DID YOUR GNN FIND ANYTHING? EXACT CERTIFICATION OF DEEP GRAPH CLUSTERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Deep graph clustering pipelines always return a partition. On Erdős-Rényi graphs, whose wiring carries nothing beyond the degree sequence, our DMoN-lite and MAGI-lite reimplementations still return partitions of 16 communities at mean modularity 0.25 (DMoN-lite) and 16 at 0.21 (MAGI-lite). We give an exact, finite-sample certification layer that wraps a trained pipeline and returns a p-value against the null that the wiring carries no structure beyond the degrees. The null is uniform over simple graphs with the observed degree sequence, sampled by a double-edge-swap chain inside the Besag-Clifford parallel construction, and validity holds at any chain length. Carrying that guarantee to trained networks under stochastic and amortized training is what we establish. Training is stochastic, so we give seeding rules under which the score stays exchangeable and the p-value stays exact, and show that the wrapped rule must read the graph rather than the ordering of its edges. A certificate costs $M+1$ fits, so we give a hub-conditional warm start that provably preserves exactness, and we show that the natural alternative an implementer would write instead certifies structure in a fraction 1.00 of structureless graphs. Most tests do not reject, so a sequential rule stops the median null test after 19 of 199 fits. Applied recursively, the same test estimates model order, with false splits controlled at level α under an explicit selection-neutrality condition. In the calibration audit's 1800 tests of three deep pipelines at $\alpha = 0.05$, the wrapper's rejection rate stays between 0.005 and 0.065, the largest cell's Wilson interval $[0.038, 0.108]$ covering α, while a Bickel-Sarkar-style threshold applied outside its stated regime rejects a fraction 1.000 of 200 structureless Erdős-Rényi graphs, every rejection a type-I error under its own null. Significance and recovery come apart: pipelines earn certificates on graphs whose planted partitions they do not recover, so a certificate belongs to a (graph, pipeline, configuration) triple rather than to a graph.

1 INTRODUCTION

What a clustering score lacks is not accuracy but a reference point. Structureless graphs admit modular partitions of their own (Guimerà et al., 2004; McDiarmid & Skerman, 2020), so the scores DMoN-lite (Tsitsulin et al., 2023) and MAGI-lite (Liu et al., 2024) report on graphs wired uniformly given their degrees (Fig. 1) read as findings only because nothing supplies the comparison. The comparison a trained network needs is narrow: the same pipeline, at the same budget, on graphs sharing everything with the observed one except its wiring. This paper supplies it exactly.

Asymptotic tests and spectral estimators of block count (Bickel & Sarkar, 2016; Lei, 2016; Saade et al., 2014; Le & Levina, 2022) say nothing about what a trained network found. On the plain sparse Erdős-Rényi graphs of our calibration audit, the one family where the competitor's own null holds and every rejection is therefore a type-I error, a Bickel-Sarkar-style decision rejects a fraction 1.000 of 200 while the exact wrapper holds level, as Anonymous (2026) found for classical statistics. That rule is used outside its stated regime, at mean degree 10 and $n = 1000$; Section 4.2 adds the same-statistic and in-regime controls that locate the gap in the threshold, not the statistic.

Our null is H_0: conditionally on its degree sequence, the graph is uniform over the simple graphs carrying that sequence, node attributes held fixed (Section 3.1). A double-edge-swap chain (Taylor, 1981) inside the Besag-Clifford parallel construction (Besag & Clifford, 1989) makes the observed and M null graphs exchangeable at any chain length, so any fixed label-free rule, up to a whole

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.

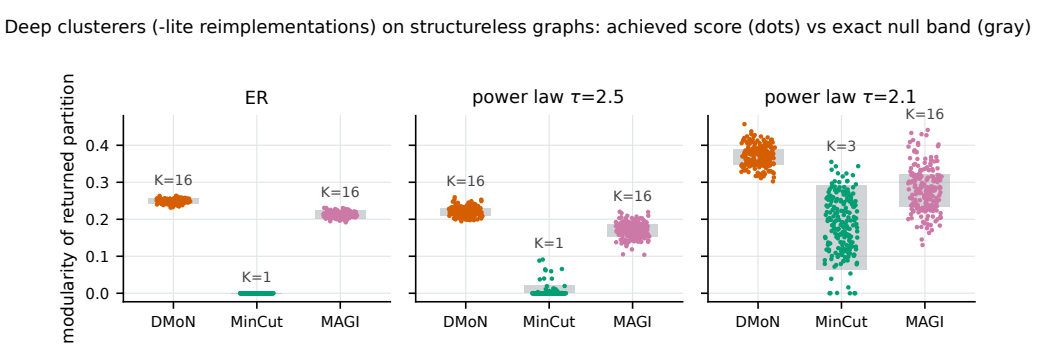

Figure 1: Deep graph clustering on pure noise (1800 logged tests). Dots are each wrapped pipeline's score on one structureless graph; the band averages the per-test fifth-to-ninety-fifth percentile interval of its null scores on rewirings of that graph, so dots above it are expected and are not rejections. See Appendices E and U.

trained pipeline, gives an exact rank p-value. The wrapper, including a trained network run at one fixed seed, is known (Anonymous, 2026); which seeding rules, warm starts and stopping rules keep it exact under stochastic training is ours.

- **C1: certification under stochastic training.** Lemma 2 gives two seeding rules that keep the score exchangeable, so non-convergence, collapse and seed variance bear on power, never on level. It also asks the wrapped rule to be invariant to the graph's *representation*, a condition that determinism at a fixed seed does not supply: we name an implementation pattern that violates it silently, contrastive positives drawn by edge slot index, and give the canonicalization and the one-permutation, one-refit check that detects it (Section 3.3, Appendix R).
- **C2: which warm starts stay exact.** Corollary 2 gives a warm start that stays exact, pretraining on the hub of the construction rather than on the observed graph, and the experiment rules out the sibling scheme an implementer would write instead, which certifies structure in a fraction 1.00 of structureless graphs, Wilson (Wilson, 1927) $[0.94, 1.00]$ (Section 4.4; per-mode costs in Table 4).
- **C3: sequential stopping.** Under the rule of Besag & Clifford (1991) the budget adapts to the answer: the median structureless test stops after 19 of 199 fits, while tests at a clearly planted contrast ($c = 5$) run to the end (Appendix M).
- **C4: model order, false splits measured.** Carrying over the recursion and level allocation of the classical line (Anonymous, 2026; Meinshausen, 2008; Kimes et al., 2017), we measure false splits on structureless graphs, where the first split is the root's and needs no extra assumption: 0.022 over 600 graphs, Wilson $[0.013, 0.037]$, against a nominal 0.05 (Appendix O). A fresh-seed replication of the accuracy grid of Anonymous (2026), with a constant-answer yardstick added, gives the best pooled mean absolute error point estimate, 1.59 against 1.72 for the Bethe-Hessian count, one of only two of the five that beat a constant; every model-order result wraps classical Leiden, and Appendix P reports where the recursion fails and why.

Two findings cut across. Significance and recovery come apart, further for the deep pipelines than for Leiden, though most of that gap is a cluster-budget artifact and the separation survives at every budget (Section 4.3 and Appendix Q). And a certificate attaches to the pipeline with its configuration rather than to the graph: on Karate one pipeline retains (does not reject H_0) at the cluster budget we fixed in advance and certifies at a smaller one, a reported misconfiguration, robust to correcting for the budgets tried (Sections 4.5 and 5).

We contribute no objective, architecture or training procedure and no ranking of wrapped pipelines, which are lite reimplementations; conditioning on degrees leaves structure expressible through degrees alone untested. Proofs, tables, per-cell readouts, further caveats and limitations are in the appendix (Appendices I and U).

2 RELATED WORK

Testing for structure in networks. The classical line asks whether a graph fits a one-block or fitted K-block model: Tracy-Widom limits for the largest eigenvalue of the centered scaled adjacency matrix (Bickel & Sarkar, 2016), spectral goodness of fit (Lei, 2016), goodness of fit for the degree-corrected mixed-membership family (Jin et al., 2025), exact conditional sampling from a fitted model's fiber, the graphs sharing its sufficient statistics (Karwa et al., 2024), and Bethe-Hessian estimators of model order (Saade et al., 2014; Le & Levina, 2022) read against detectability (Decelle et al., 2011; Abbe, 2018) and the degree-corrected block model (Karrer & Newman, 2011). Throughout, the object under test is a parametric model with an analyst-chosen statistic, never the partition a trained pipeline returned. Algorithm-agnostic significance tests for communities compare a detector's quality against randomized networks or bootstrap thresholds (Kojaku & Masuda, 2018; Yanchenko & Sengupta, 2024), and exact p-values exist for a single community under an Erdős-Rényi null (He et al., 2020); none is exact under the degree-conditioned null for a stochastic trained pipeline. Whether these asymptotics hold at practical sparsity is empirical (Section 4.2).

Selective inference and conformal prediction. Two neighboring literatures give finite-sample guarantees. Conditioning on the selection event, or splitting the data instead (Neufeld et al., 2024), keeps testing after clustering valid, for hierarchical clustering (Gao et al., 2024), k-means (Chen & Witten, 2023), whole pipelines (Miyata et al., 2026) and a graph-side result (Nishino et al., 2025), with significance analysis for hierarchical and single-cell clustering as the applied analogue (Kimes et al., 2017; Grabski et al., 2023; Hivert et al., 2024) and level allocation following Meinshausen (2008); it needs a tractable selection event, usually piecewise linear in a Gaussian feature space, which a gradient descent trajectory does not admit, and it tests differences between selected clusters, not whether structure exists. Conformal prediction gives finite-sample coverage under exchangeability, for node classification (Huang et al., 2023) and cluster assignments (Nath et al., 2026), within a wider line on exchangeability-based and resampling inference (Barber & Ramdas, 2026; Dombowsky et al., 2026; Domingo-Enrich et al., 2025); it presupposes a supervised target and exchangeable calibration units, which one unlabeled graph lacks.

Deep graph clustering and its evaluation. Modern pipelines optimize a relaxed clustering objective end to end: modularity (Tsitsulin et al., 2023), spectral cuts (Bianchi et al., 2020), differentiable pooling (Ying et al., 2018) and contrastive or dilation pretexts (Devvrit et al., 2022; Liu et al., 2023; 2024), surveyed by Liu et al. (2026). Their evaluation draws sustained criticism: large run-to-run variation, fragility under perturbation and ranking reversals under honest protocols (Wu et al., 2025; Leeney & McConville, 2024), echoing earlier findings for graph neural networks (Shchur et al., 2018; Platonov et al., 2023). That work establishes instability but supplies no inference: methods are ranked against reference labels, attaching no guarantee to one partition on one unlabeled graph.

Positioning. Wrapping a black-box clusterer in this exact degree-conditioned test is not new: a concurrent submission (Anonymous, 2026) develops it for pipelines applied by one fixed rule, chiefly Louvain (Blondel et al., 2008) and Leiden (Traag et al., 2019), notes that a trained network run at a fixed seed is admissible, and illustrates this with a small DMoN-style network; case (a) of Lemma 2 recovers that argument, and we claim no priority for it. It does not analyze what makes the deep case hard: which seeding rules preserve exchangeability under stochastic training, representation invariance, the warm start that is tempting and destroys validity, and the sequential stopping that per-fit cost makes a design requirement. For model order the recursion, its level allocation and the accuracy design are carried over; we add a direct false-split measurement and a fresh-seed replication of the accuracy study (Appendices O and P).

3 METHOD

3.1 SETUP AND NULL HYPOTHESIS

Let $G = (V, E)$ be a simple undirected graph on n vertices with m edges and degree sequence d, and let the *degree fiber* $\mathcal{G}_d$ be the set of all simple vertex-labeled graphs on V with exactly that degree sequence. The graph is drawn from an unknown law P, and the null is a statement about P in conditional form: H_0 asserts that for every degree sequence d' the conditional law of G given

$d(G) = d'$, and given X when real attributes are used, is uniform on $\mathcal{G}_{d'}$, so that once the degrees are known no wiring consistent with them is likelier than any other. Write π for that uniform law on the observed fiber. $G(n, p)$ obeys H_0 exactly, so the Erdős-Rényi cells of Section 4.1 and the $c = 1$ embedded null of Section 4.3 are exact-null draws (Appendix W on why the conditional form is the one that can hold).

A *pipeline* $\mathcal{A}$ maps a graph, an attribute matrix X, an optional initial parameter state θ and its auxiliary randomness to a partition of V, and a *label-free score* S scores that partition without ground-truth labels, here achieved modularity (Newman & Girvan, 2004). Exogenous attributes are held fixed across the observed graph and all M null graphs, entering as the fixed auxiliary input of case (a) of Lemma 2, whereas structural features, which a deep pipeline needs on our featureless synthetic graphs, must be recomputed from each graph by the same deterministic recipe: a feature matrix computed once from the observed graph and reused for the nulls breaks exactness. Conditioning on d removes degree heterogeneity as a competing explanation: a rejection says the wiring carries structure beyond what the degrees force, not that the returned partition is correct, and a non-rejection says only that this pipeline at this budget found nothing that degree-matched rewirings do not also produce (Appendix W on the fixed X).

3.2 Exchangeable null samples

Sampling π directly is infeasible, so we run a reversible chain on $\mathcal{G}_d$ inside the parallel construction of Besag & Clifford (1989). That machinery, and its use to wrap black-box pipelines, is prior work (Anonymous, 2026); our contribution begins in Section 3.3. The proposal draws an ordered pair of edge slots uniformly and an orientation coin selecting one of the two ways to rewire their four endpoints, rejecting any move that creates a self-loop or a multi-edge. Degrees are preserved (Milo et al., 2002; Fosdick et al., 2018) and every rewiring is matched by the one undoing it at equal probability, so the kernel Q is symmetric and satisfies detailed balance,

$$\pi(x)\, Q(x, y) = \pi(y)\, Q(y, x), \qquad x, y \in \mathcal{G}_d. \tag{1}$$

The chain moves an *ordered array* of edge slots, so (1) describes its Markov projection onto graphs (Appendix A); a score must therefore depend on the graph, not on the order its edges are stored in (Section 3.3). Reversibility passes to the L-step kernel.

Fix $L = \max(1, \mathrm{round}(\rho m))$ for a swaps-per-edge rate ρ, let $W_0, \ldots, W_M$ be i.i.d. randomness independent of G, and let ϕ run L steps of Q. Run the chain once from the observed graph to a hub, then M independent chains from it:

$$G^* = \phi(G, W_0), \qquad G_i = \phi(G^*, W_i), \quad i = 1, \ldots, M. \tag{2}$$

The detour through the hub is what makes the construction work. Drawing the nulls by running M independent chains straight from the observed graph is the obvious alternative and it loses exactness: that puts G at the center with every G_i one run away from it, an arrangement no permutation of the $M + 1$ graphs leaves unchanged, so the observed coordinate stays distinguishable and the rank of its score is not an exact p-value; only $\Pr(p \le \alpha) \le 2\alpha$ holds (Barber & Ramdas, 2026). Starting from a hub that all $M + 1$ graphs are one run away from removes the asymmetry.

Lemma 1 (Exchangeability of the Besag-Clifford construction)**.** *Assume H_0 and the construction* (2)*, and condition throughout on $d(G) = d$. Then for all $y, x_0, \ldots, x_M \in \mathcal{G}_d$,*

$$\Pr[G = x_0,\, G^* = y,\, G_1 = x_1, \ldots, G_M = x_M \mid d(G) = d] = \pi(y) \prod_{i=0}^{M} Q^L(y, x_i). \tag{3}$$

Consequently $G^* \sim \pi$ and, conditionally on $G^* = y$, the graphs $G, G_1, \ldots, G_M$ are i.i.d. with law $Q^L(y, \cdot)$; in particular $(G, G_1, \ldots, G_M)$ is exchangeable.

The construction and this argument are due to Besag & Clifford (1989); we state the lemma in the form the rest of the paper uses. The proof (Appendix A) turns the backward run around: under H_0 the observed graph is stationary, so reversibility rewrites the law of (G, G^*) as a hub drawn from π followed by a forward run to G, the mechanism that produced every null graph. That argument nowhere asks the chain to have converged, so (3) holds at every L, however short, and validity does not depend on mixing. What L buys is power, since too short a run returns nulls that still resemble the observed graph; connectivity of the swap moves (Taylor, 1981) and aperiodicity matter for that reason and no other.

3.3 CERTIFYING A TRAINED PIPELINE

A trained pipeline is not a function of the graph alone: initialization, data order, dropout and a k-means readout inject randomness, and partitions move between runs (Wu et al., 2025; Leeney & McConville, 2024). Lemma 1 makes the $M+1$ graphs *exchangeable*, so nothing in their joint law marks which graph was observed; scoring all $M+1$ by one fixed rule carries that over to the scores, so the observed score is no likelier than any other to rank at the top and, with ties counted against it, lands in the top α fraction with probability at most α, whatever the pipeline does with the graphs it is handed. That is what it means for a test to hold its *level*, here at finite M and finite n with no asymptotic approximation anywhere. The work is all in "one fixed rule", which a trained pipeline does not satisfy for free.

Lemma 2 (Valid score rules)**.** *Let $(V_0, \dots, V_M)$ be an exchangeable vector of random elements of $\mathcal{G}_d$ and let $s : \mathcal{G}_d \times \mathcal{U} \to \mathbb{R}$ be measurable. Define scores by either* (a) $S_i = s(V_i, u)$ *for a single fixed $u \in \mathcal{U}$, the same for every coordinate, or* (b) $S_i = s(V_i, U_i)$ *with $U_0, \dots, U_M$ i.i.d. and independent of $(V_0, \dots, V_M)$. Then $(S_0, \dots, S_M)$ is exchangeable and*

$$p = \frac{1 + \big|\{\, i \in \{1, \dots, M\} : S_i \geq S_0 \,\}\big|}{M+1} \tag{4}$$

satisfies $\Pr[p \leq a] \leq a$ for every $a \in [0, 1]$.

