

On counting satisfying assignments via polynomial separation

Scott Duke Kominers*

Preliminary Version – October 26, 2026

The number of satisfying assignments of a formula in conjunctive normal form is a sum of products of clause indicators. If the variables were partitioned into small groups and each indicator depended on just one group, this sum would factor into local sums, each computed by enumerating assignments to one group. The algorithm of OpenAI’s *The Exponential Time Hypothesis is false* [6] obtains such a factorization after deforming the inputs and fixing the values of certain integer linear forms. Low-degree polynomial interpolation then recovers the original count. We give a qualitative exposition of the construction, separating the general counting procedure from the particular construction that supplies its input.

For every fixed clause width $k \geq 2$, the OpenAI paper gives an exact-counting algorithm with time bound $(n + m)^{O_k(1)} 2^{O_k(n \log \log n / \sqrt{\log n})}$ and polynomial space [6, Theorem 1.1], which is quantitatively stronger than Theorem 1 below. Our recovery procedure and swap construction follow the ideas of [6, Sections 3–4]. For the qualitative result, we use cyclic balancing of finitely many label types and exploit the two-group support of each exceptional basis vector; these replace the estimates in [6, Lemma 4.5] and [6, equation (4.16)], respectively.

1 Statements and proof strategy

Throughout, a 3-*CNF* is a conjunction of clauses containing at most three literals, and n counts all declared variables, including unused ones. *Time* means deterministic bit complexity. Polynomial exponents in time and space bounds are absolute, independent of the fixed parameters; implicit multiplicative constants may depend on parameters indicated by subscripts. We write $[a] = \{1, \dots, a\}$ for a nonnegative integer a .

Theorem 1. *For every $\varepsilon > 0$, satisfiability of a 3-*CNF* on n variables with m clauses can be decided in time $O_\varepsilon(2^{\varepsilon n}(n + m)^{O(1)})$ and polynomial space.*

Here ε is fixed independently of the input size. The implicit constant may depend on ε , so the statement does not give a running-time bound for an arbitrary choice $\varepsilon = \varepsilon(n) \rightarrow 0$. Arbitrarily small fixed positive exponential rates suffice to contradict the Exponential Time Hypothesis of Impagliazzo and Paturi [4], which asserts that for some $\delta > 0$ there is no deterministic $O(2^{\delta n}(n +$

*HARVARD BUSINESS SCHOOL; DEPARTMENT OF ECONOMICS AND CENTER OF MATHEMATICAL SCIENCES AND APPLICATIONS, HARVARD UNIVERSITY; AND A16Z FRONTIER RESEARCH. This note was developed in the process of navigating, reading, and attempting to distill the arguments in [6] with the assistance of GPT-6 Astra and Claude Opus 5.5 (accessed in part via Poe with the support of Quora, where I am an advisor). *This draft is highly preliminary and heavily LLMathed; I expect to refine it further.* I thank Ittai Abraham, Quang Dao, Joachim Neu, Tim Roughgarden, Justin Thaler, and Kai Zhe Zheng for helpful comments. This work was conducted while I was visiting the Technological Innovation, Entrepreneurship, and Strategic Management (TIES) Group at the MIT Sloan School of Management; I greatly appreciate their hospitality.

$m)^{O(1)}$ -time algorithm for 3-SAT. The algorithmic statement we prove is the following counting result.

Theorem 2. *For every $b \geq 1$ and $\alpha > 0$, the number of satisfying assignments of a 3-CNF on n variables with m clauses, in which each variable occurs at most b times, can be computed in time $O_{b,\alpha}(2^{\alpha n} \text{poly}(n+m))$ and polynomial space.*

Proof of Theorem 1 from Theorem 2. The sparsification lemma [5], in the bounded-occurrence formulation stated in [3, Lemma 1], expresses the input as a disjunction of at most $2^{\sigma n}$ 3-CNFs on the same variables, each with occurrence bound b_σ . Unused variables can be retained as declared variables. After removing duplicate clauses, the input has $O(n^3)$ clauses. For fixed σ , sparsification takes $2^{\sigma n} n^{O_\sigma(1)}$ time; depth-first traversal generates its leaves one at a time in polynomial space. Take $\sigma = \varepsilon/3$ and count each formula at rate $\alpha = \varepsilon/3$. The original formula is satisfiable exactly when one of these counts is positive. The combined time is $2^{2\varepsilon n/3} n^{O_\varepsilon(1)}$, in addition to polynomial-time input preprocessing. The unused $\varepsilon n/3$ in the exponent absorbs the parameter-dependent polynomial in n . Processing the formulas sequentially gives the asserted bounds. $\square$

