

Some exposition of “The Exponential Time Hypothesis Is False”*

October 26, 2026

1 The plan

The goal is to count the independent sets of a v -vertex 3-regular graph in time roughly

$$2^{O(v/\sqrt{\log v})}.$$

The argument below is a specialization of Sections 2–4 of the manuscript *The Exponential Time Hypothesis Is False* [OAI26]. We will deliberately use loose, easily checked parameters rather than optimize logarithmic factors. They give the following clean, weaker bound.

Theorem 1.1. *The number of independent sets of a v -vertex 3-regular graph can be computed exactly by a Las Vegas randomized algorithm in expected time*

$$v^{O(1)} 2^{O(v/(\log v)^{1/4})} = 2^{o(v)}, \quad (1.1)$$

using polynomial space.

This already suffices for the intended complexity consequence. Liu and Chen [LC24, Theorem 11] prove that, assuming #ETH, counting vertex covers in v -vertex 3-regular graphs requires exponential time. The complementation bijection

$$S \text{ is independent} \iff V \setminus S \text{ is a vertex cover}$$

transfers their lower bound to independent sets. As ETH implies #ETH, a *deterministic* $2^{o(v)}$ exact counter would refute ETH.

We describe a randomized algorithm only to simplify one combinatorial labeling step. It always outputs the correct count; randomness affects only how quickly a suitable labeling is found. This step can be derandomized by the method of conditional expectations, without changing the time bound. We omit the derandomization, which is given in the more general setting of [OAI26, Section 4.4].

The key algebraic step is a familiar distributional approximation of parity, applied to commuting SWAP operators. The resulting low-rank correction is represented by a small number of *exact* linear measurements. Finally, formal Taylor expansion, Boolean Möbius inversion, and Hermite interpolation reconstruct the count. The latter tools are classical; we give the elementary calculation rather than treating it as a new interpolation principle.

All logarithms are base two. The number of graph vertices is v . For the auxiliary matrix we use

$$d = \text{local register dimension}, \quad n = \text{number of register pairs},$$

$$N = d^{2n} = \text{matrix dimension}.$$

The integer b will bound the degree of a polynomial and the number of exceptional SWAP locations. The quantum notation is simply convenient language for real symmetric matrices; the algorithm is classical.

*Produced by odonnell@cs.cmu.edu through interaction with LLMs.

2 A classical approximation of parity

Our only approximation ingredient is an old idea: interpolate a symmetric Boolean function on the Hamming-weight layers carrying almost all the probability mass. The historical chain deserves some care. Bhatnagar, Gopalan, and Lipton [BGL06] gave the finite-characteristic symmetric-polynomial ingredient. A proof communicated by Wigderson appears in Viola's exposition [Vio09, Section 1.2.3]. Srinivasan [Sri13, Lemma 14] formulated the product-distribution extension over arbitrary fields, and Alman and Williams [AW15, Lemma 3.1] gave a particularly useful integral interpolation formulation. We use only the following elementary rational-polynomial specialization.

Lemma 2.1 (Typical-weight interpolation). *Let $p \in [1/4, 1/2]$, and suppose $32 \leq b \leq n/6$. There is an explicitly constructible rational polynomial f of degree at most b such that, for $J \sim \text{Bin}(n, p)$,*

$$\Pr[f(J) \neq (-1)^J] \leq \rho, \quad \rho := 2 \exp\left(-\frac{b^2}{8n}\right). \quad (2.1)$$

Proof. Choose $b + 1$ consecutive integers $\{a, a + 1, \dots, a + b\} \subseteq \{0, \dots, n\}$ centered as closely as possible around np . Set

$$f(z) = (-1)^a \sum_{j=0}^b (-2)^j \binom{z-a}{j}. \quad (2.2)$$

For $0 \leq k \leq b$, the binomial theorem gives

$$f(a+k) = (-1)^a \sum_{j=0}^k (-2)^j \binom{k}{j} = (-1)^{a+k}.$$

The interval contains every integer within $b/4$ of np (using $b \geq 32$ and $p \in [1/4, 1/2]$). Hoeffding's inequality gives

$$\Pr[|J - np| > b/4] \leq 2 \exp(-b^2/(8n)),$$

which proves the claim. $\square$

The polynomial is deterministic. The probability in the lemma measures its set of *disagreement weights*, not a random choice of polynomial. In the construction below those exceptional weights will be retained, not ignored.