The proof is in Appendix A; its rank step is the classical Monte Carlo argument (Besag & Clifford, 1989). Case (a) recovers what is already available for deterministic pipelines, case (b) is what stochastic training adds, and ties count against rejection through the $\geq$ in (4). All probabilities in Lemmas 1 and 2, the corollaries and Proposition 1 are conditional on the observed degree sequence, and on X when real attributes are used; averaging gives the same bounds unconditionally. Corollary 1 draws the consequence: with the recipe and seeding rule fixed in advance and applied unchanged to all $M+1$ graphs, non-convergence, collapse and a truncated epoch budget bear on power, never on level; the cluster budget is likewise no threat to level, and in practice moves recovery rather than power (Appendix Q; Appendices J and W on what the rule excludes).

Representation invariance. Implementations hold data structures, not graphs, so Lemma 2 asks more than determinism at a fixed seed: the rule must be invariant to the graph's *representation*, in particular to the row order of the edge array, which observed and null graphs reach the pipeline in differently. Our own code violated this, one wrapped pipeline drawing its contrastive positives by edge slot index, the asymmetry the permutation invariance of Lemma 2 forbids and no seed discipline repairs; we canonicalize the edge array at every fit's entry point (Appendix R).

The whole procedure is Algorithm 1 in Appendix F: read d off G, run the backward and the M forward chains of (2), canonicalize every graph's edge array, fit $\mathcal{A}$ once per graph under the fixed seeding rule, count exceedances with ties included, and report (4).

Defaults, fixed in advance and used throughout unless a study says otherwise: $\rho = 2$ swaps per edge, $M = 49$ null samples per test ($M = 199$ for the calibration audit of Section 4.1 and the sequential-stopping study), $\alpha = 0.05$, one fixed seed shared across the tuple, and the cluster budgets and size-stratified M on real graphs of Appendix B; the sequential rule of Section 3.5 replaces steps 3 to 7 of Algorithm 1 with an exceedance target $h = 10$ against a maximum $M = 199$.

3.4 HUB-CONDITIONAL AMORTIZATION

Exactness needs $M+1$ trained pipelines, so one wants to train once and warm start the rest. The natural way to do that is wrong. Lemma 1 delivers more than exchangeability: conditionally on $G^* = y$ the graphs $G, G_1, \dots, G_M$ are i.i.d., so by Corollary 2 any measurable T scored as $S_0 = T(G, G^*)$ and $S_i = T(G_i, G^*)$ keeps (4) exact, randomized T included. We therefore pretrain once on the hub, read through its canonicalized edge array, and fine-tune $\theta(G^*)$ on all $M+1$ graphs for a short budget.

The tempting variant pretrains on the *observed* graph instead, needing no extra fit; that pins the second argument to whichever coordinate is observed, so no permutation leaves the rule unchanged and the permutation invariance Lemma 2 requires fails. A failed sufficient condition is not a proof of invalidity; the experiment settles it: that scheme rejects 1.00 of structureless graphs at $\alpha = 0.05$,

mean p-value 0.020 (Table 4), while every mode covered by Corollary 2 holds its level. At this scale hub pretraining buys validity, not speed: a cheap cold-start surrogate dominates it (Section 4.4).

3.5 SEQUENTIAL STOPPING

Most graphs will not reject, and paying $M + 1$ full fits to learn that is wasteful. We use the rule of Besag & Clifford (1991): fix the exceedance target h and maximum M of Section 3.3, draw null scores one at a time, let $C_\ell = |\{i \le \ell : S_i \ge S_0\}|$, and stop at $T = \min\{\ell : C_\ell = h\} \wedge M$, reporting $p = h/T$ if the hth exceedance arrives and the fixed-budget $p = (1 + C_M)/(M + 1)$ otherwise. Its validity, unchanged from Besag & Clifford (1991), needs the observed and null scores to follow the same law, not merely to be independent, which Lemma 1 supplies conditionally on G^* in both the full-fit and hub-pretrained modes. At our (h, M), where $h > \alpha M$, two properties follow (Appendix W): an early stop is always a non-rejection at 0.05, so the rule accelerates the null case only, and the reported value is in general not the fixed-M p-value, though our two decisions coincide on every realization (Appendix M).

3.6 CERTIFYING MODEL ORDER

Applied recursively, the same test returns a model order whose false splits are controlled at level α under the condition of Proposition 1, which is not the same as certifying a block count. The recursion and its level allocation are carried over, not contributed (Section 2). Peel-and-merge runs top down: a node v on n_v vertices is tested at level $\alpha_v = \alpha\, n_v/n$ against the local null that $G[V_v]$, relabeled, is uniform on the fiber of its own degree sequence, with a battery $\mathcal{S}$, modularity and the count of negative Bethe-Hessian eigenvalues (Saade et al., 2014; Le & Levina, 2022), on one shared set of M_v null samples, rejecting when the smallest battery p-value is at most $\alpha_v/|\mathcal{S}|$; a node that does not reject, is smaller than $n_{\min}$, or receives a one-part partition is a leaf, otherwise it splits into the parts of the wrapped algorithm's partition and recurses; a merge pass then undoes sibling splits that do not reject, and $\hat{K}$ is the number of leaves (Appendix F). Detection and division run on different machinery, battery and wrapped algorithm, so they fail separately, as both do on the bipartite blocks of Appendix P. The wrapper is classical Leiden, since the recursion costs $\sum_v (M_v + 1)$ fits; any pipeline can be substituted at that price (Appendix W records one battery deviation on the largest real graph).

Proposition 1 (Family-wise control of false splits)**.** *Call a vertex set W truly structureless if $G[W]$, relabeled, is uniform on the fiber of its own degree sequence. Enumerate nodes in the order the top-down recursion tests them, $v_1, v_2, \ldots$, and let $\mathcal{F}_{k-1}$ be the σ-field generated by the vertex sets and split outcomes of $v_1, \ldots, v_{k-1}$, the vertex set of v_k and the degree sequence of $G[V_{v_k}]$. Assume selection neutrality: for every k, on the event that v_k is tested and truly structureless, the conditional law of $G[V_{v_k}]$ given $\mathcal{F}_{k-1}$ is uniform on that fiber, and the node's null samples are drawn by* (2) *with randomness independent of $(G, \mathcal{F}_{k-1})$. Then the probability that peel-and-merge splits any truly structureless node is at most α.*

The guarantee restates that of Anonymous (2026) against the recursion's filtration; the argument (Appendix A) bounds the *first* false split: the structureless nodes tested before it are pairwise disjoint, so their levels sum to at most α. Selection neutrality is an assumption and not a consequence of Lemma 2, and a splitter built to isolate fiber-atypical vertex subsets would violate it; on structureless graphs (Appendix O) the first split is the root's, so that check measures the root test, not the assumption. And $n_{\min}$ is an honest resolution limit rather than a tuning convenience, one of two effects pushing $\hat{K}$ toward conservatism (Appendix W); every run uses the α and $n_{\min}$ of Appendix F, and $\hat{K}$ must be read against both. A test costs $M + 1$ fits plus $M + 1$ chain runs (Appendix K).

4 EXPERIMENTS

Setup. We wrap three deep clusterers, DMoN-lite (Tsitsulin et al., 2023), MinCut-lite (Bianchi et al., 2020) and MAGI-lite (Liu et al., 2024), sharing one encoder, and Leiden (Traag et al., 2019) as a classical reference; the score is the returned partition's modularity. Structureless graphs are Erdős-Rényi and power-law configuration models with exponent τ, structured graphs degree-corrected block models (Karrer & Newman, 2011) in an assortative and a bipartite-pairs pattern at contrast

c, where $c = 1$ is exactly Erdős-Rényi and doubles as an embedded null. Every design parameter, including the burn-in every heavy-tailed draw passes through (Fosdick et al., 2018) and the null sample sizes of Section 3.3, is fixed in advance, not tuned (Appendix B, which also says what the 6145 logged runs count). One correction precedes every number: an audit of our own code found the representation-invariance defect of Section 3.3 in MAGI-lite, so we canonicalized the edge order and reran every MAGI-lite arm, changing no qualitative conclusion (Appendix R).

4.1 What deep clusterers report on structureless graphs

Does the wrapper hold level around a trained pipeline? 1800 exact tests: three structureless settings, three deep pipelines, 200 replicates per cell, full training on every graph in every tuple (Table 3 in Appendix E). **At $\alpha = 0.01$ and $\alpha = 0.05$ no cell rejects significantly more often than its nominal level.** At $\alpha = 0.01$ all 9 of the 9 cells have a Wilson 95% interval covering the nominal level, with rates from 0.000 to 0.010. At $\alpha = 0.05$, 8 do, the largest cell's Wilson interval $[0.038, 0.108]$ covering α; the exception is MinCut-lite on Erdős-Rényi graphs, which sits *below* at 0.005, Wilson $[0.001, 0.028]$. At $\alpha = 0.10$, 7 do: that same cell is again below, and DMoN-lite at $\tau = 2.1$ is the one cell *above*, at 0.145, Wilson $[0.103, 0.200]$. It is the only upward miss among 9 cells checked at each of three levels, the order of miss rate sampling variation alone produces under Lemma 2. The conservative cell has a mechanism: MinCut-lite returns a single cluster on nearly every Erdős-Rényi graph, so the observed score ties with the null tuple and ties count against rejection; it un-collapses at $\tau = 2.1$, where it too reports a modal 3 clusters on noise, but still returns one cluster on a fraction 0.745 of $\tau = 2.5$ graphs (Appendix G). Meanwhile (Fig. 1), on Erdős-Rényi noise DMoN-lite returns a modal 16 communities, the full budget we gave it, at mean modularity 0.25 and MAGI-lite 16 at 0.21, scores unremarkable against their own null.

Three shorter studies are in the appendix, the first two with planted arms beside the structureless ones. Sequential stopping ends 0.95 of null tests early at level 0.045 with no decision changed (Appendix M); all three seeding rules we ran hold level (Appendix N); and on structureless graphs, where the first split is always the root's, peel-and-merge splits 0.022 of 600, Wilson $[0.013, 0.037]$, below 0.05, a check of the root test rather than of selection neutrality (Appendix O).

4.2 An asymptotic baseline on the same graphs

What does a fixed-threshold asymptotic rule do here? As Anonymous (2026) did for classical statistics, we recompute a Bickel-Sarkar-style decision (Bickel & Sarkar, 2016) on exactly the structureless graphs of Section 4.1, rejecting when $n^{2/3}(\lambda_1 - 2)$ exceeds 0.9793, the Tracy-Widom 95th percentile, on shared graphs and seeds. Only the Erdős-Rényi cell is a type-I error rate under the competitor's own hypothesis, and there the threshold rejects 1.000 at 200 replicates, every rejection false; pooled over all 600 graphs, including two heavy-tailed settings whose null is false by construction, it is 1.000. The exact wrapper holds level on those graphs. Two controls locate the failure, since the procedures differ in statistic and null at once. *Same statistic*: that eigenvalue wrapped in the exact test rejects 0.039 on 180 of those graphs (the first 60 replicates of each setting), where the fixed threshold rejects 1.000. *In regime*: on dense Erdős-Rényi graphs, its own null in the fixed-edge-probability regime its limit is proved for, the threshold is consistent with its level, rejecting 0.017 at $n = 1000$ with mean degree 200, Wilson $[0.003, 0.089]$, and 0.083 at $n = 2000$ with mean degree 300, Wilson $[0.036, 0.181]$, at 60 replicates each (Appendix C). This is not a refutation of the asymptotic theory: at mean degree 10 and $n = 1000$ the rule is outside its regime, and its authors recommend a simulated shift-and-scale correction there; that correction and exact fitted-model tests (Karwa et al., 2024) test a block model rather than a pipeline, so we evaluate the uncorrected threshold (Appendix C). The narrower point is that it gives a practitioner holding one sparse graph no signal that they have left that regime; the exact test needs none.

4.3 Power and the recovery gap

When the test rejects, has the pipeline found the planting? 1200 exact tests, two patterns, contrasts $\{1, 2, 3, 5, 8, 12\}$, 25 replicates, four wrapped algorithms, logging the exact p-value and the normalized mutual information (NMI) against the planted labels (Fig. 2 in Appendix D); the embedded null at $c = 1$ rejects at 0.02. Power and recovery then separate, as Anonymous (2026) found for Leiden. On the assortative pattern Leiden, DMoN-lite and MAGI-lite reach power 1.00 by $c = 5$ at mean

Table 1: Real graphs. Exact p per wrapped pipeline (clusters reported in parentheses; DMoN, MinCut and MAGI are the -lite reimplementations), Leiden-wrapped peel-and-merge $\hat{K}$, and two runtimes, the largest per-test time among the wrapped pipelines and the time for the peel-and-merge certificate. M is stratified by size (199, 99, 49); peel-and-merge runs at $\alpha = 0.05$ with minimum node size 30; only Cora uses native attributes. Amazon Photo's $\hat{K}$ uses modularity alone and is not on a common threshold with the other rows (Appendix W). The Leiden column and $\hat{K}$ repeat the real-graph study of Anonymous (2026) in our harness.

graph	n	m	exact p (K reported)				$\hat{K}$	seconds	
			Leiden	DMoN	MinCut	MAGI		max	$\hat{K}$
Karate	34	78	0.005 (4)	0.010 (14)	0.115 (8)	0.880 (16)	3	24	<1
Dolphins	62	159	0.005 (5)	0.005 (16)	0.005 (8)	0.005 (16)	3	25	1
Football	115	613	0.005 (10)	0.005 (16)	0.005 (8)	0.005 (16)	10	40	6
PolBooks	105	441	0.005 (5)	0.005 (16)	0.005 (8)	0.005 (16)	8	37	3
Cora	2485	5069	0.010 (32)	0.010 (16)	0.010 (8)	0.010 (16)	110	371	2650
Citeseer	2120	3679	0.010 (40)	0.010 (16)	0.010 (8)	0.010 (16)	84	263	1887
Am. Photo	7487	119043	0.020 (15)	0.020 (16)	0.020 (8)	0.020 (16)	332	565	7948

NMI 0.46, 0.40 and 0.42; MinCut-lite, still collapsed, is the exception at 0.00, so all four rejecting at $c = 5$ would be a false summary. At $c = 8$ Leiden essentially recovers the planting (NMI 0.92) while DMoN-lite sits at 0.58 and MAGI-lite at 0.61 at unchanged power. Most of that distance is the over-provisioned cluster budget these arms run at rather than the objective: matching the budget to the planted block count closes most of it without moving power at $c \geq 5$, where power is saturated at every budget, in a separate sweep of 390 exact tests at different seeds (so the change is a direction, not a paired difference); what does not move is the separation between rejecting and recovering (Appendix Q). On bipartite pairs every algorithm fires by $c = 8$, three of them through pair-merging partitions whose modularity rises while NMI stays low (Appendix D), so p-value and partition should be reported together.

4.4 Amortization and the validity boundary

Can training be reused across the tuple without breaking exactness, and does it pay? 810 exact tests wrapping DMoN-lite over six modes: a full fit per graph, a reduced-epoch cold-start surrogate, hub pretraining at three fine-tuning budgets, and the deliberately invalid mode that pretrains on the *observed* graph (Table 4 in Appendix H).

The validity boundary is sharp (Table 4). Every exact mode holds level, the full fit, every hub budget and the surrogate alike, with every Wilson interval covering α. The invalid sibling rejects every null graph, Wilson $[0.94, 1.00]$, for the reason Section 3.4 gives.

Warm-starting is exact and does not pay here. At $c = 5$ an equal-cost reduced-epoch cold-start surrogate reaches full power where hub fine-tuning at matched cost falls well short (Table 4; Appendix H, which also plots power against cost). Shortening the chain so the hub sits nearer the observed graph does not rescue it: hub power stays between 0.32 and 0.44, at a replication too small to resolve a modest trend, so we detect no effect rather than refute one (Appendix L, which also reports one level cell whose interval misses the nominal level, read as Monte Carlo noise). Two mechanisms are consistent with the loss, feature-basis misalignment across the tuple and the generalization penalty of warm-starting, and we did not run the ablation that separates them (Appendix S). At this scale, amortize by training less.

4.5 Real graphs and model order

What happens on real data? 7 graphs spanning over two orders of magnitude in n, each the largest connected component of its simple undirected version, with all four wrappers and a Leiden-wrapped peel-and-merge row (Table 1). Every graph is certified by at least one pipeline, and nearly every (graph, pipeline) pair sits at the p-value floor for its stratified M; the exceptions are on Karate, where DMoN-lite certifies one step above the floor ($p = 0.010$) while MinCut-lite and MAGI-lite retain,

MAGI-lite with the highest NMI of the three deep pipelines, 0.48 against DMoN-lite's 0.46, so reference agreement and certificate point opposite ways, and MAGI-lite certifies at both smaller cluster budgets (Appendix Q). Cora and Citeseer reject at the floor while agreeing only weakly with the reference labels, and the model orders on the three largest graphs are upper bounds on what was certified (Appendix T.1).

Model order against a planted block count. On 480 degree-corrected block models with $K \in \{2, 4, 8\}$, Leiden-wrapped peel-and-merge is competitive on assortative structure, not dominant (Appendix P): it has the best pooled mean absolute error point estimate, 1.59 against 1.72 for the Bethe-Hessian count, but the bootstrap intervals $[1.41, 1.77]$ and $[1.54, 1.90]$ overlap over most of their length, pooled exact-match does not separate the top four of five estimators, and against a constant answer, 0.333 exact-match and 2.00 error on this grid, it clears the exact-match floor by less than its own interval width, while on error it and the Bethe-Hessian count, which share a spectrum, are the only two that beat the constant (Appendix P.3). On bipartite pairs it fails for two distinct reasons, detection, no split being reached on 0.775 of the two-block graphs, and division, 0.988 of the strong-contrast splits reporting exactly $K/2$; we ran neither repair (Appendix P.2).

5 Discussion and limitations

What a certificate asserts. The null fixes the degree sequence, and attributes where real ones are used, so a small p-value says only that the *wiring* carries structure the degrees do not explain (Milo et al., 2002; Fosdick et al., 2018). It does not say the partition is correct, nor validate K: the output is a certified refinement, not a consistent block count (Le & Levina, 2022). It belongs not to the graph but to a (graph, pipeline, configuration) triple: the four pipelines split on Karate (Table 1), and MAGI-lite's retention there becomes a certificate at a smaller cluster budget (Appendix Q). That is intended, the audited object being the deployed pipeline (Wu et al., 2025; Leeney & McConville, 2024); a graph-level answer needs several pipelines and a multiplicity correction (Appendix Y).

