Electronic Colloquium on Computational Complexity, Report No. 292 (2026)

Testing Implies Distance Estimation with Polynomial Loss

Gaia Carenini* & Cameron Seth †

November 3, 2026

Abstract

A tester decides whether a graph has a property or is *far* from having it. A distance estimator asks for more: it must approximately determine how many edges need to be changed. A property is said to be *testable* if there is a tester with a number of queries bounded independently of the graph's order for each conceivable distance. Similarly, if a property admits an estimator with a number of queries bounded independently of the graph's order, it is said to be *estimable*.

Fischer and Newman proved that, in the dense graph model, every testable property is also estimable. We show that the passage from testing to estimation requires only a polynomial increase in query complexity. This resolves a long-standing open problem in property testing, previously tackled by Fiat and Ron, by Hoppen, Kohayakawa, Lang, Lefmann, and Stagni, and by Gishboliner, Kushnir, and Shapira.

1 Introduction

1.1 Property testing in the dense graph model

Deciding whether a large object satisfies a given property can be very expensive, both in computation and in the amount of information that must be read. Property testing studies a relaxation of this task: distinguishing objects that satisfy the property from those that require a substantial proportion of their entries to be changed before they satisfy it. The aim is to make this distinction by inspecting only a (very) small part of the input. We refer the reader to Goldreich's book [1] for a general introduction to the area.

We work in the *dense graph model*, also called the adjacency matrix model. A graph on a vertex set V is a function $G : \binom{V}{2} \to \{0,1\}$, and we write $\mathcal{G}_V$ for the set of such graphs. For $n = |V| \geq 2$, the normalized edit distance between $G, H \in \mathcal{G}_V$ is

$$d(G,H) = \frac{|E(G) \triangle E(H)|}{\binom{n}{2}}.$$

A *graph property* $\mathcal{P}$ is a collection of graphs closed under isomorphism. Let $\mathcal{P}_V = \mathcal{P} \cap \mathcal{G}_V$; we define

$$\operatorname{dist}(G, \mathcal{P}) = \min_{H \in \mathcal{P}_V} d(G,H)$$

when $\mathcal{P}_V \neq \emptyset$, and set $\operatorname{dist}(G,\mathcal{P}) = 1$ otherwise. Thus, $\operatorname{dist}(G,\mathcal{P})$ is the smallest fraction of pairs whose adjacency status must be changed to obtain a graph in $\mathcal{P}$ on the same vertex set. We say that G is ε-*far* from $\mathcal{P}$ if $\operatorname{dist}(G,\mathcal{P}) \geq \varepsilon$. Graphs of size less than two can be handled separately.

*Trinity College Cambridge, Department of Pure Mathematics and Mathematical Statistics, Centre for Mathematical Sciences, Wilberforce Road, Cambridge CB3 0WA, United Kingdom. Email: gc645@cam.ac.uk

†Cheriton School of Computer Science, University of Waterloo, 200 University Ave W, Waterloo, ON N2L 5Z5, Canada. Email: cjmpseth@uwaterloo.ca

ISSN 1433-8092

An algorithm receives the order n of the input graph and may query whether a given pair of vertices is an edge. An *ε-tester* for $\mathcal{P}$ accepts every graph in $\mathcal{P}$ with probability at least $2/3$ and rejects every graph that is ε-far from $\mathcal{P}$ with probability at least $2/3$. No requirement is imposed on graphs whose distance lies strictly between 0 and ε. The property is *testable* if, for every $\varepsilon > 0$, such a tester exists with a number of queries bounded independently of n.

Throughout the paper, $q(\varepsilon)$ denotes an upper bound on the number of *edge queries*, valid for every input size. We often use *canonical testers*, which sample a uniformly random set of vertices and inspect its induced subgraph. The canonization theorem of Goldreich and Trevisan [2] converts a tester making q edge queries into a canonical tester sampling $O(q)$ vertices. Conversely, inspecting the subgraph induced by s vertices uses at most $\binom{s}{2}$ edge queries. These polynomial conversions allow us to pass between the two models without changing whether a bound is polynomial. We nevertheless keep track of the distinction, using s for the size of a canonical sample.

Goldreich, Goldwasser and Ron [3] introduced graph property testing in this model, and proved that a broad class of partition properties, including k-colourability and properties prescribing a large cut, are testable. Subsequent work established general criteria for testability. In particular, Alon and Shapira [4] proved that every *hereditary* graph property, meaning one closed under taking induced subgraphs, is testable.

Regularity methods play a central role in these general results. They replace a large graph with a bounded description that preserves enough information to detect violations of the property. The quantitative cost of employing these methods can, however, be very large: Szemerédi's regularity lemma has tower-type bounds, and Gowers [5] showed that such dependence is unavoidable for the lemma itself. This leaves open the possibility that a particular testing problem admits a much better bound through a more direct argument or a weaker approximation theorem. For example, the regularity-based argument of Rödl and Duke [6] for k-colourability was followed by polynomial testing bounds obtained without the full regularity lemma [3, 8, 9].

Understanding when such improvements are possible is a central question in graph property testing. Related work of Carenini, Seth and Yoshida [7] gives a quantitative container characterization of one-sided testability, connecting prescribed testing complexity to container-like descriptions. This develops a different route to quantitative testing results, building on the hypergraph container method of Balogh, Morris and Samotij [10] and Saxton and Thomason [11]. Here, we study the quantitative relationship between testing and the stronger task of distance estimation.

1.2 Distance estimation and previous bounds

A tester distinguishes distance zero from distance at least ε. A distance estimator must approximate the distance on every input. More precisely, an *ε-estimator* for $\mathcal{P}$ outputs a number $\widehat{d}$ satisfying

$$\Pr\bigl[|\widehat{d} - \operatorname{dist}(G, \mathcal{P})| \le \varepsilon\bigr] \ge \frac{2}{3}$$

for every graph G. We call $\mathcal{P}$ *estimable* if it admits such an algorithm with query complexity depending only on ε. Estimability implies testability, after changing the accuracy by a constant factor.

Parnas, Ron and Rubinfeld [12] initiated the systematic study of tolerant testing and distance approximation. Testing need not imply distance estimation in other models [13]. In the dense graph model, however, Fischer and Newman [14] proved that every testable property is estimable. Their proof uses regularity and incurs a tower-type loss in converting a tester into an estimator. The natural quantitative question is how much of this loss is necessary.

This question was first studied for hereditary properties by Hoppen, Kohayakawa, Lang, Lefmann and Stagni [15] and by Fiat and Ron [16]. In particular, can a hereditary property that is testable with polynomially many queries also be estimated with polynomially many queries? The following formulation asks for a quantitative conversion for arbitrary testing bounds.

Problem 1 (Testing versus estimation)**.** Suppose that a graph property $\mathcal{P}$ has an ε-tester using $q(\varepsilon)$ queries. Can its distance be estimated to additive accuracy ε using $\mathrm{poly}\big(\max\{1, q(c\varepsilon)\}, 1/\varepsilon\big)$ queries, for an absolute constant $c > 0$?

For hereditary properties, earlier bounds are most precisely expressed using induced-removal lemma parameters. At proximity η, these are numbers $M = M(\eta)$, $\delta = \delta(\eta) > 0$ and $n_0 = n_0(\eta)$ such that every η-far graph G of order at least n_0 contains some $F \notin \mathcal{P}$, with $|V(F)| \le M$, for which a uniformly random injective map from $V(F)$ to $V(G)$ induces a copy of F with probability at least δ. Such parameters give a one-sided tester with vertex sample size $O(\max\{n_0, M/\delta\})$.

Hoppen, Kohayakawa, Lang, Lefmann and Stagni [15] obtained distance estimators with a doubly exponential bound in terms of these removal parameters. Gishboliner, Kushnir and Shapira [17] improved this to a singly exponential bound.

Theorem 1 (Gishboliner–Kushnir–Shapira [17])**.** *Let $\mathcal{P}$ be hereditary. For an absolute constant $c_0 > 0$, let M, δ, n_0 be induced-removal parameters at proximity $c_0\varepsilon$. Then, $\mathcal{P}$ has an ε-estimator with query complexity $2^{\mathrm{poly}(M/\delta, \log n_0)}$.*

Using the same tools, Gishboliner, Kushnir and Shapira also obtained a doubly exponential conversion for general graph properties [17]. Thus, the remaining question was whether a polynomial loss suffices.

1.3 Our main result

We prove that every tester in the dense graph model can be converted into a distance estimator with polynomial loss.

Theorem 2 (Testing implies estimation with polynomial loss)**.** *There is an absolute constant $c > 0$ with the following property. If $\mathcal{P}$ has an ε-tester with edge-query complexity $q(\varepsilon)$ for every $\varepsilon \in (0, 1)$, then it has an ε-estimator with edge-query complexity $\mathrm{poly}\big(\max\{1, q(c\varepsilon)\}, 1/\varepsilon\big)$.*

In particular, a graph property is testable with $\mathrm{poly}(1/\varepsilon)$ queries if and only if its distance can be estimated with $\mathrm{poly}(1/\varepsilon)$ queries. The theorem applies to general graph properties and allows the original tester to be adaptive and to have two-sided error.