3 The approximation lemma in matrix form

Fix $d \geq 2$ and consider the tensor-product space

$$\mathcal{H} = (\mathbb{C}^d \otimes \mathbb{C}^d)^{\otimes n}, \quad \dim \mathcal{H} = N = d^{2n}. \quad (3.1)$$

For $t \in [n]$, let S_t exchange the two registers in the t th pair. The operators S_t are real symmetric, commute, and satisfy $S_t^2 = I$. Define

$$A_t = \frac{I - S_t}{2}, \quad W = \sum_{t=1}^n A_t, \quad U = \prod_{t=1}^n S_t. \quad (3.2)$$

Here A_t is the orthogonal projector onto the antisymmetric subspace of pair t ; U swaps every pair.

Apply Lemma 2.1 with $p = (d - 1)/(2d)$. Define

$$M = Uf(W). \quad (3.3)$$

If $f(z) = \sum_j f_j z^j$, the notation $f(W)$ means simply $\sum_j f_j W^j$. The matrix M is rational and real symmetric. In quantum language it is an observable, though it need not be a unitary or a contraction.

Index the computational basis by

$$\Omega = [d]^n \times [d]^n.$$

For a row label write $\omega = (\alpha, \beta)$; for a column label write $\eta = (\alpha', \beta')$. The t th pair of the corresponding basis vector is $|\alpha_t \beta_t\rangle$.

Lemma 3.1 (The matrix construction). *The matrix M in (3.3) has the following properties.*

(i) *Its error has small rank:*

$$\text{rank}(I - M) \leq \rho N. \quad (3.4)$$

(ii) *Its support is nearly a complete swap:*

$$M_{\omega, \eta} \neq 0 \implies |\{t \in [n] : \alpha'_t \neq \beta_t\}| \leq b. \quad (3.5)$$

(iii) *There is an explicit matrix $C \in \{0, 1, -1\}^{s_0 \times N}$ with $s_0 \leq \rho N$ such that*

$$Cz = Cz' \implies (I - M)z = (I - M)z' \quad (z, z' \in \mathbb{Q}^N). \quad (3.6)$$

The data can be constructed in time polynomial in N and n .

Proof. Spectral agreement and rank. The symmetric and antisymmetric subspaces of a single register pair have dimensions

$$d_+ = \frac{d(d+1)}{2}, \quad d_- = \frac{d(d-1)}{2}.$$

Choose a tensor-product eigenbasis for the S_t . A basis vector having exactly j antisymmetric pairs satisfies

$$W\psi = j\psi, \quad U\psi = (-1)^j\psi, \quad M\psi = (-1)^j f(j)\psi.$$

The total dimension of this eigenspace is $\binom{n}{j} d_-^j d_+^{n-j}$. Consequently,

$$\begin{aligned} \frac{\text{rank}(I - M)}{N} &= \sum_{j: f(j) \neq (-1)^j} \binom{n}{j} \left(\frac{d_-}{d^2}\right)^j \left(\frac{d_+}{d^2}\right)^{n-j} \\ &= \Pr_{J \sim \text{Bin}(n, p)} [f(J) \neq (-1)^J] \leq \rho. \end{aligned} \quad (3.7)$$

Thus the distributional disagreement probability is exactly the relative rank of the matrix error.

Degree and support. Because W is an affine linear combination of the commuting involutions S_t and $\deg f \leq b$, expansion and the relations $S_t^2 = I$ give

$$f(W) = \sum_{\substack{T \subseteq [n] \\ |T| \leq b}} \gamma_T \prod_{t \in T} S_t$$

for rational coefficients γ_T . Hence

$$M = \sum_{\substack{T \subseteq [n] \\ |T| \leq b}} \gamma_T \prod_{t \notin T} S_t. \quad (3.8)$$

Every permutation in this sum swaps all pairs outside a set of size at most b . A nonzero (ω, η) entry in such a permutation requires $\alpha'_t = \beta_t$ outside that set. If more than b coordinates disagree, every summand has zero entry there. This proves (3.5).