Limitations. Four would change a reader's decision to use the method: a certificate costs $M + 1$ fits and a model order a test at every node, so the wrapped pipeline must be trainable a couple of hundred times over at the audit budget, and everything here runs on one CPU machine at thousands of nodes; the wrapped rule must be invariant to the graph's representation, a property of an implementation rather than of an objective; model order fails on disassortative structure for a detection reason and a separate division reason, and every model-order result wraps classical Leiden rather than a deep pipeline; and each certificate is conditional on the cluster budget it was run at, which should be varied, with a correction for the budgets tried, before a retention is read as a statement about the graph. Appendix I carries the full list with its numbers, and Appendix Y the repairs it points to.

Ethics statement

This work is a tool for restraint. Graph clustering generates hypotheses in biology and network security and returns a partition on any input, including pure noise, so a layer that withholds unsupported claims should reduce the false discoveries such pipelines seed. One misreading deserves stating: a small p-value certifies structure in the wiring for the pipeline wrapped, not that the clusters are right, their number correct, or that they map to any real-world category, and it should accompany rather than replace the usual evidence that a partition is meaningful.

Reproducibility statement

Every number, figure and table in this paper is generated from logged experiment output by scripts in our code repository: the experiment runners write one CSV per study, a numbers script renders every reported value as a macro into the manuscript source, and the figure and table scripts read only those CSVs, so no result is transcribed by hand. Seeds are fixed and derived arithmetically per job; every generator, chain length, training budget and test parameter is recorded in Appendix B, the proofs are in Appendix A, the peel-and-merge procedure is stated in full in Appendix F, and the real graphs are the standard public releases named in Appendix B. The code, the CSVs and the scripts that regenerate this manuscript will be released.

AI USE STATEMENT

We used a large language model as an assistant in developing the method, its hypotheses and mathematical claims, and in drafting their proofs; in designing, implementing and running the experiments, including preprocessing the real graphs, and in interpreting their results; and in drafting and editing the text. No data were produced as model output: every synthetic graph is drawn by a seeded generator in our code, and translation and qualitative analysis do not apply. The authors chose the research question; every reported number is generated from the logged runs by scripts, the code was tested end to end on the chain's invariants and on the embedded null of Section 4.3, and every reference was resolved against its bibliographic record. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Emmanuel Abbe. Community detection and stochastic block models: Recent developments. *Journal of Machine Learning Research*, 18(177):1–86, 2018. URL https://jmlr.org/papers/v18/16-480.html.

Anonymous. Exact, algorithm-agnostic tests for structure and model order in graph clustering, 2026. Anonymized concurrent submission; full reference to be supplied for the camera-ready.

Jordan T. Ash and Ryan P. Adams. On warm-starting neural network training. In *Advances in Neural Information Processing Systems 33*, 2020. arXiv:1910.08475.

Rina Foygel Barber and Aaditya Ramdas. Monte Carlo testing: non-asymptotic guarantees without joint exchangeability. *arXiv preprint arXiv:2607.23010*, 2026. URL https://arxiv.org/abs/2607.23010.

Julian Besag and Peter Clifford. Generalized Monte Carlo significance tests. *Biometrika*, 76(4): 633–642, 1989. doi: 10.1093/biomet/76.4.633.

Julian Besag and Peter Clifford. Sequential Monte Carlo p-values. *Biometrika*, 78(2):301–304, 1991. doi: 10.1093/biomet/78.2.301.

Filippo Maria Bianchi, Daniele Grattarola, and Cesare Alippi. Spectral clustering with graph neural networks for graph pooling. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 874–883, 2020. URL https://proceedings.mlr.press/v119/bianchi20a.html.

Peter J. Bickel and Purnamrita Sarkar. Hypothesis testing for automated community detection in networks. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 78(1): 253–273, 2016. doi: 10.1111/rssb.12117.

Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008 (10):P10008, 2008. doi: 10.1088/1742-5468/2008/10/P10008.

Yiqun T. Chen and Daniela M. Witten. Selective inference for k-means clustering. *Journal of Machine Learning Research*, 24(152):1–41, 2023. URL https://jmlr.org/papers/v24/22-0371.html.

Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106, 2011. doi: 10.1103/PhysRevE.84.066106.

Devvrit, Aditya Sinha, Inderjit S. Dhillon, and Prateek Jain. S3GC: Scalable self-supervised graph clustering. In *Advances in Neural Information Processing Systems 35*, 2022. URL https://papers.nips.cc/paper_files/paper/2022/hash/15972a9575e0f03bf82f00aebeb40774-Abstract-Conference.html.

Alexander Dombowsky, Barbara E. Engelhardt, and Aaditya Ramdas. Besag-Clifford e-values for unnormalized testing. *arXiv preprint arXiv:2603.15845*, 2026. URL https://arxiv.org/abs/2603.15845.

Carles Domingo-Enrich, Raaz Dwivedi, and Lester Mackey. Cheap permutation testing. *arXiv preprint arXiv:2502.07672*, 2025. URL https://arxiv.org/abs/2502.07672.

Lasse Fischer and Aaditya Ramdas. Sequential Monte-Carlo testing by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 87(4):1200–1220, 2025. doi: 10.1093/jrsssb/qkaf014. arXiv:2401.07365.

Bailey K. Fosdick, Daniel B. Larremore, Joel Nishimura, and Johan Ugander. Configuring random graph models with fixed degree sequences. *SIAM Review*, 60(2):315–355, 2018. doi: 10.1137/16M1087175.

Axel Gandy. Sequential implementation of Monte Carlo tests with uniformly bounded resampling risk. *Journal of the American Statistical Association*, 104(488):1504–1511, 2009. doi: 10.1198/jasa.2009.tm08368.

Lucy L. Gao, Jacob Bien, and Daniela Witten. Selective inference for hierarchical clustering. *Journal of the American Statistical Association*, 119(545):332–342, 2024. doi: 10.1080/01621459.2022.2116331.

Michelle Girvan and Mark E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002. doi: 10.1073/pnas.122653799.

Isabella N. Grabski, Kelly Street, and Rafael A. Irizarry. Significance analysis for clustering with single-cell RNA-sequencing data. *Nature Methods*, 20(8):1196–1202, 2023. doi: 10.1038/s41592-023-01933-9.

Roger Guimerà, Marta Sales-Pardo, and Luís A. Nunes Amaral. Modularity from fluctuations in random graphs and complex networks. *Physical Review E*, 70(2):025101, 2004. doi: 10.1103/PhysRevE.70.025101.

Zengyou He, Hao Liang, Zheng Chen, Can Zhao, and Yan Liu. Computing exact P-values for community detection. *Data Mining and Knowledge Discovery*, 34(3):833–869, 2020. doi: 10.1007/s10618-020-00681-0.

Benjamin Hivert, Denis Agniel, Rodolphe Thiébaut, and Boris P. Hejblum. Post-clustering difference testing: Valid inference and practical considerations with applications to ecological and biological data. *Computational Statistics & Data Analysis*, 193:107916, 2024. doi: 10.1016/j.csda.2023.107916.

Kexin Huang, Ying Jin, Emmanuel Candès, and Jure Leskovec. Uncertainty quantification over graph with conformalized graph neural networks. In *Advances in Neural Information Processing Systems 36*, 2023. arXiv:2305.14535.

Jiashun Jin, Zheng Tracy Ke, Jiajun Tang, and Jingming Wang. Network goodness-of-fit for the block-model family. *Journal of the American Statistical Association*, 120(551):1919–1932, 2025. doi: 10.1080/01621459.2025.2479242.

Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83(1):016107, 2011. doi: 10.1103/PhysRevE.83.016107.

Vishesh Karwa, Debdeep Pati, Sonja Petrović, Liam Solus, Nikita Alexeev, Mateja Raič, Dane Wilburne, Robert Williams, and Bowei Yan. Monte Carlo goodness-of-fit tests for degree corrected and related stochastic blockmodels. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):90–121, 2024. doi: 10.1093/jrsssb/qkad084.

Patrick K. Kimes, Yufeng Liu, David Neil Hayes, and James Stephen Marron. Statistical significance for hierarchical clustering. *Biometrics*, 73(3):811–821, 2017. doi: 10.1111/biom.12647.

Sadamori Kojaku and Naoki Masuda. A generalised significance test for individual communities in networks. *Scientific Reports*, 8:7351, 2018. doi: 10.1038/s41598-018-25560-z.

Can M. Le and Elizaveta Levina. Estimating the number of communities by spectral methods. *Electronic Journal of Statistics*, 16(1):3315–3342, 2022. doi: 10.1214/21-EJS1971.

William Leeney and Ryan McConville. Uncertainty in GNN learning evaluations: A comparison between measures for quantifying randomness in GNN community detection. *Entropy*, 26(1):78, 2024. doi: 10.3390/e26010078.

Jing Lei. A goodness-of-fit test for stochastic block models. *The Annals of Statistics*, 44(1):401–424, 2016. doi: 10.1214/15-AOS1370.

Yue Liu, Ke Liang, Jun Xia, Sihang Zhou, Xihong Yang, Xinwang Liu, and Stan Z. Li. Dink-Net: Neural clustering on large graphs. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21794–21812, 2023. arXiv:2305.18405.

Yue Liu, Jun Xia, Benyu Wu, Sihang Zhou, Xihong Yang, Ke Liang, Chenchen Fan, Yan Zhuang, Guoxian Yu, Stan Z. Li, Xinwang Liu, and Kunlun He. A survey of deep graph clustering: Taxonomy, challenge, application, and open resource. *IEEE Transactions on Knowledge and Data Engineering*, 38(6):3365–3384, 2026. doi: 10.1109/TKDE.2026.3669747. arXiv:2211.12875.

Yunfei Liu, Jintang Li, Yuehe Chen, Ruofan Wu, Ericbk Wang, Jing Zhou, Sheng Tian, Shuheng Shen, Xing Fu, Changhua Meng, Weiqiang Wang, and Liang Chen. Revisiting modularity maximization for graph clustering: A contrastive learning perspective. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1968–1979, 2024. doi: 10.1145/3637528.3671967. arXiv:2406.14288.

David Lusseau, Karsten Schneider, Oliver J. Boisseau, Patti Haase, Elisabeth Slooten, and Steve M. Dawson. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54(4):396–405, 2003. doi: 10.1007/s00265-003-0651-y.

Colin McDiarmid and Fiona Skerman. Modularity of Erdős-Rényi random graphs. *Random Structures & Algorithms*, 57(1):211–243, 2020. doi: 10.1002/rsa.20910.

Nicolai Meinshausen. Hierarchical testing of variable importance. *Biometrika*, 95(2):265–278, 2008. doi: 10.1093/biomet/asn007.

Ron Milo, Shai Shen-Orr, Shalev Itzkovitz, Nadav Kashtan, Dmitri Chklovskii, and Uri Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002. doi: 10.1126/science.298.5594.824.

Yugo Miyata, Tomohiro Shiraishi, Shuichi Nishino, and Ichiro Takeuchi. Statistical testing framework for clustering pipelines by selective inference. *arXiv preprint arXiv:2603.18413*, 2026. URL https://arxiv.org/abs/2603.18413.

Anirban Nath, YoonHaeng Hur, and Genevera I. Allen. Weighted conformal clustering. *arXiv preprint arXiv:2606.00436*, 2026. URL https://arxiv.org/abs/2606.00436.

Anna Neufeld, Ameer Dharamshi, Lucy L. Gao, and Daniela Witten. Data thinning for convolution-closed distributions. *Journal of Machine Learning Research*, 25(57):1–35, 2024. URL https://jmlr.org/papers/v25/23-0446.html.

M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004. doi: 10.1103/PhysRevE.69.026113.

Shuichi Nishino, Tomohiro Shiraishi, Teruyuki Katsuoka, and Ichiro Takeuchi. Statistical test for saliency maps of graph neural networks via selective inference. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=5NkXTCVa7F.

Oleg Platonov, Denis Kuznedelev, Michael Diskin, Artem Babenko, and Liudmila Prokhorenkova. A critical look at the evaluation of GNNs under heterophily: Are we really making progress? In *International Conference on Learning Representations*, 2023. arXiv:2302.11640.

Alaa Saade, Florent Krzakala, and Lenka Zdeborová. Spectral clustering of graphs with the Bethe Hessian. In *Advances in Neural Information Processing Systems 27*, pp. 406–414, 2014.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–106, 2008. doi: 10.1609/aimag.v29i3.2157.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *arXiv preprint arXiv:1811.05868*, 2018. URL https://arxiv.org/abs/1811.05868.

Richard Taylor. Constrained switchings in graphs. In *Combinatorial Mathematics VIII*, volume 884 of *Lecture Notes in Mathematics*, pp. 314–336. Springer, Berlin, Heidelberg, 1981. doi: 10.1007/BFb0091828.

V. A. Traag, L. Waltman, and N. J. van Eck. From Louvain to Leiden: guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, 2019. doi: 10.1038/s41598-019-41695-z.

Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Müller. Graph clustering with graph neural networks. *Journal of Machine Learning Research*, 24(127):1–21, 2023. URL https://jmlr.org/papers/v24/20-998.html.

Edwin B. Wilson. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927. doi: 10.1080/01621459.1927.10502953.

Benyu Wu, Yue Liu, Qiaoyu Tan, Xinwang Liu, Wei Du, Jun Wang, and Guoxian Yu. DGCBench: A deep graph clustering benchmark. In *Advances in Neural Information Processing Systems 38: Datasets and Benchmarks Track*, 2025. URL https://openreview.net/forum?id=dKVUUZfcW9.

Eric Yanchenko and Srijan Sengupta. A generalized hypothesis test for community structure in networks. *Network Science*, 12(2):122–138, 2024. doi: 10.1017/nws.2024.1.

Zhilin Yang, William W. Cohen, and Ruslan Salakhutdinov. Revisiting semi-supervised learning with graph embeddings. In *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 40–48, 2016. URL https://proceedings.mlr.press/v48/yanga16.html.

Zhitao Ying, Jiaxuan You, Christopher Morris, Xiang Ren, William L. Hamilton, and Jure Leskovec. Hierarchical graph representation learning with differentiable pooling. In *Advances in Neural Information Processing Systems 31*, pp. 4805–4815, 2018.

Wayne W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33(4):452–473, 1977. doi: 10.1086/jar.33.4.3629752.

GUIDE TO THE APPENDIX

This appendix holds the proofs, the reproducibility detail, the per-cell readouts behind the body's summary statements, and the secondary studies whose results the body states in a paragraph each. Appendix A proves Lemma 1, Lemma 2 and Proposition 1, whose statements are in Section 3, states and proves Corollaries 1 and 2, which Section 3 cites, and records the one inductive step about the chain those proofs consume. Appendix B records the reproducibility detail behind Section 4: the wrapped pipelines' training budgets and hyperparameters, the generators, the test parameters, and the provenance of the real graphs. The sections from Appendix C to Appendix T.1 carry one study or one caveat apiece: the per-cell readings behind the calibration audit and the asymptotic comparison, the chain-length, stopping, seeding and false-split studies, the model-order study against a planted block count and the cluster-budget sweep, what the model-order baselines actually return, the limitations in full, the representation-invariance defect we found in our own code, and the arithmetic behind reading the real-graph model orders as upper bounds. The five closing sections hold the long form of the introduction and of the related work, notes on what the null holds fixed and what the lemmas license, the experimental grids and per-cell readings, and the longer discussion of scope, cost and directions.

Table 2 is the index into all of it, section by section. A reader following the argument needs none of this; a reader checking a particular claim needs the row belonging to the section that makes it.

Table 2: Where to look. For each body section, the appendix sections that support it.

Body section		Appendix sections
1	Introduction	E, I, M, O, P, Q, R, U
2	Related work	O, P, V
3.1	Setup and null hypothesis	W
3.2	Exchangeable null samples	A, W
3.3	Certifying a trained pipeline	A, B, F, J, R, Q, W
3.4	Hub-conditional amortization	A, H, L, S, W
3.5	Sequential stopping	M, W
3.6	Certifying model order	A, F, K, O, P, W
4	Experiments, common setup	B, R, X
4.1	Deep clusterers on structureless graphs	E, G, M, N, O, X
4.2	An asymptotic baseline	C, X
4.3	Power and the recovery gap	D, Q, Q.1, X
4.4	Amortization and the validity boundary	H, L, S, X
4.5	Real graphs and model order	T, T.1, P, P.2, P.3, P.1, Q, X
5	Discussion and limitations	I, Q, R, Y

A PROOFS

Projection of the slot chain onto graphs. The implemented chain is carried on an ordered array of edge slots and a move overwrites the two chosen slots in place, so the state space of the implemented process is slot assignments rather than graphs. Edges are stored with a canonical orientation, so two slot arrays of the same graph differ only by a permutation of slots, and the proposal law is invariant under slot permutations; the probability of moving from an array of x into the arrays of y is therefore the same for every array of x (strong lumpability), and the process projected onto graphs is Markov with the symmetric kernel Q of (1). That projection is what (1) describes and what every validity argument uses, and it is why a wrapped score must be a function of the graph and not of the ordering in which its edges happen to be stored.

Reversibility of the L-step kernel. Reversibility passes to the L-step kernel, by induction on L with (1) as the base case: assuming $\pi(x)Q^{L-1}(x,y) = \pi(y)Q^{L-1}(y,x)$ and applying this hypothesis

and then (1) once,

$$\pi(x)Q^L(x,y) = \sum_z \pi(x)Q^{L-1}(x,z)Q(z,y) = \sum_z \pi(z)Q^{L-1}(z,x)Q(z,y)$$
$$= \sum_z \pi(y)Q(y,z)Q^{L-1}(z,x) = \pi(y)Q^L(y,x). \tag{5}$$

Proof of Lemma 1. The streams $W_1, \dots, W_M$ are independent of each other and of (G, W_0), and $G_i = \phi(G^*, W_i)$ has, given (G, W_0), graph law $Q^L(G^*, \cdot)$ by the projection paragraph, which depends on (G, W_0) only through the graph G^*. Hence conditionally on $(G, G^*) = (x_0, y)$ the graphs $G_1, \dots, G_M$ are independent with law $Q^L(y, \cdot)$, so the joint probability in (3) equals $\Pr[G = x_0, G^* = y] \prod_{i=1}^M Q^L(y, x_i)$. Under H_0 the observed graph is drawn from the stationary law, so $\Pr[G = x_0, G^* = y] = \pi(x_0)Q^L(x_0, y)$, and reversibility of the L-step kernel (5) rewrites this as $\pi(y)Q^L(y, x_0)$; substituting gives (3). Summing (3) over $x_0, \dots, x_M$ gives $\Pr[G^* = y] = \pi(y)$, and dividing by $\pi(y)$ leaves $\prod_{i=0}^M Q^L(y, x_i)$, the law of $M + 1$ i.i.d. draws from $Q^L(y, \cdot)$. Exchangeability follows because a product of identical conditional laws is permutation invariant, and survives integration over G^*. □

Proof of Lemma 2. Step 1: the scores are exchangeable. In case (a), let f be the coordinatewise map $(v_0, \dots, v_M) \mapsto (s(v_0, u), \dots, s(v_M, u))$. For any permutation σ, $(V_{\sigma(0)}, \dots, V_{\sigma(M)}) \overset{d}{=} (V_0, \dots, V_M)$, and applying the fixed measurable map f to both sides preserves equality in law, which is exactly $(S_{\sigma(0)}, \dots, S_{\sigma(M)}) \overset{d}{=} (S_0, \dots, S_M)$. In case (b), the law of the pairs $((V_i)_i, (U_i)_i)$ is a product of an exchangeable law and an i.i.d. law, so permuting the pairs leaves it unchanged; apply $(v, u) \mapsto s(v, u)$ coordinatewise as before.