To apply the counting procedure, we express the formula algebraically: Remove tautological clauses and repeated literals, retaining unused variables. An empty clause gives count 0; if no clauses remain, the count is 2^n . This preprocessing takes polynomial time in $n+m$. If no empty clause is present, the occurrence bound implies that at most bn clauses remain, so the subsequent counting algorithm has polynomial factors in n alone. For each remaining clause e , let $I_e \subseteq [n]$ be its set of variables. For $v \in I_e$, put $\ell_{ev}(y) = y$ for a positive literal and $\ell_{ev}(y) = 1 - y$ for a negative literal. The polynomial

$$\Phi_e((y_{ev})_{v \in I_e}) = 1 - \prod_{v \in I_e} (1 - \ell_{ev}(y_{ev}))$$

is 1 on assignments satisfying clause e and 0 on the others. Thus the desired count is

$$\sum_{x \in \{0,1\}^n} \prod_e \Phi_e((x_v)_{v \in I_e}).$$

Each occurrence of x_v in clause e is replaced by its own form $L_{ev}(T, x)$, linear in x and polynomial in T , with $L_{ev}(0, x) = x_v$. At each of Q designated points $T = t$, fixing a few integer linear forms (*measurements*) makes every clause depend only on variables in one small group. The count then factors into local sums. Section 2 shows how Taylor coefficients at these points reduce to such sums, so Hermite interpolation recovers the value at $T = 0$. Sections 3–4 construct the forms by writing the identity operator as a sum of terms that use variables outside their assigned groups at only a few designated points, plus an exact remainder whose value on an assignment is determined by measurements. Factors $1 - T/t$ cancel each term at those points. The number of vanishing factors bounds the degree, and hence the Taylor order required for recovery. Assignments remain Boolean, although the deformed inputs need not be.

2 Separated deformations and recovery

We work with a general finite family of factors, indexed by e . For each factor, let $I_e \subseteq [n]$ and let $F_e \in \mathbb{Q}[(y_{ev})_{v \in I_e}]$ be a nonzero polynomial, supplied as a list of monomials and rational coefficients. Write

$$Z_F = \sum_{x \in \{0,1\}^n} \prod_e F_e((x_v)_{v \in I_e}), \quad M = \sum_e \deg F_e.$$

A zero factor gives $Z_F = 0$ and is handled separately. For clause indicators, $M = \sum_e |I_e| \leq bn$ when the occurrence bound is b .

Definition 3. Let ρ, w, μ be nonnegative integers and Q, B positive integers. A *separated deformation* of (F_e) with parameters (ρ, Q, w, μ, B) consists of forms

$$L_{ev}(T, x) = \sum_{i=1}^n \lambda_{evi}(T) x_i, \quad \lambda_{evi} \in \mathbb{Q}[T], \quad \deg \lambda_{evi} \leq \rho,$$

with $L_{ev}(0, x) = x_v$, and the following data at each of Q distinct designated rational points t :

1. a finite indexed family of disjoint groups $\mathcal{V}_{t,g}$, $g \in \mathcal{G}_t$, with union $[n]$ and sizes at most w , allowing empty groups, and an assigned group $g_t(e) \in \mathcal{G}_t$ for each factor;
2. integer linear forms $m_{t,1}, \dots, m_{t,\mu_t}$ in x , called *measurements*, where $\mu_t \leq \mu$, each supported in one group (that is, with zero coefficients outside that group) and taking values in $[-B, B]$ on the Boolean cube;
3. separation: if $x, x' \in \{0, 1\}^n$ agree on $\mathcal{V}_{t,g_t(e)}$ and have the same measurement values, then $L_{ev}(t, x) = L_{ev}(t, x')$ for every $v \in I_e$.