Small-integer measurements. For one pair, use the unnormalized orthogonal eigenbasis

$$|aa\rangle \quad (a \in [d]), \quad |ab\rangle + |ba\rangle, \quad |ab\rangle - |ba\rangle \quad (a < b). \quad (3.9)$$

The first two kinds are symmetric and the last is antisymmetric. Tensoring these vectors gives an orthogonal basis of $\mathbb{Q}^N$, all of whose entries lie in $\{0, 1, -1\}$.

Let the rows of C be the transposes of those basis vectors on which M does not act as the identity. Equation (3.7) shows that there are at most ρN such vectors. If $C(z - z') = 0$, then $z - z'$ is orthogonal to every exceptional basis vector. It is therefore a linear combination of vectors fixed by M , proving (3.6).

Enumerating this basis, constructing the polynomial in (2.2), and evaluating it on the $N \times N$ matrix W all take polynomial time in N and n . $\square$

The rank bound alone would not suffice for the algorithm: a rational linear measurement with unrestricted coefficients can distinguish all Boolean vectors. The useful extra conclusion is the explicit $\{0, 1, -1\}$ measurement matrix. On Boolean inputs, each of its rows takes one of at most $2N + 1$ integer values.

Lemma 3.1 is the combination of Lemmas 4.1–4.2 of [OAI26]. The exceptional directions are not thrown away. They will be retained through Cz , so all subsequent calculations remain exact.

4 Low-degree replacement forms

At point $t \in [n]$, interpret the first label α_t of $z_{\alpha, \beta}$ as its group, and the second label β_t as the group in which we would like to evaluate its replacement. The groups of auxiliary coordinates are

$$\Omega_{t,g} = \{(\alpha, \beta) \in \Omega : \alpha_t = g\} \quad (g \in [d]).$$

For a nonzero entry $M_{\omega, \eta}$, define its bad locations and the corresponding root polynomial by

$$B_{\omega, \eta} = \{t : \alpha'_t \neq \beta_t\}, \quad \phi_{\omega, \eta}(T) = \prod_{t \in B_{\omega, \eta}} \left(1 - \frac{T}{t}\right). \quad (4.1)$$

The support condition gives $\deg \phi_{\omega, \eta} \leq b$. Now put

$$K_{\omega}(T, z) = ((I - M)z)_{\omega} + \sum_{\eta: M_{\omega, \eta} \neq 0} M_{\omega, \eta} \phi_{\omega, \eta}(T) z_{\eta}. \quad (4.2)$$

This is linear in z , with polynomial coefficients in T .

Lemma 4.1. *For every $\omega = (\alpha, \beta) \in \Omega$, the form K_ω satisfies*

$$K_\omega(0, z) = z_\omega, \quad \deg_T K_\omega \leq b.$$

At every $t \in [n]$, its value $K_\omega(t, z)$ is determined by the shared measurements Cz and the coordinates $z|_{\Omega_{t, \beta_t}}$ in the desired group.

Proof. At zero every root polynomial equals one, so the two terms in (4.2) sum to z_ω . The degree bound is immediate. At $T = t$, every term whose column is outside Ω_{t, β_t} vanishes because its root polynomial contains $1 - T/t$. The remaining part of the sum is local to that group, and $((I - M)z)_\omega$ is determined by Cz by Lemma 3.1. $\square$

This is the passage from the matrix construction to the polynomial replacements in [OAI26, Section 4.3]. It explains why both parts of the matrix lemma are needed: low-rank error gives few shared measurements, while near-swap support gives low-degree root polynomials.

5 Putting the graph into the auxiliary arrays

Let $G = (V, E)$ be 3-regular, with $|V| = v$ and $|E| = 3v/2$. For each edge e , label its two endpoint occurrences by 1, 2 and write $u(e, i)$ for the corresponding vertex. Thus the incidence set

$$\mathcal{I} = E \times \{1, 2\}$$

has size $3v$. We work with simple graphs; no further generality is needed.

For a Boolean vector $x \in \{0, 1\}^V$, let $x_u = 1$ mean that u is selected. The count is

$$Z(G) = \# \text{IS}(G) = \sum_{x \in \{0, 1\}^V} \prod_{e \in E} (1 - x_{u(e, 1)} x_{u(e, 2)}). \quad (5.1)$$

Each factor indicates that the two endpoints of its edge are not both selected.