Step 2: the rank bound. For $j = 0, \dots, M$ set $R_j = 1 + |\{i \neq j : S_i \geq S_j\}|$, so $p = R_0/(M + 1)$. We first record a deterministic fact: for any reals $s_0, \dots, s_M$ and any integer $r \geq 0$,

$$\big|\{\, j : R_j \leq r \,\}\big| \leq r. \tag{6}$$

For $r = 0$ the set is empty because $R_j \geq 1$ always, so assume $r \geq 1$. Let $A = \{j : R_j \leq r\}$ and $q = |A|$, and assume $q \geq 1$. Choose $j^* \in A$ minimizing s_j over A, ties broken arbitrarily. Every $j \in A \setminus \{j^*\}$ has $s_j \geq s_{j^*}$, so the count defining R_{j^*} includes at least $q - 1$ indices and $R_{j^*} \geq q$; since $j^* \in A$ we get $q \leq R_{j^*} \leq r$, which is (6). By Step 1, swapping coordinates 0 and j leaves the law of the score vector unchanged and carries R_0 to R_j, so $R_j \overset{d}{=} R_0$ for every j; taking expectations in (6) gives $(M + 1)\Pr[R_0 \leq r] = \sum_j \Pr[R_j \leq r] = \mathbb{E}|\{j : R_j \leq r\}| \leq r$, so $\Pr[R_0 \leq r] \leq r/(M + 1)$. Finally, for $a \in [0, 1]$ the event $\{p \leq a\}$ is $\{R_0 \leq \lfloor a(M + 1) \rfloor\}$, which is covered by (6) at $r = \lfloor a(M + 1) \rfloor \geq 0$, hence $\Pr[p \leq a] \leq \lfloor a(M + 1) \rfloor/(M + 1) \leq a$. □

Corollary 1 (Training pathologies affect power, not level)**.** *Assume H_0 and* (2)*. Let $\mathcal{A}$ be trained by any fixed recipe, any architecture, optimizer, epoch budget, or early-stopping rule, that is a function of the graph and not of its representation (for example, one applied to the canonicalized edge array, Appendix R), applied identically to every graph, with the training seed either fixed across graphs or drawn i.i.d. per graph from a stream independent of the graphs. Then* (4) *is valid. In particular non-convergence, collapse to a single cluster, and a truncated epoch budget bear on power only. They may move the separation between S_0 and the null scores in either direction, and we claim nothing about which; what they cannot do is inflate the type-I error.*

Proof of Corollary 1. The recipe is one fixed map from a graph (not its storage order) and a seed to a partition, and under H_0 and (2) Lemma 1 makes the $M + 1$ graphs exchangeable, so under a seed fixed across graphs the scores are case (a) of Lemma 2 and under i.i.d. per-graph seeds they are case (b); either way Lemma 2 applies and (4) is valid. Nothing in that argument constrains what the recipe does internally, which is why non-convergence, collapse and a truncated budget cannot affect the bound. □

Corollary 2 (Hub-conditional statistics stay exact)**.** *Assume H_0 and* (2)*. Let $T : \mathcal{G}_d \times \mathcal{G}_d \to \mathbb{R}$ be measurable and set $S_0 = T(G, G^*)$, $S_i = T(G_i, G^*)$. Then* (4) *satisfies $\Pr[p \leq a] \leq a$ for all $a \in [0, 1]$. The same holds for randomized T under either condition of Lemma 2, with the auxiliary randomness independent of $(G, G^*, G_1, \dots, G_M)$.*

Proof of Corollary 2. Condition on $G^* = y$. By Lemma 1 the graphs $G, G_1, \ldots, G_M$ are i.i.d., hence exchangeable, and $T(\cdot, y)$ is a single fixed measurable map applied to every coordinate; Lemma 2 applied conditionally gives $\Pr[p \le a \mid G^* = y] \le a$ for every y. Integrate over the law of G^*. □

Proof of Proposition 1. Let D_k be the event that v_k is tested, is truly structureless, and is split, and let $B_k = \bigcap_{j \le k} D_j^c$. The event that some false split occurs is the disjoint union over k of $B_{k-1} \cap D_k$, the event that the *first* false split happens at step k, so it suffices to bound $\sum_k \Pr[B_{k-1} \cap D_k]$.

Fix k and work conditionally on $\mathcal{F}_{k-1}$. On the event that v_k is tested and structureless, the assumption places $G[V_{v_k}]$ uniformly on its own fiber given $\mathcal{F}_{k-1}$ and draws its null samples by (2), so Lemma 1 applies within that fiber and each battery p-value is conditionally valid by Lemma 2. Each of the $|\mathcal{S}|$ statistics therefore falls below $\alpha_{v_k}/|\mathcal{S}|$ with conditional probability at most $\alpha_{v_k}/|\mathcal{S}|$, and a union bound over that fixed finite battery gives $\Pr[D_k \mid \mathcal{F}_{k-1}] \le \alpha_{v_k}$ on the event in question. The indicator of B_{k-1}, the indicator that v_k is tested and structureless, and α_{v_k} are all $\mathcal{F}_{k-1}$-measurable (the truth is fixed, so being truly structureless is a property of the vertex set, and whether v_k is tested depends only on earlier outcomes, n_{v_k} and its edge count), so

$$\Pr[B_{k-1} \cap D_k] \le \mathbb{E}\big[\mathbf{1}\{B_{k-1}\}\, \mathbf{1}\{v_k \text{ tested, structureless}\}\, \alpha_{v_k}\big].$$

Summing over k and writing K for the index of the first false split (with $K = \infty$ if there is none), so that B_{k-1} holds exactly for $k \le K$,

$$\Pr[\text{some false split}] \le \mathbb{E}\Big[\textstyle\sum_{k \le K} \mathbf{1}\{v_k \text{ tested, structureless}\}\, \alpha_{v_k}\Big].$$

The remaining step is pathwise. On any realization, consider the structureless nodes tested at steps $k \le K$. All of them except possibly v_K were not split, since a split would have been an earlier false split, and so have no descendants in the recursion tree, and v_K has none among them because it is tested last. Two nodes of the tree intersect only if one is an ancestor of the other, since the children of a node partition it, so these vertex sets are pairwise disjoint and their sizes sum to at most n. The sum inside the expectation is therefore at most $\alpha \sum_k n_{v_k}/n \le \alpha$ on every realization, and the claim follows. □

B WRAPPED PIPELINES, TRAINING BUDGETS, GENERATORS, AND DATASETS

Pipelines and training budgets. These are lite reimplementations: faithful to the originals' objectives and optimization shape, but not the authors' released code, so no number in this paper should be read as a benchmark of the published systems. What is under test is the certification layer, which treats each pipeline as a black box, so any fixed implementation serves. The training budget is part of the fixed rule A, not a tuning knob: 200 epochs for DMoN-lite and MAGI-lite, 400 for MinCut-lite, which below that budget sits in its degenerate one-cluster optimum on every graph; one hard-coded initialization seed shared by every graph in the tuple; $K_{\max} = 16$ for DMoN-lite and MAGI-lite and $K_{\max} = 8$ for MinCut-lite, whose orthogonality term spreads mass across all available slots. These cluster budgets are chosen by us in advance and held fixed, while Leiden selects its own number of parts; we did not vary them, and where that choice could be doing work in a comparison we say so in Section 4. Features on synthetic graphs are the top-16 truncated SVD of the adjacency matrix, sign-canonicalized so they are a function of the graph and not of the solver. Of the real graphs only Cora uses its native attributes, held fixed across the tuple, which is exactly the null the test states; Citeseer and Amazon Photo use the same per-graph SVD recipe as the synthetic graphs.

Generators and test parameters. The erased configuration model is not uniform given degrees, so every heavy-tailed draw passes through a long swap burn-in before it is used as an H_0 draw; without that step a calibration study measures the generator's bias rather than the test's level (Fosdick et al., 2018). A finite burn-in reduces that non-uniformity on the fiber without eliminating it, so of the nine setting-by-pipeline cells of Table 3 the three Erdős-Rényi cells are exact-null draws while the six heavy-tailed cells are approximate-null at the burn-in length we use, and any residual miscalibration in those six is attributable to the generator rather than to the test. The level claim itself is untouched by this: level is the content of Lemmas 1 and 2 and holds whenever the graph handed to the wrapper is an H_0 draw, uniform on its fiber, so it is a theorem about such draws rather than an empirical

finding; the exact-null cells check it directly, and the heavy-tailed cells only up to the burn-in's residual non-uniformity. Unless stated otherwise the chain runs at 2.0 swaps per edge with $M = 49$ null samples per test ($M = 199$ for the calibration audit and the sequential-stopping study), each sample a full training run, at $\alpha = 0.05$; on real graphs M is stratified by size (199, 99, 49) and seeds are derived arithmetically per job.

Battery size and the per-node sample size. The rule $M_v = \max(99, \lceil 2/\alpha_v \rceil)$ of Appendix F keeps the smallest attainable p-value $1/(M_v + 1)$ below a node's rejection threshold $\alpha_v/|\mathcal{S}|$ for $|\mathcal{S}| \leq 2$, which covers every battery we run, and no further. A general battery needs $M_v \geq \lceil |\mathcal{S}|/\alpha_v \rceil$, and $M_v = \max(99, \lceil 2|\mathcal{S}|/\alpha_v \rceil)$ keeps the same factor-two margin, since the threshold falls with $|\mathcal{S}|$ while the attainable floor does not.

What the reported total counts. The total of 6145 logged runs counts runs of a wrapped clustering pipeline, on null and planted graphs alike, with each peel-and-merge run counted once however many node tests it makes; Section 4.2 is outside it on both of its arms. Its fixed-threshold decisions are recomputations on graphs already drawn for the other runs or on separately drawn dense controls, and cost no fits. Its same-statistic arm is not a recomputation at all but 180 freshly run exact tests; they are excluded because the rule they wrap is a raw eigenvalue rather than a clustering pipeline, not because they were cheap.

Real graphs. The 7 graphs of Section 4.5 are Karate (Zachary, 1977), Dolphins (Lusseau et al., 2003), Football (Girvan & Newman, 2002) and PolBooks (V. Krebs, unpublished, distributed with Newman's network data), four small graphs in long-standing use as community detection benchmarks and taken here from their standard public copies, Cora and Citeseer (Sen et al., 2008; Yang et al., 2016) in their standard public releases, and Amazon Photo (Shchur et al., 2018). Each is reduced to the largest connected component of its simple undirected version.

C THE ASYMPTOTIC BASELINE, PER SETTING (E1B)

The full result. The fixed-threshold decision rejects 1.000 of the plain sparse Erdős-Rényi graphs at 200 replicates. Pooled with the two heavy-tailed settings below, the rate over all 600 graphs is 1.000, Wilson 95% $[0.994, 1.000]$; that pooled figure is the least informative and we quote it only because it is the one a reader would otherwise compute. The Erdős-Rényi rejections are not marginal: there the statistic has median 7.7 and minimum 3.8 against a threshold of 0.9793; on the heavy-tailed settings it never falls below 355.

Design and the two controls. We recompute the fixed-threshold decision on exactly the structureless graphs of Section 4.1, regenerated from the same seeds, so the two procedures see identical draws and the comparison is paired rather than a comparison of independent samples. Two controls run alongside it, and they are what make the comparison interpretable, because the fixed-threshold rule and the exact wrapper differ in their statistic, a leading eigenvalue against the wrapped pipelines' modularity, and in their null, a fixed asymptotic reference against a degree-conditioned resampling one, so either difference could carry the result on its own. The first control fixes the statistic: wrapping the same largest eigenvalue in the exact test, on 180 of those sparse structureless graphs (the first 60 replicates of each setting), rejects 0.039, Wilson 95% $[0.019, 0.078]$, while the fixed threshold applied to that same statistic on those same graphs rejects 1.000. Same statistic, same graphs, opposite behavior, so what separates the two is the procedure and not the choice of statistic. On the 60 Erdős-Rényi graphs among them, the only ones where the threshold's rejections are type-I errors under its own hypothesis, the exact wrapper rejects 0.033 and the threshold 1.000. The second control fixes the regime: on dense Erdős-Rényi graphs, the competitor's own null in the fixed-edge-probability regime its limit is proved for, the fixed threshold is consistent with its level, rejecting 0.017 at $n = 1000$ with mean degree 200, Wilson $[0.003, 0.089]$, and 0.083 at $n = 2000$ with mean degree 300, Wilson $[0.036, 0.181]$, at 60 replicates each. The in-regime arm draws its own graphs and records a threshold decision only; we do not pair it with an exact rate, because we ran no wrapped test on those dense graphs.

The heavy-tailed settings. Section 4.2 reports the Erdős-Rényi rejection rate of the fixed-threshold decision, which is the only one of its four numbers that is a type-I error rate under the competitor's

own hypothesis, together with the pooled rate. The two heavy-tailed settings are recorded here for completeness, subordinated because a power-law configuration model is not an Erdős-Rényi graph, so the competitor's null is false there by construction and its rejections are not type-I errors under its own hypothesis. The rate is 1.000 at $\tau = 2.5$ on 200 replicates and 1.000 at $\tau = 2.1$ on 200. Those rejections still say something, since degree heterogeneity alone is not the community structure a practitioner asks about, but they are not calibration evidence and we do not use them as such.

D Power and recovery, both patterns (E3)

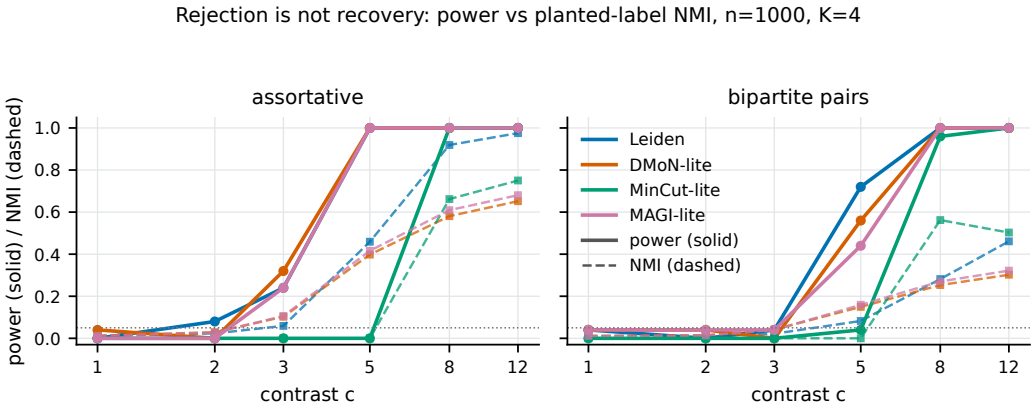

Figure 2: Power and recovery come apart (E3, 1200 tests). Solid: rejection rate at $\alpha = 0.05$ against contrast c. Dashed: mean NMI against the planted labels. Left, assortative; right, bipartite pairs. Every deep pipeline runs at the over-provisioned cluster budget fixed in advance (Appendix Q).

On bipartite pairs every wrapped algorithm fires by $c = 8$, at 25 replicates per cell: power is 1.00 for Leiden, 1.00 for DMoN-lite, 1.00 for MAGI-lite and 0.96 for MinCut-lite. Three of the four fire through partitions that merge each disassortative pair into one part, so achieved modularity rises while NMI against the planting stays low: 0.28 for Leiden, 0.25 for DMoN-lite and 0.27 for MAGI-lite. MinCut-lite is the exception, at NMI 0.56 in the same cell, roughly twice the other three, which is what its normalized-cut objective is expected to do on a planting the modularity-driven wrappers cannot express. It buys that at slightly lower power, and the comparison is one cell at 25 replicates, so we state it as measured and do not generalize it beyond this pattern and contrast.

E Calibration audit, cell by cell (E1)

Table 3: Calibration audit (E1). Rejection rate with a Wilson 95% interval at three nominal levels, per structureless setting and wrapped pipeline, at 200 replicates per cell. With $M = 199$ null samples the smallest attainable p-value is 0.005, so all three levels shown lie on the attainable grid.

setting	pipeline	rej. at α=0.01	rej. at α=0.05	rej. at α=0.10
ER	DMoN-lite	0.000 [0.000, 0.019]	0.040 [0.020, 0.077]	0.080 [0.050, 0.126]
	MinCut-lite	0.005 [0.001, 0.028]	0.005 [0.001, 0.028]	0.005 [0.001, 0.028]
	MAGI-lite	0.010 [0.003, 0.036]	0.065 [0.038, 0.108]	0.120 [0.082, 0.172]
power law τ=2.5	DMoN-lite	0.005 [0.001, 0.028]	0.055 [0.031, 0.096]	0.120 [0.082, 0.172]
	MinCut-lite	0.010 [0.003, 0.036]	0.060 [0.035, 0.102]	0.105 [0.070, 0.155]
	MAGI-lite	0.010 [0.003, 0.036]	0.030 [0.014, 0.064]	0.080 [0.050, 0.126]
power law τ=2.1	DMoN-lite	0.010 [0.003, 0.036]	0.065 [0.038, 0.108]	0.145 [0.103, 0.200]
	MinCut-lite	0.005 [0.001, 0.028]	0.045 [0.024, 0.083]	0.080 [0.050, 0.126]
	MAGI-lite	0.005 [0.001, 0.028]	0.040 [0.020, 0.077]	0.095 [0.062, 0.144]

Table 3 gives the per-cell rejection rates behind the level claim of Section 4.1, at three nominal levels, with Wilson 95% intervals. Read across the Erdős-Rényi, $\tau = 2.5$ and $\tau = 2.1$ families at $\alpha = 0.05$, the rates are 0.040 (Wilson $[0.020, 0.077]$), 0.055 and 0.065 for DMoN-lite; 0.005, 0.060 and 0.045 for MinCut-lite; and 0.065, 0.030 and 0.040 for MAGI-lite, whose arms were rerun in full after the invariance fix of Appendix R.

