distribution and 142 under the triangular distribution. Both require 255 breakpoints for exact fidelity.

The accompanying code computes the complete optimum by integer dynamic programming, reconstructs the corresponding interval classifiers, and checks the uniform closed form $K/2 + \lfloor (r+1)/2 \rfloor$ correct degrees. It covers $2 \leq n \leq 8$ and all budgets $0 \leq r < K$. The triangular curve complements the uniform-case theorem by showing how a distribution concentrated on smaller degrees lowers the required budget without removing the rapid growth.

Interpretation of the separation. The upper bound classifies nodes on every finite directed graph, and hence on every undirected graph represented by a symmetric edge relation. Every node broadcasts the same constant $h = K^{-1}$, making the aggregate depend only on its out-degree, while the remaining coordinates hold the local clock and guard. The empty proposition alphabet and uniform initialization therefore isolate degree arithmetic from identifiers, ordering, randomization, and graph-size advice. Stars are needed only as witnesses for the logical lower bound. On undirected stars, two-way messages alter the leaves' states, but Lemma 3.1 shows that those states still depend on the same finite vector of degree-threshold bits and create no new breakpoint.

We provide an explicit $\Theta(K)$-modality GML formula in the appendix that places the separation within the common expressive fragment. The lower bound remains valid with binary threshold encoding, Boolean DAG sharing, synchronous recursion, and global counts. The constructions then expose a time–size tradeoff within the source model: recurrence reuses one transition for as many as 2^n rounds, whereas the layered network extracts the degree bits in $\mathcal{O}(n)$ rounds. In Section 4, we remove the temporal reuse altogether and obtain an exponential separation at depth one.

The family uses the standard non-uniform circuit-complexity convention. Its nth member has $n + \mathcal{O}(1)$ precision bits, which are included in the encoded size. At fixed precision, the aggregation cap κ_S bounds the alternating degree pattern available to this particular construction. This points toward growing dimension or different graph families as routes to a fixed-precision separation.

4 A DEPTH-ONE SEPARATION FROM WEIGHTED COUNTING

The parity constructions trade description size against time. Weighted aggregation reveals a second mechanism that needs neither long recurrence nor growing depth: one layer can compare a linear combination of neighbor counts, while graded modalities can test only a small set of threshold directions. For a node v, let $c_p(v)$ and $c_q(v)$ be the numbers of its out-neighbors satisfying p and q, respectively, and define

$$W_n(v) = [c_p(v) + 2c_q(v) \geq 2^n]. \tag{9}$$

Thus, a neighbor satisfying both propositions contributes to both counts. For $c_p, c_q \in \mathbb{N}$, let $\mathrm{Star}^{\rightarrow}_{c_p,c_q}$ have an unlabeled root, c_p leaves labeled $\{p\}$, and c_q leaves labeled $\{q\}$, with edges directed from the root. Let $\mathrm{Star}^{\leftrightarrow}_{c_p,c_q}$ be its undirected counterpart, and write Star_{c_p,c_q} for either. On these stars, W_n accepts when the weighted count reaches 2^n. This is the counting-perceptron family studied for graded modal logic by Galliani et al. (2023). The geometric proof below also covers arbitrary recursive GMSC programs.

Lemma 4.1 (Slope lemma)**.** *On either two-type star family, the verdict of a GMSC program is a Boolean function of atoms $[c_p \geq a]$, $[c_q \geq a]$, and $[c_p + c_q \geq a]$, with at most three atoms per graded-modality occurrence. For GML on directed stars, each occurrence contributes at most one atom.*

Proof. For directed stars, each leaf type evolves uniformly and independently of (c_p, c_q). For undirected stars, jointly track the root and the two uniform leaf types. A modality at a leaf sees only the root, so it introduces no count threshold. At the root, its argument holds at neither leaf type, only the p type, only the q type, or both. The modality is therefore constant or one of the three stated atoms. Induction over rule bodies and rounds proves the claim in both cases. □

Theorem 4.2 (Weighted thresholds)**.** *W_n is GML-definable and is represented by a depth-one R-simple GNN of constant dimension and $\mathcal{O}(n)$ encoded parameter size. Every equivalent GML formula contains at least 2^{n-1} graded modalities, and every equivalent GMSC program contains at least 2^{n-2} graded modalities. Under binary grading indices, either has size $\Omega(n2^n)$. When equivalence is restricted to undirected graphs, both GML and GMSC require at least 2^{n-2} graded modalities and $\Omega(n2^n)$ encoded size.*

Proof idea. The upper bound implements the desired oblique boundary with one weighted sum. The lower bound then counts adjacent lattice pairs separated by that boundary and asks how many such changes the three logical slopes can witness.

For the upper bound, initialize separate p- and q-indicator coordinates. With $n + 10$ bits of precision and exponent range, one affine combine computes a positive multiple of $c_p + 2c_q - 2^n + \frac{1}{2}$. The half-integer offset separates the integer grid exactly, while saturation can occur only after the weighted count is already above the threshold. The construction therefore has constant dimension and $\mathcal{O}(n)$ encoded size. Appendix C.3 gives the full finite-precision calculation.

For $v = 0, \ldots, 2^{n-1} - 1$, compare the adjacent points $(2^n - 2v - 1, v)$ and $(2^n - 2v, v)$. The classifier changes across every one of these edges. Only the c_p and $c_p + c_q$ slopes change on such an edge, and a threshold atom on either slope witnesses at most one value of v. Thus, 2^{n-1} edges force at least 2^{n-1} GML modalities on directed stars. For GMSC, and for GML on undirected stars, one occurrence can supply both relevant slopes, giving at least 2^{n-2} modalities. For $0 \leq v < 2^{n-2}$, both possible witnessing indices are at least 2^{n-1}. Each occurrence witnesses at most two such edges, so $\Omega(2^n)$ occurrences have $\Omega(n)$-bit indices. This proves the bit bound. Recursion and undirected return edges are covered by the slope lemma. The full slope factorization and finite-precision construction appear in Theorem C.8. □

Theorem 4.2 recovers the established nonrecursive GML/ALCQ separation and adds the recursive case. It establishes an architectural separation at depth one: a single weighted neighborhood aggregation can be exponentially more compact than threshold-only logical counting. In the final comparison, we turn from arithmetic to a different bottleneck, namely the information lost when node labels are mapped to automaton states.

5 A STATE-COMPLEXITY OBSERVATION FOR MESSAGE-PASSING AUTOMATA

Unlike GMSC, an unrestricted CMPA transition can process neighborhood multiplicities arithmetically. Its limitation arises earlier at the initialization map, which must first encode each node label into a single state.

For $n \geq 1$, write $[n] = \{1, \ldots, n\}$ and let $\Pi_n = \{s_1, \ldots, s_n, p_1, \ldots, p_n\}$ and define

$$\varphi_n = \bigvee_{i=1}^{n} (\Diamond s_i \wedge \Diamond p_i). \tag{10}$$

The property has modal depth one and formula size $\mathcal{O}(n)$.

Lemma 5.1 (Initialization bottleneck)**.** *Suppose two label sets L, L' satisfy $\pi(L) = \pi(L')$ in a CMPA. Replacing the label L of any node by L' leaves the complete run unchanged.*

Proof. At round zero, all node states agree: at the changed node this follows from $\pi(L) = \pi(L')$, and elsewhere the labels are unchanged. If the global configurations agree at round t, every node has the same current state and the same multiset of out-neighbor states in both runs. Applying the deterministic transition δ therefore gives agreement at round $t + 1$. Induction proves that the complete runs coincide. Labels are not read again after initialization. □

Theorem 5.2 (Automata state lower bound)**.** *Every CMPA representing the property in Eq.* (10) *has at least 2^n states. The property has an $\mathcal{O}(n)$-size GMSC program and a polynomial-size recurrent finite-precision GNN. The state lower bound holds already when equivalence is restricted to finite undirected graphs.*

Proof. For $S \subseteq [n]$, let $L_S = \{p_i : i \in S\}$. If the automaton has fewer than 2^n states, then, by the pigeonhole principle, $\pi(L_S) = \pi(L_{S'})$ for distinct S, S'. Choose $i \in S \triangle S'$ and construct a root with two children, one labeled $\{s_i\}$ and the other labeled L_S or $L_{S'}$. Exactly one graph satisfies φ_n, but Lemma 5.1 makes their runs identical, a contradiction. Symmetrizing the two star edges gives the same contradiction over undirected graphs. Equation (10) is the logical upper bound, and Lemma D.1 gives the GNN construction. □

The explicit recurrent GNN in Theorem D.3 has dimension $5n + 1$, constant-precision weights, linear-size circuit initialization, and accepts from round three.

Thus, the lower bound isolates initialization width rather than the later transition dynamics. Moreover, the standard translation from an m-size GMSC program to a bounded FCMPA uses $2^{\mathcal{O}(m)}$ states (Ahvonen et al., 2024), making the explicit-state gap tight. The construction compresses 2^n explicit states into symbolic initialization over a proposition alphabet of size $2n$. Establishing the corresponding state gap over a fixed alphabet remains an open direction.

6 DISCUSSION AND CONCLUSION

Expressive equivalence can conceal a large representation cost. The finite-precision GNNs constructed here are exact and low-dimensional, yet their recursive counting descriptions are exponentially larger. The proofs separate three resources that should be accounted for independently: arithmetic on multiplicities, initialization width, and temporal reuse. Temporal reuse has two consequences here. A shared recurrent transition compresses repeated arithmetic in Section 3, while unbounded recurrence also marks the verification frontier: non-emptiness is undecidable in the recurrent model but decidable at bounded depth (Appendices F and G).

For threshold-logic compilation on directed or undirected graphs, the gap already appears in feedforward architectures and persists at every fixed fidelity above $1/2$. The computational case study quantifies the effect of the input distribution and motivates distribution-aware extraction.

Future work includes richer arithmetic explanation languages, fixed-precision and fixed-alphabet separations, and extraction from trained GNNs. We discuss these directions in detail in Appendix H.

AI USE STATEMENT

In this work, we used generative AI tools for literature search, alongside exhaustive manual search and verification of AI results, writing code, refining theoretical claims, and helping verify code and claims. We also used generative AI tools to edit the manuscript and keep the main text within the page limit. We reviewed all AI-assisted work and take responsibility for the final content of the paper. We did not use generative AI tools to provide critical ingredients for proofs, write proofs, design experiments, or interpret results. Tasks involving synthetic datasets, translation, dataset cleaning or reformatting, and qualitative or thematic data analysis are not applicable to this work.

ETHICS STATEMENT

This work is theoretical and uses only synthetic graph instances. It does not involve human participants, personal data, or deployed systems. The results characterize worst-case limits of symbolic extraction and verification for GNNs. The reliability and safety of explanations for deployed models remain outside the scope of the study. We are not aware of additional ethical concerns arising directly from this work.

REPRODUCIBILITY STATEMENT

The complete proofs, including the matrix form and finite-precision analysis of both parity networks, are provided in Appendix C. The accompanying code, included in the supplementary material, uses exact arithmetic to validate the recurrent and four-coordinate layered constructions exhaustively for $1 \leq n \leq 10$, including degrees above the cutoff. It also optimizes threshold-surrogate fidelity exactly, reconstructs the extracted classifiers, and emits the data underlying Figure 2.

REFERENCES

Micah Adler and Neil Immerman. An $n!$ lower bound on formula size. *ACM Transactions on Computational Logic*, 4(3):296–314, July 2003. ISSN 1529-3785. doi: 10.1145/772062.772064.

Veeti Ahvonen, Damian Heiman, Antti Kuusisto, and Carsten Lutz. Logical characterizations of recurrent graph neural networks with reals and floats. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=atDcnWqG5n.