5.1 Random group labels

Independently choose uniform labels

$$\alpha_u \in [d]^n \quad (u \in V), \quad \beta_e \in [d]^n \quad (e \in E).$$

At point t , vertex u belongs to group $\alpha_u(t)$ and the entire edge factor for e is assigned to group $\beta_e(t)$. The assigned group need not contain either endpoint. Give incidence (e, i) the auxiliary label

$$\omega(e, i) = (\alpha_{u(e, i)}, \beta_e) \in \Omega.$$

We need the vertex groups to be small and the incidence labels not to be too crowded.

Lemma 5.1 (Balanced labels). *Suppose $N^2 + nd^2 = o(v)$. With probability $1 - o(1)$, the random labels satisfy*

$$|\{u \in V : \alpha_u(t) = g\}| \leq \frac{2v}{d} \quad (t \in [n], g \in [d]), \quad (5.2)$$

$$|\{(e, i) \in \mathcal{I} : \omega(e, i) = \omega\}| \leq \frac{6v}{N} \quad (\omega \in \Omega). \quad (5.3)$$

The conditions can be checked in polynomial time, so repeatedly sampling until they hold gives a Las Vegas construction of expected polynomial cost.

Proof. Let X_ω count the incidences with label ω , and set $m_0 = |\mathcal{I}| = 3v$. Then $\mathbb{E}X_\omega = m_0/N$ and

$$\sum_{\omega} (X_\omega - m_0/N)^2 = \sum_{\iota, \iota' \in \mathcal{I}} \mathbf{1}\{\omega(\iota) = \omega(\iota')\} - \frac{m_0^2}{N}.$$

Two incidence labels are independent unless the incidences share a graph vertex or a graph edge. Since all degrees are three, only $O(v)$ ordered pairs share one of these. Independent pairs have collision probability $1/N$; every other probability is at most one. Therefore

$$\mathbb{E} \sum_{\omega} (X_\omega - m_0/N)^2 = O(v). \quad (5.4)$$

If some $X_\omega > 2m_0/N$, this sum exceeds $(m_0/N)^2$. Markov's inequality bounds the failure probability of (5.3) by $O(N^2/v)$.

Next let $Y_{t,g} = |\{u : \alpha_u(t) = g\}|$. Independence of the vertex labels gives

$$\mathbb{E} \sum_{t,g} (Y_{t,g} - v/d)^2 = nv(1 - 1/d) \leq nv. \quad (5.5)$$

If any $Y_{t,g} > 2v/d$, this sum exceeds $(v/d)^2$. Thus the failure probability of (5.2) is at most nd^2/v . The sum of the two failure probabilities tends to zero. $\square$

This second-moment argument is the randomized form of the balancing construction in [OAI26, Section 4.4]. From now on, fix labels that pass the two checks.

5.2 Copies of the auxiliary array

Let $h = \max_{\omega} X_\omega$. Take h copies of Ω and place each incidence (e, i) at coordinate $\omega(e, i)$ of one copy, placing incidences with the same label in different copies. Write $j(e, i)$ for its copy. Fill its position with the formal variable $x_{u(e,i)}$, and put zero in every unused position. Denote the resulting vector in copy j by

$$z^{(j)}(x) \in \mathbb{Q}^\Omega.$$

The load bound gives

$$hN \leq 6v. \quad (5.6)$$

In particular the total number of auxiliary positions is $O(v)$.

These are repeated positions for the original variables, not new Boolean variables: all positions containing x_u always have the same value. Apply the replacement form in the copy containing each incidence:

$$L_{e,i}(T, x) = K_{\omega(e,i)}(T, z^{(j(e,i))}(x)). \quad (5.7)$$

Lemma 4.1 immediately gives

$$L_{e,i}(0, x) = x_{u(e,i)}, \quad \deg_T L_{e,i} \leq b. \quad (5.8)$$

5.3 Splitting the measurements by group

Keep the measurements $Cz^{(j)}(x)$ for every copy j . There are $hs_0 \leq h\rho N \leq 6\rho v$ of them. At point t , the graph-vertex groups are

$$V_{t,g} = \{u \in V : \alpha_u(t) = g\} \quad (g \in [d]).$$

For each measurement $\ell(x)$, split it into the sum of its restrictions to these groups:

$$\ell(x) = \sum_{g=1}^d \ell|_{V_{t,g}}(x|_{V_{t,g}}).$$

This splitting is important: fixing a total such as $x_u + x_w$ would still couple groups containing u and w . Fixing the separate group contributions gives independent conditions within the groups.