F THE CERTIFICATION PROCEDURE AND THE PEEL-AND-MERGE PROCEDURE IN FULL

Algorithm 1: Exact certification of a trained pipeline

Input: graph G; pipeline $\mathcal{A}$ with training recipe, seeding rule and cluster budget fixed before G is examined; score S; rate ρ; sample size M; level α.

1. Read d off G and set $L = \max(1, \mathrm{round}(\rho m))$.
2. Backward run: $G^* = \phi(G, W_0)$, that is L proposed swaps from G.
3. Forward runs: $G_i = \phi(G^*, W_i)$, $i = 1, \ldots, M$, on independent randomness.
4. Canonicalize each graph's edge array, hiding the order it was produced in.
5. Fit $\mathcal{A}$ once on each of $G, G_1, \ldots, G_M$ under the fixed seeding rule and score its partition, giving $S_0, \ldots, S_M$.
6. Count exceedances $C = \big|\{i \geq 1 : S_i \geq S_0\}\big|$, ties included.
7. Report $p = (1 + C)/(M + 1)$; certify structure when $p \leq \alpha$.

Section 3.6 summarizes the recursion that Proposition 1 and Appendices O and P measure; this is the full statement, carried over from Anonymous (2026). Peel-and-merge runs top down from the root. A node v with vertex set V_v of size n_v carries the local null that $G[V_v]$, relabeled to $\{1, \ldots, n_v\}$, is uniform on the fiber of its own degree sequence. Node v is tested at level $\alpha_v = \alpha\, n_v/n$ with a battery $\mathcal{S}$ of statistics evaluated on one shared set of null samples, and H_v is rejected when the smallest battery p-value is at most $\alpha_v/|\mathcal{S}|$, a Bonferroni correction inside the node. The sample size is $M_v = \max(99, \lceil 2/\alpha_v \rceil)$, so that the smallest attainable p-value $1/(M_v + 1)$ stays below the node's threshold; without this a small node could never reject however strong its structure. As written that rule delivers the property for $|\mathcal{S}| \leq 2$, which covers every battery we run, and no further; a general battery needs $M_v \geq \lceil |\mathcal{S}|/\alpha_v \rceil$, and $M_v = \max(99, \lceil 2|\mathcal{S}|/\alpha_v \rceil)$ keeps the same factor-two margin, since the threshold falls with $|\mathcal{S}|$ while the attainable floor does not. If H_v is not rejected, if $n_v < n_{\min}$, or if the partition has a single part, the node is a leaf; otherwise it is split into the parts of the wrapped algorithm's partition of $G[V_v]$ and the procedure recurses. A bottom-up merge pass then visits each parent, tests each sibling-pair union at the union's own level, and while some union fails to reject merges the pair with the largest p-value and retests. Then $\hat{K}$ is the number of leaves. The merge pass is a power device and carries no part of the guarantee: it can only reduce the number of leaves, so it can never create a false split, and it is included because the wrapped algorithm's partition is typically finer than the structure the test can certify. Every run uses $\alpha = 0.05$ and $n_{\min} = 30$.

G MINCUT-LITE'S COLLAPSE ON STRUCTURELESS GRAPHS (E1)

MinCut-lite behaves differently from the other two deep wrappers on structureless graphs, and the difference is worth stating rather than idealizing. On Erdős-Rényi graphs the observed graph collapses to a single cluster in all but 1 of the 200 tests, a partition whose modularity is zero by definition, and the null graphs usually collapse with it, so the scores tie across the tuple: 58 of 200 tests return $p = 1$, which requires every null sample to score at least as high as the observed one. The smallest p-value anywhere in that cell is 0.005, from the 1 test where the wrapper escaped the degenerate optimum and scored strictly above its whole null tuple; that test supplies the cell's entire rejection rate 0.005. Because ties are counted against rejection the collapse is absorbed gracefully: the cell is conservative and still valid, not broken. At $\tau = 2.1$ the same pipeline un-collapses and reports a modal 3 clusters on pure noise, while at $\tau = 2.5$ it still returns a single cluster on a fraction 0.745 of graphs, a second flavor of the same behavior. Neither threatens validity, as Corollary 1 guarantees, and both cost power here, which it neither predicts nor rules out.

H AMORTIZATION COST, MODE BY MODE (E4)

Table 4: Amortization modes (E4, DMoN-lite): validity, level on 60 null graphs, power at two contrasts, mean cost. Only the last row is not a valid test.

mode	valid	type-I [Wilson 95%]	power c=5	power c=8	s/test
full fit, 200 ep	yes	0.08 [0.04, 0.18]	1.00	1.00	9.7
surrogate, 50 ep	yes	0.10 [0.05, 0.20]	1.00	1.00	3.6
hub pretrain, ft 25	yes	0.05 [0.02, 0.14]	0.08	0.36	2.7
hub pretrain, ft 50	yes	0.05 [0.02, 0.14]	0.40	0.56	3.7
hub pretrain, ft 100	yes	0.08 [0.04, 0.18]	0.76	0.96	5.8
observed warm-start, ft 50	**no**	1.00 [0.94, 1.00]	1.00	1.00	3.8

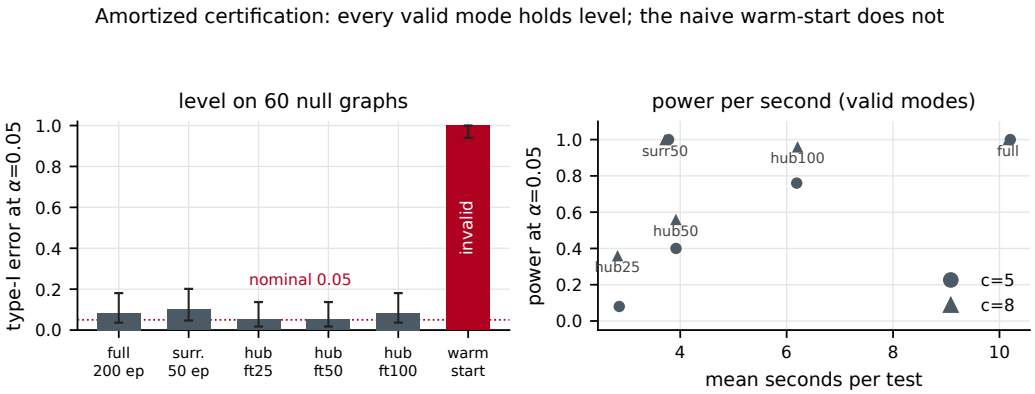

Figure 3: Amortization (E4, 810 tests, DMoN-lite). Left: type-I error on 60 null graphs per mode; every exact mode covers the nominal level and the observed-graph warm-start rejects every null. Right: power against mean seconds per test, exact modes only, at $c = 5$ (circles) and $c = 8$ (triangles).

Section 4.4 reports that hub amortization does not pay at this scale; the per-mode numbers are these. At contrast $c = 5$ hub pretraining reaches power 0.40 at 3.7 seconds per test with 50 fine-tuning epochs and 0.76 at 5.8 seconds with 100, while the reduced-epoch cold-start surrogate reaches 1.00 at 3.6 seconds, comparable cost for strictly more power, matching the full fit at 9.7 seconds. The practical recommendation at this scale is therefore the cheap surrogate: amortize by training less, not by warm-starting. The balance may shift at larger scale, where one pretraining pass dominates the per-fit cost and a short fine-tune is the only affordable option; we have not run that regime and do not claim it.

I LIMITATIONS, IN FULL

Section 5 lists the limitations in compact form; this is the same list with the numbers and the reasoning attached. Nothing here is new: every figure quoted is reported in the section it belongs to.

- The wrapped deep methods are compact reimplementations, faithful to the published objectives of DMoN, MinCutPool and MAGI (Tsitsulin et al., 2023; Bianchi et al., 2020; Liu et al., 2024) but not the authors' original code; absolute scores are not comparable to published numbers.
- The comparison against a fixed-threshold asymptotic decision (Section 4.2) is the most contestable result in the paper. The competitor is applied outside the regime it is stated for, since its threshold comes from a limit proved for fixed edge probability, that is, mean degree growing linearly in n (Bickel & Sarkar, 2016), and we apply it to sparse graphs chosen to match practice, so a high rejection rate there is misapplication and not evidence against the theorem (Bickel & Sarkar, 2016). It also differs from the exact rates in statistic and

in null at once, which is why the two controls are what license the conclusion: the same eigenvalue wrapped in the exact test on the same graphs rejects 0.039 against 1.000 for the threshold, and on dense graphs the threshold is consistent with its level. Bickel and Sarkar recommend a simulation-based shift-and-scale (Bartlett-type) correction for small or sparse graphs; we evaluate the uncorrected fixed-quantile threshold, and that correction is itself a parametric bootstrap under a fitted Erdős-Rényi null, not the degree-conditioned null used here. We claim only that a fixed threshold gives no signal that its regime has been left, and nothing about the asymptotic theory itself, a corrected version of it, or the other asymptotic procedures we cite but did not run.

- Everything runs on one CPU machine and the largest graph tested has thousands of nodes; scaling to millions is untested.
- A wrapped rule must be invariant to the representation of the graph, not merely deterministic given a seed. Our contrastive pipeline failed this until we corrected it, so any result produced before that fix would not have been covered by Lemma 2. We report only post-fix runs for it, and we did not audit invariance beyond edge order and the fixed feature recipe.
- Model order fails on disassortative structure, structurally rather than incidentally. Against a known planted K (Appendix P) peel-and-merge is competitive on assortative structure and has the best pooled mean absolute error point estimate of the five, 1.59 against 1.72 for the Bethe-Hessian count (Saade et al., 2014; Le & Levina, 2022), whose interval it overlaps, so the two are not distinguishable at this sample size; both columns are read against what a constant answer attains, 0.333 on exact-match and 2.00 on error, a yardstick applied to our own estimator as well as the baselines, and on exact-match ours clears that floor by less than the width of its interval. On bipartite pairs it matches only 0.029, and both halves of the recursion fail there while only one is the splitter: on the two-block planting the battery mostly does not reject at the node's allocated level, so the recursion never reaches a split, and where it does reject a modularity-driven splitter merges each disassortative pair rather than separating it, so 0.988 of those graphs report half the planted count. A splitter matched to the pattern would repair the second failure and not the first. The scope is narrower still, in a way that matters in a paper about deep pipelines: every model-order result we report runs peel-and-merge with the classical Leiden wrapper and not with a deep pipeline, because the recursion pays a full test at every node and a deep wrapper is unaffordable at that price (Section 3.6), so the split rule is not ablated and what the results establish is a property of the recursion demonstrated with a classical splitter. What transfers across patterns is the false-split control of Proposition 1, under its selection-neutrality condition; on structureless graphs, where any split starts at the root, it is measured at 0.022 of 600 null graphs against a nominal 0.05.
- The cluster budget of each deep pipeline is fixed by us in advance (Appendix Q). Power is insensitive to it wherever power is saturated ($c \geq 5$), but recovery moves a long way, and much of the recovery gap between classical and deep wrappers reported in Section 4.3 is an artifact of the budget rather than of the objectives; the Karate contrast moves the same way. $K_{\max}$ should be treated as a modeling choice the certificate is conditional on, reported alongside it, and varied, with a correction for the budgets tried, before a retention is read as a statement about the graph. We ablated it for DMoN-lite and MAGI-lite on one synthetic pattern and one real graph, not for MinCut-lite and not across the rest of the grid.
- The recursion cannot resolve structure below its minimum node size, an honest resolution limit that $\hat{K}$ must be read against. It binds hardest on the largest real graphs, where the average leaf falls below that minimum, so many counted leaves were never tested and those model orders are upper bounds on what was certified rather than certified counts (Section 4.5, Appendix T.1).
- Threshold-regime power estimates are noisy at the replication counts we could afford, 25 per cell on the power grid, and should be read as indicative. The level results, which carry the guarantees, are unaffected.
- Hub amortization is exact, but a reduced-epoch cold-start surrogate dominates it on power per second at this scale (Table 4). Across the chain lengths we ran its power spans only 0.32 to 0.44 at 25 replicates per cell, no evidence for the intuition that a short chain helps and too coarse to have resolved a modest trend, and level over those chain lengths is consistent with α in three of the four cells and not in the fourth, which we flag rather than average away (Appendix L). The contribution is the validity boundary it marks, since the naive sibling pretrained on the observed graph rejects 1.00 of null graphs.

- We implement the classical sequential rule of Besag & Clifford (1991); the anytime-valid betting scheme of Fischer & Ramdas (2025) is not implemented.
- Attribute permutation, testing whether attributes rather than wiring carry the structure, is deliberately out of scope; our null holds attributes fixed.

J AGGREGATION RULES THE LEMMA LICENSES AND FORBIDS

Corollary 1 says training pathologies bear on power and not on level. Aggregation is licensed on the same grounds: a median over a fixed seed list is again a fixed function of the graph, case (a) of Lemma 2, and a median over i.i.d. per-graph seeds is case (b). What is *not* covered is auxiliary randomness whose law depends on which coordinate holds the observed graph. Training the observed graph longer than the null graphs, taking the best of several seeds on it alone, or tuning the cluster budget by looking at it all break the permutation invariance of Step 1 of the proof. We therefore fix architecture, cluster budget, learning rate and epoch count in advance and apply them unchanged to all $M + 1$ graphs. On synthetic graphs the input features are recomputed from each graph by one fixed recipe, again a fixed function of that coordinate; where a real graph is tested with its own attributes those are held fixed and enter as part of the constant u of case (a), and the remaining real graphs use the same recipe as the synthetic ones.

K COST MODEL

One test costs $M + 1$ pipeline fits plus $M + 1$ runs of the chain, each of $L = \max(1, \mathrm{round}(\rho m))$ proposals, so with c_{fit} the cost of a fit and c_{swap} the cost of a proposal a fixed-budget test costs about $(M + 1)(c_{\mathrm{fit}} + c_{\mathrm{swap}} L)$, plus one pretraining fit in the amortized mode. The chain term is linear in m and independent of the pipeline, so it dominates only for very cheap wrapped algorithms. The null sample is shared: a battery of k statistics costs k evaluations per graph but only one chain, and each p-value is individually exact, with Bonferroni combining them when a single decision is required.

The two efficiency contributions attack the two factors separately. Hub-conditional amortization (Section 3.4) reduces c_{fit} by replacing $M + 1$ full fits with one pretraining fit and $M + 1$ short fine-tunes; per-test seconds by mode are in Table 4. Sequential stopping (Section 3.5) reduces the effective M: on structureless graphs the median test uses 19 of 199 available fits and takes 10 seconds on average, against 44 seconds at the planted contrast $c = 5$, where the full budget was spent every time. Model order costs substantially more than existence, since peel-and-merge pays $\sum_v (M_v + 1)$ fits over all tested nodes rather than $M + 1$ fits once, and the two are reported separately in Table 1: across our real-graph suite the most expensive existence certificate took 565 seconds and the most expensive model-order certificate 7948 seconds, about 2.2 hours, more than an order of magnitude above it. Anyone budgeting for $\hat{K}$ should budget from the second number.

L CHAIN LENGTH AND HUB AMORTIZATION (E4B)

The natural rescue for the power deficit of hub amortization is to shorten the chain, so that the hub sits near the observed graph and its optimum, and it does not work either. Varying the chain length over 0.25, 0.5, 1.0 and 2.0 swaps per edge with everything else held fixed, hub power at $c = 5$ stays between 0.32 and 0.44 across the four settings at 25 replicates per cell. We detect no trend, at a replication count that could not have resolved a modest one, which is weaker than the refutation of the small-chain intuition we expected to be able to state. The mechanism we expect is the same symmetry that makes the scheme valid, since the null graphs are drawn from the hub by the identical mechanism and inherit the identical head start.

Level in that ablation, on 40 null graphs per chain length, is 0.000 $[0.000, 0.088]$, 0.025 $[0.004, 0.129]$, 0.125 $[0.055, 0.261]$ and 0.025 $[0.004, 0.129]$ with increasing chain length. Of the four intervals, three cover the nominal level and one, the worst cell at 0.125, does not. We flag that rather than average it away, and we do not read it as evidence against exactness, which Lemma 1 supplies at any chain length: at 40 replicates per cell an excursion of this size across four cells is ordinary Monte Carlo noise. The level estimates that carry weight are the ones with larger replication, in Table 4 and Table 3.

Table 5: Seeding rules (E6, DMoN-lite): all three are exact by Lemma 2; the difference is power near the detectability edge.

seed rule	type-I	power c=3	power c=5	cost
fixed seed	0.03	0.48	1.00	1×
fresh i.i.d. seeds	0.03	0.32	1.00	1×
median of 3 seeds	0.00	0.64	1.00	3×

M SEQUENTIAL STOPPING (E5)

Question. Each null sample is a full training run, so the fixed budget is the dominant expense; how much of it can be recovered when the evidence is already hopeless? **Design.** The Besag-Clifford rule (Besag & Clifford, 1991) draws fits until h exceedances or M samples and reports h/ℓ if the hth exceedance arrives at sample ℓ, which is valid at the data-dependent stopping time; we ran $h = 10$, $M = 199$, DMoN-lite, on structureless graphs and at $c = 5$.

Level on 200 structureless graphs is 0.045, Wilson 95% $[0.024, 0.083]$, and power at $c = 5$ is 1.00, where the rule correctly never stops early.