For natural properties, the explicit dependence on $1/\varepsilon$ can be absorbed into the testing complexity. Following [17], call $\mathcal{P}$ *unnatural* if there exists $\varepsilon_0 > 0$ such that, for every $0 < \eta < \varepsilon_0$ and all sufficiently large n, either $\mathcal{P}_{[n]} = \emptyset$ or every n-vertex graph has distance at most η from $\mathcal{P}$. A property is *natural* if it is not unnatural. A hereditary property is *non-trivial* if it contains graphs of arbitrarily large order and omits at least one graph.

Corollary 1. *If $\mathcal{P}$ is natural, or is a non-trivial hereditary property, then the bound in Theorem 2 simplifies to $\mathrm{poly}\big(q(c\varepsilon)\big)$ for all sufficiently small ε.*

Indeed, the query lower bounds in [17, Propositions 1.7 and 1.10] give $q(\varepsilon) = \Omega(1/\varepsilon)$ for the edge-query complexity in these cases. The factor $1/\varepsilon$ in Theorem 2 can therefore be absorbed into the polynomial.

2 Main technical contribution and comparison with previous approaches

2.1 Previous approaches: estimating through partitions

A sufficiently regular partition of a graph, together with the edge densities between its parts, approximately determines the densities of small induced subgraphs. For a hereditary property, the

induced removal lemma turns this information into a way of detecting graphs that are far from the property. This suggests estimating distance by searching over modifications of the partition's density values. The search asks how cheaply those values can be changed so that they are compatible with a graph close to the property.

To justify this procedure, one must relate changes in the description to changes in the graph itself. Suppose that the prescribed densities change by γ on average, weighted by the sizes of the corresponding pairs of parts. One needs to realize the new densities with approximately $\gamma\binom{n}{2}$ edge changes while retaining the regularity required for counting. This *continuity* between editing the description and editing the graph is a key ingredient in the Fischer–Newman argument [14]. The large number of parts required by regularity accounts for its tower-type bounds.

Hoppen et al. [15] used the weaker approximation supplied by the Frieze–Kannan lemma [18], whose partitions have size exponential in a polynomial of the inverse accuracy. Weak regularity still controls the density of each fixed induced graph, but transferring edits while preserving the required approximation is more delicate. Their method combines weakly regular partitions with partitions that have no regularity requirement, leading to a doubly exponential bound. Gishboliner, Kushnir and Shapira [17] established an appropriate continuity statement: a slightly stronger form of weak regularity before editing is enough to retain ordinary weak regularity afterwards. This underlies their improved bounds.

2.2 Our approach

Our proof begins by expressing the distance as the approximate minimum of a single objective function built from the tester. We then show that this minimum is attained approximately by a weighted graph described using short labels on its vertices. Optimizing over these labels is a dense constraint satisfaction problem, whose optimum can be estimated from a small random sample. We overview each of these steps below.

Replacing membership by a penalty. Fix a target accuracy ε and let $\tau = \varepsilon/100$. After amplification and canonization, the tester samples $s = O\big(\max\{1, q(\tau)\}\log(1/\tau)\big)$ vertices and rejects according to a symmetric rule $f : \mathcal{G}_s \to [0,1]$. For an n-vertex graph J, consider the following rejection rule $\mathrm{rej}(J) = \mathbb{E}_{\mathbf{v}} f(J[\mathbf{v}])$, where $\mathbf{v}$ is a uniformly random ordered tuple of s distinct vertices. The rule f may depend on n; throughout the argument, n and its associated rule are fixed. The tester satisfies

$$\begin{cases} J \in \mathcal{P} & \implies \mathrm{rej}(J) \le \tau, \\ \mathrm{dist}(J, \mathcal{P}) \ge \tau & \implies \mathrm{rej}(J) \ge \frac{2}{3}. \end{cases}$$

For the input graph G on V, consider $M(G) = \min_{J \in \mathcal{G}_V} \big\{ d(G,J) + 2\,\mathrm{rej}(J) \big\}$. When $\mathcal{P}_V \neq \emptyset$, this minimum satisfies

$$\mathrm{dist}(G,\mathcal{P}) - \tau \le M(G) \le \mathrm{dist}(G,\mathcal{P}) + 2\tau. \tag{1}$$

For the upper bound, take a closest graph in $\mathcal{P}$; its edit cost is exactly $\mathrm{dist}(G,\mathcal{P})$ and its penalty is at most 2τ. For the lower bound, a graph J within distance τ of $\mathcal{P}$ has $d(G,J) \ge \mathrm{dist}(G,\mathcal{P}) - \tau$, by the triangle inequality. Every other J pays a penalty of at least $4/3$, which already exceeds the maximum possible distance. Thus the tester supplies all the information about $\mathcal{P}$ needed by this objective.

We next allow the edges of a candidate graph to have weights in $[0,1]$. For a weighted graph W, let $\mathbb{G}(W)$ be the random graph obtained by rounding its edges independently, with edge probabilities given by W. Extend the objective by setting

$$\mathrm{cost}_G(W) = \mathbb{E}\, d(G, \mathbb{G}(W)), \qquad \mathrm{rej}(W) = \mathbb{E}\ \mathrm{rej}(\mathbb{G}(W)), \qquad \mathcal{E}_G(W) = \mathrm{cost}_G(W) + 2\,\mathrm{rej}(W).$$

Then,

$$\mathcal{E}_G(W) = \mathbb{E}\big[d(G, \mathbb{G}(W)) + 2\,\mathrm{rej}(\mathbb{G}(W))\big].$$

Consequently, allowing weights does not change the minimum: every weighted objective value is an average of ordinary graph objective values. This exact relaxation makes it possible to approximate candidate graphs by bounded real-valued functions.

Approximating a candidate using short vertex labels. The approximation only needs to preserve two quantities: the edit cost and the rejection probability of the fixed tester. The first is affine in the edge weights:

$$\mathrm{cost}_G(W) = \mathbb{E}_{uv}\big[G(uv) + (1 - 2G(uv))W(uv)\big].$$

Thus, its change is measured by correlation with the single function $\mathbf{1} - 2G$.

To control rejection probability, we compare two weighted graphs by changing the $\binom{s}{2}$ edge probabilities in a sampled graph one at a time. Each change is an edge-weight difference multiplied by a discrete derivative of the tester's rejection rule. The important point is how these changes can be described.

Suppose the changing edge occupies positions 1 and 2 in the sample. Call positions $3, \dots, s$ the *anchors*. After rounding the other edges, the change, which we call *the derivative*, is determined by the graph b on the anchors and the two adjacency vectors x, y from the endpoints to the anchors:

$$\partial_b f(x, y) = f\big(F_1(b, x, y)\big) - f\big(F_0(b, x, y)\big),$$

where $F_t(b, x, y)$ is the graph with these prescribed adjacencies and with the edge 12 set to t. In the comparison between two weighted graphs, a vertex may need a different adjacency vector for each of the two endpoint roles. Recording both vectors, and symmetrizing in the endpoints, gives a bounded test of an unordered pair.

In this way we obtain a family $\mathcal{T}$ of functions on vertex pairs. Each derivative test is specified by $O(s^2)$ bits of common data and $O(s)$ bits at each vertex; we also include the cost test $\mathbf{1} - 2G$. With $\|R\|_{\mathcal{T}} = \max_{A \in \mathcal{T}} \big|\mathbb{E}_{uv} R(uv)A(uv)\big|$, the edge-by-edge comparison gives, for sufficiently large n,

$$|\mathcal{E}_G(W) - \mathcal{E}_G(Z)| \le \left(1 + 4\binom{s}{2}\right) \|W - Z\|_{\mathcal{T}}. \tag{2}$$

This is the counting lemma needed by the proof.

The use of restrictions of the tester is inspired by Dwork and Tankala [19, Definition 3.1 and Lemma 3.2]. Here, the tester observes an induced graph, so its queried pairs share vertices. The endpoint labels retain these shared incidences and make the resulting tests compatible with sampling vertices.

A weak regularity argument now provides the required approximation. Starting from G, maintain a sum of signed tests and use its clipped value as the current approximation. Whenever a test distinguishes this approximation from a target weighted graph H, add a small signed multiple of that test to the sum. A bounded potential shows that fewer than γ^{-2} updates suffice to obtain

$$W = \mathrm{clip}\left(G + \gamma \sum_{j=1}^{t} B_j\right), \qquad t < \gamma^{-2}, \qquad \|H - W\|_{\mathcal{T}} \le \gamma,$$

where $B_j \in \mathcal{T} \cup (-\mathcal{T})$ and $\mathrm{clip}(a) = \min\{1, \max\{0, a\}\}$. This is a form of weak regularity adapted to the tester, following the energy-increment principle of Frieze and Kannan [18] and the boosting perspective of Trevisan, Tulsiani and Vadhan [20].

Combining the descriptions of the tests gives a label z_v at each vertex, drawn from a finite set Λ with $\log|\Lambda| = O(s^2\gamma^{-2})$. The *approximating weighted graph* is recovered by a fixed symmetric function D:

$$W_z(uv) = D\big(G(uv), z_u, z_v\big). \tag{3}$$

Thus, the labels describe the approximation relative to the original graph G. Common values are included in each label, and D is defined for every pair of labels, including pairs whose common values disagree. Every labeling therefore specifies a valid weighted graph, while the boosting argument guarantees that some labeling approximates any given target H.