The groups, factor assignments, and measurements may depend on t .

The algorithm enumerates measurement values. Once they are fixed, each factor depends only on its assigned group, although the forms L_{ev} may initially involve variables anywhere in $[n]$.

All data are listed explicitly, with sparse encodings permitted. Let N be their total binary encoding length plus $n + M + \rho + Q$.

2.1 The recovery algorithm

Suppose a separated deformation is given. Define

$$H(T) = \sum_{x \in \{0,1\}^n} \prod_e F_e((L_{ev}(T, x))_{v \in I_e}).$$

Then $H(0) = Z_F$ and $\deg H \leq M\rho$. The values at the Q designated points need not determine H , but its Taylor coefficients through order $q = \lfloor M\rho/Q \rfloor$ at each point do, since $M\rho < Q(q+1)$. We retain only these orders, performing computations in $\mathcal{T} = \mathbb{Q}[z]/(z^{q+1})$.

To recover Taylor coefficients, we keep x in the base inputs $L_{ev}(t, x)$ and introduce auxiliary variables X in the perturbations $L_{ev}(t + z, X) - L_{ev}(t, X)$. Setting $X = x$ restores the desired inputs; setting X to a fixed Boolean vector instead makes the perturbations independent of x , preserving separation. Each auxiliary variable brings a factor of z , so each monomial in the order- j coefficient involves at most j auxiliary variables. The proof below shows how inclusion–exclusion converts this observation into local sums with at most j original variables fixed to 1.

The local routine takes a designated point t , a set $J \subseteq [n]$, and shifts $c = (c_{ev})$ in $\mathcal{T}$, independent of x . It computes

$$\mathcal{S}_t(J, c) = \sum_{\substack{x \in \{0,1\}^n \\ x_i = 1 \ (i \in J)}} \prod_e F_e((L_{ev}(t, x) + c_{ev})_{v \in I_e}). \quad (1)$$

Assign each measurement to a group containing its support. Enumerate $\beta = (\beta_1, \dots, \beta_{\mu_t}) \in \{-B, \dots, B\}^{\mu_t}$, where β_j is the proposed value of $m_{t,j}$. A group assignment is *compatible with* β if it realizes every measurement assigned to that group. Initialize a running sum to zero, and for each β :

1. Enumerate assignments to each group $\mathcal{V}_{t,g}$ and retain a compatible representative ξ_g^* , without imposing the constraints $x_i = 1$. If any group has no compatible assignment, discard β .
2. For each group, enumerate its compatible assignments ξ satisfying $\xi_i = 1$ for $i \in J \cap \mathcal{V}_{t,g}$. Let $y^{g,\xi}$ combine ξ with the representatives in the other groups, and compute

$$Z_g(\beta) = \sum_{\substack{\xi \text{ compatible with } \beta \\ \xi_i=1 \ (i \in J \cap \mathcal{V}_{t,g})}} \prod_{e: g_t(e)=g} F_e((L_{ev}(t, y^{g,\xi}) + c_{ev})_{v \in I_e}).$$

The representatives provide values outside the current group; separation makes their choice irrelevant to the factors assigned to this group. They need not satisfy the constraints in the other groups: those constraints are imposed when computing the other local sums.

3. Add $\prod_g Z_g(\beta)$ to the running sum.

Return the running sum. An empty product is 1, so a group assigned no factors still contributes its number of compatible assignments satisfying the constraints.

For recovery, evaluations at indicator vectors determine the auxiliary multilinear coefficients by inclusion–exclusion. Write $\mathbf{1}_U$ for the indicator vector of $U \subseteq [n]$ and proceed as follows.

1. At each designated point t , initialize $H_t(z) = 0$ in $\mathcal{T}$. For every $E \subseteq [n]$ with $|E| \leq q$ and every $U \subseteq E$, form

$$c_{ev}^{t,U}(z) = L_{ev}(t + z, \mathbf{1}_U) - L_{ev}(t, \mathbf{1}_U) \quad \text{in } \mathcal{T}$$

and add $(-1)^{|E|-|U|} \mathcal{S}_t(E, c^{t,U})$ to $H_t(z)$.