One observation makes the level estimate quoted in Section 4 easier to read. At the triple we ran, $\alpha M < h \leq \alpha(M + 1)$, and in that range the sequential decision coincides with the fixed-budget one: an early stop means at least h exceedances occurred, so the fixed-M p-value is above α, while no early stop means fewer than h exceedances in M samples, so the fixed-M p-value is at most α. Early stopping therefore cannot change any rejection reported in this paper, and the level reported there is equally the fixed-budget estimate at 200 replicates, carrying no sequential-specific inflation. Disagreement between the sequential and the fixed-budget decision is therefore impossible at this (h, M, α) and becomes possible only when $h \leq \alpha M$; this is not the resampling risk of Gandy (2009), which compares a Monte Carlo decision with the one the exact p-value would give and is not uniformly bounded for any bounded-budget rule, ours included. Testing noise is cheap; testing structure pays full price, and at $c = 5$ it rejected every time. The modern upgrade path is anytime-valid testing by betting (Fischer & Ramdas, 2025); we have not implemented it and report no numbers for it.

N SEEDING RULES AND TRAINING STOCHASTICITY (E6)

Question. Deep clustering is unstable across seeds (Wu et al., 2025; Leeney & McConville, 2024), and Corollary 1 says instability cannot cost level. Does that match? **Design.** Three rules wrapping DMoN-lite: one fixed seed shared by every graph; an i.i.d. seed per graph from a stream independent of the chain; and the median score over three fixed seeds, again a fixed rule at three times the compute. Level on 40 structureless graphs, power at contrasts 3 and 5 (Table 5).

Answer. All three hold level: 0.03 fixed, 0.03 fresh, 0.00 median-of-three. They separate only at the detectability edge, where the median rule reaches power 0.64 against 0.48 fixed and 0.32 fresh; by contrast 5 all three saturate. Aggregating over seeds reduces score variance and sharpens the ranking of the observed graph against the nulls, and fresh per-graph randomness does the opposite. Use a fixed seed by default. Where a study is expected to sit near the detectability edge, choose the median-of-three rule before seeing any data, where the extra factor of three in compute buys power; switching to it after seeing a near-threshold fixed-seed p-value is a data-dependent choice between two tests that Lemma 2 does not cover.

One caveat belongs in the text and not a footnote, and it cuts against the precision of the effect just reported. Threshold-regime power is noisy at the replication we could afford, 25 replicates per cell. The three nominally similar contrast-3 cells in E3, E4 and E6 use different graph draws and span 0.20 to 0.48, a spread comparable in size to the 0.32 to 0.64 range that separates the three seed rules themselves. The ordering of the rules is therefore indicative rather than a precisely estimated effect; at least 100 replicates in that regime is the right fix and we have not run it.

O FALSE SPLITS ON STRUCTURELESS GRAPHS (E8)

Question. Proposition 1 bounds by α the probability that peel-and-merge splits any truly structureless node, but its proof consumes a selection-neutrality assumption that we state rather than prove, because the recursion chooses the vertex sets it tests using the data. The proof is sound given that assumption and says nothing without it. On structureless graphs the whole vertex set is one structureless node and so is every node the recursion reaches inside it, so *any* split anywhere in the recursion is a false split, and any split requires a root split first. The fraction of graphs receiving at least one split is therefore the level of the root battery test, which is at most α by Lemmas 1 and 2 and Bonferroni without the assumption (Appendix W); this study measures that level, not selection neutrality.

Design. 600 structureless graphs, each certified by Leiden-wrapped peel-and-merge at $\alpha = 0.05$ with the modularity and Bethe-Hessian battery and the default minimum node size, in three settings of 200 replicates each: Erdős-Rényi at $n = 1000$, a heavy-tailed configuration model at $\tau = 2.5$ and $n = 1000$, and a smaller Erdős-Rényi setting at $n = 300$ with mean degree 8. Every draw passes through the swap burn-in used everywhere else in this paper, so what is measured is the level of the procedure and not the bias of the generator. We log whether the top-down pass split the graph at all, which on these graphs is the event that the root rejects, alongside the model order the full procedure finally reports.

Answer. Every point estimate is below the bound, and the pooled interval lies entirely below it. Of the 600 graphs, 13 received at least one split, a rate of 0.022 with Wilson 95% interval $[0.013, 0.037]$, an interval lying entirely below the nominal 0.05. Per setting, at 200 replicates each, the rates are 0.010 $[0.003, 0.036]$ for Erdős-Rényi at $n = 1000$, 0.030 $[0.014, 0.064]$ for the heavy-tailed configuration model, and 0.025 $[0.011, 0.057]$ for the smaller Erdős-Rényi setting. Every point estimate is below 0.05, and the pooled interval excludes it; the two per-setting intervals for the heavy-tailed and small-n settings do extend slightly above 0.05, which is the resolution 200 replicates buy and not a sign of anticonservatism.

The merge pass is not what makes this work. A secondary observation about these runs deserves recording and is easy to overread. The top-down pass produced a split on 13 graphs, and after the merge pass 0 graphs had model order above one, so every one of those splits was undone and the reported model order was one on all 600 null graphs. The quantity Proposition 1 bounds is the top-down split rate, which is the rate reported above, and the merge pass carries no part of that argument or of the root test's level. It is a power device. It can only reduce the number of leaves, so it can never create a false split, but it equally cannot be relied on to remove one: a merge pass cannot repair a validity failure in general, and that it repaired every false split in these 600 runs is an observation about these runs, not a guarantee. The pooled rate 0.022 is the check on the root test; the post-merge count is a bonus on top of it and carries no guarantee of its own.

What this settles and what it does not. These runs do not test selection neutrality, and no number of them could, since the headline rate is the root test's level. The only direct evidence on the assumption here is weak: in the 13 graphs whose root split, the recursion went on to test data-selected children, all structureless, and 0 of those graphs had a further split. The study covers three structureless regimes with one splitter, one wrapped algorithm and one battery. It also speaks only to validity: nothing here measures whether $\hat{K}$ recovers a block count when structure is present, which is the separate question Appendix P takes up.

P MODEL ORDER AGAINST A PLANTED BLOCK COUNT (E9)

How accurate is the recursion when K is known? This study replicates the accuracy design of Anonymous (2026) at fresh seeds in our harness, with a constant-answer yardstick added. 480 degree-corrected block models in a recovery-feasible regime, $K \in \{2, 4, 8\}$, contrasts $\{3, 5, 8, 12\}$, 20 replicates per cell in both patterns, scored against Leiden-wrapped peel-and-merge and four baselines: the Bethe-Hessian negative eigenvalue count (Saade et al., 2014; Le & Levina, 2022), the silhouette maximum, Leiden's part count, and the normalized Laplacian eigengap (Table 6). Two disclosures. Every model-order result here wraps Leiden and not a deep pipeline (Section 3.6). And the Bethe-Hessian spectrum sits in our battery and in the strongest baseline, so their near-tie on the assortative pattern reads partly shared information twice (Appendix P.3).

Table 6: Model order against a planted block count (E9, 480 graphs). Exact-match rate against the planted K and mean absolute error, pooled and split by mixing pattern, for peel-and-merge and four standard estimators. Higher exact-match and lower error are better. Both columns should be read against what a constant answer attains on this grid, 0.333 and 2.00, discussed in the text for every row including ours.

estimator	pooled		assortative		bipartite pairs	
	exact	MAE	exact	MAE	exact	MAE
peel-and-merge	0.404	1.59	0.779	0.78	0.029	2.40
Bethe-Hessian count	0.398	1.72	0.796	0.81	0.000	2.62
silhouette max	0.406	2.38	0.479	2.16	0.333	2.59
Leiden's own K	0.360	2.56	0.675	0.98	0.046	4.14
Laplacian eigengap	0.125	3.42	0.250	3.25	0.000	3.58

Competitive on assortative structure, not dominant. Intervals are bootstrap over 2000 resamples; pooled exact-match does not separate the top four. On mean absolute error peel-and-merge has the best point estimate, $[1.41, 1.77]$ against $[1.54, 1.90]$ for the Bethe-Hessian count, but those intervals overlap over most of their length, so the leading pair is not distinguishable here. Both columns must also be read against a constant answer, which exact-match on this grid rewards and which attains 0.333 exact-match and 2.00 error: on that yardstick, our own row included, peel-and-merge clears the exact-match floor by less than its own interval width, and on error it and the Bethe-Hessian count are the only two that beat a constant (Appendix P.1).

On bipartite pairs it fails, for two distinct reasons. Peel-and-merge almost never matches the planted K on those graphs, and no estimator does well, the best being the silhouette maximum (Table 6). The first is detection: on the two-block planting the battery mostly does not reject at the root, so no split is reached on 0.775 of those graphs, and not only at weak contrast. The second is division, and appears only where detection succeeds: at four and eight planted blocks with strong contrast the pass fires on 1.000 of those graphs, and 0.988 of them report exactly $K/2$, the pair-merging signature of a modularity-driven splitter. Neither is repaired by a different p-value: the first needs a battery statistic that fires on a purely disassortative two-block graph, the second a splitter matched to the structure; we ran neither (Appendix P.2).

P.1 What the baselines return, and the constant-answer yardstick

This section expands the reading of Table 6 that Appendix P states in brief: what the four baseline estimators actually return on this grid, and how every row of that table fares against the best a constant answer achieves.

What the baselines are actually doing, which qualifies the fact that the silhouette maximum edges peel-and-merge on pooled exact-match (Table 6). Two of the four are close to constant on this grid. The silhouette maximum returns 2 on 0.890 of the 480 graphs, and the eigengap returns 1 on 0.833 of them, almost regardless of the planted K, the contrast or the mixing pattern. That matters for how the pooled exact-match column is read. A planted K of two accounts for 0.333 of the grid, so an estimator that answers 2 nearly always collects most of that stratum for free. An unconditional answer of 2 would score 0.333 by construction, and the silhouette's pooled 0.406 is only modestly above that; on the bipartite half its 0.333, the best of a poor field there, is exactly the two-block share and not a detection of the bipartite planting. The eigengap's constant answer 1 is never a planted value on this grid, so it scores poorly on both columns. Mean absolute error is what separates a near-constant answer from an estimator that tracks K, and there the silhouette sits at 2.38 against peel-and-merge's 1.59. So that fact stands about the exact-match column, that the silhouette maximum edges peel-and-merge pooled, and it should be read with the mechanism attached: exact-match on a grid with a large two-block stratum rewards a constant, and the same estimator loses badly on error. We report both columns in Table 6 for that reason and would not want either read alone.

The same yardstick, applied to every estimator including ours. A constant-answer discount aimed only at the baselines would be a double standard, so we state the floor once and hold all five rows

of Table 6 to it. The best exact-match rate any constant answer attains on this grid is 0.333, and the best mean absolute error any constant answer attains is 2.00. Read against the first, peel-and-merge's pooled 0.404 $[0.360, 0.448]$ clears the floor, but by less than the width of its own interval; the silhouette maximum and the Bethe-Hessian count clear it by margins of the same size; Leiden's own count 0.360 $[0.319, 0.404]$ has an interval that covers the floor, so it is not distinguishable from a constant at all; and the eigengap falls below it. The pooled exact-match column therefore does not establish that any of the top four does much better than naming the modal planted K every time, and that verdict applies to our estimator exactly as it applies to the baselines we discount with it. Read against the second, the picture changes and the change is the substantive one: peel-and-merge at 1.59 $[1.41, 1.77]$ and the Bethe-Hessian count at 1.72 $[1.54, 1.90]$ both sit entirely below 2.00, while the silhouette maximum, Leiden's own count and the eigengap are all worse on error than a constant answer would be. That is the defensible reading of Table 6: on this grid two of the five estimators track K and three do not, and the two that do are not separated from each other.

The five estimators, column by column. Appendix P states the readings; the point estimates and bootstrap intervals over 2000 resamples of the 480 graphs are these. Pooled exact-match: silhouette 0.406 $[0.362, 0.452]$, peel-and-merge 0.404 $[0.360, 0.448]$, Bethe-Hessian 0.398 $[0.356, 0.442]$, Leiden's own count 0.360 $[0.319, 0.404]$ and eigengap 0.125 $[0.098, 0.156]$; the leading three intervals overlap one another almost completely and Leiden's overlaps all of them, so that column orders the top four estimators without separating them. Pooled mean absolute error: peel-and-merge 1.59 $[1.41, 1.77]$, Bethe-Hessian 1.72 $[1.54, 1.90]$, silhouette 2.38 $[2.16, 2.60]$, Leiden 2.56 $[2.32, 2.81]$ and eigengap 3.42 $[3.18, 3.66]$; ours is separated from the silhouette's, Leiden's and the eigengap's and overlaps the Bethe-Hessian count's over most of its length. On the assortative pattern alone the exact-match rates are 0.779 for peel-and-merge against 0.796 for the Bethe-Hessian count, 0.675 for Leiden, 0.479 for the silhouette and 0.250 for the eigengap; on the bipartite half they are 0.029, 0.000, 0.046, 0.333 and 0.000.

Where peel-and-merge fails on the assortative pattern. Its failures there are confined to the cells where blocks stop being separable rather than spread across the grid. Two planted blocks are recovered from the weakest contrast we ran, 0.90 at $c = 3$ and 1.00 at $c = 5$; four blocks need more contrast, 0.25 at $c = 3$ rising to 1.00 at $c = 5$; eight blocks need more again, 0.00 at $c = 3$, 0.25 at $c = 5$, 0.95 at $c = 8$ and 1.00 at $c = 12$. More blocks at fixed mean degree means less signal per block, and the recursion stops where its tests stop rejecting, which is the resolution limit behaving as designed rather than a defect.

P.2 The two bipartite failures in detail

Appendix P reports that peel-and-merge fails on bipartite pairs for two distinct reasons and gives the rates; this records the mechanisms. The first failure is detection. The battery's Bethe-Hessian component is meant to respond to disassortative as well as assortative structure, and on a purely disassortative two-block graph, where no partition has appreciable modularity and the coarse structure the recursion would key on does not exist, it mostly does not fire at the node's allocated level. The whole-graph rejections of Section 4.3 do not carry over to these cells: those are single tests on a different pattern instance at a different mean degree, scored by a wrapped pipeline's modularity rather than by a Bonferroni-corrected battery at a node's allocated level. Raising the contrast does not rescue it, which is why we cannot claim the recursion detects disassortative structure reliably. This is unlike the four-block whole-graph setting of Anonymous (2026), where the battery detects the planting and only the split fails.

The second failure is division, and it appears only where detection succeeds. The recursion divides using the wrapped algorithm's partition, and a modularity-driven algorithm merges each disassortative pair into a single part instead of separating it, so the recursion resolves the graph down to pairs, finds nothing further inside a pair, and stops; where there are several pairs the pairs themselves sit in an assortative arrangement the battery can see, which is why detection succeeds at four and eight planted blocks and fails at two. Neither failure is repaired by a different p-value. The first would need a battery statistic that fires on a purely disassortative two-block graph, and the second a splitter matched to the structure. We ran neither, so the exact-match rate we report there is the sum of the two effects rather than a single defect.

P.3 THE BETHE-HESSIAN OVERLAP BETWEEN PROCEDURE AND BASELINE

The Bethe-Hessian spectrum appears twice in Appendix P, once inside the procedure and once as the baseline the procedure is measured against. Peel-and-merge runs the two-statistic battery of modularity and the negative Bethe-Hessian eigenvalue count at every node, and the strongest of the four baselines is that same count computed on the whole graph. The two are not independent methods and we do not present them as such. They are also not the same procedure, which is why the comparison is still informative rather than circular. The baseline reads the raw count off one spectrum and returns it as a point estimate of K, with no reference distribution and no notion of level. The battery never uses the count as an estimate at all: at each node it turns the count into an exact p-value by comparing that node's value against values from degree-preserving rewirings of that node's induced subgraph, and compares the p-value to the level allocated to the node. What is being compared is a point estimate against a sequence of calibrated tests that happen to consume one of the same spectra, and the false-split guarantee of Proposition 1 attaches to the second and has no analogue for the first. The honest consequence is that the near-tie between the two on the assortative pattern should be read as two readings of partly shared information, not as two independent estimators agreeing.

Q THE CLUSTER BUDGET (E10)

How much of the recovery gap of Section 4.3 is an artifact of an over-provisioned $K_{\max}$? 390 exact tests varying $K_{\max}$ for DMoN-lite and MAGI-lite over the assortative grid and Karate. Validity is not at stake: every budget defines its own exact test. Power is robust to it, reaching 1.00 at the two stronger contrasts for both pipelines at every budget (Appendix Q.1). What the budget moves is recovery, and on one real graph the verdict itself.

Recovery moves with the budget. At $c = 8$ DMoN-lite's mean NMI rises from 0.59 at the largest budget to 0.90 at the smallest, close to the Leiden reference of Section 4.3, and MAGI-lite's from 0.61 to 0.86. The deep-versus-classical comparison is therefore a statement about these pipelines at an over-provisioned budget rather than about their objectives; the two arms are separate runs, so it is a direction of movement and not a paired difference. What does not move is that a rejection is not a certificate of recovery.

On Karate the budget decides the verdict. MAGI-lite retains at the largest budget, p from 0.895 to 0.930 over 5 replicates, and certifies at both smaller ones, the largest p there being 0.005; DMoN-lite certifies at every budget, its largest p at the largest budget being 0.015. The budget account therefore holds for MAGI-lite and not as a general rule about budget relative to n, and it does not change what the certificate meant: at $K_{\max} = 16$ that partition carried no evidence beyond the degree sequence and the test reported so. A different budget is a different pipeline entitled to a different verdict, so the certificate belongs to a (graph, pipeline, configuration) triple.

Q.1 PER CELL

Appendix Q states that power is robust to the cluster budget and that recovery is not. The per-cell numbers behind the first claim are these. At $c = 5$ DMoN-lite reaches 1.00, 1.00 and 1.00 at budgets 4, 8 and 16, and MAGI-lite 1.00, 1.00 and 1.00; at $c = 8$ DMoN-lite reaches 1.00, 1.00 and 1.00, and MAGI-lite 1.00, 1.00 and 1.00. At contrast 3 the cells do spread, from 0.10 to 0.40 across the six budget-by-pipeline combinations, but that is the threshold regime in which Appendix N already documents power estimates at this replication being noisy, and at 20 replicates per cell we read no ordering into them.