Taking γ to be a sufficiently small constant multiple of ε/s^2, equation (2) gives

$$M(G) \le \widetilde{M}(G) := \min_{z\in\Lambda^V} \mathcal{E}_G(W_z) \le M(G) + O(\varepsilon), \qquad \log|\Lambda| = O(s^6\varepsilon^{-2}).$$

The number of possible labels may be exponential, but their length is polynomial in s and $1/\varepsilon$.

Estimating the optimum by sampling vertices. For a fixed labeling, both terms of $\mathcal{E}_G(W_z)$ can be written as averages of losses on ordered s-tuples of distinct vertices. The rejection term already has this form; for the edit term, average the edge-edit losses over the $\binom{s}{2}$ pairs in the tuple. By equation (3), the loss on a tuple depends only on its induced subgraph in G and the labels of its vertices.

It follows that minimizing $\mathcal{E}_G(W_z)$ is a dense s-ary constraint satisfaction problem with alphabet Λ. Each vertex is a variable, its possible labels are the alphabet, and each tuple contributes one bounded loss. Sampling a set U of vertices restricts this same instance to U; the decoder and the rejection rule f_n remain fixed. This is why labels on vertices are useful.

The subsampling theorem of Bernasconi, Castiglioni, Celli, Farina, and Malavolta [21, Theorem 3.3] approximates the optimum in expectation using a sample whose size is polynomial in the arity, the inverse accuracy, and the logarithm of the alphabet size. The theorem also requires a lower bound on the original instance size, polynomial in the same parameters. Both requirements are satisfied here. Small input graphs can be read in full; for larger graphs, averaging a polynomial number of independent sample optima gives the required success probability.

The sampled optimum can be computed by exhaustive search over labels, using only the edges inside the sampled vertex sets. Combining this estimate with the short-description approximation and equation (1) yields an estimate of $\mathrm{dist}(G, \mathcal{P})$ with query complexity

$$\mathrm{poly}(s, 1/\varepsilon) = \mathrm{poly}\big(\max\{1, q(c\varepsilon)\}, 1/\varepsilon\big)$$

. The polynomial dependence on the arity and the logarithmic dependence on the number of labels are both essential: the arity grows with the tester, and the label set is exponentially large. The boosting argument establishes the existence of a suitable description, and the estimator only computes optimum values on its samples.

3 Preliminaries

We first put the tester into a form that can be applied to weighted graphs. The weights let us describe gradual changes to a candidate repair. Independent rounding will then show that introducing weights does not change the optimum of the objective function introduced in Section 2.

For every integer $k \ge 0$, let $[k] = \{1, \dots, k\}$, with $[0] = \emptyset$. For integers a, b, let $[a, b] = \{j \in \mathbb{Z} : a \le j \le b\}$; in particular, $[3, 2] = \emptyset$. Given a set X, write $\binom{X}{k}$ for its k-element subsets and $X^k_{\neq}$ for its ordered k-tuples of pairwise distinct elements. The set $X^0_{\neq}$ consists of the empty tuple. For $0 \le k \le n$, write $(n)_k = n(n-1)\cdots(n-k+1)$, with $(n)_0 = 1$. For distinct $u, v \in X$, write

$uv = \{u, v\}$. All logarithms have base 2. Unless stated otherwise, averages over a finite set are uniform. The empty-set conventions allow a tester with $s = 2$ to have no anchors.

Let V be a finite set. A *graph* on V is a function $G : \binom{V}{2} \to \{0, 1\}$. The set of these is $\mathcal{G}_V$. A *weighted graph* on V is a function $W : \binom{V}{2} \to [0, 1]$. The value $W(uv)$ represents the probability that uv is an edge. The set of these is $\mathcal{W}_V$. We have that $\mathcal{G}_V \subseteq \mathcal{W}_V$, indeed unweighted graphs are a special case of weighted graphs. Note that when $|V| < 2$, $\binom{V}{2} = \emptyset$, and $\mathcal{G}_V = \mathcal{W}_V$ consist of the unique empty function.

For $V = [s]$, we use the following shorthands $\mathcal{G}_s, \mathcal{W}_s$. For $W \in \mathcal{W}_V$, $\mathbb{G}(W)$ is the random graph in $\mathcal{G}_V$ whose edges are sampled independently at random with $\Pr[\mathbb{G}(W)(e) = 1] = W(e)$. If W is a graph, $\mathbb{G}(W) = W$. For $\mathbf{v} = (v_1, \ldots, v_s) \in V^s_{\neq}$, the *induced weighted graph* $W[\mathbf{v}] \in \mathcal{W}_s$ is $W[\mathbf{v}](ab) = W(v_a v_b)$.

Let $n = |V| \geq 2$. For functions $X, Y : \binom{V}{2} \to \mathbb{R}$, let $\mathbb{E}_e$ denote expectation over a uniformly random unordered pair $e \in \binom{V}{2}$, and define $\langle X, Y \rangle = \mathbb{E}_e[X(e)Y(e)]$, and $\|X\|_2 = \big(\langle X, X \rangle\big)^{1/2}$. Thus, inner products measure average correlation, independently of the order of the graph. Averages indexed by uv use the same unordered-pair convention; when ordered pairs are needed, we write them as (u, v). We use $\mathbf{1}$ to denote the constant function 1, and $\mathbf{1}[\mathcal{B}]$ for the indicator of a condition $\mathcal{B}$. The *distance* between graphs $G, H \in \mathcal{G}_V$ is given by

$$d(G, H) = \mathbb{E}_e|G(e) - H(e)| = \frac{|E(G) \triangle E(H)|}{\binom{n}{2}} \in [0, 1].$$

A *graph property* $\mathcal{P}$ is a set of graphs closed under isomorphism, and $\mathcal{P}_V = \mathcal{P} \cap \mathcal{G}_V$. The *distance to* $\mathcal{P}$ is

$$\operatorname{dist}(G, \mathcal{P}) = \begin{cases} \min_{H \in \mathcal{P}_V} d(G, H) & \mathcal{P}_V \neq \emptyset,\ n \geq 2, \\ 1 & \mathcal{P}_V = \emptyset, \\ 0 & \mathcal{P}_V \neq \emptyset,\ n < 2. \end{cases}$$

G is *ε-far* from $\mathcal{P}$ if $\operatorname{dist}(G, \mathcal{P}) \geq \varepsilon$. For graphs G, J with $\mathcal{P}_V \neq \emptyset$, $\operatorname{dist}(G, \mathcal{P}) \leq d(G, J) + \operatorname{dist}(J, \mathcal{P})$.

The next proposition replaces the internal operation of the tester by a single rejection rule on a small induced graph. This is useful because we can then study how that rule changes when one sampled edge changes. Amplification makes rejection rare on graphs in the property; this will make the penalty small for a genuine repair. The original input order n remains available to the tester, so its rule is allowed to depend on n.

We use the canonization theorem of Goldreich and Trevisan [2] in the following functional form.

Proposition 1. Let $\mathcal{P}$ be a property. Fix $\tau \in (0, 1/3]$ and a τ-tester for $\mathcal{P}$ that requires at most $q(\tau)$ queries. Then, there exist an integer $s = O(\max\{1, q(\tau)\} \log(1/\tau)) \geq 2$ and, for each $n \geq s$, a function $f = f_n : \mathcal{G}_s \to [0, 1] \cap \mathbb{Q}$, where $f(J)$ is the probability of rejecting when the induced graph on the s sampled vertices is J, such that the following two conditions hold:

(N1) $f(J \circ \sigma) = f(J)$ for all $J \in \mathcal{G}_s$ and $\sigma \in \operatorname{Sym}([s])$, where $(J \circ \sigma)(ab) = J(\sigma(a)\sigma(b))$;

(N2) for every V with $|V| = n$ and $J \in \mathcal{G}_V$, writing $\operatorname{rej}(J) = \mathbb{E}_{\mathbf{v}} f(J[\mathbf{v}])$ with $\mathbf{v}$ uniform in $V^s_{\neq}$,

$$\begin{cases} J \in \mathcal{P} & \Rightarrow\ \operatorname{rej}(J) \leq \tau, \\ \operatorname{dist}(J, \mathcal{P}) \geq \tau & \Rightarrow\ \operatorname{rej}(J) \geq \frac{2}{3}. \end{cases} \tag{4}$$

Proof. Amplify the tester by $O(\log(1/\tau))$ independent repetitions and a majority vote, so that its rejection probability is at most τ on members and at least $2/3$ on τ-far graphs. Apply the canonization theorem [2] to obtain a tester inspecting $s = O(\max\{1, q(\tau)\} \log(1/\tau))$ uniformly sampled

vertices, padding to $s \geq 2$ if necessary. Average its decision rule over all permutations of the sampled positions. This does not change the rejection probability on any input graph, and gives (N1). The function f_n is the resulting conditional rejection probability, giving (N2). With the usual finite-randomness convention for a randomized tester, these probabilities are rational. □

From now on, n and its associated rule $f = f_n$ are fixed whenever we compare graphs on V. In particular, restricting an optimization problem to a smaller vertex set later in the proof will not replace f_n by the tester for that smaller order.

Recall that a function $g : \mathcal{W}_V \to \mathbb{R}$ is *multilinear* if it is affine in each coordinate $W(e)$ separately. Such a function treats an edge of weight p exactly as an average of an absent edge and a present edge, with weights $1 - p$ and p. The next fact applies this observation to all edges: a weighted input cannot have a value smaller than every ordinary graph obtained by rounding it.