2. By Hermite interpolation, find the unique polynomial $\tilde{H}$ of degree less than $Q(q+1)$ satisfying

$$\tilde{H}(t + z) \equiv H_t(z) \pmod{z^{q+1}} \quad \text{at every designated point } t.$$

3. Return $\tilde{H}(0)$.

Process the measurement tuples and pairs (E, U) sequentially, retaining the Taylor coefficient arrays for interpolation.

2.2 Correctness and cost

We verify the local routine and the recovery step in turn, then bound the arithmetic cost and storage.

Lemma 4. *Suppose a separated deformation of (F_e) with parameters (ρ, Q, w, μ, B) is given. Put $M = \sum_e \deg F_e$ and $q = \lfloor M\rho/Q \rfloor$, and let N be the binary encoding length of the factors and deformation data plus $n+M+\rho+Q$. The recovery algorithm returns $Z_F = \sum_{x \in \{0,1\}^n} \prod_e F_e((x_v)_{v \in I_e})$ in time*

$$\text{poly}(N) 2^w (2B + 1)^\mu \sum_{j=0}^{\min(q,n)} \binom{n}{j} 2^j \quad (2)$$

and polynomial space in N .

Proof. Fix t and a feasible measurement tuple β . Since each measurement is supported in its assigned group, the compatible assignments satisfying $x_i = 1$ for $i \in J$ form a Cartesian product. Separation makes each shifted factor depend only on its assigned group and on β . Thus the product of the local sums $Z_g(\beta)$ is exactly the contribution of β to (1). Summing over β proves correctness of the local routine.

For the recovery step, introduce auxiliary variables $X = (X_1, \dots, X_n)$ and put

$$G_x(X, z) = \prod_e F_e \left((L_{ev}(t, x) + L_{ev}(t + z, X) - L_{ev}(t, X))_{v \in I_e} \right) \quad \text{in } \mathcal{T}[X].$$

At $X = x$ the inputs become $L_{ev}(t + z, x)$, so $\sum_x G_x(x, z) = H(t + z)$. Each correction is linear in X and divisible by z . Consequently the coefficient $[z^j]G_x$ has X -degree at most j .

Reduce G_x modulo $X_i^2 - X_i$ for every i , obtaining a multilinear polynomial $\bar{G}_x$ with the same Boolean values. Write

$$\bar{G}_x(X, z) = \sum_{E \subseteq [n]} g_{x,E}(z) \prod_{i \in E} X_i.$$

The degree bound implies $g_{x,E} = 0$ for $|E| > q$. At $X = \mathbf{1}_U$, only monomials indexed by subsets of U survive, so $G_x(\mathbf{1}_U, z) = \sum_{E \subseteq U} g_{x,E}(z)$. Inverting these subset sums by inclusion-exclusion gives

$$g_{x,E}(z) = \sum_{U \subseteq E} (-1)^{|E|-|U|} G_x(\mathbf{1}_U, z).$$

For Boolean x we have

$$G_x(x, z) = \sum_{|E| \leq q} g_{x,E}(z) \prod_{i \in E} x_i.$$

The monomial on the right is 1 exactly when $x_i = 1$ for every $i \in E$. Moreover,

$$G_x(\mathbf{1}_U, z) = \prod_e F_e((L_{ev}(t, x) + c_{ev}^{t,U}(z))_{v \in I_e}).$$

The shifts $c_{ev}^{t,U}(z)$ are independent of x , so fixing the measurements still separates these factors. Summing over x and substituting the coefficient formula therefore yields

$$H(t + z) = \sum_{\substack{E \subseteq [n] \\ |E| \leq q}} \sum_{U \subseteq E} (-1)^{|E|-|U|} \mathcal{S}_t(E, c^{t,U}) = H_t(z) \quad \text{in } \mathcal{T}. \quad (3)$$

Both H and $\tilde{H}$ have degree less than $Q(q + 1)$, and their difference is divisible by $(T - t)^{q+1}$ at each of the Q distinct designated points. It must therefore be zero. This is the uniqueness step in Hermite interpolation [7, Chapter 5], and proves that the algorithm returns $H(0) = Z_F$.