Veeti Ahvonen, Reijo Jaakkola, and Antti Kuusisto. Formula size game and model checking for modal substitution calculus, May 2025.

Burouj Armgaan, Manthan Dalmia, Sourav Medya, and Sayan Ranu. Graphtrail: Translating GNN predictions into human-interpretable logical rules. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=fzlMza6dRZ.

Steve Azzolin, Antonio Longa, Pietro Barbiero, Pietro Liò, and Andrea Passerini. Global explainability of GNNs via logic combination of learned concepts. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=OTbRTIY4YS.

Franz Baader. A New Description Logic with Set Constraints and Cardinality Constraints on Role Successors. In *Frontiers of Combining Systems*, volume 10483 of *Lecture Notes in Computer Science*, pp. 43–59. Springer International Publishing, 2017. ISBN 978-3-319-66167-4. doi: 10.1007/978-3-319-66167-4_3.

Franz Baader. Expressive cardinality constraints on ALCSCC concepts. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (SAC 2019)*, pp. 1123–1130. ACM, April 2019. doi: 10.1145/3297280.3297390.

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*, 4(2):1–19, May 2026. ISSN 2836-6573. doi: 10.1145/3801914.

Pablo Barceló, Egor V. Kostylev, Mikaël Monet, Jorge Pérez, Juan Reutter, and Juan Pablo Silva. The logical expressiveness of graph neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=r1lZ7AEKvB.

Timon Barlag, Vivian Holzapfel, Laura Strieker, Jonni Virtema, and Heribert Vollmer. Recurrent Graph Neural Networks and Arithmetic Circuits. In *Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning (KR 2026)*, pp. 826–836. International Joint Conferences on Artificial Intelligence Organization, July 2026. doi: 10.24963/kr.2026/78.

Michael Benedikt, Chia-Hsuan Lu, and Tony Tan. Decidability of Graph Neural Networks via Logical Characterizations. *ACM Transactions on Computational Logic*, 27(2):1–65, April 2026. ISSN 1529-3785. doi: 10.1145/3800944.

Pascal Bergsträßer, Ryan Cotterell, and Anthony Widjaja Lin. Transformers are inherently succinct. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Yxz92UuPLQ.

Jeroen Bollen, Jan Van den Bussche, Stijn Vansummeren, and Jonni Virtema. Halting Recurrent GNNs and the Graded μ-Calculus. In *Proceedings of the 22nd International Conference on Principles of Knowledge Representation and Reasoning (KR 2025)*, pp. 175–184. International Joint Conferences on Artificial Intelligence Organization, November 2025. doi: 10.24963/kr.2025/18.

Bernardo Cuenca Grau, Eva Feng, and Przemysław Andrzej Wałęga. The Correspondence Between Bounded Graph Neural Networks and Fragments of First-Order Logic. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(23):19135–19142, March 2026. ISSN 2374-3468. doi: 10.1609/aaai.v40i23.38987.

Stéphane Demri and Denis Lugiez. Complexity of modal logics with Presburger constraints. *Journal of Applied Logic*, 8(3):233–252, September 2010. ISSN 1570-8683. doi: 10.1016/j.jal.2010.03.001.

Grégoire Fournier and György Turán. A Game for Counting Logic Formula Size and an Application to Linear Orders. In Stefano Guerrini and Barbara König (eds.), *34th EACSL Annual Conference on Computer Science Logic (CSL 2026)*, volume 363 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pp. 36:1–36:22, Dagstuhl, Germany, 2026. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-411-6. doi: 10.4230/LIPIcs.CSL.2026.36. URL https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CSL.2026.36.

Tim French, Wiebe van der Hoek, Petar Iliev, and Barteld Kooi. On the succinctness of some modal logics. *Artificial Intelligence*, 197:56–85, April 2013. ISSN 0004-3702. doi: 10.1016/j.artint.2013.02.003.

Pietro Galliani, Oliver Kutz, and Nicolas Troquard. Succinctness and Complexity of ALC with Counting Perceptrons. In *Proceedings of the 20th International Conference on Principles of Knowledge Representation and Reasoning (KR 2023)*, pp. 291–300. International Joint Conferences on Artificial Intelligence Organization, September 2023. doi: 10.24963/kr.2023/29.

Chuqin Geng, Ziyu Zhao, Zhaoyue Wang, Haolin Ye, Yuhe Jiang, and Xujie Si. LogicXGNN: Grounded logical rules for explaining graph neural networks. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=nqZe43tRY9.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 1263–1272. JMLR.org, 2017.

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, June 2021. doi: 10.1109/lics52264.2021.9470677.

Martin Grohe. The Descriptive Complexity of Graph Neural Networks. *TheoretiCS*, 3, December 2024. ISSN 2751-4838. doi: 10.46298/theoretics.24.25.

Stan P Hauke and Przemysław Andrzej Wałęga. Aggregate-Combine-Readout GNNs Can Express Logical Classifiers Beyond the Logic C2. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(26):21594–21601, March 2026. ISSN 2374-3468. doi: 10.1609/aaai.v40i26.39308.

Lauri Hella and Miikka Vilander. Formula size games for modal logic and μ-calculus. *Journal of Logic and Computation*, 29(8):1311–1344, December 2019. ISSN 0955-792X. doi: 10.1093/logcom/exz025.

Markus Holzer and Martin Kutrib. Descriptional and computational complexity of finite automata—A survey. *Information and Computation*, 209(3):456–470, March 2011. ISSN 0890-5401. doi: 10.1016/j.ic.2010.11.013.

Yevgeny Kazakov and Ian Pratt-Hartmann. A Note on the Complexity of the Satisfiability Problem for Graded Modal Logics. In *2009 24th Annual IEEE Symposium on Logic In Computer Science (LICS)*, pp. 407–416. IEEE, August 2009. doi: 10.1109/lics.2009.17.

Niklas Kemper, Tom Wollschläger, and Stephan Günnemann. What Expressivity Theory Misses: Message Passing Complexity for GNNs. In *Advances in Neural Information Processing Systems*, volume 38, pp. 49574–49617. Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2025. doi: 10.52202/085713-1477.

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. URL https://openreview.net/forum?id=lsvGqR6OTf.

Orna Kupferman, Ulrike Sattler, and Moshe Y. Vardi. The Complexity of the Graded μ-Calculus. In *Automated Deduction—CADE-18*, volume 2392 of *Lecture Notes in Computer Science*, pp. 423–437. Springer Berlin Heidelberg, 2002. ISBN 978-3-540-45620-9. doi: 10.1007/3-540-45620-1_34.

Antti Kuusisto and Fabian Reiter. Emptiness problems for distributed automata. *Information and Computation*, 272:104503, June 2020. ISSN 0890-5401. doi: 10.1016/j.ic.2019.104503.

Huan Luo and Jonni Virtema. Unifying Approach to Uniform Expressivity of Graph Neural Networks. In *Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning (KR 2026)*, pp. 1030–1040. International Joint Conferences on Artificial Intelligence Organization, July 2026. doi: 10.24963/kr.2026/96.

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):4602–4609, July 2019. ISSN 2374-3468. doi: 10.1609/aaai.v33i01.33014602.

Pierre Nunn, Marco Sälzer, François Schwarzentruber, and Nicolas Troquard. A Logic for Reasoning about Aggregate-Combine Graph Neural Networks. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, pp. 3532–3540. International Joint Conferences on Artificial Intelligence Organization, August 2024. doi: 10.24963/ijcai.2024/391.

Tamara Pereira, Erik Nascimento, Lucas E. Resck, Diego Mesquita, and Amauri Souza. Distill n' explain: explaining graph neural networks using simple surrogates. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent (eds.), *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 6199–6214. PMLR, 25–27 Apr 2023. URL https://proceedings.mlr.press/v206/pereira23a.html.

Maximilian Pflueger, David Tena Cucala, and Egor V. Kostylev. Recurrent Graph Neural Networks and Their Connections to Bisimulation and Logic. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(13):14608–14616, March 2024. ISSN 2374-3468. doi: 10.1609/aaai.v38i13.29377.

Alexander Pluska, Pascal Welke, Thomas Gärtner, and Sagar Malhotra. Logical Distillation of Graph Neural Networks. In *Proceedings of the 21st International Conference on Principles of Knowledge Representation and Reasoning (KR 2024)*, pp. 920–930. International Joint Conferences on Artificial Intelligence Organization, November 2024. doi: 10.24963/kr.2024/86.

Eran Rosenbluth and Martin Grohe. Repetition Makes Perfect: Recurrent Graph Neural Networks Match Message Passing Limit. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40 (30):25168–25175, March 2026. ISSN 2374-3468. doi: 10.1609/aaai.v40i30.39707.

Marco Sälzer and Martin Lange. Fundamental limits in formal verification of message-passing neural networks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WlbG820mRH-.

Marco Sälzer, François Schwarzentruber, and Nicolas Troquard. Verifying Quantized Graph Neural Networks is PSPACE-complete. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence (IJCAI-25)*, pp. 4660–4668. International Joint Conferences on Artificial Intelligence Organization, September 2025. doi: 10.24963/ijcai.2025/519.

Marco Sälzer, Pascal Bergsträßer, and Anthony W. Lin. The Polynomial Counting Capabilities of Message Passing Neural Networks. In *Proceedings of the 23rd International Conference on Principles of Knowledge Representation and Reasoning (KR 2026)*, pp. 1121–1131. International Joint Conferences on Artificial Intelligence Organization, July 2026. doi: 10.24963/kr.2026/105.

François Schwarzentruber. Lecture Notes on Verifying Graph Neural Networks, October 2025.

Stephan Tobies. PSPACE Reasoning for Graded Modal Logics. *Journal of Logic and Computation*, 11(1):85–106, February 2001. ISSN 0955-792X. doi: 10.1093/logcom/11.1.85.

Przemysław Andrzej Wałęga and Bernardo Cuenca Grau. Structural Preservation and the Logical Expressiveness of Graph Neural Networks, June 2026.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

Tuo Xu. Towards a complete logical framework for GNN expressiveness. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=pqOjj90Vwp.

Andy Yang, David Chiang, and Dana Angluin. Masked hard-attention transformers recognize exactly the star-free languages. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9798331314385.

Zhitao Ying, Dylan Bourgeois, Jiaxuan You, Marinka Zitnik, and Jure Leskovec. Gnnexplainer: Generating explanations for graph neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/d80b7040b773199015de6d3b4293c8ff-Paper.pdf.

APPENDIX CONTENTS

A Preliminaries **16**

A.1 Graphs and Logics 16

A.2 Counting Message-Passing Automata 16

A.3 Recurrent GNNs over Floats 17

A.4 Succinctness 18

B Related Work and Positioning **18**

C Exponential Succinctness over the Logic **19**

C.1 Bounded Parity 19

C.2 Approximate Extraction 22

C.3 Weighted Counting at Depth One 23

D A State-Complexity Observation for Message-Passing Automata **25**

E Structural Features of Graph Message Passing **27**

F Undecidability of Verification **28**

G Decidability at Bounded Depth **29**

H Future Work **31**

We begin by collecting the definitions used throughout the paper and giving a fuller account of related work. We then prove the two logical succinctness results in the same order as in the main text. We first treat bounded parity and then turn to weighted counting. In the remaining sections, we develop the automata comparison and study the verification consequences of recurrence. These results concern different sources of complexity and do not depend on the central arithmetic-versus-threshold separation.