Fact 1. If $g : \mathcal{W}_V \to \mathbb{R}$ is multilinear, then $g(W) = \mathbb{E}\, g(\mathbb{G}(W))$ for every $W \in \mathcal{W}_V$. Hence, $\min_{\mathcal{W}_V} g = \min_{\mathcal{G}_V} g$.

Proof. We round one edge at a time. Affinity in the edge being rounded preserves the expected value at each step, and after all steps we have an ordinary graph. List the pairs $e_1, \dots, e_N$, $N = \binom{n}{2}$, and write $W_i = W(e_i)$. For $0 \leq k \leq N$, let

$$g_k(W) = \sum_{x \in \{0,1\}^k} g(x, W_{k+1}, \dots, W_N) \prod_{i \leq k} W_i^{x_i} (1 - W_i)^{1 - x_i}.$$

Affinity in W_{k+1} gives $g_k = g_{k+1}$, and therefore $g = g_0 = g_N = \mathbb{E}\, g(\mathbb{G}(W))$. This is a weighted average of values $g(J)$, $J \in \mathcal{G}_V$, so it is at least their minimum. The minimum over $\mathcal{W}_V$ exists by compactness. □

We now extend the tester by interpreting each edge weight as an independent edge probability. The function ϕ is the rejection probability on one weighted sample of size s; the function rej averages this quantity over all samples in the full graph. Keeping these two levels separate will let us analyze one sampled edge before averaging over the input graph. Define $\phi : \mathcal{W}_s \to [0,1]$ and $\mathrm{rej} : \mathcal{W}_V \to [0,1]$ by

$$\begin{cases} \phi(Y) = \mathbb{E}\, f(\mathbb{G}(Y)) = \sum_{J \in \mathcal{G}_s} f(J) \prod_{ab \in \binom{[s]}{2}} Y(ab)^{J(ab)} \big(1 - Y(ab)\big)^{1 - J(ab)}, \\ \mathrm{rej}(W) = \mathbb{E}_{\mathbf{v} \in V^s_{\neq}}\, \phi\big(W[\mathbf{v}]\big). \end{cases}$$

The next fact says that the extension retains relabeling symmetry and is compatible with rounding. Thus we may either round the whole graph and run the tester, or sample first and then round the sampled edges.

Fact 2. The following properties hold:

(a) $\phi(Y \circ \sigma) = \phi(Y)$ for $Y \in \mathcal{W}_s$, $\sigma \in \mathrm{Sym}([s])$;

(b) ϕ and rej are multilinear;

(c) $\mathrm{rej}(W) = \mathbb{E}\ \mathrm{rej}(\mathbb{G}(W))$, and on graphs this rej agrees with that of (N2).

Proof. Each assertion follows because relabeling and taking an induced sample preserve the independent-edge model. More explicitly:

(a) $\mathbb{G}(Y \circ \sigma)$ has the law of $\mathbb{G}(Y) \circ \sigma$; use (N1).

(b) Each summand of ϕ is a product of factors affine in distinct coordinates, and rej is an average of ϕ composed with coordinate projections.

(c) Fact 1; on graphs $\phi(J[\mathbf{v}]) = f(J[\mathbf{v}])$.

□

4 Proofs

4.1 Derivatives of the tester, tests, and the counting lemma

The first step is to identify which features of a candidate repair need to be preserved. There are only two relevant quantities: its expected edit cost and its rejection probability. We will construct a family of pair functions whose correlations control changes in both quantities.

Fix $0 < \tau \le 1/3$ and $s, f = f_n$ as in Proposition 1. Let $G \in \mathcal{G}_V$ with $n \ge s$, and put $K = \binom{s}{2}$, the number of pairs inspected in a canonical sample. Recall the definitions of $\mathrm{cost}_G : \mathcal{W}_V \to [0,1]$ and $\mathcal{E}_G : \mathcal{W}_V \to [0,3]$:

$$\mathrm{cost}_G(W) = \mathbb{E}\, d\big(G, \mathbb{G}(W)\big), \qquad \mathcal{E}_G(W) = \mathrm{cost}_G(W) + 2\ \mathrm{rej}(W), \qquad M(G) = \min_{W \in \mathcal{W}_V} \mathcal{E}_G(W).$$

The objective charges for editing the graph and for leaving a graph that the tester is likely to reject. We first observe that the edit part is particularly simple: on a non-edge of G, increasing the weight increases the cost, while on an edge of G it decreases the cost. The function $\mathbf{1} - 2G$ records exactly these two signs.

Fact 3. The function $\mathrm{cost}_G : \mathcal{W}_V \to [0,1]$ satisfies the equality $\mathrm{cost}_G(W) = \langle G, \mathbf{1}\rangle + \langle W, \mathbf{1} - 2G\rangle$. In particular, cost_G is affine and $\mathrm{cost}_G(J) = d(G,J)$ for $J \in \mathcal{G}_V$.

Proof. Compute the cost for one pair, then average over pairs and over the rounding. For $a, b \in \{0,1\}$, $|a-b| = a + b - 2ab$. So $d(G,H) = \langle G, \mathbf{1}\rangle + \langle H, \mathbf{1} - 2G\rangle$ for graphs H. Take expectations, using $\mathbb{E}\,\mathbb{G}(W)(e) = W(e)$. □

The next lemma explains why we can minimize this objective instead of searching only over graphs in $\mathcal{P}$. A genuine repair has a small penalty. A graph that remains far from $\mathcal{P}$ pays more than the largest possible edit distance, while a graph already close to $\mathcal{P}$ cannot save much on the cost of a genuine repair. Independent rounding also shows that fractional edge weights introduce no better minimum.

Lemma 1. The weighted and unweighted minima agree, i.e., $M(G) = \min_{J \in \mathcal{G}_V} \big\{ d(G,J) + 2\,\mathrm{rej}(J) \big\}$. Further, if $\mathcal{P}_V \neq \emptyset$, then $\mathrm{dist}(G,\mathcal{P}) - \tau \le M(G) \le \mathrm{dist}(G,\mathcal{P}) + 2\tau$.

Proof. We first remove the weights using rounding, then compare an arbitrary candidate with a closest genuine repair. By Facts 3 and 2, $\mathcal{E}_G$ is multilinear, so Fact 1 gives the equality. For the distance bounds, assume $\mathcal{P}_V \neq \emptyset$. For the upper bound, take $H \in \mathcal{P}_V$ with $d(G,H) = \mathrm{dist}(G,\mathcal{P})$. By (4), $\mathcal{E}_G(H) \le \mathrm{dist}(G,\mathcal{P}) + 2\tau$. For the lower bound, let $J \in \mathcal{G}_V$. If $\mathrm{dist}(J,\mathcal{P}) < \tau$, then $d(G,J) \ge \mathrm{dist}(G,\mathcal{P}) - \mathrm{dist}(J,\mathcal{P}) > \mathrm{dist}(G,\mathcal{P}) - \tau$. If $\mathrm{dist}(J,\mathcal{P}) \ge \tau$, then $2\,\mathrm{rej}(J) \ge 4/3 > 1 \ge \mathrm{dist}(G,\mathcal{P})$ by (4). Either way $\mathcal{E}_G(J) \ge \mathrm{dist}(G,\mathcal{P}) - \tau$. □

The coefficient 2 makes the lower-bound comparison work: a τ-far graph has penalty at least $4/3 > 1 \ge \mathrm{dist}(G,\mathcal{P})$. With coefficient 1, the soundness guarantee would give a penalty of only $2/3$, which could be too small to prevent such a graph from undercutting the true distance.

We next isolate the information needed to measure the effect of one sampled edge. Designate its endpoints as positions 1 and 2, and call positions $3, \dots, s$ the *anchors*. Once the graph on the anchors and the two endpoint-to-anchor adjacency vectors are fixed, the sampled graph is completely determined except for the edge 12. The gadget below simply records this decomposition; it adds no further graph structure.

Definition 1 (Gadget F_t)**.** For a graph $b \in \mathcal{G}_{[3,s]}$, vectors $x, y \in \{0,1\}^{[3,s]}$ and integer $t \in \{0,1\}$, let $F_t(b,x,y) : \binom{[s]}{2} \to \{0,1\}$ be the graph on $[s]$ given by:

- $F_t(b,x,y)(\{1,2\}) = t$;

- $F_t(b,x,y)(\{1,a\}) = x_a$ for every $a \in [3,s]$;
- $F_t(b,x,y)(\{2,a\}) = y_a$ for every $a \in [3,s]$;
- $F_t(b,x,y)(\{a,c\}) = b(\{a,c\})$ for every $a \neq c \in [3,s]$.

So x_a encodes whether position 1 is adjacent to anchor a, y_a does the same for position 2, b is the graph induced on the anchors, and t is the value of the edge $\{1,2\}$.

The derivative below measures how much the tester notices a change to the remaining edge. It can be positive or negative, since adding an edge need not make rejection more likely. This discrete difference is also the coefficient of the edge probability when the other sampled edges have been fixed.

Definition 2 (Derivative of f)**.** The *derivative of f in the edge* 12 *at* b is $\partial_b f : \{0,1\}^{[3,s]} \times \{0,1\}^{[3,s]} \to [-1,1]$:

$$\partial_b f(x,y) = f\big(F_1(b,x,y)\big) - f\big(F_0(b,x,y)\big).$$

To compare two weighted graphs, we will replace the K sampled edge probabilities one at a time. At an intermediate stage, some positions use weights from the first graph and others from the second. A vertex can therefore need one adjacency vector when it occupies endpoint position 1 and another when it occupies position 2. We store both vectors in a single *color*. The extra color $\perp$ marks the anchors themselves, so that a test ignores pairs meeting an anchor.