Collect the restricted measurements on group g as the rows of an integer matrix $C_{t,g}$. Including zero rows, the total number of measurements at a point is

$$s = \sum_{g=1}^d \# \text{rows}(C_{t,g}) = dhs_0 \leq 6\rho dv. \quad (5.9)$$

Every row of C has at most N nonzero entries, each equal to 1 or -1 . Substitution of graph variables, combination of repeated occurrences, and restriction to a group cannot increase the absolute coefficient sum. Hence every row of $C_{t,g}$ has absolute coefficient sum at most N . On a Boolean assignment, each measurement therefore takes an integer value in $[-N, N]$.

Lemma 5.2 (Locality after fixing measurements). *Fix $t \in [n]$. After the values of all $C_{t,g}x|_{V_{t,g}}$ have been fixed, both forms $L_{e,1}(t, x)$ and $L_{e,2}(t, x)$ depend only on the variables in $V_{t,\beta_e(t)}$.*

Proof. The group-restricted measurements determine each total $Cz^{(j)}(x)$. By Lemma 4.1, the only other coordinates needed to evaluate $L_{e,i}(t, x)$ have first group label $\beta_e(t)$. Every occupied auxiliary coordinate containing x_u has first label α_u , so these coordinates contain only variables from $V_{t,\beta_e(t)}$. Both endpoint occurrences of e have second label β_e and therefore use the same group. $\square$

This completes the transfer from auxiliary coordinates to graph incidences; it is the specialization of [OAI26, Section 4.5] used here.

6 A polynomial with easily evaluated points

Define

$$H(T) = \sum_{x \in \{0,1\}^V} \prod_{e \in E} (1 - L_{e,1}(T, x)L_{e,2}(T, x)). \quad (6.1)$$

At zero the replacements restore the original variables, while each edge contributes degree at most $2b$. Thus

$$H(0) = Z(G), \quad \deg H \leq 2|E|b = 3vb. \quad (6.2)$$

At a nonzero point $t \in [n]$, Lemma 5.2 makes the factors separate once the group measurements are fixed.

Here is the evaluation procedure explicitly. Write $x_g = x|_{V_{t,g}}$ and let

$$F_{t,g}(x) = \prod_{e: \beta_e(t)=g} (1 - L_{e,1}(t, x)L_{e,2}(t, x)).$$

Enumerate all tuples of possible measurement values

$$y = (y_1, \dots, y_d), \quad y_g \in \{-N, \dots, N\}^{\# \text{rows}(C_{t,g})}.$$

For each group put

$$\mathcal{X}_g(y_g) = \{x_g \in \{0, 1\}^{V_{t,g}} : C_{t,g}x_g = y_g\}.$$

If some $\mathcal{X}_g(y_g)$ is empty, skip the tuple. Otherwise choose one representative $\bar{x}_g \in \mathcal{X}_g(y_g)$ for every group. To evaluate $F_{t,g}$ on a local assignment x_g , combine x_g with the representatives in the other groups. Locality guarantees that the chosen representatives do not affect the value; denote it by $F_{t,g}(x_g; y)$.

The assignments with measurement tuple y form the Cartesian product $\prod_g \mathcal{X}_g(y_g)$. Distributivity therefore gives

$$H(t) = \sum_y \prod_{g=1}^d \left(\sum_{x_g \in \mathcal{X}_g(y_g)} F_{t,g}(x_g; y) \right). \quad (6.3)$$

Every Boolean assignment has exactly one tuple of measurement values, so the formula counts its contribution exactly once.