A Preliminaries

A.1 Graphs and Logics

Fix a countably infinite supply of proposition symbols, and let Π be a finite proposition alphabet. A Π-*labeled graph* is a tuple $G = (V, E, \lambda)$ with $E \subseteq V \times V$ and $\lambda\colon V \to \mathcal{P}(\Pi)$. A *pointed graph* is a pair (G, w) with $w \in V$. The *out-neighbors* of w are the nodes v with $(w, v) \in E$. A *node property over* Π is an isomorphism-closed class of pointed Π-labeled graphs. Undirected graphs are represented by symmetric E. In that case, out-neighbors are simply neighbors. We take $\mathbb{N} = \{0, 1, \ldots\}$. The GNN constructions work on all finite directed graphs, and the parity and weighted-counting lower bounds also hold when equivalence is required only on the undirected subclass (Sections C and C.3).

Graded modal logic (GML) over Π is given by $\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Diamond_{\geq k}\varphi$ with $p \in \Pi$, $k \geq 1$. $G, w \models \Diamond_{\geq k}\varphi$ iff at least k out-neighbors of w satisfy φ. We write $\Diamond$ for $\Diamond_{\geq 1}$. The *size* $\|\varphi\|$ counts occurrences of symbols, with $\Diamond_{\geq k}$ contributing $\lceil \log_2(k{+}1) \rceil$. The indices are binary coded. The modality-count lower bounds are coding-independent, while binary coding strengthens them to the bit bounds stated below.

Graded modal substitution calculus (GMSC) (Ahvonen et al., 2024; 2025) extends GML with recursion. A *program* Λ over $(\Pi, \mathcal{T})$ consists of a base rule $X(0) :- \varphi_X$ and an induction rule $X :- \psi_X$ for each head predicate $X \in \mathcal{T}$ (where φ_X is a GML formula over Π and ψ_X may additionally use head predicates as atoms), together with accepting predicates $\mathcal{A} \subseteq \mathcal{T}$. Round 0 truth values are given by the base rules. Round $t{+}1$ values are given by the induction rules evaluated at round-t values. Λ *accepts* (G, w) iff some $X \in \mathcal{A}$ holds at w in some round. Program size is $|\Pi| + |\mathcal{T}|$ plus the sizes of all rule bodies. Global-counting GMSC (GGMSC) additionally permits the counting global modality $\langle E \rangle_{\geq k}\varphi$, true at a node iff at least k nodes of the whole graph satisfy φ. Its round semantics and size measure are otherwise the same as for GMSC (Ahvonen et al., 2025).

Both logics evolve synchronously and accept when a designated predicate becomes true. Counting message-passing automata provide the corresponding state-based view of this round-by-round computation.

A.2 Counting Message-Passing Automata

We use the counting message-passing automata of Ahvonen et al. (2024, Appendix B). A *counting message-passing automaton* (CMPA) over Π is a tuple $A = (Q, \pi, \delta, F)$: a non-empty set of states Q, an *initialization function* $\pi\colon \mathcal{P}(\Pi) \to Q$, a *transition function* $\delta\colon Q \times \mathcal{M}(Q) \to Q$ over multisets of states, and accepting states $F \subseteq Q$. On a Π-labeled graph G the *run* is the sequence of global configurations $g_t\colon V \to Q$ with $g_0(w) = \pi(\lambda(w))$ and $g_{t+1}(w) = \delta\big(g_t(w), \{\!\{g_t(v) : v \text{ an out-neighbor of } w\}\!\}\big)$. A *accepts* (G, w) iff $g_t(w) \in F$ for some $t \in \mathbb{N}$. A is κ-*bounded* if $\delta(q, M) = \delta(q, M_{|\kappa})$ for all q, M, where $M_{|\kappa}$ caps all multiplicities at κ. *Bounded* means κ-bounded for some κ. A *finite counting message-passing automaton* (FCMPA) is a CMPA with finite Q.

Throughout, *time* means the round index t. A depth-L network runs for exactly L rounds, recurrent networks run indefinitely, and GMSC programs and automata use the same round convention. Two features of this model will be useful later. First, node labels affect a run only through π at round zero, because δ never reads λ. Second, once the initial configuration is fixed, δ determines the rest of the run uniquely.

Remark A.1 (Size of automata)**.** We measure an FCMPA by its number of states $|Q|$, as is standard for automata and as Bergsträßer et al. (2026) measure DFAs. (The look-up table measure of Ahvonen et al. (2024) additionally counts $|\mathcal{P}(\Pi)|$ and the transition table. They note explicitly in their

Appendix A.1 that "there is no single way to define the sizes".) All lower bounds below concern the state-space cardinality $|Q|$, independently of how π, δ, and F are represented. They are not lower bounds on description length: a small circuit can denote an automaton with many states, and the GNNs below are themselves such descriptions. The point is that no automaton with few states exists at all.

A.3 Recurrent GNNs over Floats

We now replace discrete automaton states by finite floating-point vectors. The finiteness of the numeric domain preserves the connection with bounded automata, while affine aggregation exposes the arithmetic operations used by our compact constructions. Following Ahvonen et al. (2024), a recurrent $\mathrm{GNN}[F]$ over Π has state space S^d for a finite floating-point system S, an initialization $\pi\colon \mathcal{P}(\Pi) \to S^d$, and an aggregate-combine transition. An ***R-simple*** transition is exactly

$$x_v^{t+1} = \mathrm{ReLU}^*\big(Cx_v^t + A\,\mathrm{SUM}_S\{\{x_u^t : (v,u) \in E\}\} + b\big)\,, \tag{11}$$

where $C, A \in S^{d\times d}$ and $b \in S^d$. The matrix C acts on the node's current state, while A acts on the aggregated neighbor state and b is the bias. More explicitly, if $y_v^t = \mathrm{SUM}_S\{\{x_u^t : (v,u) \in E\}\}$, then the neighbor term is the ordinary matrix–vector product Ay_v^t. Here, A is a network parameter, and not the graph's adjacency matrix. The function SUM_S adds the multiset of neighbor vectors coordinatewise in the finite floating-point system S. Finally, $\mathrm{ReLU}^*(z) = \min(\max(0,z),1)$ componentwise, with every operation evaluated in S. In each affine coordinate, products are summed in increasing coordinate order. The resulting Cx, Ay, and b terms are then added in the displayed order, with rounding after every primitive operation. Acceptance is by an explicitly described affine threshold on the current feature vector. A recurrent $\mathrm{GNN}[F]$ accepts a pointed graph when that threshold holds at the distinguished node in some round. A layered $\mathrm{GNN}[F]$ reads out its final layer. Float summation makes boundedness automatic. By Ahvonen et al. (2024, Prop. 2.3 and App. A.2), for every S, there is κ_S such that every finite multiset M satisfies $\mathrm{SUM}_S(M) = \mathrm{SUM}_S(M_{|\kappa_S})$, and $\kappa_S = \mathcal{O}(\beta^{p+1})$ for radix β and precision p.

Our *parametric size* includes $d + |\Pi|$, the binary description of the float format, and every entry of C, A, b at the format's word width. We represent π by a straight-line circuit whose inputs are the bits $[p \in L]$ for $p \in \Pi$ and whose gates are affine maps followed by ReLU^* over S. Circuit gates, wires, and constants contribute their encoded lengths. The same applies to the affine-threshold readout. This explicitly accounts for precision and initialization. The parity construction has a constant initialization, while the automata witness uses $2n$ direct input wires for the characteristic vector.

This is the analogue of encoded parameter count for the transformers of Bergsträßer et al. (2026). It differs from the look-up-table measure used by Ahvonen et al. (2024), which treats a state space U of size N and counting cap κ as a table of size $|\mathcal{P}(\Pi)| + N(\kappa+1)^N$ and can therefore be doubly exponential in our parametric size.

Remark A.2 (Precision accounting)**.** The exact clock, bit-extraction, and weighted-threshold constructions used in our upper bounds (Theorems C.2, C.8) use a mantissa of $n + \mathcal{O}(1)$ bits. With constant dimension and constantly many weights, this gives parametric size $\mathcal{O}(n)$. The measure includes this linear precision in the same way as the transformer measure of Bergsträßer et al. (2026). Scaling bit-width with the family index is also standard in circuit complexity. TC^0 families and the relevant GNN characterizations use the same convention (Grohe, 2024).

At a fixed width, the construction extends to the largest representable n. For example, a 23-bit `float32` mantissa supports roughly $n \le 21$. The theorem is therefore an asymptotic non-uniform family with precision linear in its index. Each member fixes its float system explicitly and computes exactly on every state relevant to acceptance, using monotonicity and saturation above the cutoff.

The aggregation cap also explains the role of growing precision in this witness. For fixed S, the aggregate of identical positive leaf features stabilizes once $c \ge \kappa_S$, bounding the available alternating degree pattern. Fixed-precision separations based on growing dimension or different graph families form a complementary direction.

A.4 Succinctness

With the three representation models fixed, we can state the quantitative comparison used throughout the paper. We follow Definitions 2 and 3 of Bergsträßer et al. (2026). For classes $\mathcal{C}^{(1)}, \mathcal{C}^{(2)}$ of representations of node properties and $f\colon \mathbb{N} \to \mathbb{N}$, $\mathcal{C}^{(1)}$ is *f-more succinct* than $\mathcal{C}^{(2)}$ if there is a family $\{P_n\}_{n\geq 1}$ of node properties with representations $\mathcal{R}_n^{(1)} \in \mathcal{C}^{(1)}$ such that every $\mathcal{R}_n^{(2)} \in \mathcal{C}^{(2)}$ representing P_n satisfies $\|\mathcal{R}_n^{(2)}\| \geq f(\|\mathcal{R}_n^{(1)}\|)$. It is *exponentially more succinct* if there are constants $c, e > 0$ and m_0 such that $f(m) \geq 2^{cm^e}$ for every $m \geq m_0$. Dually, $\mathcal{C}^{(1)}$ has *g-bounded expansion* over $\mathcal{C}^{(2)}$ if every property representable in $\mathcal{C}^{(2)}$ by $\mathcal{R}$ has a $\mathcal{C}^{(1)}$-representation of size $\leq g(\|\mathcal{R}\|)$.

Remark A.3. Recall that, in the string setting, the transformers of Bergsträßer et al. (2026) recognize exactly the LTL languages (Yang et al., 2024). Our logical test families, E_n and W_n, are definable already in basic GML and therefore lie in the common expressive fragment of the models compared in their respective theorems.

B Related Work and Positioning

The paper lies at the intersection of logical characterizations of GNNs, descriptional complexity, arithmetic counting logics, and symbolic explanation. These literatures often compare expressive power or reasoning complexity. Our question is instead quantitative: once two formalisms can express the same classifier, how large must the target representation be?

Message-passing neural networks were placed in a common framework by Gilmer et al. (2017), and their relation to Weisfeiler–Leman refinement was established by Xu et al. (2019) and Morris et al. (2019). Logical characterizations of aggregate-combine and recurrent GNNs are due to Barceló et al. (2020); Grohe (2021); Ahvonen et al. (2024). Subsequent work develops descriptive-complexity, structural-preservation, and uniform expressivity perspectives (Grohe, 2024; Xu, 2025; Cuenca Grau et al., 2026; Wałęga & Cuenca Grau, 2026; Luo & Virtema, 2026). Hauke & Wałęga (2026) show that aggregate-combine-readout GNNs can express logical classifiers beyond the two-variable counting logic C^2. Recurrent GNNs have also been related to fixpoint logics, arithmetic circuits, and computation completeness (Pflueger et al., 2024; Barlag et al., 2026; Rosenbluth & Grohe, 2026), while Barceló et al. (2026) analyze how linear maps and activation functions affect GNN-style computation. We complement these expressive characterizations with lower bounds on representation size.