Let $\Omega = \{\perp\} \sqcup \big(\{0,1\}^{[3,s]} \times \{0,1\}^{[3,s]}\big)$. For each $b \in \mathcal{G}_{[3,s]}$, define the symmetric function $\kappa_b : \Omega \times \Omega \to [-1,1]$ by

$$\begin{cases} \kappa_b\big((x,y),(x',y')\big) = \frac{1}{2}\big(\partial_b f(x,y') + \partial_b f(x',y)\big), \\ \kappa_b(\perp,\omega) = \kappa_b(\omega,\perp) = 0. \end{cases}$$

By (N1), $\partial_b f(x,y) = \partial_b f(y,x)$: exchanging positions 1 and 2 is a relabeling. The reason for averaging in κ_b is different. The role-dependent expression $\partial_b f(x,y')$ need not equal $\partial_b f(x',y)$ when the two vertices have different vectors for their two roles. Averaging these two orientations makes the test a well-defined function of an unordered pair.

A *coloring* $\chi : V \to \Omega$ assigns this local information to every vertex. For each b and χ, define the *test* $A_{b,\chi}(uv) = \kappa_b\big(\chi(u),\chi(v)\big)$. Thus a single coloring describes the response to every possible endpoint pair, with the same anchor graph b. We allow all colorings; they need not come from actual neighborhoods in one graph. This larger family contains the random colorings used in the counting proof and is easy to encode by vertex labels.

Set

$$\mathcal{A} = \{A_{b,\chi} : b \in \mathcal{G}_{[3,s]},\ \chi : V \to \Omega\}, \qquad \mathcal{T} = \mathcal{T}(G) = \{\mathbf{1} - 2G\} \cup \mathcal{A}.$$

For $R : \binom{V}{2} \to \mathbb{R}$, define

$$\|R\|_{\mathcal{T}} = \max_{A \in \mathcal{T}} |\langle R, A\rangle|.$$

This seminorm measures the largest average discrepancy visible to any of these tests. A small value of $\|W - Z\|_{\mathcal{T}}$ means that the tests have little ability to distinguish W from Z; it does not assert that every edge weight is close. The cost test $\mathbf{1} - 2G$ is included because its correlation is exactly the change in edit cost.

Every test takes values in $[-1,1]$. A derivative test is specified by $O(s^2)$ common bits for b and $O(s)$ bits per vertex for $\chi(v)$. The cost test is instead recovered directly from the input edge $G(uv)$. This distinction will explain why the decoder later reads both endpoint labels and the original edge.

Example ($s = 3$, triangles). Consider the rejection rule $f(J) = \mathbf{1}[J \text{ is a triangle}]$. There is one anchor, so b has no edges and $\partial_b f(x,y) = xy$ for $x,y \in \{0,1\}$. Fix an anchor vertex w in an

ordinary graph J, and write $N_J(w)$ for its neighborhood. Give w color $\perp$ and every other vertex v the color $(\mathbf{1}[v \in N_J(w)], \mathbf{1}[v \in N_J(w)])$. Then

$$A_{b,\chi}(uv) = \mathbf{1}[u \in N_J(w)]\mathbf{1}[v \in N_J(w)].$$

The test detects changes on pairs inside the neighborhood of w. Summing these indicators over w counts common neighbors of u and v: these are exactly the vertices that can complete the changing edge to a triangle. The general derivative tests play the same role for an arbitrary rejection rule.

The counting lemma says that these tests capture everything needed to control the objective. For some intuition towards the proof of the counting lemma, we continue with the triangle example. Let W and Z be two graphs that have a large difference in their rejection probabilities (and for simplicity assume the edge weights are from $\{0, 1\}$). In other words, taking a random sample S of 3 vertices, the probability that the induced subgraph is a triangle is significantly different for W compared to Z. We wish to show that there exists $A \in \mathcal{T}$ for which $\langle W - Z, A\rangle$ is large.

To this end, consider a specific ordering e_1, e_2, e_3 of the edge locations between the 3 sampled vertices S, and consider interpolating between $W[S]$ and $Z[S]$ by starting with $W[S]$ and changing one edge at a time, in the order e_1, e_2, e_3, until reaching $Z[S]$. For the sets S for which the rejection rule differs for W and Z, there is some point in the interpolation for which the output of the rejection rule changes. By considering all such sets S, there exists a step in the interpolation at which the rejection rule changes for a significant fraction of these sets.

After selecting an anchor graph b on a single vertex w, we construct $A_{b,\chi}$ using an interpolated version of W and Z to define the labeling χ. Roughly speaking, the entries in $A_{b,\chi}$ correspond to pairs of vertices u, v for which the output of the rejection rule on the interpolated subgraph changes when flipping the edge u, v. These entries in $A_{b,\chi}$ line up with non-zero entries in $W - Z$, resulting in $\langle W - Z, A_{b,\chi}\rangle$ being large.

More generally, s is the canonical sample size and $K = \binom{s}{2}$. There are K possible edge changes in a sample; bounding each one by the seminorm and summing gives the dependence on K.

Lemma 2 (Counting)**.** If $n \geq 4s$ then for all $W, Z \in \mathcal{W}_V$,

$$\begin{cases} |\operatorname{rej}(W) - \operatorname{rej}(Z)| \leq 2K \|W - Z\|_{\mathcal{T}}, \\ |\mathcal{E}_G(W) - \mathcal{E}_G(Z)| \leq (1 + 4K) \|W - Z\|_{\mathcal{T}}. \end{cases}$$

Proof. We interpolate between Z and W on one sampled pair at a time. For each step, we express the tester's response as an average of derivative tests. The only normalization loss comes from excluding the anchors as endpoints.

List the pairs of positions $e_1, \ldots, e_K$ of $[s]$. For $0 \leq i \leq K$ and $\mathbf{v} \in V_{\neq}^s$ define $H_i(\mathbf{v}) \in \mathcal{W}_s$ by $H_i(\mathbf{v})(e_j) = W(v_a v_b)$ if $j \leq i$ and $Z(v_a v_b)$ if $j > i$, where $e_j = ab$. Let $\rho_i = \mathbb{E}_{\mathbf{v}}\phi(H_i(\mathbf{v}))$. Then $\rho_0 = \operatorname{rej}(Z)$ and $\rho_K = \operatorname{rej}(W)$.

Fix $i \geq 1$, $e_i = ab$. For $Y \in \mathcal{W}_s$ let $Y^{[t]}$ be Y with its value on ab replaced by $t \in \{0, 1\}$, and $\Delta(Y) = \phi(Y^{[1]}) - \phi(Y^{[0]}) \in [-1, 1]$. Since ϕ is affine in $Y(ab)$ and $H_i(\mathbf{v}), H_{i-1}(\mathbf{v})$ differ only on ab,

$$\rho_i - \rho_{i-1} = \mathbb{E}_{\mathbf{v}}\Big[(W - Z)(v_a v_b)\, \Delta\big(H_i(\mathbf{v})\big)\Big].$$

By Fact 2(a), we may relabel both the changing pair and the sampled positions without changing the derivative value or the sampling law. Thus we may assume $e_i = 12$; each other pair cd then carries a label $m_{cd} \in \{W, Z\}$ saying which function is used on it. Write $\mathbf{v} = (p, q, w)$ with $p = v_1$, $q = v_2$, $w = (w_c)_{c \in [3,s]}$, and let $\Delta(p, q, w) = \Delta(N)$ where $N(cd) = m_{cd}(v_c v_d)$ for $cd \neq 12$ and, say, $N(12) = 0$. The last value is immaterial because the derivative overwrites it. Then

$$\rho_i - \rho_{i-1} = \mathbb{E}_{(p,q,w)}\big[(W - Z)(pq)\, \Delta(p, q, w)\big]. \tag{5}$$

Fix $w \in V_{\neq}^{s-2}$. To realize the derivative by our tests, round the anchor graph once and independently round the two role-dependent adjacency vectors of each non-anchor vertex. Different pairs may share these random bits; we only need the correct distribution for each individual pair.

Let b be a random graph on $[3, s]$ with independent edges, $\Pr[b(cd) = 1] = m_{cd}(w_c w_d)$. For each $u \notin \{w_c\}$ let $X_u, Y_u \in \{0,1\}^{[3,s]}$ have independent coordinates with $\Pr[X_u(c) = 1] = m_{1c}(uw_c)$ and $\Pr[Y_u(c) = 1] = m_{2c}(uw_c)$, all variables independent. Let $\chi(u) = (X_u, Y_u)$ for $u \notin \{w_c\}$, $\chi(w_c) = \perp$, and $A = A_{b,\chi}$, a random element of $\mathcal{A} \subseteq \mathcal{T}$. For distinct non-anchors p, q, the graph $F_t(b, X_p, Y_q)$ has independent edges with exactly the probabilities of $N^{[t]}$. Hence $\mathbb{E} f(F_t(b, X_p, Y_q)) = \phi(N^{[t]})$, so $\mathbb{E}\, \partial_b f(X_p, Y_q) = \Delta(p, q, w)$ and

$$\mathbb{E} A(pq) = \tfrac{1}{2}\big(\Delta(p, q, w) + \Delta(q, p, w)\big),$$

while $A(pq) = 0$ if p or q is an anchor.