There are at most $(2N + 1)^s$ tuples. For each tuple, finding representatives and summing within the groups examines at most

$$2 \sum_{g=1}^d 2^{|V_{t,g}|} \leq 2d 2^{2v/d}$$

local assignments. Consequently an evaluation costs, up to polynomial factors,

$$(2N + 1)^s 2^{2v/d}. \quad (6.4)$$

Two extensions of the same procedure. We may require any prescribed subset of the Boolean variables to equal one: this simply restricts the local summations. We may also add to each input $L_{e,i}(t, x)$ a fixed element of a commutative ring, independent of x . Such shifts do not alter the dependence on other groups, so separation still holds. In particular, the same procedure works over $\mathbb{Q}[z]/(z^{q+1})$, with fixed polynomial shifts, at polynomial overhead in q . The representatives used to evaluate a factor need only satisfy the measurements, not the additional fixed-bit restrictions.

7 Taylor recovery by standard algebraic tools

We know how to evaluate $H(T)$ at $T = 1, \dots, n$, but $\deg H \leq 3vb$, so ordinary interpolation is insufficient. We use three familiar tools: Hermite interpolation, formal Taylor expansion (equivalently Hasse derivatives), and Möbius inversion on the Boolean lattice. Closely related derivative-recovery methods appear in Bhargava–Ghosh–Kumar–Mohapatra [BGKM23, Lemmas 4.10 and 6.2]; subset Möbius transforms have long been used in exact graph algorithms, for example by Björklund–Husfeldt–Kaski–Koivisto [BHKK08]. Here they allow us to use the *same* separated summation algorithm for all the required derivatives.

7.1 How much Taylor data is enough?

Set

$$q = \left\lfloor \frac{3vb}{n} \right\rfloor. \quad (7.1)$$

We will compute

$$H(t + z) \pmod{z^{q+1}} \quad (t = 1, \dots, n).$$

These data determine H : if two polynomials of degree at most $3vb$ have the same first $q+1$ Taylor coefficients at every t , their difference is divisible by

$$\prod_{t=1}^n (T - t)^{q+1},$$

whose degree $n(q+1)$ exceeds $3vb$. Hermite interpolation therefore recovers $H(0)$ using polynomially many rational operations.

The problem is that the deformed factors at $t+z$ no longer separate across the groups at t . We keep the changes as independent formal inputs.

7.2 Give the changes their own variables

Fix t . Let $x \in \{0, 1\}^V$ be the Boolean assignment being summed over, and introduce a separate formal vector $X = (X_u)_{u \in V}$. Define

$$\Delta L_{e,i}(z, X) = L_{e,i}(t+z, X) - L_{e,i}(t, X) \quad (7.2)$$

and

$$F_{t,x}(X, z) = \prod_{e \in E} (1 - (L_{e,1}(t, x) + \Delta L_{e,1}(z, X))(L_{e,2}(t, x) + \Delta L_{e,2}(z, X))). \quad (7.3)$$

Setting $X = x$ restores the shifted inputs, so

$$H(t+z) = \sum_{x \in \{0,1\}^V} F_{t,x}(x, z). \quad (7.4)$$

But if we fix X , every $\Delta L_{e,i}(z, X)$ becomes a constant polynomial shift, independent of x . Then the separated computation from Section 6 still works.

Every $\Delta L_{e,i}(z, X)$ is linear in X and divisible by z . Therefore every occurrence of an X_u contributes a factor of z , and

$$\deg_X [z^j] F_{t,x}(X, z) \leq j. \quad (7.5)$$

In particular, modulo z^{q+1} , the polynomial $F_{t,x}(X, z)$ has degree at most q in X . We now eliminate the diagonal substitution $X = x$ by Boolean finite differences.

7.3 Boolean discrete Taylor expansion

For a polynomial $P(X)$ and a set $A \subseteq V$, write

$$\Delta_A P(0) := \sum_{B \subseteq A} (-1)^{|A|-|B|} P(\mathbf{1}_B),$$

where $\mathbf{1}_B$ is the Boolean vector with support B . This is the A th discrete derivative of P at zero. Möbius inversion says that on Boolean inputs y ,

$$P(y) = \sum_{A \subseteq V} \Delta_A P(0) \prod_{u \in A} y_u. \quad (7.6)$$

If $\deg P \leq q$, only sets A of size at most q appear.