The comparison with Weisfeiler–Leman becomes quantitative once the decoder is counted. On a star, one refinement round gives the root a color determined by its degree. An arithmetic decoder can recover bounded parity from that color compactly, whereas an explicit lookup or a collection of GMSC thresholds is exponentially larger. This representation-level question is complementary to efficient simulations of bounded-degree logical queries by GNNs (Khalife & Tonelli-Cueto, 2025) and to task-level message-passing complexity (Kemper et al., 2025). Our size conventions follow the descriptional-complexity treatment of transformers by Bergsträßer et al. (2026). General modal lower-bound techniques use formula-size games (Adler & Immerman, 2003; French et al., 2013; Hella & Vilander, 2019; Ahvonen et al., 2025; Fournier & Turán, 2026); the star families here admit more direct breakpoint and slope arguments.

Arithmetic successor constraints have a longer history in modal and description logics. Presburger modal logic permits arithmetic constraints on successor counts (Demri & Lugiez, 2010), and ALCSCC combines Boolean-algebra and Presburger constraints over role successors (Baader, 2017; 2019). Arithmetic counting logic $\mathrm{K}^{\#}$ has polynomial translations into two-way aggregate-combine GNNs (Nunn et al., 2024), and Sälzer et al. (2026) extend this landscape to polynomial constraints. Linear inequalities compactly express the oblique boundary of W_n, but on one-type stars they still induce only ordinary degree thresholds and do not compactly express the alternating family E_n. The parity gap can instead be removed by adding modular counting, divisibility, or full Presburger constraints. Thus, different arithmetic extensions address the two separations for different reasons. The closest earlier succinctness theorem is the positive counting-perceptron/ALCQ separation of Galliani et al. (2023). Our depth-one theorem extends its lower-bound mechanism to

recursive GMSC. The bounded-parity theorem gives a different family, an exact finite-float realization, tight GMSC and GGMSC bounds, and approximate as well as undirected variants.

The extraction interpretation connects these formal bounds to explanation. Local and surrogate approaches include Ying et al. (2019) and Pereira et al. (2023). Global methods build formulas from concepts (Azzolin et al., 2023), regress rules over subgraphs (Armgaan et al., 2024), or distill counting and grounded rules (Pluska et al., 2024; Geng et al., 2026). Those methods optimize empirical fidelity. Our stated teacher and graph distribution identify a family on which every threshold-only global surrogate of fidelity above one half remains exponentially large.

Finally, verification provides the broader setting for the later results in the appendix. Undecidability for general message-passing neural networks was shown by Sälzer & Lange (2023); bounded-depth logical verification and quantized models are studied by Benedikt et al. (2026); Nunn et al. (2024); Sälzer et al. (2025). We use distributed-automata emptiness (Kuusisto & Reiter, 2020) to locate the corresponding recurrent frontier for the finite-precision model considered here.

C Exponential Succinctness over the Logic

This section proves the paper's central separation in three steps. First, a breakpoint lemma describes everything that recursive threshold counting can observe on stars. We then construct recurrent and layered GNNs for bounded parity and compare their encoded sizes with the resulting logical lower bound. Finally, we extend the argument to approximate extraction and to a depth-one weighted-counting family.

C.1 Bounded Parity

For $c \in \mathbb{N}$, let $\mathrm{Star}_c^{\rightarrow}$ be the pointed graph consisting of a root with c out-neighbors and no return edges, all labels empty, and let $\mathrm{Star}_c^{\leftrightarrow}$ be the corresponding undirected star. We write Star_c for either family when the statement holds for both.

Lemma C.1 (Breakpoint lemma)**.** *Let Λ be a* GMSC *program (in particular, a* GML *formula) over any Π, and let $A \subseteq \mathbb{N}$ be the set of grading indices appearing in diamonds of Λ. On either star family, the verdict of Λ, as a function of c, is constant on every interval between consecutive elements of A. Hence the verdict has at most $|A|$ breakpoints, and $|A|$ is at most the number of diamond occurrences in Λ (distinct diamond subformulas, for directed acyclic graph (DAG) representations), regardless of how grading indices are coded. In particular, the bound also covers an arbitrary Boolean circuit over shared modal atoms. For a* GGMSC *program, with A containing the indices of local and global counting modalities, the verdict has at most $2|A|$ breakpoints.*

Proof. For $\mathrm{Star}_c^{\rightarrow}$, the leaves have empty out-neighborhoods, and the familiar one-sided induction gives the claim. For $\mathrm{Star}_c^{\leftrightarrow}$, simultaneously induct on rounds and rule bodies for the root and the uniform leaves. At the root, $\Diamond_{\geq a}\psi$ is either false or $[c \geq a]$, according to the common truth of ψ at the leaves. At a leaf, the same modality sees only the root. It is false for $a > 1$ and, for $a = 1$, equals the inductively known truth of ψ at the root. Thus, the complete joint root/leaf valuation in either star family is a function only of $([c \geq a])_{a \in A}$. Such a function can change only at elements of A. Two-way messages can reuse the threshold bits but cannot create a new one.

For GGMSC, simultaneously induct on rounds and rule bodies for the root and the uniform leaves. A global modality counts either ρ or $c + \rho$ nodes, where $\rho \in \{0, 1\}$ records whether its argument holds at the root. Its truth is therefore determined by $[c \geq a]$ and $[c \geq a - 1]$, together with the inductively determined truth values of its argument. Thus, every root and leaf truth value factors through $([c \geq a], [c \geq a - 1])_{a \in A}$, which changes at no more than $2|A|$ positions. □

Theorem C.2 (Bounded parity)**.** *For $n \geq 1$, let E_n be the node property "the out-degree is even and less than 2^n" (over $\Pi = \emptyset$). Then:*

1. *E_n is* GML*-definable, and is expressed by a recurrent R-simple* $\mathrm{GNN}[F]$ *of dimension* 6 *and $\mathcal{O}(n)$ parametric size, and by a depth-$(2n + 2)$ layered R-simple AC-*$\mathrm{GNN}[F]$ *of $\mathcal{O}(n^2)$ parametric size.*

2. *every* GMSC *program expressing* E_n*, in particular every* GML *formula, contains at least* $2^n - 1$ *graded modalities, and every* GGMSC *program contains at least* 2^{n-1} *graded modalities. With binary-coded gradings, every such* GMSC *or* GGMSC *program has size* $\Omega(n2^n)$*, in tree or DAG representation.*

The lower bounds in item (2) also hold when equivalence is required only over finite undirected graphs.

Proof. We prove the logical lower bound first, then give the recurrent and layered upper bounds. This order makes clear that the lower bound depends only on the alternating degree pattern, not on details of either neural construction.

(2) On Star_c the verdict of E_n flips between every consecutive $c, c+1$ with $c+1 < 2^n$, hence has $2^n - 1$ breakpoints. Lemma C.1 forces at least $2^n - 1$ local modalities in GMSC, and at least $\lceil (2^n - 1)/2 \rceil = 2^{n-1}$ local or global modalities in GGMSC.

For the bit bound, put $K = 2^n$. In GMSC, the breakpoint set is contained in the set A of grading indices. Therefore, $\{1, \dots, K-1\} \subseteq A$, and the indices alone cost

$$\sum_{j=1}^{K-1} \lceil \log_2(j+1) \rceil = \Theta(nK). \tag{12}$$

In GGMSC, the breakpoint set lies in $A \cup \{a - 1 : a \in A\}$. Each index covers at most two breakpoints. Covering those in $[K/2, K-1]$ therefore requires $\Omega(K)$ indices of value at least $K/2$, each costing $\Omega(n)$ bits. Hence the same $\Omega(nK)$ bound holds with global counting and under DAG sharing.

Having established the lower bound, we turn to the two representations that attain the compact side of the separation.

(1) The formula

$$\neg\Diamond_{\geq 1}\top \ \vee \bigvee_{\substack{0<j<2^n \\ j \text{ even}}} (\Diamond_{\geq j}\top \wedge \neg\Diamond_{\geq j+1}\top) \tag{13}$$

defines E_n. We next give explicit recurrent and layered constructions.

Recurrent construction. Put $K = 2^n$ and $\epsilon = K^{-1}$. Choose the binary floating-point system $S_n = ((n+3, n+3, 2), +, \cdot)$ of Ahvonen et al. (2024). It has precision $n+3$, exponents between $-(n+3)$ and $n+3$, round-to-nearest with ties-to-even, and saturating overflow. All of the values ϵ, K, $K-1$, and $3/2$ are representable.

The six coordinates separate the roles of the construction. The coordinate h broadcasts the grid spacing, τ and b form a clock with an alternating bit, q and r detect the clock's first crossing of the degree, and g rejects degrees at or above the cutoff. Use the constant initialization

$$\pi(\emptyset) = (\epsilon, 0, 0, 1, 0, 0)^\top. \tag{14}$$

For $y = \mathrm{SUM}_{S_n}\{\{x_u : (v,u) \in E\}\}$, define the R-simple transition $x' = \sigma(Cx + Ay + d)$, using d for the bias vector to avoid confusion with the state coordinate b. Here $\sigma = \mathrm{ReLU}^*$ and

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & -K & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \qquad A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ K & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ K & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \qquad d = \begin{pmatrix} 0 \\ \epsilon \\ 1 \\ 0 \\ 0 \\ -(K-1) \end{pmatrix}. \tag{15}$$

Equivalently, its coordinate equations are

$$\begin{aligned} h' &= \sigma(h), & \tau' &= \sigma(\tau + \epsilon), & b' &= \sigma(1 - b), \\ q' &= \sigma(Ky_h - K\tau), & r' &= \sigma(q), & g' &= \sigma(Ky_h - (K-1)). \end{aligned} \tag{16}$$

The affine readout accepts precisely when

$$r - q + b - g > 3/2. \tag{17}$$

Fix a node of degree c. Since every node keeps $h = \epsilon$, y_h is the increasing-order float sum of c copies of ϵ. For $c \le K$ every partial sum $j\epsilon$, $0 \le j \le c$, has at most $n+1$ significant binary digits and is therefore exact in S_n. Hence $y_h = c/K$. For $c > K$, the positive sum first reaches 1 after K additions and can never decrease. Therefore, $y_h \ge 1$ regardless of later rounding or saturation.

We can now verify the two degree regimes. Suppose first that $c < K$. Direct induction on t, for all rounds relevant before the clock saturates, gives

$$\begin{aligned} &\tau_t = \min(t/K, 1), \qquad b_t = t \bmod 2, \qquad q_t = [c \ge t], \\ &r_t = q_{t-1} \quad (t \ge 1), \quad g_t = 0 \quad (t \ge 1). \end{aligned} \tag{18}$$

Indeed, $q_{t+1} = \sigma(c - t)$, an exact integer calculation. Before round $c+1$ we have $q = 1$, after it we have $r = 0$, and in round $c+1$ alone we have $(r, q) = (1, 0)$. The score in Eq. (17) can exceed $3/2$ only in that round, where it equals $1 + b_{c+1}$. This is 2 exactly when c is even. Round zero rejects, and for $c = 0$, the unique accepting round is round one.

Above the cutoff, the guard takes over. If $c \ge K$, then $g_t = 1$ from round one and the readout score is at most 1. Round zero also rejects. Importantly, the guard is essential. Without it, at $c = K$ the round $K+1$ values would satisfy $(r, q, b) = (1, 0, 1)$ and the readout would accept. Thus, the network expresses E_n on every finite directed graph. Stars serve as the lower-bound witness family. All calculations used for $c < K$ are exact differences of integers of magnitude at most K or exact multiples of K^{-1}. The two matrices have constant size and constantly many nonzero entries, each stored in an $\mathcal{O}(n)$-bit word. Therefore, the parametric size is $\mathcal{O}(n)$. This gives an explicit six-dimensional realization.