Each enumeration of a group examines at most 2^w assignments, and there are at most $(2B + 1)^\mu$ measurement tuples. Each call to the local routine uses $2^w(2B + 1)^\mu \text{poly}(N)$ arithmetic operations. There are $\sum_{j=0}^{\min(q,n)} \binom{n}{j} 2^j$ pairs (E, U) at each point; the number Q of points is included in N . This proves the operation bound in (2).

The numerical terms added to the encoding length in N bound the degrees and coefficient-array sizes polynomially. The operands also have polynomial bit length in N : the shifts are obtained from the input polynomials. A common denominator for the input data, raised to a polynomially bounded power, clears the denominators in the truncated evaluations; their numerators also have polynomial bit length. Summation adds at most the logarithm of the number of terms, again polynomial in N . The Hermite system has polynomial dimension and entry bit length, so exact rational linear algebra has polynomial bit cost. Sequential enumeration stores only the current assignments, representatives, running sums, and coefficient arrays, using polynomial space. $\square$

The three nonpolynomial factors in (2) account for local enumeration, measurement values, and Taylor recovery. The following criterion bounds their combined exponential rate.

Corollary 5. *Fix $b \geq 1$. Suppose that for every $\delta \in (0, 1]$ one can construct a separated deformation of the clause indicators of each preprocessed 3-CNF with occurrence bound b , at least one clause, and no empty clause, in time $O_{b,\delta}(\text{poly}(n))$, with $N = O_{b,\delta}(\text{poly}(n))$, such that*

$$\begin{aligned} w &\leq \delta n + O_{b,\delta}(1), \\ \mu \log_2(2B + 1) &\leq \delta n + O_{b,\delta}(1), \\ M\rho &\leq \delta Qn. \end{aligned} \tag{4}$$

Then Theorem 2 holds for this b .

Proof. Here $q \leq \delta n$. Since $0 < \delta \leq 1$, the binomial theorem gives

$$\sum_{j=0}^{\min(q,n)} \binom{n}{j} 2^j \leq \delta^{-q} \sum_{j=0}^n \binom{n}{j} (2\delta)^j \leq \delta^{-\delta n} (1 + 2\delta)^n.$$

Lemma 4 bounds the time by

$$O_{b,\delta} \left(2^{n(2\delta + \delta \log_2(1/\delta) + \log_2(1+2\delta))} \text{poly}(n) \right).$$

The exponential rate tends to zero with δ , so it can be made smaller than any prescribed $\alpha > 0$. Adding the input preprocessing cost gives the $\text{poly}(n + m)$ factor in Theorem 2. $\square$

The second bound in (4) concerns the ranges of the measurements as well as their number. A single integer linear form, such as $\sum_i 2^{i-1} x_i$, can encode the entire Boolean assignment. Thus controlling the number of measurements alone does not suffice; their ranges must also be controlled.

3 Balanced labels

Fix an occurrence bound $b \geq 1$ and a formula after the preprocessing in Section 1, with at least one clause and no empty clause. An *incidence* is a pair (e, v) with $v \in I_e$; their number is $M = \sum_e |I_e| \leq bn$.

Fix an integer $s \geq 6$ independently of n , and use $Q = s^6$ designated points. The parameter choice is explained in Section 5. Let $\mathcal{A} = [s]^Q$, with $\Lambda = s^Q$ elements. Assign a label vector $d_e \in \mathcal{A}$ to each clause and $a_v \in \mathcal{A}$ to each variable. At designated point $t \in [Q]$, variable v belongs to group $a_v(t)$ and clause e is assigned to group $d_e(t)$. The incidence label is the pair (a_v, d_e) , with $D = \Lambda^2$ possible values. Balancing the variable labels makes the groups small; balancing incidence labels allows coordinate swaps after bounded padding.