Lemma 7.1 (Diagonal-sum identity). *Let $\mathcal{R}$ be a commutative ring. For each $x \in \{0,1\}^V$, let $P_x(X)$ be a polynomial over $\mathcal{R}$ of degree at most q . Then*

$$\sum_x P_x(x) = \sum_{\substack{A \subseteq V \\ |A| \leq q}} \sum_{x: x_u=1 \ (u \in A)} \Delta_A P_x(0). \quad (7.7)$$

Proof. Replace every positive power of X_u by X_u to obtain the multilinear representative of each P_x . The standard Möbius-inversion formula (7.6) gives its coefficients. Upon setting $X = x$, the monomial $\prod_{u \in A} x_u$ selects exactly the assignments whose bits in A are all one. Sum over x and interchange the two sums. $\square$

Expanding Δ_A yields the form directly used by the algorithm:

$$\sum_x P_x(x) = \sum_{\substack{A \subseteq V \\ |A| \leq q}} \sum_{B \subseteq A} (-1)^{|A|-|B|} \sum_{x: x_u=1 \ (u \in A)} P_x(\mathbf{1}_B). \quad (7.8)$$

This is Lemma 3.3 of [OAI26]; its proof is exactly the ordinary discrete Taylor formula.

7.4 Computing the Taylor coefficients

Apply (7.8) with $P_x(X) = F_{t,x}(X, z) \bmod z^{q+1}$ over $\mathcal{R} = \mathbb{Q}[z]/(z^{q+1})$. By (7.5),

$$H(t+z) \equiv \sum_{\substack{A \subseteq V \\ |A| \leq q}} \sum_{B \subseteq A} (-1)^{|A|-|B|} \sum_{x: x_u=1 \ (u \in A)} F_{t,x}(\mathbf{1}_B, z) \pmod{z^{q+1}}. \quad (7.9)$$

Each innermost sum is an instance of the separated computation: $X = \mathbf{1}_B$ fixes the correction polynomial, while the conditions $x_u = 1$ for $u \in A$ restrict the allowed assignments within their groups. In particular, we have not needed to construct a polynomial in all the formal variables X .

The number of pairs (A, B) is

$$\mathcal{C}_q = \sum_{j=0}^{\min(q,v)} 2^j \binom{v}{j}. \quad (7.10)$$

For $1 \leq q \leq v/2$,

$$\mathcal{C}_q \leq (q+1) \left(\frac{2ev}{q} \right)^q = 2^{O(q \log(2v/q))}. \quad (7.11)$$

Thus all the necessary Taylor coefficients at one point are computable at the cost of only this additional factor. Computing them at every point and performing Hermite interpolation gives the exact value $H(0)$.

For context, Barvinok's Taylor method and its graph-polynomial refinements by Patel–Regts [PR22] also exploit low-order coefficients, but use zero-free regions for *approximate* counting. Here Hermite interpolation recovers the whole polynomial *exactly*. The device that makes it efficient is the separation-preserving use of the second vector X .

8 Running time and completion of the proof

Suppose the auxiliary parameters are polynomially bounded in v and $b/n = o(1)$. Then eventually $1 \leq q \leq v/2$. By (5.9), (6.4), and (7.11), the total time, up to polynomial factors, is

$$2^{O\left(\frac{v}{d} + \rho dv \log(2N+1) + v \frac{b}{n} \log \frac{2n}{b}\right)}. \quad (8.1)$$

The three terms count local Boolean assignments, measurement values, and Boolean finite differences. The number of interpolation points and the exact Hermite interpolation itself contribute polynomial factors.

Let $L = \lceil \log v \rceil$. To avoid log-log bookkeeping, choose simply

$$\boxed{d = \lfloor L^{1/4} \rfloor, \quad n = \lfloor L^{4/5} \rfloor, \quad b = \lceil L^{1/2} \rceil.} \quad (8.2)$$

These parameters satisfy $d \geq 2$ and $32 \leq b \leq n/6$ for all sufficiently large v . The typical-weight lemma gives

$$\rho \leq 2 \exp\left(-\Omega(L^{1/5})\right), \quad (8.3)$$

where we use $b^2/n = \Theta(L^{1/5})$. Moreover,

$$\log N = 2n \log d = O(L^{4/5} \log L) = o(L). \quad (8.4)$$

Hence $N = v^{o(1)}$; in particular $N^2 + nd^2 = o(v)$, so the randomized labeling lemma applies and all the auxiliary matrices have polynomial size.