Layered construction. While the recurrent network waits until its clock crosses the stored degree, the layered network uses its depth differently: each pair of layers extracts and removes one binary digit of the normalized degree. Use four coordinates (h, z, b, a) and the same S_n and initialization $h = \epsilon$, all other coordinates zero. Again write y for the aggregated neighbor state. A first R-simple layer has the only nonzero coefficients

$$C_{hh} = 1, \qquad A_{zh} = 1. \tag{19}$$

This layer defines $z_0 = \sigma(y_h)$. For each $i = 1, \dots, n$, a threshold layer preserves h, z and uses

$$C_{bz} = K, \qquad d_b = -(K/2 - 1), \tag{20}$$

thereby computing $b_i = \sigma(K z_{i-1} - (K/2 - 1))$. The following remainder layer preserves h, b and has $C_{zz} = 2, C_{zb} = -1$, computing $z_i = \sigma(2z_{i-1} - b_i)$. All unspecified matrix and bias entries are zero. Below the cutoff, writing $c/K = 0.b_1 b_2 \cdots b_n$, induction gives $z_{i-1} = 0.b_i b_{i+1} \cdots b_n$ in binary. The threshold is exactly 0 at the largest grid point below $1/2$ and exactly 1 at $1/2$, while the remainder update deletes the leading bit. Thus, b_n is the parity bit. For $c \ge K$, clipping gives $z_0 = 1$, after which every threshold layer sets $b_i = 1$ and every remainder layer restores $z_i = 1$. The last bit therefore rejects without a guard. A final layer with $C_{ab} = -1, d_a = 1$ accepts iff $a = 1$, equivalently iff $b_n = 0$. The depth is $1 + 2n + 1 = 2n + 2$. There are $\mathcal{O}(n)$ constant-size matrices over $\mathcal{O}(n)$-bit words, hence $\mathcal{O}(n^2)$ encoded parameter bits. Thus, for source encoded size m, this construction yields a target lower bound $2^{\Omega(n)} = 2^{\Omega(\sqrt{m})}$ with depth $\mathcal{O}(n)$.

The $\mathcal{O}(n^2)$ count follows the explicit-layer convention in Section A, which includes every instantiated layer. A source language that encoded the repeated two-layer block once, together with its repetition count, would give the same computation an $\mathcal{O}(n)$-bit description.

The accompanying code exhaustively checks both recurrences with exact rational arithmetic for the supplied small instances, including all cutoff boundary cases. □

Corollary C.3. *For the recurrent family, encoded source size $m = \Theta(n)$ forces GMSC size $2^{\Omega(m)}$. For the layered family, $m = \mathcal{O}(n^2)$ forces $2^{\Omega(\sqrt{m})}$. Both statements hold already over the empty alphabet. The explicit formula in the proof has size $\mathcal{O}(n2^n)$ under binary threshold encoding. Consequently, the minimum binary-encoded* GMSC *and* GGMSC *size for E_n is $\Theta(n2^n)$.*

This corollary settles the optimal size for the family itself. But it does not settle the worst-case translation rate between the two formalisms. The general translation of Ahvonen et al. (2024) is polynomial in their look-up-table measure $|\mathcal{P}(\Pi)| + N(\kappa+1)^N$, where $N = |S|^d$. For the six-dimensional

S_n construction, $|S_n| = 2^{\Theta(n)}$ and $\kappa = 2^{\mathcal{O}(n)}$. That measure is therefore $2^{2^{\Theta(n)}}$, yielding a doubly exponential general upper bound in the parametric size used here. A singly exponential general $\mathrm{GNN}[F] \to \mathrm{GMSC}$ translation under this encoding is an open problem.

C.2 Approximate Extraction

Exact equivalence is the strongest extraction objective. We next retain the same teacher but allow errors under a distribution over degrees. Because a threshold surrogate is constant between consecutive breakpoints, its best possible fidelity is determined by how the alternating labels are partitioned into intervals.

Theorem C.4 (Approximate bounded parity)**.** *Put $K = 2^n$, fix either star convention, and let $\mathcal{D}_n$ be uniform on $\mathrm{Star}_0, \dots, \mathrm{Star}_{K-1}$. If a* GMSC *program with r graded-modality occurrences has fidelity at least $1 - \varepsilon$ with E_n under $\mathcal{D}_n$, then*

$$r \geq (1 - 2\varepsilon)K - 1. \tag{21}$$

For GGMSC *the corresponding bound is $r \geq ((1 - 2\varepsilon)K - 1)/2$. In particular, for every fixed $\varepsilon < 1/2$, approximate extraction still requires $2^{\Omega(n)}$ modalities.*

More precisely, among all binary classifiers on $\{0, \dots, K-1\}$ that are constant on at most $r+1$ consecutive intervals, the maximum number of agreements with parity is

$$\frac{K}{2} + \left\lfloor \frac{\min(r+1, K)}{2} \right\rfloor. \tag{22}$$

Proof. The factorization in Lemma C.1 partitions the degree line into at most $s = r + 1$ intervals on each of which a GMSC verdict is constant. If their nonzero lengths are $\ell_1, \dots, \ell_s$, a constant label agrees with alternating parity at most $\lceil \ell_i/2 \rceil$ times on interval i. Therefore

$$\#\text{agreements} \leq \sum_i \left(\frac{\ell_i}{2} + \frac{1}{2} \right) = \frac{K}{2} + \frac{s}{2} \leq \frac{K}{2} + \frac{r+1}{2}. \tag{23}$$

Dividing by K and rearranging yields the stated GMSC bound. A GGMSC program has at most $2r$ breakpoints and hence at most $2r + 1$ intervals, which gives the second bound.

For the exact optimum, cap s at K. Since K is even, the number of odd parts among $\ell_1, \dots, \ell_s$ is even. Consequently $\sum_i \lceil \ell_i/2 \rceil$ is at most $K/2 + \lfloor s/2 \rfloor$. This is attained when s is even by choosing all s lengths odd, and when s is odd by choosing $s - 1$ odd lengths and one even length. Singleton initial intervals give explicit such partitions. Label each interval by its majority parity. This proves Eq. (22). The main theorem's simpler bound is therefore tight up to at most one half-agreement before integer rounding. □

Computational case study. The code independently solves the more general weighted problem. For nonnegative integer degree weights w_c, let $W_b(i, j)$, for $b \in \{0, 1\}$, be the total weight of degrees of parity b in the half-open interval $[i, j)$. The dynamic program

$$D(s, j) = \max_{s-1 \leq i < j} \big(D(s-1, i) + \max\{W_0(i, j), W_1(i, j)\} \big), \tag{24}$$

for $1 \leq s \leq j \leq K$, with $D(0, 0) = 0$ and all other states with $s = 0$ or $j = 0$ set to $-\infty$. Here $D(s, j)$ is the maximum agreement weight on the prefix $[0, j)$ using exactly s nonempty intervals. Prefix sums make each candidate evaluation constant time. Retaining backpointers reconstructs the actual threshold list and interval labels. Computing the complete frontier uses $\mathcal{O}(K^3)$ time and $\mathcal{O}(K^2)$ memory in the simple reference implementation. All scores and target comparisons are integer exact.

Table 2 reports the minimum number of breakpoints needed for 0.9 fidelity. The uniform column matches Eq. (22). The triangular distribution uses $w_c = K - c$, favoring lower degrees. Both grow quickly, while the smaller triangular constants quantify the effect of concentrating probability on lower degrees.

Table 2: Minimum number of breakpoints needed to reach 0.9 fidelity for bounded parity under uniform and triangular degree weights.

n	$K = 2^n$	Uniform	Triangular $w_c = K - c$
4	16	13	9
5	32	25	18
6	64	51	36
7	128	103	71
8	256	205	142

Corollary C.5 (Bounded-depth separation)**.** *Depth-$\mathcal{O}(n)$ AC-$\mathrm{GNN}[F]$ models of polynomial parametric size express* GML*-definable classifiers whose smallest* GML *formulas have size $2^{\Omega(n)}$. This bounded-depth succinctness is breakpoint-driven. E_n has witnesses of size $\mathcal{O}(1)$, while its alternating degree pattern forces exponentially many logical thresholds.*

Theorem C.2 also resolves the dynamics-versus-width question for the *logic*. The separation already holds over the empty alphabet. For automata, the transition function changes the picture.

Remark C.6. A 2-state FCMPA with counting bound $\kappa = 2^n$ computes E_n. At each round, its transition sums the multiplicities in the capped multiset. If $c < \kappa$, this sum recovers c exactly. If $c \geq \kappa$, the sum remains at least κ. The transition can therefore decide whether c is even and below κ. The constant-alphabet automata question (Open Problem D.7) asks how broadly this multiset-processing power can substitute for states.

C.3 Weighted Counting at Depth One

The layered network in Theorem C.2 uses depth $\Theta(n)$ and encoded size $m = \mathcal{O}(n^2)$, giving a $2^{\Omega(n)} = 2^{\Omega(\sqrt{m})}$ target lower bound. The recurrent network instead reuses one transition for up to 2^n rounds. We now show that constant depth supports a second exponential mechanism. On one-type stars an R-simple depth-L GNN's verdict is piecewise-linear in c with few pieces, whereas two leaf types introduce a *slope* separation. Graded modal logic cuts the count plane along $\{0, 1\}$-coefficient functionals, while one weighted aggregation cuts along arbitrary-coefficient ones.

For $c_p, c_q \in \mathbb{N}$, let $\mathrm{Star}^{\rightarrow}_{c_p,c_q}$ be the pointed *two-type star* with an unlabeled root, c_p leaves labeled $\{p\}$, c_q leaves labeled $\{q\}$, and edges directed from the root. Let $\mathrm{Star}^{\leftrightarrow}_{c_p,c_q}$ be its undirected counterpart, and write Star_{c_p,c_q} when a statement holds for either family. The two leaf types are determined by their labels.

The nonrecursive case of the next result is due to Galliani et al. (2023, Theorem 11 and Corollary 12), whose counting perceptron $2c_1 + c_2 \geq k$ is the same family up to swapping leaf types. We include the slope proof to isolate the additional statement needed here. The lower bound persists for arbitrary recursive GMSC programs.

Figure 3 illustrates the geometric obstruction. The GNN uses the target normal vector $(1, 2)$, whereas every logical boundary has one of three allowed directions. The marked adjacent pair forces one of those logical boundaries to account for a change in the target verdict.