The color $\perp$ makes all anchor-incident pairs contribute zero. Our inner product nevertheless averages over all pairs in V, whereas the hybrid step averages only over pairs avoiding the anchors. We now account for this normalization explicitly. Summing over unordered pairs, and using that $(W - Z)(pq)$ is a function of the unordered pair,

$$\mathbb{E}_A \langle W - Z, A\rangle = \frac{1}{n(n-1)} \sum_{(p,q)} (W - Z)(pq)\, \Delta(p, q, w),$$

the sum over ordered pairs of distinct non-anchors. There are $(n - s + 2)(n - s + 1)$ of them, and $(n)_s = (n)_{s-2}(n - s + 2)(n - s + 1)$. Averaging over uniform w and comparing with (5),

$$\mathbb{E}\langle W - Z, A\rangle = \alpha\, (\rho_i - \rho_{i-1}), \qquad \alpha = \frac{(n - s + 2)(n - s + 1)}{n(n-1)}.$$

Here α is the proportion of ordered pairs avoiding the anchors. Since $n \geq 4s$,

$$\alpha \geq \Big(\frac{n - s + 1}{n}\Big)^2 > \frac{1}{2}.$$

Every realization of A lies in $\mathcal{T}$, so $|\langle W - Z, A\rangle| \leq \|W - Z\|_{\mathcal{T}}$, and therefore $|\rho_i - \rho_{i-1}| \leq 2\|W - Z\|_{\mathcal{T}}$. Summing over i gives $|\operatorname{rej}(W) - \operatorname{rej}(Z)| \leq 2K\|W - Z\|_{\mathcal{T}}$. By Fact 3 and $\mathbf{1} - 2G \in \mathcal{T}$, $|\operatorname{cost}_G(W) - \operatorname{cost}_G(Z)| = |\langle W - Z, \mathbf{1} - 2G\rangle| \leq \|W - Z\|_{\mathcal{T}}$. Add twice the first bound. □

4.2 Boosting

The counting lemma reduces the task to making every test correlation small. We do this by repeatedly correcting a correlation that is still too large. The sum of the corrections is allowed to be real-valued; clipping converts it into valid edge probabilities. Keeping the un-clipped sum is useful because it is described by a short list of tests.

Define $\operatorname{clip} : \mathbb{R} \to [0, 1]$ by $\operatorname{clip}(u) = \min\{1, \max\{0, u\}\}$, and apply it pointwise to real-valued functions. To track progress, we use a function whose derivative is exactly this clipping operation. Define $\psi : \mathbb{R} \to [0, \infty)$ by

$$\psi(u) = \begin{cases} 0 & u \leq 0, \\ u^2/2 & 0 \leq u \leq 1, \\ u - \frac{1}{2} & u \geq 1. \end{cases}$$

By direct differentiation, $\psi \in C^1$, and $\psi' = \operatorname{clip}$ is 1-Lipschitz. The following two properties explain this choice. The first will make our potential nonnegative; the second bounds the cost of taking an update step:

$$\psi(u) = \sup_{0 \leq h \leq 1} \big(uh - \tfrac{1}{2}h^2\big), \qquad \psi(u + r) \leq \psi(u) + r\, \psi'(u) + \tfrac{1}{2}r^2. \tag{6}$$

Indeed, the supremum is attained at $h = \mathrm{clip}(u)$, and the inequality follows from

$$\psi(u+r) - \psi(u) - r\psi'(u) = \int_0^r (\psi'(u+t) - \psi'(u))\, dt.$$

For a target $H \in \mathcal{W}_V$, define the potential $\Phi_H : \mathbb{R}^{\binom{V}{2}} \to [0,\infty)$, where Q is the un-clipped sum of the current corrections, by

$$\Phi_H(Q) = \mathbb{E}_e\Big[\psi\big(Q(e)\big) - Q(e)H(e) + \tfrac{1}{2}H(e)^2\Big].$$

It is nonnegative by (6) with $h = H(e)$.

When Q already takes values in $[0,1]$, this potential is $\frac{1}{2}\|Q - H\|_2^2$. Outside that interval, it remains nonnegative and its derivative with respect to Q is $\mathrm{clip}(Q) - H$. This is the residual we can detect using tests. Thus the potential continues to track useful progress even when the un-clipped sum leaves the interval of valid weights.

The next lemma gives a short description of an approximation to any target H: fewer than γ^{-2} signed tests suffice. The number of vertices and the number of available tests do not enter this bound.

Lemma 3. Let $\mathcal{T}$ be a finite nonempty family of functions $\binom{V}{2} \to [-1,1]$, let $G, H \in \mathcal{W}_V$ and $0 < \gamma < 1$. There exist an integer $0 \le t < \gamma^{-2}$ and $B_1, \dots, B_t \in \mathcal{T} \cup (-\mathcal{T})$ such that $W := \mathrm{clip}\Big(G + \gamma \sum_{j=1}^t B_j\Big) \in \mathcal{W}_V$ satisfies $\|H - W\|_{\mathcal{T}} \le \gamma$.

Proof. If a test sees an error larger than γ, move a distance γ in the sign of that test which reduces the error. The resulting decrease in the potential is at least $\gamma^2/2$, so only a bounded number of such moves is possible.

Set $Q_0 = G$. Given Q_j, let $W_j = \mathrm{clip}(Q_j)$. If $\|H - W_j\|_{\mathcal{T}} \le \gamma$, stop. Otherwise some $A \in \mathcal{T}$ has $|\langle H - W_j, A\rangle| > \gamma$; choose $B_{j+1} \in \{\pm A\}$ with $\langle H - W_j, B_{j+1}\rangle > \gamma$ and set $Q_{j+1} = Q_j + \gamma B_{j+1}$. Applying (6) pointwise, with $\psi'(Q_j) = W_j$ and $\|B_{j+1}\|_2 \le 1$,

$$\Phi_H(Q_{j+1}) \le \Phi_H(Q_j) - \gamma\langle H - W_j, B_{j+1}\rangle + \tfrac{\gamma^2}{2}\|B_{j+1}\|_2^2 < \Phi_H(Q_j) - \tfrac{\gamma^2}{2}.$$

Also $\Phi_H(Q_0) = \frac{1}{2}\|G - H\|_2^2 \le \frac{1}{2}$ and $\Phi_H \ge 0$. After any positive number t of updates,

$$0 \le \Phi_H(Q_t) < \tfrac{1}{2} - \tfrac{t\gamma^2}{2},$$

so $t < \gamma^{-2}$. The process must therefore stop within this bound. If no update is needed, $t = 0$ already has the required properties. □

This is gradient descent for the convex potential Φ_H, restricted to test directions. The step count γ^{-2} is independent of n.

We now turn the short list of corrections into information stored at vertices. In each step there is common information, i.e., the kind of test, its sign, and possibly its anchor graph, and, for a derivative test, a color for each vertex. We call the common information a *symbol*. A vertex label stores one symbol and one color for every possible step, padding unused steps by an empty symbol. This is why the label length depends on the number of boosting steps and on s, but not on n.

Here and below, fix $0 < \gamma < 1$, and use the family $\mathcal{T}(G)$ constructed in the preceding subsection.

Let $L = \lceil \gamma^{-2} \rceil$. The set of *symbols* is $\mathcal{C} = \{\varnothing\} \sqcup \big(\{\mathrm{cost}\} \times \{\pm 1\}\big) \sqcup \big(\{\mathrm{der}\} \times \{\pm 1\} \times \mathcal{G}_{[3,s]}\big)$, and the *label set* is $\Lambda = (\mathcal{C} \times \Omega)^L$. A label $z \in \Lambda$ has L components $(c_\ell(z), \omega_\ell(z)) \in \mathcal{C} \times \Omega$, and $\log|\Lambda| = L(\log|\mathcal{C}| + \log|\Omega|) = O(s^2\gamma^{-2})$.

The decoder must recover an edge weight from the two endpoint labels and the original edge. A genuine boosting description has matching common symbols at every vertex. We nevertheless

define the decoder on arbitrary pairs of labels; the later CSP optimization must be allowed to choose any label at any vertex. If the common symbols disagree, keeping the original edge is a convenient way to return a valid weight.

Define the *decoder* $D : \{0,1\} \times \Lambda \times \Lambda \to [0,1]$ as follows. For $g \in \{0,1\}$ and $z, z' \in \Lambda$:

- if $c_\ell(z) \neq c_\ell(z')$ for some ℓ, then $D(g, z, z') = g$;
- otherwise, let $c_\ell = c_\ell(z) = c_\ell(z')$ and set

$$\mu_\ell = \begin{cases} 0 & c_\ell = \varnothing, \\ \sigma(1-2g) & c_\ell = (\mathrm{cost}, \sigma), \\ \sigma\,\kappa_b\big(\omega_\ell(z), \omega_\ell(z')\big) & c_\ell = (\mathrm{der}, \sigma, b), \end{cases} \qquad D(g, z, z') = \mathrm{clip}\Big(g + \gamma \sum_{\ell=1}^{L} \mu_\ell\Big).$$

The decoder D is symmetric in (z, z'). For a labeling $z : V \to \Lambda$, write $z_v = z(v)$ and define $W_z \in \mathcal{W}_V$ by $W_z(uv) = D\big(G(uv), z_u, z_v\big)$.