Finally, the three terms of (8.1) are bounded by

$$\begin{aligned} \frac{v}{d} &= O(v/L^{1/4}), \\ \rho dv \log(2N+1) &= v \exp(-\Omega(L^{1/5})) L^{O(1)} = o(v/L^{1/4}), \\ v \frac{b}{n} \log \frac{2n}{b} &= O\left(\frac{v \log L}{L^{3/10}}\right) = O(v/L^{1/4}). \end{aligned}$$

This proves the time bound in Theorem 1.1. A sample labeling is checked before use, succeeds with probability $1 - o(1)$, and can be resampled independently on failure; all subsequent steps are exact. Small input sizes are handled by exhaustive search.

We omit the standard exact-arithmetic estimates and the proof of polynomial space. The interpolation polynomials have rational coefficients of polynomial bit length; after clearing common denominators, even exponentially many summands can be accumulated with polynomially many bits. The loops are sequential and store only polynomially many Taylor coefficients. These details are carried out in [OAI26, Lemma 4.6 and Appendix B.4].

Conditional expectations derandomize the auxiliary labeling, as discussed at the start, yielding a deterministic algorithm with the same asymptotic time bound. Liu–Chen’s result [LC24] then gives the stated contradiction to $\#ETH$, and hence to ETH . $\square$

Perspective. The spectral approximation is not used to approximate the count. It makes $I - M$ have small rank; the exceptional eigendirections are retained as exact integer measurements. After these measurements are fixed, the deformed edge factors separate. Ordinary finite differences and Hermite interpolation then recover the original partition function exactly.

References

- [AW15] Josh Alman and Ryan Williams. Probabilistic Polynomials and Hamming Nearest Neighbors. In *56th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2015. Lemma 3.1. <https://arxiv.org/abs/1507.05106>.
- [BGL06] Nayantara Bhatnagar, Parikshit Gopalan, and Richard J. Lipton. Symmetric Polynomials over $\mathbb{Z}_m$ and Simultaneous Communication Protocols. *Journal of Computer and System Sciences* 72(2):252–285, 2006. <https://doi.org/10.1016/j.jcss.2005.06.007>.
- [BGKM23] Vishwas Bhargava, Sumanta Ghosh, Mrinal Kumar, and Chandra Kanta Mohapatra. Fast, Algebraic Multivariate Multipoint Evaluation in Small Characteristic and Applications. *Journal of the ACM* 70(6), Article 42, 2023. Lemmas 4.10 and 6.2. <https://doi.org/10.1145/3625226>.
- [BHK08] Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Trimmed Möbius Inversion and Graphs of Bounded Degree. In *STACS 2008*, pages 85–96, 2008. <https://doi.org/10.4230/LIPIcs.STACS.2008.1336>.
- [LC24] Ying Liu and Shiteng Chen. Sub-Exponential Time Lower Bounds for $\#VC$ and $\#Matching$ on 3-Regular Graphs. In *41st Symposium on Theoretical Aspects of Computer Science (STACS)*, LIPIcs 289, 49:1–49:18, 2024. Theorem 11. <https://doi.org/10.4230/LIPIcs.STACS.2024.49>.
- [OAI26] OpenAI. *The Exponential Time Hypothesis Is False*. Manuscript dated October 17, 2026; revised October 25, 2026. Sections 2–4 and Appendix B.
- [PR22] Viresh Patel and Guus Regts. Approximate Counting Using Taylor’s Theorem: A Survey. *Bulletin of the EATCS*, 2022. <https://arxiv.org/abs/2212.08143>.
- [Sri13] Srikanth Srinivasan. On Improved Degree Lower Bounds for Polynomial Approximation. In *FSTTCS 2013*, LIPIcs 24, pages 201–212, 2013. Lemma 14. <https://doi.org/10.4230/LIPIcs.FSTTCS.2013.201>.
- [Vio09] Emanuele Viola. On the Power of Small-Depth Computation. *Foundations and Trends in Theoretical Computer Science* 5(1):1–72, 2009. Section 1.2.3, including an argument communicated by Avi Wigderson. <https://doi.org/10.1561/04000000033>.