Lemma C.7 (Slope lemma)**.** *Let $\Pi = \{p, q\}$. On either two-type star family, the verdict of any* GML *formula φ is a Boolean function of atoms of the forms $[c_p \geq a]$, $[c_q \geq a]$, $[c_p + c_q \geq a]$, with at most one atom per diamond subformula on directed stars and at most one atom per diamond and per slope on undirected stars. For a* GMSC *program the same three-atoms-per-diamond bound holds on either family.*

Proof. On directed stars, the two leaf classes evolve uniformly and independently of c_p, c_q. On undirected stars, we instead track the root and both uniform leaf classes simultaneously. A diamond at a leaf sees only the root. It is false when $a > 1$ and, when $a = 1$, has the inductively determined truth of its argument at the root, so it introduces no count threshold. At the root, the argument holds at neither leaf class, only the p class, only the q class, or both. The diamond is therefore constant or one of the three threshold atoms with index a. Induction on formulas, and on rule bodies and rounds for GMSC, proves the factorization. On directed stars, a GML occurrence contributes one atom,

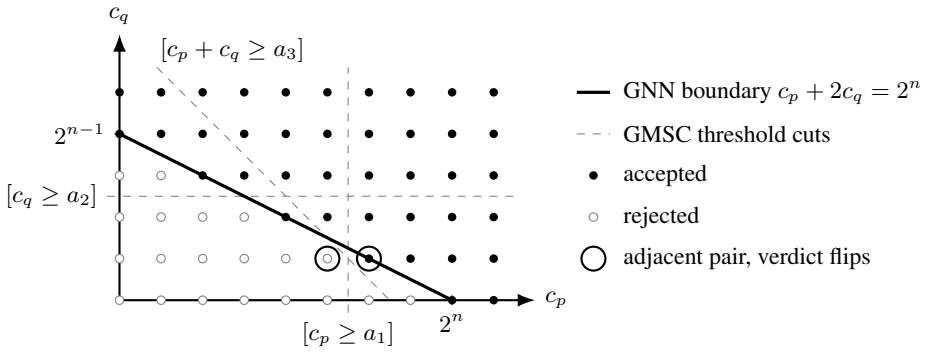


Figure 3: Geometry of the depth-one weighted-counting separation for $n = 3$. Filled points satisfy $c_p + 2c_q \geq 2^n$, whose oblique boundary is realized by one weighted GNN aggregation. On two-type stars, GMSC thresholds induce only vertical, horizontal, or slope--1 cuts. The circled adjacent points have opposite target verdicts, but no horizontal threshold separates them; thus, a vertical or slope--1 threshold must account for the change. Counting such boundary-crossing pairs gives the lower bound of Theorem C.8.

because leaf truth is independent of the counts. On undirected stars, it can contribute one atom per slope, as can a GMSC occurrence across all rounds. □

Theorem C.8 (Weighted thresholds at depth one)**.** *For a node v, let $c_p(v)$ and $c_q(v)$ count its out-neighbors satisfying p and q, respectively, and let $W_n(v)$ hold when $c_p(v) + 2c_q(v) \geq 2^n$. A neighbor satisfying both propositions contributes to both counts. Then:*

1. *W_n is* GML*-definable, and is expressed by a* depth-one *R-simple AC-*GNN$[F]$ *with dimension $\mathcal{O}(1)$ and $\mathcal{O}(n)$ parametric size.*
2. *every* GML *formula expressing W_n contains at least 2^{n-1} graded modalities, and every* GMSC *program contains at least 2^{n-2} graded modalities. With binary-coded gradings, either has size $\Omega(n2^n)$.*

When equivalence is required only over finite undirected graphs, both GML and GMSC require at least 2^{n-2} graded modalities and $\Omega(n2^n)$ encoded size.

Proof. The proof separates the geometric lower bound from the finite-precision upper bound. For the lower bound, we follow adjacent lattice points across the target boundary and count how many crossings one logical atom can explain.

(2) For $v \in \{0, \ldots, 2^{n-1}-1\}$ put $c_p^* := 2^n - 2v$ and consider the adjacent stars $\mathrm{Star}_{c_p^*-1,\,v} \notin W_n$ and $\mathrm{Star}_{c_p^*,\,v} \in W_n$. Crossing this edge increments c_p only. Among the atoms of Lemma C.7, exactly those of the forms $[c_p \geq c_p^*]$ and $[c_p + c_q \geq c_p^* + v]$ change value. Since the verdict changes, an atom with threshold $2^n - 2v$ (first slope) or $2^n - v$ (third slope) must occur. Both maps $v \mapsto 2^n - 2v$ and $v \mapsto 2^n - v$ are injective. Thus, each atom serves at most one v per slope. The 2^{n-1} values of v force at least 2^{n-1} atoms. On directed stars, this gives 2^{n-1} diamonds for GML. A GMSC occurrence can supply both relevant slopes, as can a GML occurrence on undirected stars, so those cases require at least 2^{n-2} diamonds. For $0 \leq v < 2^{n-2}$, both possible witnessing indices are at least 2^{n-1}. One modal occurrence can witness at most two such crossings, so $\Omega(2^n)$ occurrences have $\Omega(n)$-bit indices. This gives the asserted $\Omega(n2^n)$ encoded-size bound.

For the upper bound, one weighted aggregate realizes the target normal vector directly.

(1) Put $K = 2^n$. The following staircase formula defines W_n and has $\mathcal{O}(nK)$ binary-encoded size:

$$\Diamond_{\geq K} p \;\vee\; \bigvee_{j=1}^{K/2-1} \left(\Diamond_{\geq j} q \wedge \Diamond_{\geq K-2j} p\right) \;\vee\; \Diamond_{\geq K/2} q. \tag{25}$$

Together with item (2), this also gives minimum GML and GMSC size $\Theta(n2^n)$. For the GNN, use the binary float system $S'_n = ((n+10, n+10, 2), +, \cdot)$. This choice specifies both the mantissa precision and the exponent range. The aggregation yields $\hat{y}_p = \min(c_p, B_n)$ and $\hat{y}_q = \min(c_q, B_n)$ exactly, where the saturation value satisfies $B_n \geq 2^{n+5}$. Saturated instances satisfy W_n anyway. The single combine layer computes $u = \hat{y}_p \cdot 2^{-(n+6)} + \hat{y}_q \cdot 2^{-(n+5)} + \left(\frac{1}{2} - 2^n\right)2^{-(n+6)}$. Put $T = c_p + 2c_q$. If no aggregate has saturated and $T \leq 2^{n+8}$, then

$$2^{n+7}u = 2T - 2^{n+1} + 1, \qquad |2T - 2^{n+1} + 1| < 2^{n+9}. \tag{26}$$

Under the evaluation convention of Section A, the p term is added to the q term before the bias. Each operand and partial sum lies on the $2^{-(n+7)}$ grid. When $T \leq 2^{n+8}$, every intermediate numerator fits within the $n+10$-bit mantissa. Therefore, the complete calculation is exact. If instead $T > 2^{n+8}$ without saturation, the unrounded value satisfies $u > 4 - 2^{-6} > 0$, and monotone round-to-nearest or positive overflow cannot change its sign. Finally, if either aggregate has saturated, its scaled positive contribution is at least $2^{n+5}2^{-(n+6)} = 1/2$, while the negative bias has magnitude below 2^{-6}. This case is positive as well. Thus, the linear-threshold acceptance fires iff $\mathrm{ReLU}^*(u) > 0$, i.e. iff $c_p + 2c_q - 2^n + \frac{1}{2} > 0$, i.e. iff $c_p + 2c_q \geq 2^n$ (the half-integer bias breaks the tie on integers). The construction therefore has depth one, constant dimension, and $\mathcal{O}(n)$-bit weights. □

Corollary C.9 (Exponential succinctness at depth one)**.** *Depth-one R-simple AC-*$\mathrm{GNN}[F]$ *models are exponentially more succinct than* GML *and* GMSC*. In particular, no polynomial translation into* GML *exists even at constant depth. The minimum accepted-graph size for W_n is $\lceil 2^n/3 \rceil$ when self-loops are allowed. Consequently, the depth-one witness bound of Lemma G.2 is tight up to constants in the exponent.*

Proof. The succinctness claim follows from Theorem C.8. Each out-neighbor contributes at most 3 to $c_p + 2c_q$, so every accepted graph has at least $\lceil 2^n/3 \rceil$ nodes. This bound is attained by labeling every node $\{p, q\}$ and making the pointed node reach all of them, including itself through a self-loop. If self-loops are disallowed, adding a separate unlabeled pointed node gives a witness with one more node. □

Remark C.10 (Arithmetic counting logics)**.** The classifier W_n is a *single atom* of the logic $\mathrm{K}^{\#}$ of Nunn et al. (2024), which permits counting modalities inside one linear inequality. Its representation has size $\mathcal{O}(n)$. Theorem C.8(2) therefore recovers its known exponential advantage over graded modal logic (Galliani et al., 2023) and extends the comparison to recursive GMSC. The verification literature raises broader succinctness questions for these GNN-motivated counting logics as well (Schwarzentruber, 2025).

We next turn to the automata comparison and then to the verification consequences of unbounded recurrence.

D A State-Complexity Observation for Message-Passing Automata

The logical separations above arise, because threshold modalities cannot compress arithmetic relationships among neighbor counts. CMPAs have a more powerful transition interface. Consequently, bounded parity itself gives no state lower bound. Instead, their bottleneck occurs at initialization: once two label sets enter the same state, no later message passing can separate them. The family below turns that observation into a tight exponential comparison.

For $n \geq 1$, write $[n] = \{1, \ldots, n\}$, let $\Pi_n = \{s_1, \ldots, s_n, p_1, \ldots, p_n\}$, and define

$$\varphi_n \;=\; \bigvee_{i=1}^{n} \left(\Diamond s_i \wedge \Diamond p_i\right), \tag{27}$$

a GML formula of modal depth 1 and size $\Theta(n)$. Let P_n be the node property it defines.

Lemma D.1 (Upper bounds)**.** *P_n is expressed by a* GMSC *program of size $\mathcal{O}(n)$ and by a recurrent* $\mathrm{GNN}[F]$ *of dimension $\mathcal{O}(n)$ and parametric size* $\mathrm{poly}(n)$.

Proof. The logical representation is immediate, while the neural representation stores each proposition bit and combines the corresponding neighbor existence tests over three rounds. For GMSC, use the program $X(0) :- \varphi_n,\ X :- X$, with $\mathcal{A} = \{X\}$. For the GNN$[F]$, use coordinates $(s_i, p_i, u_i, v_i, c_i)_{i\in[n]}$ and a, hence $d = 5n + 1$. The initialization circuit sends each proposition bit directly to its s_i or p_i coordinate and initializes u_i, v_i, c_i, a to zero. Preserve s_i, p_i, and, writing y for the aggregated neighbor state, use the single shared R-simple transition

$$\begin{aligned} u_i' &= \mathrm{ReLU}^*(u_i + y_{s_i}), & v_i' &= \mathrm{ReLU}^*(v_i + y_{p_i}), \\ c_i' &= \mathrm{ReLU}^*(c_i + u_i + v_i - 1), & a' &= \mathrm{ReLU}^*(a + \textstyle\sum_{i\in[n]} c_i). \end{aligned} \tag{28}$$

All coordinates are Boolean and u_i, v_i, c_i, a are sticky. After round one, u_i, v_i record whether an out-neighbor carries s_i, p_i. After round two, c_i records their conjunction. From round three, $a = 1$ exactly for P_n; use the affine readout threshold $a > 1/2$. The initialization has $2n$ direct wires, all weights have constant precision, and the dense matrix encoding costs $\mathcal{O}(n^2)$. □

Lemma D.2 (Initialization bottleneck)**.** *Let $A = (Q, \pi, \delta, F)$ be a* CMPA *over Π and let G, G' be Π-labeled graphs on the same (V, E), with labelings agreeing except at a single node u, where $\pi(\lambda(u)) = \pi(\lambda'(u))$. The runs of A on G and G' coincide, and A accepts (G, w) iff it accepts (G', w), for every $w \in V$.*

Proof. We proceed by induction on t. At $t = 0$, $g_0(v) = \pi(\lambda(v)) = \pi(\lambda'(v)) = g_0'(v)$ for all v (at $v = u$ by hypothesis, else labels agree). At $t + 1$, g_{t+1} is determined by g_t and (V, E) alone, since δ does not read labels. Acceptance depends only on the run. □

Theorem D.3 (Initialization state bound)**.** *Every* CMPA *expressing P_n, in particular every* FCMPA *(bounded or not), has at least 2^n states. The same lower bound holds when equivalence is required only over finite undirected graphs.*

Proof. The proof assigns a distinct label set to every subset of $[n]$. If two such sets share an initial state, a three-node star exposes a proposition on which their required verdicts differ. Formally, let $A = (Q, \pi, \delta, F)$ express P_n. For $S \subseteq [n]$ put $L_S := \{p_i : i \in S\}$, and for $i \in [n]$ consider the pointed star $G_{i,S}$ with nodes $\{v, x, y\}$, edges $(v, x), (v, y)$, labels $\lambda(v) = \emptyset$, $\lambda(x) = \{s_i\}$, $\lambda(y) = L_S$, pointed at v. Since x is the unique out-neighbor satisfying any $s_{i'}$ (namely $i' = i$) and y the unique one satisfying any $p_{i'}$ (namely $i' \in S$), we have $(G_{i,S}, v) \in P_n \iff i \in S$.