In particular, every labeling defines a valid weighted graph, even if its symbols are inconsistent. Consistency is needed only for encoding the specific boosting approximation whose existence we have proved.

The next lemma connects the two descriptions: each boosting approximation can be reconstructed from vertex labels. Its final assertion is what makes sampling possible, since a sampled edge can be decoded without knowing any labels outside its endpoints.

Lemma 4. For every $G \in \mathcal{G}_V$ and $H \in \mathcal{W}_V$ there is $z \in \Lambda^V$ with $\|H - W_z\|_{\mathcal{T}} \leq \gamma$, where $\mathcal{T} = \mathcal{T}(G)$. Moreover $W_z(uv)$ depends only on $G(uv)$, z_u, z_v.

Proof. Record the boosting sequence component by component, duplicating each step's common symbol at every vertex and keeping its vertex-dependent color locally. Apply Lemma 3 to get $B_j = \sigma_j A_j$, $j \leq t \leq L$. In component j, give every vertex the symbol $(\mathrm{cost}, \sigma_j)$ and color $\perp$ if $A_j = \mathbf{1} - 2G$. Otherwise $A_j = A_{b_j, \chi_j}$; give every vertex the symbol $(\mathrm{der}, \sigma_j, b_j)$ and give vertex v the color $\chi_j(v)$. Components $j > t$ get $(\varnothing, \perp)$. All symbols agree across vertices, so D is in its second case with $\mu_j = B_j(uv)$ for every uv. Thus $W_z = \mathrm{clip}(G + \gamma \sum_j B_j)$, the function of Lemma 3. The last claim is the definition of D. □

We can now restrict the search for a repair to weighted graphs specified by labels. Define

$$\widetilde{M}(G) = \min_{z \in \Lambda^V} \mathcal{E}_G(W_z).$$

The next proposition says that this restriction loses little: encode a true minimizer of $M(G)$ and use the counting lemma to bound the change in its objective value. Arbitrary labelings cannot artificially improve on $M(G)$, because they still decode to genuine weighted graphs.

Proposition 2. For $n \geq 4s$, $M(G) \leq \widetilde{M}(G) \leq M(G) + (1 + 4K)\gamma$, and if $\mathcal{P}_V \neq \emptyset$,

$$\mathrm{dist}(G, \mathcal{P}) - \tau \;\leq\; \widetilde{M}(G) \;\leq\; \mathrm{dist}(G, \mathcal{P}) + 2\tau + (1 + 4K)\gamma.$$

Proof. The lower bound is immediate from inclusion of the feasible sets; the upper bound follows by approximating one minimizing weighted graph. Since $W_z \in \mathcal{W}_V$, we have $\widetilde{M}(G) \geq M(G)$. Let H^* attain $M(G)$ and take z from Lemma 4 for $H = H^*$. Lemma 2 gives $\widetilde{M}(G) \leq \mathcal{E}_G(W_z) \leq M(G) + (1 + 4K)\gamma$. The distance bounds follow from Lemma 1. □

Observe that the boosting sequence is never computed by the algorithm. It only proves that a short-description repair of nearly optimal energy exists.

4.3 Estimating the optimum from a sample

The labels have reduced our search to a finite optimization problem. To apply a sampling theorem, we must express its objective as an average of bounded functions of a few vertex labels. The tester term already depends on s vertices. We write the edit term on the same tuples by averaging its pair costs over the K pairs in each tuple.

A *dense s-ary constraint satisfaction problem (CSP)* consists of variables V, a label set Λ, and for each $\mathbf{v} \in V_{\neq}^s$ a predicate $P_{\mathbf{v}} : \Lambda^s \to [0,1]$. For $U \subseteq V$ with $|U| = m$ set $\mathrm{Val}(U) = \max_{z \in \Lambda^U} \; m^{-s} \sum_{\mathbf{v} \in U_{\neq}^s} P_{\mathbf{v}}(z_{v_1}, \dots, z_{v_s})$, and let $\mathrm{OPT} = \mathrm{Val}(V)$. Here a predicate is a score in $[0,1]$, so the goal is to maximize the average score. We take $m \geq s$ whenever using this value. The normalization is m^{-s}, rather than $1/(m)_s$: equivalently, tuples with a repeated vertex have score zero. This convention is responsible for the factor a_n below.

For our graph G, a tuple score will be one minus a rescaled local loss. The division by 3 is enough because the edit loss is at most 1 and the rejection penalty at most 2. For $\mathbf{v} \in V_{\neq}^s$ and $\zeta = (\zeta_1, \dots, \zeta_s) \in \Lambda^s$ let $Y \in \mathcal{W}_s$ be $Y(ab) = D\big(G(v_a v_b), \zeta_a, \zeta_b\big)$. Define the loss $\ell_{\mathbf{v}} : \Lambda^s \to [0,3]$ and the predicate $P_{\mathbf{v}}$ by $P_{\mathbf{v}} := 1 - \frac{1}{3}\ell_{\mathbf{v}}$, with

$$\ell_{\mathbf{v}}(\zeta) = \frac{1}{K} \sum_{ab \in \binom{[s]}{2}} \Big[G(v_a v_b)\big(1 - Y(ab)\big) + \big(1 - G(v_a v_b)\big) Y(ab) \Big] + 2\,\phi(Y).$$

The loss depends only on $G[\mathbf{v}]$ and ζ. Denote this CSP by $\mathcal{I}(G)$. In $\mathrm{Val}(U)$ we always restrict this same instance to U: the alphabet Λ, decoder D, and rejection rule f_n are kept fixed. This restriction need not be the instance one would construct by treating $G[U]$ as a new input graph.

Let $a_n = (n)_s / n^s$, the probability that s independent uniformly chosen vertices of V are all distinct. The next lemma verifies that the local losses average to the objective we want, with precisely this normalization factor when losses are converted to CSP scores.

Lemma 5. For every $z \in \Lambda^V$, $\mathbb{E}_{\mathbf{v}}\, \ell_{\mathbf{v}}(z_{v_1}, \dots, z_{v_s}) = \mathcal{E}_G(W_z)$, with $\mathbf{v}$ uniform in $V_{\neq}^s$. Hence

$$\mathrm{OPT}(\mathcal{I}(G)) = a_n\big(1 - \widetilde{M}(G)/3\big).$$

Proof. A fixed pair of positions in a uniform injective tuple is a uniform pair of vertices. Consequently, distributing the edit loss over larger tuples does not change its mean; the rejection term agrees by definition. For uniform $\mathbf{v}$ and fixed ab, $v_a v_b$ is a uniform pair, so the ab-term of the first part has expectation $\mathbb{E}_e[G(e)(1 - W_z(e)) + (1 - G(e))W_z(e)] = \mathrm{cost}_G(W_z)$ (Fact 3). Averaging over ab gives $\mathrm{cost}_G(W_z)$. Since $Y = W_z[\mathbf{v}]$ here, the second part has expectation $2\,\mathrm{rej}(W_z)$. For OPT: $m = n$ and $|V_{\neq}^s| = a_n n^s$, so $n^{-s} \sum_{\mathbf{v}} P_{\mathbf{v}} = a_n(1 - \mathcal{E}_G(W_z)/3)$; maximize over z. □

We use the following external sampling theorem. It controls the optimum *after* optimizing the labels on the sample, not just the sampled value of a fixed labeling. This distinction matters: the best labeling of a sample can depend on the sample and need not extend to an equally good labeling of the whole graph. The guarantee is in expectation; independent repetition will turn it into the desired probability guarantee.

Theorem 3 (Bernasconi–Castiglioni–Celli–Farina–Malavolta [21, Theorem 3.3])**.** *There are absolute constants $c_*, C_* > 0$ such that for every dense k-ary CSP with label set of size Q_0 and every $0 < \theta < 1$, if*

$$m = \big\lceil c_*\, k^{12} \theta^{-6} \log^2(Q_0 k^2/\theta) \big\rceil,$$
$$n \geq c_*^2 k^{24} \theta^{-13} \log^4(Q_0 k^2/\theta),$$

$n \geq m$, and $U \subseteq V$ is a uniformly random m-subset, then $\big|\mathbb{E}\,\mathrm{Val}(U) - \mathrm{OPT}\big| \leq C_* \theta$.

The dependence on the label set is only through $\log Q_0$. This is essential, since $|\Lambda|$ is exponential in $\mathrm{poly}(s, 1/\varepsilon)$.

The penalty and label approximation account for a small deterministic error; the sampled CSP optima account for the remaining random error. The threshold N_0 ensures that all counting and sampling hypotheses hold. Graphs below that threshold can be read in full at a polynomial query cost.