On the recovery side the intermediate budget behaves as the endpoints suggest: at $c = 8$ DMoN-lite's mean NMI passes through 0.70 at a budget of 8 and MAGI-lite's through 0.70. At $c = 5$, where power is already at its maximum at every budget, mean NMI at the matched budget of 4 is 0.63 for DMoN-lite and 0.62 for MAGI-lite, which is the separation between rejecting and recovering persisting at the budget most favorable to recovery. On Karate, DMoN-lite certifies at every budget, its largest p in each cell being 0.005, 0.005 and 0.015 at budgets 4, 8 and 16, and MAGI-lite's largest p at a budget of 8 is 0.005.

R REPRESENTATION INVARIANCE: THE DEFECT WE FOUND AND HOW WE CHECKED IT

Lemma 2 asks that one fixed map be applied to every coordinate of the tuple. In an implementation the coordinates are not graphs but data structures, so the requirement is stronger than determinism at a fixed seed: the rule must be invariant to the representation of the graph, and in particular to the order of the rows of the edge array it is handed.

Our own code violated this and the failure is instructive enough to report in full. MAGI-lite drew its contrastive positives by edge slot index, so its score was a function of the row order of the edge list and not of the edge set alone. The two kinds of coordinate arrive in systematically different orders: the observed graph comes from the generators and the dataset loaders in lexicographic order, while every null sample leaves the swap chain in chain order, with rewired edges written back into the slots they replaced. The rule was therefore silently different on the observed coordinate than on the null coordinates, which is exactly the asymmetry Step 1 of the proof of Lemma 2 forbids, and no amount of seed discipline repairs it, because the defect is in the map and not in the randomness. We confirmed the dependence directly: permuting the rows of an identical edge set changed that pipeline's score on every instance we checked, while DMoN-lite, MinCut-lite and the classical Leiden wrapper were invariant.

The remedy is one line at the entry point of every fit: canonicalize the edge array, we sort it lexicographically, before anything reads it. We applied it, then reran every MAGI-lite arm from scratch at full replication rather than patching the affected cells, and the corrected results change no qualitative conclusion of this paper. Anyone wrapping an existing implementation should check the same property for any rule that reads an edge ordering, an adjacency-list traversal order built from it, or a mini-batch order inherited from storage layout. The corresponding check on the null side is that the chain is used only through its projection to graphs, as Section 3.2 sets out. The test itself is cheap: permute the edge list, refit, and confirm the score is unchanged.

S WHY HUB AMORTIZATION MAY COST POWER

Section 4.4 reports that hub amortization holds level and loses on power per second to a reduced-epoch cold-start surrogate, and that shortening the chain does not rescue it. Two mechanisms are consistent with what we measured and we did not run the ablation that separates them, so we name both and commit to neither.

The first is specific to the feature recipe. On graphs without native attributes the input features are leading singular vectors of each graph's own adjacency matrix, so every graph in the tuple receives its own basis; those bases carry no alignment across graphs beyond the leading degree-like direction that a degree-preserving rewiring nearly fixes, and even the sign of each singular vector is arbitrary. A pretrained encoder is then applied to an essentially unrelated input basis on every coordinate, which costs power and, being symmetric across coordinates, can never cost level. The second is the generalization penalty of warm-starting (Ash & Adams, 2020), which is consistent with what we observe but is not needed to explain it. Feature-basis misalignment is our leading candidate. The distinction is testable and predicts that hub amortization behaves differently whenever the feature map is graph independent, as it is when a real graph is tested with its own attributes.

The same symmetry argument explains why we did not expect a short chain to help: the null graphs sit as close to the hub as the observed graph does and inherit the same head start, so shortening the chain moves all $M + 1$ coordinates together. We offer that as an explanation of what we measured, not as a demonstrated fact. Whether the balance reverses when pretraining dominates per-fit cost is open, and entangled with the feature question, since a graph-independent feature map is what would give a warm start something transferable.

T REAL GRAPHS, IN FULL (E7)

What happens on real data? 7 graphs spanning over two orders of magnitude in n, each the largest connected component of its simple undirected version, with all four wrappers and a Leiden-wrapped peel-and-merge row (Table 1, in Section 4.5). Every graph is certified by at least one pipeline,

and nearly every (graph, pipeline) pair sits at the p-value floor for its stratified M. The exceptions are on Karate, where DMoN-lite certifies one step above the floor ($p = 0.010$) while MinCut-lite retains and MAGI-lite sits further from a certificate than any other pair in the suite. MAGI-lite's retention is not a worse partition: its NMI 0.48 is above DMoN-lite's 0.46 and MinCut-lite's 0.47, so reference agreement and certificate point opposite ways. The row measures only that a pipeline whose modularity is not extreme against rewirings of its degree sequence gets no certificate, and the verdict belongs to the configuration rather than to the graph: on this same graph MAGI-lite certifies at both smaller cluster budgets (Appendix Q). The recovery gap reappears: Cora and Citeseer reject at the floor while agreeing only weakly with the reference labels. Model order here has no ground truth, and the counts on the three largest graphs are upper bounds on what was certified, since many leaves fell below the minimum node size and went untested (Appendix T.1). Cost separates sharply by what is certified (Appendix X).

T.1 Why the real-graph model orders are upper bounds

Section 4.5 reports peel-and-merge model orders of 110 on Cora and 332 on Amazon Photo and calls them upper bounds on what was certified. The arithmetic behind that reading is worth stating. A leaf is a leaf for one of two reasons: its exact test did not reject, or it fell below the minimum node size of 30 and was never tested at all. Only the first kind has a test behind it, and only the first kind can hold as many vertices as that minimum. Set the reported counts against the vertex counts in Table 1 and the average leaf on each of the three large graphs holds fewer vertices than the minimum node size, which is arithmetically impossible unless many of the leaves are of the second kind. On graphs this size the minimum node size, and not the evidence in the wiring, is much of what fixes the scale of the reported model order. That is the honest resolution limit of the procedure rather than a property of these graphs, and it is listed among the limitations.

U Extended motivation, the asymptotic comparison, and notes on the contributions

This part collects the material the introduction states in short form: why an achieved score is uninformative on its own, how the asymptotic comparison of Section 4.2 should and should not be read, the adjacent literatures we do not draw on, and the qualifiers that travel with each contribution.

Why a respectable score is not a finding. The scores that DMoN-lite (Tsitsulin et al., 2023) and MAGI-lite (Liu et al., 2024) report on structureless graphs are not implausibly high, and that is the point. Structureless graphs admit partitions of nontrivial modularity (Guimerà et al., 2004; McDiarmid & Skerman, 2020), so a respectable score is uninformative until it is compared against what the same pipeline achieves on graphs that are structureless by construction. Compared that way the scores of Fig. 1 are unremarkable: in 8 of the 9 cells of the calibration audit the cell-mean achieved score sits at or below the cell-mean upper end of the null band, the exception being the collapsing MinCut-lite Erdős-Rényi cell, which is the most conservative in the audit rather than the least. Individual graphs reject at about the nominal rate, between 0.005 and 0.065 across the 9 setting-by-pipeline cells at the nominal five percent level (Section 4.1, Table 3, 1800 logged tests). This is why the band in Fig. 1 is drawn as an average of per-test intervals rather than as a decision boundary: a dot above it is an ordinary outcome, not a rejection. Benchmarking work documents how unstable deep clustering pipelines are (Wu et al., 2025; Leeney & McConville, 2024; Liu et al., 2026), which sharpens the question without answering it, since instability tells us the output moves and not whether it means anything.

How to read the asymptotic comparison. The recomputation in Section 4.2 is not a refutation of the asymptotic theory. The limit behind a Tracy-Widom threshold is proved for fixed edge probability, mean degree linear in n (Bickel & Sarkar, 2016), and the graphs of the calibration audit have mean degree 10 at $n = 1000$, so the rule is being applied outside the regime its authors state (Bickel & Sarkar, 2016; Lei, 2016). Two controls in Appendix C keep the comparison fair. The same-statistic control wraps the identical leading eigenvalue in the exact construction on 180 of those graphs, the first 60 replicates of each setting: the fixed threshold rejects 1.000 of them and the exact wrapper 0.039, so the discrepancy is the threshold and not the statistic. The in-regime control raises the

density until the asymptotics apply, and the same threshold then rejects 0.017 at mean degree 200 with $n = 1000$, that is, it is consistent with its level in its own regime. What a practitioner cannot do is check from one finite graph whether the regime has been left, and the median of $n^{2/3}(\lambda_1 - 2)$ on our sparse Erdős-Rényi graphs, 7.7 against a threshold of 0.9793, shows how far outside it a routine sparse graph can sit. The exact wrapper needs no such judgment because its validity is finite-sample.

Adjacent literatures. Selective inference after clustering (Gao et al., 2024; Chen & Witten, 2023; Neufeld et al., 2024; Hivert et al., 2024; Miyata et al., 2026; Nishino et al., 2025) and conformal prediction on graphs (Huang et al., 2023; Nath et al., 2026) solve problems next to this one, the first conditioning on a selection event under a distributional model for the data matrix and the second calibrating node-level predictions under exchangeability of nodes rather than of graphs. Section 2 says why neither delivers a finite-sample statement about the wiring of one observed graph read through a trained pipeline.

Notes on C2 and C3. The warm start that Corollary 2 does not cover, and that the experiment shows to be invalid, is the one an implementer would write without it, namely pretraining on the observed graph and fine-tuning on each null graph. Its failure is severe rather than marginal (Section 4.4). On the other side, hub power is close to flat in chain length, moving only between 0.32 and 0.44 across the chain lengths of Appendix L, which is a further reason to read C2 as a validity boundary rather than as a way to buy power. For C3 the reading is that testing noise is cheap while testing structure pays full price, which is the desirable direction: the cases that stop early are the ones a practitioner runs many of.

Note on C4. The recursive estimator and its level allocation follow the hierarchical significance testing line (Meinshausen, 2008; Kimes et al., 2017) and are carried over here from Anonymous (2026) without change. Our contribution is the direct false-split measurement on structureless graphs; the accuracy grid replicates the design of Anonymous (2026) at fresh seeds.

The karate reading in full. MAGI-lite retains on the karate club network at the largest cluster budget we ran (p between 0.895 and 0.930 over the 5 replicates of Appendix Q; 0.880 in the single run of Table 1) and certifies at the smallest ($p \leq 0.005$). This is the layer working as intended rather than failing. Splitting a graph of a few dozen nodes into as many clusters as that budget allows yields a partition no more modular than degree-preserving rewirings of the same graph produce, so the test reports a misconfiguration the practitioner would otherwise never learn about. The cluster budget is therefore part of the object audited, and a retention at a large budget is a reason to rerun at a smaller one and report both; a certificate found at a budget chosen after seeing a retention is level-α only after correcting for the budgets tried, for example by Bonferroni, which the Karate certificate survives because its p-values at both smaller budgets sit at the floor $1/(M + 1)$ (Appendix Q).

Two further scope statements. The wrapped deep pipelines are compact reimplementations of DMoN (Tsitsulin et al., 2023), MinCutPool (Bianchi et al., 2020) and MAGI (Liu et al., 2024) rather than the original authors' code, so their absolute scores are not comparable to published numbers and no ranking among them is intended. Near-threshold power estimates are also noisy at our replication counts: the three nominally similar weakest-contrast cells of E3, E4 and E6 span 0.20 to 0.48, so single-cell power differences of that size should not be read as real (Appendix N). Finally, a small p-value remains evidence about the wiring and not a recovery guarantee.

V Related work in full: the four axes, and the null-model and Monte Carlo lineage the body compresses

This part expands Section 2, which states each axis and the gap it leaves but not the individual results behind it. Nothing here is needed to follow the method; it is here so that a reader who wants the lineage of a particular construction can find it.

Testing for structure, in more detail. Bickel & Sarkar (2016) use the Tracy-Widom limit of the largest eigenvalue of the centered scaled adjacency matrix to test a one-block null, Lei (2016) extend the spectral argument to goodness of fit for a fitted stochastic block model, and Jin et al. (2025)

develop goodness of fit for the broader degree-corrected mixed-membership family. Algorithm-agnostic significance tests for communities compare a detector's quality against randomized networks or bootstrap thresholds (Kojaku & Masuda, 2018; Yanchenko & Sengupta, 2024), and He et al. (2020) compute exact p-values for a single community under an Erdős-Rényi null. Karwa et al. (2024) take an exact conditional route instead, sampling from the fiber of a fitted block model, which is the closest existing construction in spirit to ours: exactness bought by conditioning rather than by a limit. It still conditions on a fitted parametric model and tests that model, where we condition on the observed degree sequence and test a pipeline. Model order has a parallel literature, most prominently spectral estimators built on the Bethe Hessian (Saade et al., 2014; Le & Levina, 2022), read against the detectability threshold (Decelle et al., 2011; Abbe, 2018) and the degree-corrected block model (Karrer & Newman, 2011), which is also the generator we plant structure with. In all of this the object under test is a parametric model and the statistic is chosen by the analyst, so a practitioner holding a trained pipeline and one graph has no way in.

Post-selection and selective inference, in more detail. Gao et al. (2024) test for hierarchical clustering and Chen & Witten (2023) for k-means, both testing differences between the means of selected clusters in a Gaussian feature matrix, with Neufeld et al. (2024) splitting the data instead of conditioning, Miyata et al. (2026) generalizing the conditioning argument to whole pipelines, and Nishino et al. (2025) giving the closest graph-side result. The recurring requirement is a tractable description of the selection event, typically piecewise linearity in a Gaussian feature space, and the recurring target is a difference between clusters that are presupposed to be meaningful. Neither survives contact with a gradient descent trajectory over a graph with no features under test.

Conformal prediction, in more detail. Conformal prediction transfers finite-sample coverage to node classification (Huang et al., 2023) and to cluster assignments (Nath et al., 2026), and belongs to a continuing line on exchangeability-based and resampling inference (Barber & Ramdas, 2026; Dombowsky et al., 2026). Cheap permutation testing (Domingo-Enrich et al., 2025) lowers resampling cost but is specific to U-statistics and V-statistics and does not transfer to fitted pipelines. Our construction shares the exchangeability device and inverts what is held fixed: conformal methods hold the fitted rule fixed and resample the units, while we hold the pipeline fixed and resample the graph itself.

Null models, Monte Carlo testing, and significance for clusterings. The null used here is the uniform distribution over simple graphs with the observed degree sequence, sampled by double-edge swaps, which connect that space (Taylor, 1981) and underpin motif and rewiring analyses (Milo et al., 2002; Fosdick et al., 2018). Exactness comes from the parallel Monte Carlo construction of Besag & Clifford (1989); its sequential companion (Besag & Clifford, 1991) stops early on unpromising tests and is what Appendix M implements, Gandy (2009) gives an open-ended rule whose resampling risk, the chance of disagreeing with the decision the ideal p-value would give, is uniformly bounded, and Fischer & Ramdas (2025) obtain anytime-valid Monte Carlo tests by betting, which we do not implement. The nearest applied analogue is significance analysis for hierarchical and single-cell clustering (Kimes et al., 2017; Grabski et al., 2023; Hivert et al., 2024), built for feature matrices and particular clustering routines rather than for graph wiring or trained networks. Our recursive level allocation follows Meinshausen (2008), by way of the peel-and-merge recursion of Anonymous (2026).

Deep graph clustering, in more detail. The benchmark critiques we rely on (Wu et al., 2025; Leeney & McConville, 2024) report run-to-run variation large enough to reorder published rankings, fragility under small perturbations of the graph, and reversals once tuning budgets are equalized across methods, echoing the earlier and better known findings for graph neural network evaluation generally (Shchur et al., 2018; Platonov et al., 2023). That literature is the reason a certificate is worth having: it establishes that the partition returned by a deep clusterer (Tsitsulin et al., 2023; Bianchi et al., 2020; Ying et al., 2018; Devvrit et al., 2022; Liu et al., 2023; 2024; 2026) is unstable, without offering any way to ask whether a particular returned partition on a particular graph reflects anything in the wiring. The classical wrapper line (Blondel et al., 2008; Traag et al., 2019; Anonymous, 2026) answers that question for a pipeline run under one fixed rule, including a trained network at a fixed seed, and Lemma 2 is what extends it to the seeding, warm-start and stopping choices a trained pipeline forces.

W NOTES ON THE NULL, THE LEMMAS, AND THE MODEL-ORDER CAVEATS

This part collects the expository material that Section 3 compresses: the reach of the null hypothesis, the wider conditions the proof of Lemma 2 actually establishes, the two elementary properties of the sequential rule, and two caveats attached to the model-order recursion.

Why the null is stated conditionally. The conditional form of H_0 is the one that can hold: read instead as a point null naming one fixed measure, it would be a hypothesis whose reference measure is computed from the realization, and no $G(n, p)$ draw would obey it. Conditionally $G(n, p)$ obeys it exactly, its density depending on the graph only through the edge count and so flat on every fiber, which makes the Erdős-Rényi cells of Section 4.1 and the $c = 1$ embedded null of Section 4.3 exact-null draws rather than approximations to one.

What the null holds fixed, and what it does not test. The attribute matrix X is held fixed across the observed graph and all M null graphs, so any difference in score is attributable to the wiring rather than to the attributes. This makes the test a statement about wiring given degrees and given attributes. Whether the attributes carry information about the partition once the wiring is known is a different null, requiring X to be permuted while G is held, and it is outside the scope of this paper. A second idealization is worth naming: the driving randomness $W_0, \dots, W_M$ of the chain and the training seeds are distinct pseudorandom streams, and as is standard for Monte Carlo tests we treat them as ideal independent sources.

The condition the proof of Lemma 2 really uses. Cases (a) and (b) of Lemma 2 are two instances of a single condition that the proof establishes without change: that $(U_0, \dots, U_M)$ be exchangeable and independent of $(V_0, \dots, V_M)$. That condition also covers drawing one seed once and sharing it across all $M + 1$ graphs, which is neither of the two stated cases but is the most convenient rule in practice. Case (a) alone recovers the construction already available for pipelines that are deterministic functions of the graph; case (b), and Corollary 1 with it, is what the stochastic-training setting adds. The tie rule matters for the same reason: counting ties against rejection through the $\geq$ in (4) keeps the bound valid for scores with atoms, and a pipeline that collapses to a single cluster scores every graph in the tuple alike, so the test reports $p = 1$ rather than anything anti-conservative.