Suppose $|Q| < 2^n$. The map π is not injective on $\{L_S : S \subseteq [n]\}$. Thus, there are $S \neq S'$ with $\pi(L_S) = \pi(L_{S'})$. Pick $i \in S \triangle S'$. The graphs $G_{i,S}$ and $G_{i,S'}$ differ only in the label of y, and π identifies the two labels. By Lemma D.2, A gives the same verdict on $(G_{i,S}, v)$ and $(G_{i,S'}, v)$. But exactly one of them is in P_n, a contradiction.

For the undirected restriction, replace the two edges of each $G_{i,S}$ by their symmetric closures. The paired instances still have the same edge relation and initialization, so Lemma D.2 again makes their runs identical. The depth-one property P_n still distinguishes the two roots. □

Corollary D.4 (Explicit-state comparison)**.** *In the state-count measure,* GMSC *(already modal-depth-one* GML*) and recurrent* GNN$[F]$ *models have exponentially more compact symbolic representations than bounded FCMPAs. The family $\{P_n\}$ has* GMSC *(resp.* GNN$[F]$*) representations of size $\mathcal{O}(n)$ (resp. $\mathrm{poly}(n)$) while every FCMPA for P_n has at least 2^n states. The gap is tight. Every* GMSC *program of size m translates to a bounded* FCMPA *with $2^{\mathcal{O}(m)}$ states (Ahvonen et al., 2024). Thus, symbolic GMSC descriptions can encode an exponentially larger explicit state space.*

The collision proof targets the only irreversible step in the model. In the following remarks, we delimit that argument.

Remark D.5 (Label-re-reading automata)**.** In the original label-indexed presentation of Kuusisto & Reiter (2020), nodes re-read their labels every round. This differs from the label-storing, initialization-only presentation used in Lemma E.1; converting between them can enlarge the state space by a factor depending on the alphabet. In the original presentation, trajectory counting gives an $\Omega(n/\log n)$ bound. A blind node's state sequence is eventually periodic with pre-period plus period at most s. Each label induces a unary update $f_a \colon Q \to Q$. Its trajectory is determined by the pair consisting of its initial state and f_a. There are at most s^{s+1} state–map pairs (and at most s^s in the original presentation, where the initial state is common to all labels). The proof of Theorem D.3

then needs the 2^n data-labels to induce pairwise distinct trajectories (orienting the star's edges so the leaves remain blind in that model), giving $s = \Omega(n/\log n)$. Closing the gap between $\Omega(n/\log n)$ and 2^n in that model is still an open problem.

Remark D.6 (Reach of the lower-bound methods)**.** Bergsträßer et al. (2026) obtain a *doubly* exponential gap over DFAs from witness lengths. Proposition E.3 separates the graph setting from that approach, while the initialization method scales with the $2^{|\Pi|}$ available labels. Together they yield the exponential gap above. Larger separations call for a different graph family or lower-bound invariant.

Open Problem D.7 (Constant alphabet, vs. automata)**.** *Over a fixed finite proposition set, are recurrent* $\mathrm{GNN}[F]$ *models—or, equivalently,* GMSC *programs—super-polynomially more succinct than bounded FCMPAs in the state-count measure? Our lower bound uses label width. Over a constant alphabet, conveyor replay serializes label information over time, while counting formulas require correlations among neighboring labels. The question appears related to open state-complexity problems for one-way cellular automata, which is what FCMPAs on directed paths amount to. (Against the* logic*, the constant-alphabet question is answered affirmatively in Section C, but the mechanisms there, bounded parity and weighted thresholds, are free for an automaton's transition function. See Remark C.6.)*

E Structural Features of Graph Message Passing

The preceding state lower bound used an initialization collision rather than a large minimum witness. This section explains why the more familiar witness-size route cannot replace that argument for recurrent graph models. For strings, a useful lower-bound principle is that a deterministic finite automaton with s states and a nonempty language accepts some word of length less than s (Bergsträßer et al., 2026). A large minimum witness therefore forces many states. Graph message passing breaks this connection more sharply. Recurrent models admit nonempty languages whose minimum witness size has no computable bound in the representation size.

Lemma E.1. *For every distributed automaton A in the model of Kuusisto & Reiter (2020), one can compute a* 1*-bounded* FCMPA A' *(over a proposition set encoding A's alphabet) whose language is non-empty iff that of A is.*

Proof. The original presentation of Kuusisto & Reiter (2020) uses a label-indexed transition in every round. For non-forgetful automata, they also give an equivalent label-storing form in which the label is read once and retained in the state. Apply that effective conversion first and write $A = (Q, \delta_0, \delta, F)$, where each node is initialized by $\delta_0 \colon \Sigma \to Q$ from its label, and each round updates by $\delta \colon Q \times 2^Q \to Q$ from the node's state and the *set* of states of its incoming neighbors. It accepts a pointed digraph if the distinguished node ever visits F. Encode Σ injectively into $\mathcal{P}(\Pi)$ for $|\Pi| = \lceil \log_2 |\Sigma| \rceil$ and set $\pi := \delta_0 \circ \mathrm{decode}$, where decode maps non-code labels to a fixed default letter $a_0 \in \Sigma$. Every Π-labeled graph then decodes to a Σ-labeled one, and the correspondence is onto. Set $\delta'(q, M) := \delta(q, \mathrm{supp}(M))$. Since $\mathrm{supp}(M) = \mathrm{supp}(M_{|1})$, A' is 1-bounded. Finally, CMPAs read *out*-neighbors, whereas the automata of Kuusisto & Reiter (2020) read *in*-neighbors. For every pointed digraph (G, w), the run of A' on (G, w) is therefore the run of A on (G^{rev}, w) after decoding labels, where $G^{\mathrm{rev}} = (V, E^{-1}, \lambda)$. Edge reversal is an involution on finite digraphs, so $L(A')$ is non-empty exactly when $L(A)$ is. □

Theorem E.2. *The non-emptiness problem for* 1*-bounded FCMPAs is undecidable.*

Proof. Emptiness for the distributed automata of Kuusisto & Reiter (2020) is undecidable by their main theorem, already for automata on directed paths. Compose with Lemma E.1. □

Proposition E.3 (No computable witness bound)**.** *Fix any effective binary encodings of bounded FCMPAs (including $\Pi, Q, \kappa, \pi, \delta, F$),* GMSC *programs, and recurrent* $\mathrm{GNN}[F]$ *models. For each of the three representation classes, there is no computable $f \colon \mathbb{N} \to \mathbb{N}$ such that every non-empty encoded object of description length m accepts a pointed graph with at most $f(m)$ nodes.*

Proof. Suppose such a computable f exists. Given a bounded FCMPA A of description length m, with s states over finite Π, enumerate the finitely many pointed Π-labeled graphs with at most

$f(m)$ nodes up to isomorphism. Observe that, on a fixed graph with vertex set V, the run $(g_t)_{t\in\mathbb{N}}$ is a deterministic sequence in the finite space Q^V, hence eventually periodic with pre-period plus period at most $s^{|V|}$. Therefore, A accepts (G, w) iff it does so within the first $s^{|V|}$ rounds, which is decidable by simulation. This decides non-emptiness, contradicting Theorem E.2. If such a bound existed for GMSC or GNN$[F]$, first translate the input FCMPA effectively into that formalism using Ahvonen et al. (2024). Its encoded size is computable from the input, and the same finite search would again decide non-emptiness. □

Thus, minimum witness size is unbounded by every computable function of the description size for both FCMPAs and GMSC programs. This confirms why the state lower bound must use initialization rather than witness size. Two other features reinforce the distinction. Acceptance is existential over rounds. An automaton may therefore trade time for communication across graph cuts. Moreover, information stored in node labels can be rebroadcast indefinitely.

Conveyor replay (construction sketch). For every finite Σ and $n \in \mathbb{N}$, there is an FCMPA with $\mathcal{O}(|\Sigma|^3 + |\Sigma|\, n)$ states such that on every Σ-labeled directed path $v_1 \to v_2 \to \cdots \to v_\ell$ with $\ell \le n$, the emission coordinates of v_1 over any full period from round ℓ onward spell $\lambda(v_1)\cdots\lambda(v_\ell)$ followed by blanks, cyclically with period $n{+}1$. Every node with label σ starts in the relay state $(\sigma, \mathrm{blank}, \mathrm{blank})$. At round one, a node whose out-neighbor multiset is empty enters the blind-end state $(\sigma, 0)$; otherwise it remains a relay. Transitions on multisets outside the directed path case may be fixed arbitrarily. The end node v_ℓ detects that it is blind (its out-neighbor multiset is empty), runs a mod-$(n{+}1)$ clock, and emits its label at clock 0 and a blank otherwise. Every other node performs a delay-one relay of the incoming stream using one buffered symbol. The only exception occurs when the previously read symbol was blank and the newly read one is not. It then emits its own label in place of the buffered blank. The insertion consumes one blank slot, while v_ℓ's cycle provides $n \ge \ell - 1$ such slots. Blanks therefore never run out, and the period $n{+}1$ is preserved. By induction on $j = \ell, \ldots, 1$, node v_j emits $\lambda(v_j)\cdots\lambda(v_\ell)$ followed by $n{+}1 - (\ell{-}j{+}1)$ blanks, cyclically with period $n{+}1$, starting in round $\ell - j + 1$. The relay-node state carries the node's own label (over Σ), the current emission, and a one-symbol buffer. Each ranges over $\Sigma \cup \{\mathrm{blank}\}$. The buffer always holds the last-read symbol. Hence, no separate last-read coordinate is needed. The blind end carries its label jointly with the mod-$(n{+}1)$ clock, using exactly $|\Sigma|(n + 1)$ such states. The construction identifies the initialization map as the durable source of state complexity.

Consequently, a polynomial-state automaton can replay labels placed far from the point of decision and process them over time. By contrast, label sets identified by the initialization function have identical state trajectories. The lower bound exploits precisely this irreversible compression. The same unbounded temporal behavior also determines the verification frontier studied after the automata comparison.

F Undecidability of Verification

The previous section showed that recurrence defeats every computable bound on minimum witness size. We now translate the underlying automata undecidability into the GNN and logical models. This yields a sharp contrast with the bounded-depth regime considered in the next section.

Theorem F.1. *The non-emptiness (equivalently, satisfiability) problem for recurrent* GNN$[F]$ *models and for* GMSC *programs is undecidable.*

Proof. By Theorem E.2, non-emptiness of bounded FCMPAs is undecidable. The translations between bounded FCMPAs, GMSC programs, and recurrent GNN$[F]$ models of Ahvonen et al. (2024) are effective in both directions (their main characterization theorem, proved via the equivalence of GMSC with bounded FCMPAs). The problem therefore transfers. □

Undecidability for GMSC is known from the substitution-calculus literature and is summarized by Ahvonen et al. (2025). The theorem records the effective transfer to the exact recurrent float model. The decidability results of Benedikt et al. (2026) concern non-recurrent GNNs. These results are consistent with the recent uniform computation-completeness of recurrent GNNs up to color-refinement invariance (Rosenbluth & Grohe, 2026). Recurrence provides full algorithmic power.

It also makes $\exists$-graph questions undecidable. The following proposition gives the corresponding semidecidability classification.