Proof of Theorem 2. Let $\varepsilon \in (0,1)$, put $\tau = \varepsilon/100$, and choose s as in Proposition 1, so that

$$s = O\big(\max\{1, q(\varepsilon/100)\}\log(100/\varepsilon)\big).$$

Set $K = \binom{s}{2}$ and $\gamma_0 = \varepsilon/[100(1+4K)]$. Choose a rational $\gamma \in [\frac{1}{2}\gamma_0, \gamma_0]$. Then $(1+4K)\gamma \le \varepsilon/100$ and

$$\log|\Lambda| = O(s^2\gamma^{-2}) = O(s^6\varepsilon^{-2}). \tag{7}$$

For $n \ge 4s$ and $\mathcal{P}_V \neq \emptyset$, Proposition 2 gives

$$|\widetilde{M}(G) - \mathrm{dist}(G, \mathcal{P})| \le \tfrac{3\varepsilon}{100}. \tag{8}$$

Apply Theorem 3 with $k = s$, $Q_0 = |\Lambda|$ and $\theta = c_0\varepsilon$, where c_0 is an absolute constant with $\theta < 1$ and $C_*\theta \le \varepsilon/24$. Let m be the resulting sample size and N_0 the maximum of the theorem's threshold on n, of m, of $4s$ and of $2s^2$ (rounded up). By (7), both m and N_0 are $\mathrm{poly}(s, 1/\varepsilon)$ with absolute degrees. Let $T = \lceil 432\varepsilon^{-2}\rceil$ be the number of independent samples.

We may identify the input vertex set with $[n]$. The rational choice of γ, together with the rational values of f_n, makes the finite sample objectives rational, so their optima can be computed by exhaustive search without an additional approximation error.

The algorithm

1. By finite enumeration decide whether $\mathcal{P}_{[n]} = \emptyset$; if so output 1. If $n < 2$ output 0.
2. If $2 \le n \le N_0$: query all pairs and output $\mathrm{dist}(G, \mathcal{P})$ exactly.
3. If $n > N_0$: draw independent uniform m-subsets $U_1, \dots, U_T \subseteq [n]$ and query all pairs inside each. By exhaustive search compute $\mathrm{Val}_i = \mathrm{Val}(U_i) \in [0,1]$ for the CSP $\mathcal{I}(G)$ restricted to U_i (each predicate depends only on $G[\mathbf{v}]$, the function f_n and D). With $\overline{\mathrm{Val}} = T^{-1}\sum_{i=1}^{T}\mathrm{Val}_i$, output $\widehat{d} = \mathrm{clip}\big(3(1 - \overline{\mathrm{Val}}/a_n)\big)$.

Correctness for $n > N_0$ The sampling theorem bounds the bias of each sampled optimum. Averaging independent copies controls its fluctuation around that mean. We then invert the exact CSP identity from Lemma 5; the lower bound on a_n ensures that this inversion enlarges the error by only a constant factor.

By Theorem 3 and Lemma 5, $|\mathbb{E}\,\mathrm{Val}_i - \mathrm{OPT}(\mathcal{I}(G))| \le \varepsilon/24$. The Val_i are independent with values in $[0,1]$, so $\mathrm{Var}(\mathrm{Val}_i) \le 1/4$ and $\mathrm{Var}(\overline{\mathrm{Val}}) \le 1/(4T)$. Chebyshev's inequality gives

$$\Pr\big[|\overline{\mathrm{Val}} - \mathbb{E}\,\mathrm{Val}_i| > \varepsilon/24\big] \le \frac{144}{T\varepsilon^2} \le \frac{1}{3}.$$

So with probability at least $2/3$, $|\overline{\mathrm{Val}} - \mathrm{OPT}(\mathcal{I}(G))| \le \varepsilon/12$. Since $n > 2s^2$, $a_n = \prod_{j<s}(1-j/n) \ge 1 - \binom{s}{2}/n \ge \frac{1}{2}$. By Lemma 5, $\widetilde{M}(G) = 3(1 - \mathrm{OPT}(\mathcal{I}(G))/a_n)$, so the quantity inside the clip differs from $\widetilde{M}(G)$ by at most $3 \cdot \frac{\varepsilon}{12} \cdot 2 = \frac{\varepsilon}{2}$. With (8) the total error is below $\frac{\varepsilon}{2} + \frac{3\varepsilon}{100} < \varepsilon$, and clipping to $[0,1] \ni \mathrm{dist}(G, \mathcal{P})$ cannot increase it. For $n \le N_0$ the output is exact.

Query complexity All graph queries are made inside the sampled vertex sets, or inside the whole graph when its order is at most N_0. Searching through labels and evaluating their predicates uses no further graph queries. Thus the number of queries is at most

$$\max\left\{\tbinom{N_0}{2},\ T\tbinom{m}{2}\right\} = \mathrm{poly}(s, 1/\varepsilon),$$

and substituting $s = O(\max\{1, q(\varepsilon/100)\}\log(100/\varepsilon))$ proves the theorem with $c = 1/100$. □

For the corollary, the point is that a natural property already requires many queries to test at small proximity. Thus the explicit accuracy parameter in the estimator's bound can be paid for by the tester's own query complexity. We use the definitions of natural and non-trivial hereditary properties from Section 1.

Proof of Corollary 1. For natural properties, [17, Proposition 1.10] gives the edge-query lower bound $q(\eta) = \Omega(1/\eta)$. For non-trivial hereditary properties, [17, Proposition 1.7] gives this lower bound for the canonical vertex sample size. The Goldreich–Trevisan conversion [2] turns a tester with $q(\eta)$ edge queries into one with $O(q(\eta))$ sampled vertices, so the same lower bound follows for our edge-query notation. Taking $\eta = c\varepsilon$, we obtain $1/\varepsilon = O(q(c\varepsilon))$ for sufficiently small ε. This also absorbs the $\max\{1, \cdot\}$ in Theorem 2, proving the corollary. □

AI disclosure

AI-assisted tools, specifically *Quillbot*, were utilized only to enhance the exposition's clarity and language. The mathematical arguments were not produced using them. Additionally, the author made use of the optional AI feedback for STOC 2027. The system found a few typos and some formatting errors, which the author personally corrected.

Acknowledgments

This project started while the authors were participating in the 2025 NII International Internship Program. The authors are grateful to the program for its support. G.C. is supported by the CB European PhD Studentship funded by Trinity College, Cambridge, and would like to thank the Isaac Newton Institute for Mathematical Sciences, Cambridge, for support and hospitality during the programme Logical Foundations of Complexity Theory, where part of the work on this paper was undertaken. This work was supported by EPSRC grant EP/Z000580/1.

References

[1] O. Goldreich, *Introduction to Property Testing*, Cambridge University Press, 2017.

[2] O. Goldreich and L. Trevisan, Three theorems regarding testing graph properties, *Random Structures & Algorithms* 23 (2003), 23–57.

[3] O. Goldreich, S. Goldwasser and D. Ron, Property testing and its connection to learning and approximation, *Journal of the ACM* 45 (1998), 653–750.

[4] N. Alon and A. Shapira, A characterization of the (natural) graph properties testable with one-sided error, *SIAM Journal on Computing* 37 (2008), 1703–1727.

[5] W. T. Gowers, Lower bounds of tower type for Szemerédi's uniformity lemma, *Geometric and Functional Analysis* 7 (1997), 322–337.

[6] V. Rödl and R. Duke, On graphs with small subgraphs of large chromatic number, *Graphs and Combinatorics* 1 (1985), 91–96.

[7] G. Carenini, C. Seth and Y. Yoshida, A quantitative container characterization of one-sided testability, 2026, arXiv:2608.01523.

[8] E. Blais and C. Seth, Testing graph properties with the container method, *SIAM Journal on Computing* 54 6 (2025), 1626-1642.

[9] N. Kushnir and A. Shapira, Towards tight bounds for testing k-colorability, *Proceedings of APPROX/RANDOM* (2026), 35:1–35:13.

[10] J. Balogh, R. Morris and W. Samotij, Independent sets in hypergraphs, *Journal of the American Mathematical Society* 28 (2015), 669–709.

[11] D. Saxton and A. Thomason, Hypergraph containers, *Inventiones Mathematicae* 201 (2015), 925–992.

[12] M. Parnas, D. Ron and R. Rubinfeld, Tolerant property testing and distance approximation, *Journal of Computer and System Sciences* 72 (2006), 1012–1042.

[13] E. Fischer and L. Fortnow, Tolerant versus intolerant testing for Boolean properties, *Theory of Computing* 2 (2006), 173–183.

[14] E. Fischer and I. Newman, Testing versus estimation of graph properties, *SIAM Journal on Computing* 37 (2007), 482–501.

[15] C. Hoppen, Y. Kohayakawa, R. Lang, H. Lefmann and H. Stagni, On the query complexity of estimating the distance to hereditary graph properties, *SIAM Journal on Discrete Mathematics* 35 (2021), 1238–1251.

[16] N. Fiat and D. Ron, On efficient distance approximation for graph properties, *Proceedings of SODA 2021*, 1618–1637.

[17] L. Gishboliner, N. Kushnir and A. Shapira, Testing versus estimation of graph properties, revisited, *Random Structures & Algorithms* 65 (2024), 460–487.

[18] A. Frieze and R. Kannan, Quick approximation to matrices and applications, *Combinatorica* 19 (1999), 175–220.

[19] C. Dwork and P. Tankala, Efficient and private property testing via indistinguishability, arXiv:2511.03653v2, 2026.

[20] L. Trevisan, M. Tulsiani and S. Vadhan, Regularity, boosting, and efficiently simulating every high-entropy distribution, *Proceedings of IEEE CCC*, 2009.

[21] M. Bernasconi, M. Castiglioni, A. Celli, G. Farina and G. Malavolta, A subsampling theorem for constraint satisfaction problems with large arity, 2026, arXiv:2608.00203.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il