Lemma 6. *In polynomial time one can assign labels so that each variable label occurs $n/\Lambda + O_{b,s}(1)$ times and each incidence label occurs $M/D + O_{b,s}(1)$ times.*

Proof. For each clause size, distribute clauses cyclically among the vectors in $\mathcal{A}$. If M_d counts incidences in clauses with label d , then $M_d = M/\Lambda + O(1)$, with error at most $1 + 2 + 3 = 6$. Classify each variable by the multiset of labels on its incident clauses, including the empty multiset for unused variables. There are at most $(\Lambda + 1)^b$ types. For any clause label d , every variable of a given type has the same number of incidences with clauses labeled d . Distributing the variables of each type cyclically among $\mathcal{A}$ therefore balances all these incidence counts at once. A type κ with

n_κ variables contributes $n_\kappa/\Lambda + O(1)$ to each variable label. Summing proves the first bound. If $\nu_{\kappa,d}$ is the multiplicity of d in type κ , the incidence count for label (a,d) is

$$\sum_{\kappa} \nu_{\kappa,d} (n_\kappa/\Lambda + O(1)) = M_d/\Lambda + O_{b,s}(1) = M/D + O_{b,s}(1).$$

For fixed b, s the numbers of labels and types are constant, so the assignments take polynomial time. $\square$

We give every incidence label the same number of positions so that coordinate swaps preserve the coordinate set. Write n_{ad} for the number of incidences with label (a,d) and put $h = \max_{a,d} n_{ad}$. Place those incidences in positions $(a,d,1), \dots, (a,d,n_{ad})$; the remaining positions $(a,d,n_{ad}+1), \dots, (a,d,h)$ are padding. Thus the coordinate set is

$$\Omega = \mathcal{A} \times \mathcal{A} \times [h], \quad |\Omega| = Dh = M + O_{b,s}(1).$$

For $\omega = (a,d,r)$ write $a_\omega = a$ and $d_\omega = d$. Define $\zeta(x) \in \mathbb{Q}^\Omega$ by placing x_v at the incidence position of (e,v) and zero at each padding position. No Boolean variables are added. Exactly Λ/s label vectors have any prescribed value at coordinate t . Lemma 6 therefore gives $|\mathcal{V}_{t,g}| = n/s + O_{b,s}(1)$ for the groups $\mathcal{V}_{t,g} = \{v : a_v(t) = g\}$, $g \in [s]$. Unused variables remain in these groups and contribute their factors of 2 through the local sums.

4 The tensor construction

Use the groups and clause assignments from Section 3, with designated points $1, \dots, Q$. We first identify a sufficient operator decomposition, then construct it using swaps.

4.1 From a decomposition to a deformation

Call a linear operator A on $\mathbb{Q}^\Omega$ *local at t* if, at each incidence position ω of clause e , the form $(A\zeta(x))_\omega$ depends only on variables in group $d_e(t)$. Linear combinations of operators local at t remain local at t . We will decompose the identity into such operators and an exact remainder Δ , called the *residual operator*. Its value $\Delta\zeta(x)$ is the *residual vector*: the correction needed to recover $\zeta(x)$ from the other terms.

Lemma 7. *Suppose we have an exact decomposition into linear operators on $\mathbb{Q}^\Omega$,*

$$I = \Delta + \sum_{\substack{S \subseteq [Q] \\ |S| \leq \rho}} \gamma_S A_S, \tag{5}$$

where $\gamma_S \in \mathbb{Q}$, each A_S is local at every $t \notin S$, and $\Delta\zeta(x)$ is determined at each point by bounded integer measurements, each supported in one group. Then

$$P(T) = \Delta + \sum_{|S| \leq \rho} \gamma_S \prod_{t \in S} \left(1 - \frac{T}{t}\right) A_S \tag{6}$$

defines forms $L_{ev}(T, x) = (P(T)\zeta(x))_\omega$, where ω is the position of (e,v) , with degree at most ρ , $L_{ev}(0, x) = x_v$, and separation after fixing the measurements.

Proof. Every multiplier equals 1 at zero, so $P(0) = I$ and $L_{ev}(0, x) = x_v$. At $T = t$, the terms with $t \in S$ vanish, making $P(t) - \Delta$ local at t . Specifying the measurements determines the residual vector $\Delta\zeta(x)$ and gives the separation condition for the input forms. Also $\deg P \leq \rho$. $\square$

It remains to construct a decomposition with few possible locality failures per summand and measurements satisfying Corollary 5.