Proposition F.2. *Non-emptiness of recurrent* $\mathrm{GNN}[F]$ *models,* GMSC *programs, and bounded FCMPAs is recursively enumerable.*

Proof. Enumerate pairs (pointed Π-labeled graph, round bound T) and simulate T rounds. Accept upon finding an accepting round. Runs are deterministic and effective. This procedure therefore semi-decides non-emptiness. □

The *halting* recurrent GNN model of Bollen et al. (2025) provides a complementary frontier. Every formula of the graded μ-calculus ($\mu\mathrm{GML}$) is expressible by a halting recurrent GNN, and satisfiability of the graded μ-calculus is decidable in EXPTIME (Kupferman et al., 2002). Halting recurrent GNNs are strictly more expressive.

Observation F.3. *Halting recurrent GNNs (over the reals, as in Bollen et al. (2025)) express node classifiers not definable in the graded* μ*-calculus.*

Proof. Consider "strictly more out-neighbors satisfy p than q". A halting recurrent GNN computes it in one round. It aggregates the sums of the p- and q-indicator coordinates as y_p, y_q, outputs $[y_p - y_q > 0]$ via a clipped-ReLU unit, and raises a halting flag at every node after the first transition. Real-valued GNNs express such majority-style classifiers (Barceló et al., 2020). The classifier lies outside the graded μ-calculus. Indeed, $\mu\mathrm{GML} \subseteq \mathrm{MSO}$, monadic second-order logic (fixpoints and fixed-threshold counting are MSO-definable), and majority is not MSO-definable already over star graphs. □

Thus, the halting model strictly extends this decidable logic. Its verification behavior therefore cannot be inferred from graded μ-calculus alone. The preservation theorem of Pflueger et al. (2024) (recurrent GNN classifiers *expressible in local monadic monotone fixpoint logic, LocMMFP,* are $\mu\mathrm{GML}$-expressible) applies to the LocMMFP-expressible fragment. A halting guarantee (Bollen et al., 2025) bounds the number of rounds to termination by a function of the *graph size*. On any *fixed* graph, one can therefore simulate to the halting configuration and read off the verdict, making membership decidable per graph. However, non-emptiness can also quantify over graph size. Halting controls the temporal axis (how long to run), while decidability additionally requires control of the spatial axis (how large a witness to search). A computable witness bound would provide that spatial control.

Open Problem F.4 (Halting frontier)**.** *Is non-emptiness decidable for halting recurrent GNNs (over the reals with rational weights, or over floats)? More generally, which restrictions on recurrence restore decidability? Runs of halting GNNs terminate on every input graph. Therefore, the problem is recursively enumerable with graph enumeration alone. A computable witness bound would upgrade this semidecision procedure to a decision procedure.*

G Decidability at Bounded Depth

Bounding the number of layers removes the unbounded temporal search used in the undecidability proof. The remaining question is spatial: how large must an accepted graph be? Unravelling converts any witness into a tree, and using finite-precision saturation, we then bound its branching.

A *depth-L aggregate-combine* $\mathrm{GNN}[F]$ *(AC-*$\mathrm{GNN}[F]$*)* is defined as in Section A but with L (possibly distinct) AC transitions applied once each, with acceptance read off the layer-L feature of the pointed node. Write $U = S^d$ for the feature space and $\kappa = \kappa_S$ for the saturation bound. The parametric size of a layered network sums the parameter counts of its layers. If float words have k bits, the explicit matrix encoding gives $Ld^2k = \mathcal{O}(m)$; in particular, $L, k \leq m$.

Lemma G.1 (Unravelling invariance)**.** *The layer-t feature of w in G equals the layer-t feature of the root of the depth-t unravelling of G at w (the tree of directed paths from w of length $\leq t$).*

Proof. We proceed by induction on t. The layer-t feature of w depends on w's label, its layer-$(t-1)$ feature, and the multiset of layer-$(t-1)$ features of its out-neighbors. In the unravelling, the children

of a path-node ending at w are its extensions by one edge, whose subtrees are the depth-$(t-1)$ unravellings at the corresponding out-neighbors. □

Lemma G.2 (Small witnesses at bounded depth)**.** *If a depth-L AC-$\mathrm{GNN}[F]$ N accepts some finite directed pointed graph, it accepts a directed tree with at most $(L\,\kappa\,|U|)^{L+1}$ nodes. In terms of the parametric size m of N, this is $2^{\mathcal{O}(m \log m)}$, and it is $2^{\mathcal{O}(m)}$ when L is fixed.*

Proof. Let N accept (G, w). By Lemma G.1, N accepts the depth-L unravelling T at its root. Prune T top-down. At a kept node u at depth j, the only data its children contribute to u's features at layers $1, \dots, L-j$ are, for each layer $t \le L-j-1$, the κ-capped multiset of layer-t features of its children. For each relevant pair (t, α), choose a set $K_{t,\alpha}(u)$ of $\min(\kappa, \text{count})$ children having feature $\alpha \in U$ at layer t, and retain the union $K(u) = \bigcup_{t,\alpha} K_{t,\alpha}(u)$. Delete the other children. If the original multiplicity of α at layer t is below κ, all such children remain; otherwise at least κ remain. Thus, the capped child-feature multiset is unchanged simultaneously at every relevant layer. If M_t and M'_t are the original and retained multisets at layer t, then $(M'_t)_{|\kappa} = (M_t)_{|\kappa}$ and hence $\mathrm{SUM}_S(M'_t) = \mathrm{SUM}_S(M_t)$ coordinatewise. The children's features at every layer are preserved, because their subtrees are pruned in the same way. Induction now shows that the root's layer-L feature is unchanged and that N still accepts. The pruned tree has branching at most $L\kappa|U|$ and depth L. Finally $|U| = |S|^d \le 2^{kd}$ for k-bit floats and $\kappa \le 2^{\mathcal{O}(k)}$. The encoding includes all L pairs of $d \times d$ transition matrices at word width k, so $Ld^2k = \mathcal{O}(m)$ and $d^2k = \mathcal{O}(m)$. Consequently, $(L+1)(\mathcal{O}(k) + kd) = \mathcal{O}(m)$, while $(L+1)\log_2 L = \mathcal{O}(m \log m)$. Therefore

$$\log_2(L\kappa|U|)^{L+1} \le (L+1)\big(\log_2 L + \mathcal{O}(k) + kd\big) = \mathcal{O}(m \log m). \tag{29}$$

For fixed L, the logarithmic term is constant and the bound is $2^{\mathcal{O}(m)}$. □

Corollary G.3. *Non-emptiness of depth-L AC-$\mathrm{GNN}[F]$ models over finite directed graphs is decidable (guess a directed tree of size $2^{\mathcal{O}(m \log m)}$ and evaluate). Decidability in this regime, and indeed the exact complexity, is known for closely related bounded-depth models. Satisfiability is* PSPACE*-complete for quantized GNNs (Sälzer et al., 2025) and for the logic $\mathrm{K}^{\#}$, which has polynomial two-way GNN translations (Nunn et al., 2024). The explicit float-model witness bound in Lemma G.2 connects these decidability results to the succinctness analysis in Proposition G.5.*

Remark G.4 (Witness-bound gap)**.** The depth-one family W_n of Theorem C.8, whose minimal witnesses have $2^{\Theta(n)} = 2^{\Theta(m)}$ nodes (Corollary C.9), gives a $2^{\Omega(m)}$ lower bound. At depth one, Lemma G.2 gives the matching $2^{\mathcal{O}(m)}$ upper bound, so the witness-size scale is tight up to constants in the exponent. For growing depth, the general upper bound is $2^{\mathcal{O}(m \log m)}$; its sharp dependence on m remains open.

Proposition G.5 (Limit of witness-size arguments)**.** *Every satisfiable* GML *formula ψ (binary-coded gradings) has a model with at most $2^{\mathcal{O}(\|\psi\|^2)}$ nodes. Consequently, for any non-empty node property recognized by a depth-L AC-$\mathrm{GNN}[F]$ of parametric size m, the witness-size method ("the smallest witness is large, and small formulas have small models") can produce from this general model bound a* GML *lower bound of at most $\mathcal{O}(\sqrt{m \log m})$, and at most $\mathcal{O}(\sqrt{m})$ when L is fixed.*

Proof. For the model bound, GML over arbitrary frames has the tree-model property with depth bounded by the modal depth $\le \|\psi\|$ and branching bounded by the sum of grading indices over diamond subformulas, $\le \|\psi\|\, 2^{\|\psi\|}$ (Tobies, 2001; Kazakov & Pratt-Hartmann, 2009), giving $\big(\|\psi\| 2^{\|\psi\|}\big)^{\|\psi\|} = 2^{\mathcal{O}(\|\psi\|^2)}$ nodes. Let $W_{\min}(N)$ be the minimum number of nodes in a graph accepted by N. If ψ defines the same non-empty property, the small-model bound gives

$$W_{\min}(N) \le 2^{\mathcal{O}(\|\psi\|^2)}, \qquad \|\psi\| = \Omega\Big(\sqrt{\log W_{\min}(N)}\Big). \tag{30}$$

Lemma G.2 gives $W_{\min}(N) \le 2^{\mathcal{O}(m \log m)}$, or $2^{\mathcal{O}(m)}$ at fixed depth. Hence the numerical lower bound obtainable from minimum witness size and this general small-model theorem is at most $\mathcal{O}(\sqrt{m \log m})$, or $\mathcal{O}(\sqrt{m})$ at fixed depth. □

At bounded depth, this general witness-size argument therefore yields at most $\mathcal{O}(\sqrt{m \log m})$ formula lower bounds, while the initialization argument of Theorem D.3 scales with label width. The central results proved earlier avoid both limitations. Their exponential logical lower bounds come from arithmetic access to multiplicities and the resulting breakpoint or slope structure.

Remark G.6 (Three mechanisms)**.** The paper identifies three independent resources. *Initialization width* determines which label distinctions remain available to an automaton (Theorem D.3). *Arithmetic access to multiplicities* supports exact clocks and bit extraction in the parity constructions and weighted sums already at depth one in Theorem C.8. *Recurrence* provides unbounded time and drives the verification frontier in Appendix F. Arithmetic aggregation is the graph-specific resource. Unique-hard-attention transformers in the string setting of Bergsträßer et al. (2026) select individual positions, whereas a GNN aggregates weighted occurrence counts in its first layer.

H Future Work

The results suggest several ways to sharpen the boundary between compact neural computation and symbolic description. The first is to identify the weakest useful extension of threshold counting. Linear counting inequalities, such as those in $\mathrm{K}^{\#}$, express the weighted family W_n compactly, whereas modular, divisibility, or Presburger-style predicates compactly express bounded parity. A precise hierarchy of target languages would show which arithmetic operations are necessary for compact extraction. It would also help resolve the gap between the family-specific exponential lower bound proved here and the doubly exponential upper bound obtained from the general look-up-table translation.

A second direction concerns the neural resource responsible for the parity construction. Its precision grows linearly with n, and that bit width is included in the encoded size. A fixed-precision separation could instead use growing dimension, a different graph family, or another arithmetic invariant. For bounded-depth networks, Lemma G.2 gives witnesses of size $2^{\mathcal{O}(m \log m)}$ in general and $2^{\mathcal{O}(m)}$ at fixed depth. The depth-one family W_n matches the latter scale. Determining the sharp bound when depth grows with the network size remains open.

The automata comparison leaves a complementary fixed-alphabet question. The state lower bound of Theorem D.3 uses $2n$ proposition symbols to expose collisions in the initialization map. Over a fixed alphabet, recurrent automata can serialize label information over time, as the conveyor construction illustrates. A super-polynomial state separation in that regime would require a lower-bound invariant that survives such temporal reuse.

Finally, the extraction results can be tested beyond the explicit hard families studied here. One empirical direction is to compare exact and approximate extraction from trained GNNs under several graph distributions, recording both fidelity and description size. Comparing threshold-only targets with languages that permit weighted or modular counting would reveal when the worst-case arithmetic bottleneck appears in learned models and when the data distribution makes a smaller description possible.