Read the other way, the instability documented for deep graph clustering motivates the test rather than threatening it. Non-convergence, collapse, seed variance and a truncated epoch budget change which statistic is being computed, not whether the p-value is valid, so a short fixed training budget is a legitimate statistic rather than a corner cut. The same reading applies to the cluster budget $K_{\max}$, the hyperparameter most likely to be mistaken for a threat to validity. It is part of the rule $\mathcal{A}$, so each budget defines its own exact test rather than a family among which something must be chosen, and the only requirement is that it be fixed before the observed graph is looked at. What it moves is recovery rather than power, measured in Appendix Q.

What hub amortization buys, and what it does not. The scheme licensed by Corollary 2 lets one expensive pretraining fit buy $M + 1$ cheap fine-tunes with exactness untouched. Its value is in the boundary it draws rather than in the saving: the warm start that pretrains on the hub is exact, the warm start that pretrains on the observed graph is not, and the two differ by which graph the second argument of the statistic is pinned to. That boundary is narrow and invisible without Corollary 2, which is why we present it as a validity result. Note also that the proof of Lemma 1 behind it uses only stationarity and reversibility, nothing about mixing or the size of L, which is the reason validity is free of the chain length while power is not.

Two properties of the sequential rule. First, the early-stopping branch reports $p = h/T$ with $T \leq M$, so it never returns a value below h/M. At $h = 10$ and $M = 199$ that lower bound exceeds 0.05, so an early stop is always a non-rejection at that level: the rule accelerates the null case and never the alternative, and the saving it delivers is a saving on graphs that were not going to reject. Second, the stopping time is data dependent, so the reported value is in general not the fixed-M p-value that the same null samples would have produced. In the configuration we run the two decisions nevertheless coincide on every realization, which holds deterministically whenever $\alpha M < h \leq \alpha(M+1)$, as here (Appendix M), and not in general. This is distinct from the resampling risk of Gandy (2009), the chance of disagreeing with the ideal infinite-sample p-value, which no

bounded-budget rule, ours included, controls uniformly. Anytime-valid testing by betting (Fischer & Ramdas, 2025) is the natural upgrade, and we do not implement it.

The battery deviation on the largest real graph. On the largest graph in the real-graph suite the Bethe-Hessian eigenvalue count is prohibitive at every node of the recursion, so that graph is run with modularity alone. Its nodes are therefore tested at threshold α_v rather than $\alpha_v/2$, and without the statistic that responds to disassortative structure. Proposition 1 still applies to it, since the level allocation is unchanged, but its $\hat{K}$ is not on a common threshold with the other rows and should not be compared with them as though it were.

What selection neutrality buys, and the one guarantee without it. For a vertex set fixed in advance, the local null of a node is a consequence of the global null; what the assumption of Proposition 1 buys is the data-dependence of the node, that is, permission to treat a subgraph the recursion itself chose as though it were uniform on the fiber of its own degree sequence given the selection history. Hierarchies fixed in advance (Meinshausen, 2008) avoid the issue entirely; data-dependent ones (Kimes et al., 2017) share it. One guarantee survives without the assumption: under the global null, and with the root larger than $n_{\min}$, reporting $\hat{K} > 1$ requires the root test to reject, and the root's vertex set is not data selected, so $\Pr[\hat{K} > 1] \leq \alpha$ for any splitter whatsoever. Appendix O measures this root-level guarantee rather than the assumption, on which it gives only the weak child-level evidence reported there.

Finally, the mechanism behind the conservatism of the recursion is worth stating explicitly. The level $\alpha_v = \alpha\, n_v/n$ shrinks with the size of the node while the required sample size M_v grows like $1/\alpha_v$, so deep nodes are held to increasingly stringent levels at increasing cost, and nodes smaller than $n_{\min}$ are never tested at all. Both effects push $\hat{K}$ downward, which is why $n_{\min}$ belongs in the limitations as an honest resolution limit rather than being presented as a tuning convenience.

X Experimental detail: grids, per-cell readings, and the caveats compressed out of the experiments section

This part carries the material that Section 4 states in one line each: the synthetic grids in full, the limits of the calibration audit, the per-pipeline readings behind the power and recovery figures, the amortization and stopping studies at the level of their designs, the real-graph readout, the output distributions of the model-order baselines, and the two bipartite failure modes cell by cell. Nothing here is new evidence; every number is the one the corresponding results file already carries.

The two synthetic grids. Structureless graphs are Erdős-Rényi and power-law configuration models with $\tau \in \{2.5, 2.1\}$, at $n = 1000$ and mean degree 10; every heavy-tailed draw passes through the swap burn-in of Appendix B before it is used, since the erased configuration model is not uniform given degrees (Fosdick et al., 2018). Structured graphs are degree-corrected block models (Karrer & Newman, 2011) at $n = 1000$, $K = 4$ and mean degree 10, in an assortative and a bipartite-pairs pattern, with the contrast c setting the ratio of within-pattern to background edge probability, so that $c = 1$ is exactly Erdős-Rényi and doubles as an embedded null needing no burn-in. The model-order grid of Appendix P moves to mean degree 15, a regime in which recovery is feasible, and plants $K \in \{2, 4, 8\}$ at contrasts $\{3, 5, 8, 12\}$ in both patterns. The MAGI-lite rerun forced by the representation-invariance defect was carried out at full replication, not on a reduced grid (Appendix R).

What the calibration audit can and cannot establish (E1). One limit sits on the audit of Section 4.1. The Wilson intervals in Table 3 are several percentage points wide, so the audit is consistent with correct calibration and cannot verify it more sharply than that. That limit is a property of the budget we chose, not of the construction: with $M = 199$ null samples the smallest attainable p-value is 0.005, which puts all three nominal levels of Table 3 on the attainable grid. The MinCut-lite cells deserve their own sentence for the same reason. Its collapse to a single cluster on Erdős-Rényi graphs makes the observed graph tie with the whole null tuple, and since ties are counted against rejection the cell comes out conservative rather than broken; at $\tau = 2.1$ it un-collapses and reports communities on pure noise, while at $\tau = 2.5$ it still collapses on most graphs; the un-collapsed cell is the behavior

the exhibit of Section 4.1 is about. Neither threatens validity, both cost power, and Appendix G gives the counts.

The asymptotic comparison, stated at length (E1b). The Erdős-Rényi rate of Section 4.2 is the one to read first because it is the only cell of the four in which the graphs really are drawn from the competitor's own null, so that every rejection there is false by construction. The two heavy-tailed settings are subordinated deliberately: the competitor's hypothesis is false on them before any test is run, so a rejection rate there is not a type-I error rate and we do not present it as one, even though the pooled figure includes it. The dense controls are the mirror image, the threshold applied where its limit theory asks it to be applied, and Appendix C reports both control sizes with their intervals. The conclusion we draw from the pair is about information available to a user rather than about the theorem: a fixed threshold returns a verdict in the same form whether or not its regime holds, so whether the asymptotics apply is a judgment the practitioner must make from outside the procedure, and that judgment is precisely what the exact construction removes.

Power and recovery, per pipeline (E3). The embedded null at $c = 1$ rejecting at 0.02 is a sanity check on the pipeline plumbing rather than calibration evidence; the calibration evidence is Section 4.1. Power and recovery separate per pipeline rather than uniformly, and the $c = 8$ cells give the fullest picture: Leiden reaches power 1.00 at NMI 0.92, DMoN-lite 1.00 at 0.58, MAGI-lite 1.00 at 0.61, and MinCut-lite comes out of collapse to 1.00 at NMI 0.66. The comparison between the deep pipelines and the classical reference is conditional on the cluster budgets of Appendix B, $K_{\max} = 16$ for DMoN-lite and MAGI-lite and $K_{\max} = 8$ for MinCut-lite against a planting of four blocks, while Leiden chooses its own number of parts and NMI penalizes over-partitioning mechanically; Appendix Q measures how much of the gap that explains. On bipartite pairs at $c = 8$ the four wrapped algorithms reach power 1.00, 1.00, 1.00 and 0.96 at NMI 0.28, 0.25, 0.27 and 0.56. MinCut-lite is the exception there, at roughly twice the others' agreement and slightly lower power, which is what a normalized-cut objective is expected to do on a planting the modularity-driven wrappers cannot express; the comparison is a single cell at 25 replicates, so we do not generalize it.

Amortization, stopping, and seeding, at the level of their designs (E4 to E6). The amortization modes of Section 4.4 are compared at 60 null graphs per mode for level and at $c = 5$ and $c = 8$ for power (Table 4), with the contrast-3 cells entering only the spread of Appendix N. The two hub levels Section 4.4 reports are fine-tuning budgets of 50 and 100 epochs; Table 4 carries the third along with the per-mode wall-clock costs, and Appendix H sets the surrogate against hub fine-tuning at matched cost. Two mechanisms could explain why hub pretraining loses power at this scale, and Appendix S states both; the chain-length ablation of Appendix L tested a different rescue, a nearer hub, and found no trend, and the ablation that separates the two mechanisms was not run. On the sequential rule, stopping ends 0.95 of tests early at a median of 19 fits out of 199 across 200 structureless graphs while holding level at 0.045, and at the exceedance target and budget we ran it coincides with the fixed-budget decision on every realization, so early stopping changes no rejection reported anywhere in this paper. On seeding, the three rules covered by Lemma 2 all hold level and separate only at the detectability edge, where the median-of-three rule reaches power 0.64 against 0.48 for a single fixed seed and 0.32 for fresh per-graph seeds; the replication makes that ordering indicative rather than precisely estimated. The false-split measurement of Appendix O speaks only to validity: it says nothing about whether $\hat{K}$ is right when structure is present, which is what Appendix P is for.

The real graphs, read out (E7). Provenance for all 7 graphs is in Appendix B. On Karate, DMoN-lite certifies one step above the attainable floor at $p = 0.010$ with NMI 0.46 against the reference labels, so that graph alone receives four different p-values and both verdicts from four pipelines, which is what it means for a certificate to belong to a (graph, pipeline, configuration) triple rather than to a graph. The recovery gap on real data is visible on Cora, where NMI against the reference class labels is 0.20 for Leiden, 0.21 for DMoN-lite and 0.21 for MAGI-lite while all four pipelines reject at the floor, and where the deep pipelines recover Football's conference structure much more closely. The Amazon Photo model-order row runs with modularity alone as its battery, so it is not on a common threshold with the other rows and should not be compared to them directly. Neither reported model order is scored for accuracy, since these graphs carry no known block count; what the runs support is the narrower statement that every split performed is backed by an exact test at that node's allocated level, and Appendix T.1 shows why the counts are upper bounds on what was

certified. Cost separates sharply by what is certified: the most expensive existence certificate took 565 seconds and the most expensive model-order certificate 7948 seconds, about 2.2 hours.

Model order: the assortative cells, the baselines, and the two bipartite failures (E9). Peel-and-merge and the Bethe-Hessian count are not independent methods, since the same spectrum appears inside our battery and in that baseline, though they are not the same procedure either (Appendix P.3). On the assortative pattern alone peel-and-merge matches the planted K on 0.779 of graphs and the Bethe-Hessian count edges it at 0.796, our failures there being confined to the cells where blocks stop being separable. Pooled exact-match fails to separate the top four because the three leading intervals overlap almost completely and Leiden's own count, at 0.360 $[0.319, 0.404]$, overlaps all of them; on error, our interval is separated from the silhouette's, Leiden's and the eigengap's, and overlaps only the Bethe-Hessian count's. The constant-answer yardstick is what makes those columns readable. Two of the four baselines are close to constant on this grid: the silhouette maximum returns 2 on 0.890 of graphs and the eigengap returns 1 on 0.833, while the planted two-block stratum is 0.333 of the grid, so a constant answer alone attains 0.333 on exact-match. Applied to every row including ours, that floor leaves Leiden's own count indistinguishable from a constant and leaves peel-and-merge and the Bethe-Hessian count as the only two of the five that beat a constant on error, which is the sense in which two of the five track K and three do not (Appendix P.1). On the bipartite pattern the Bethe-Hessian count matches on 0.000 of graphs, and the two failures behind 0.029 separate cleanly. Detection: pooled over the four contrasts the top-down pass produces no split at all on 0.775 of the 80 two-block graphs, with the final reported order one on 0.950 of them, and this is not a weak-contrast effect, since at $c = 12$ the no-split rate is still 0.700 and the same non-detection covers the eight-block cells at the weakest contrast at 1.000. Division: at four and eight planted blocks with contrast at least eight the pass fires on 1.000 of the 80 graphs in those cells, and 0.988 of them report exactly $K/2$. The two are additive, so 0.029 is the sum of two effects rather than a single defect (Appendix P.2).

The cluster budget, per cell (E10). The budget sweep runs $K_{\max} \in \{4, 8, 16\}$ for DMoN-lite and MAGI-lite, on the assortative grid at contrasts 3, 5 and 8 with 20 replicates per cell and on Karate with 5 replicates. At the weakest contrast the cells do spread, but that is the threshold regime, where power estimates at 20 replicates are noisy and we read no ordering into them. The revision this study forces on Section 4.3 is qualitative rather than quantitative for one reason worth restating: the two arms are separate runs at different seeds and different replication, so the NMI values are compared as a direction of movement and not as a paired difference. Appendix Q.1 reports every cell.

Y READING A CERTIFICATE: SCOPE, MULTIPLICITY, COST, AND DIRECTIONS

Section 5 states the scope of a certificate in compact form. This part carries the longer version of that discussion: what a small p-value is and is not a statement about, why we report per-pipeline certificates rather than a single graph-level one, how far the invariance requirement of Lemma 2 reaches beyond the defect we found in our own code, what the method costs in practice, and the work the limitations point to.

What the certificate is a statement about. A rejection is a statement about a wrapped pipeline and not about clustering in general. On the assortative grid at moderate contrast, Leiden, DMoN-lite and MAGI-lite all reject at rate 1.00, while agreeing with the planted labels at only 0.40 and 0.42 for the two deep pipelines, and MinCut-lite has power 0.00 in the same cell. Three of the four wrapped pipelines reject in every replicate and the fourth in none, on the same graphs and against the same null, so no single number summarizes what the test does at that contrast. Agreement with reference labels neither rescues a retention nor is what the test measures. On Karate, MAGI-lite reaches NMI 0.48, above DMoN-lite's 0.46, and is the pipeline furthest from a certificate at $p = 0.880$ against DMoN-lite's $p = 0.010$. A pipeline can land close to the reference partition and still fail to separate itself from what the degree sequence alone produces, which is the quantity the certificate reports. Reading NMI as a proxy for significance, or significance as a proxy for NMI, inverts the relationship the paper is built to keep apart.

Multiplicity, and why we do not report a graph-level p-value. The awkwardness of the (graph, pipeline, configuration) triple is real. A practitioner who wants a graph-level answer, rather than an answer about one deployment, must wrap several pipelines and correct for the multiplicity of

that family, which costs power. We report per-pipeline certificates rather than the minimum p-value across pipelines, for two reasons. A minimum over a family is not itself a valid p-value without a correction, so reporting it uncorrected would be exactly the error the paper exists to prevent. And the per-pipeline reading is the one that matches the deployment question, since the object a practitioner ships is a particular pipeline at a particular configuration, not the union of every pipeline that might have been tried. Where a graph-level statement is genuinely wanted, a Bonferroni correction over the wrapped family is valid by the same argument that licenses the battery in Section 3.6, and the resulting loss of power should be budgeted for in advance rather than discovered afterward.

How far the invariance requirement reaches. The defect described in Section 3.3 and measured in Appendix R is one instance of a general failure mode. The point generalizes to any quantity a pipeline reads that is an artifact of the encoding rather than of the graph and differs between the observed and null coordinates: edge ordering, the layout of an adjacency list built from it, any sampling scheme keyed to an index, any tie-break resolved by position. Each of these is a fixed function of the file, not of the graph, and a pipeline that reads one is not the fixed function of the graph that Lemma 2 requires. The audit is inexpensive relative to the test itself, since it costs a permutation and one extra fit rather than $M + 1$ of them, and we recommend it as a standing check for anyone wrapping a pipeline they did not write.

What the method costs. An existence certificate costs $M + 1$ pipeline fits, 565 seconds for the most expensive graph in our suite on one CPU machine, while a model-order certificate costs 2.2 hours on the largest graph because the recursion pays a full test at every node it visits. Sequential stopping helps only where help is already cheap, stopping 0.95 of structureless graphs early, which is the case in which the full run would have been least informative anyway. The practical reading is that the method is affordable for anyone who can train their pipeline a couple of hundred times over at the audit budget, or a few dozen at the compute-bound budget of the remaining studies, and is not affordable when a single run takes hours. Appendix K gives the cost model, and Table 1 and Appendix H the measured figures.

Future work. The two repairs the limitations name are separate pieces of work. A split rule not driven by modularity would let the recursion express the partitions it cannot currently produce, and would leave the level allocation of Proposition 1 unchanged, though its selection-neutrality condition depends on the splitter and would have to be re-examined. On its own, however, it would fix only the cells where the battery already rejects; the cells where it does not need a battery statistic with power against a purely disassortative two-block graph, which is a separate problem in statistic design rather than in splitting. Anytime-valid betting (Fischer & Ramdas, 2025) would allow stopping at any time and extending a run after seeing its result, which suits a setting where each sample costs a training run (Gandy, 2009) and where the analyst will be tempted to look at the running p-value whether or not the procedure licenses it. Larger pipelines, where pretraining dominates per-fit cost, are the regime in which hub amortization should pay, and scalable contrastive methods (Liu et al., 2024; Devvrit et al., 2022) are the natural test bed for that claim, since the surrogate that dominates hub amortization at our scale (Table 4) loses its advantage exactly when a reduced-epoch fit stops being cheap. Weighted and hypergraph nulls would widen applicability at the price of a new sampler with the same exchangeability property, which is the only property Lemma 2 actually consumes. Finally, single-cell analysis is an application where a pipeline shape the wrapper covers unchanged, a k-nearest neighbor graph followed by Leiden, is standard, and where the danger of reading clusters out of noise is well documented (Grabski et al., 2023; Hivert et al., 2024). The certificate there would be conditional on the graph construction as well as on the clustering, which is a limitation and also a fair description of what such pipelines actually commit to.

A misreading worth naming. Reporting a certificate as though it validated a partition would turn a tool for restraint into a source of unwarranted confidence. Our own experiments pair rejection at the p floor with poor agreement with reference labels, on Karate and on the synthetic grids alike, so the two readings come apart in the data we report and not only in principle.