4.2 Why swaps give the decomposition

For each fixed position number r , pair the coordinates $a(t)$ and $d(t)$ of the two labels. This identifies

$$\mathbb{Q}^{A \times A} \cong (\mathbb{Q}^s \otimes \mathbb{Q}^s)^{\otimes Q}.$$

For $t \in [Q]$, let Π_t exchange $a(t)$ and $d(t)$, leaving all other coordinates and r unchanged. At an incidence position of clause e , this permutation reads a position whose variable label has t th coordinate $d_e(t)$. If that position contains a variable, the variable belongs to the assigned group; if it is padding, the value is zero. Thus Π_t is local at t . Swapping other coordinate pairs does not change this t th coordinate, so it preserves that property.

The swaps commute and satisfy $\Pi_t^2 = I$. Write $\Pi_S = \prod_{t \in S} \Pi_t$, with $\Pi_\emptyset = I$, and put $V = \Pi_{[Q]}$. The full swap is local at every designated point, but feeding $V\zeta(x)$ to the clauses generally gives them values of other variables or padding zeros. It therefore cannot replace the original inputs. Its usefulness comes from the identity

$$V\Pi_S = \Pi_{[Q] \setminus S}.$$

This operator performs every swap except those in S , and is local at every designated point outside S . Here the designated point $T = t$ corresponds to coordinate pair t : the multiplier $1 - T/t$ cancels each term that omits that swap and may therefore fail to be local there. Consequently, multiplying a polynomial of total degree at most ρ in the swaps by V gives a linear combination of operators with at most ρ possible locality failures each. It remains to find such a polynomial whose product with V agrees with the identity on a subspace of small codimension. The complementary components will supply the measurements. The bounds below require their proportion to be $o(1/Q)$, since each measurement value uses $O(Q)$ bits.

For one coordinate pair, the standard vectors $\mathbf{b}_{ij}$ of $\mathbb{Q}^{[s] \times [s]}$ give the orthogonal eigenbasis

$$\mathbf{b}_{ii} \ (i \in [s]), \quad \mathbf{b}_{ij} + \mathbf{b}_{ji}, \quad \mathbf{b}_{ij} - \mathbf{b}_{ji} \ (1 \leq i < j \leq s).$$

The eigenvalues of the swap are respectively $+1, +1, -1$. The antisymmetric vectors comprise $s(s-1)/2$ of the s^2 basis vectors, a proportion $p = (s-1)/(2s)$. Tensor this basis over the Q coordinate pairs and take a copy for each $r \in [h]$, obtaining a common eigenbasis $\mathcal{H}$ for all swaps. Set

$$W = \frac{1}{2} \sum_{t=1}^Q (I - \Pi_t).$$

On a basis vector with k antisymmetric factors, V multiplies the Q swap eigenvalues and therefore acts by $(-1)^k$. Each summand $(I - \Pi_t)/2$ acts by 1 on an antisymmetric factor and by 0 otherwise, so W acts by k . A uniformly chosen tensor basis vector chooses its factors independently, each antisymmetric with probability p . Hence $k \sim \text{Bin}(Q, p)$, with standard deviation at most $\sqrt{Q}/2$. Set $R = s^5$ and choose the *central window*

$$k_0 = \lfloor Qp - (R-1)/2 \rfloor, \quad I_0 = \{k_0, \dots, k_0 + R - 1\}.$$

This window contains the values of k for most basis vectors, yet $R/Q = 1/s$ is small. Let $f \in \mathbb{Q}[\tau]$ be the unique polynomial of degree less than R with $f(k) = (-1)^k$ for $k \in I_0$, and put $K = Vf(W)$. Then K is the identity on the span of those basis vectors. A term in W^j is a product of at most j swaps; cancelling repeated swaps by $\Pi_t^2 = I$ cannot increase that number. Thus we can expand $f(W) = \sum_{|S| < R} \gamma_S \Pi_S$. Multiplication by V gives the exact decomposition

$$I = (I - K) + \sum_{|S| < R} \gamma_S \Pi_{[Q] \setminus S}.$$

Here the residual operator is $I - K$, and its value is $\zeta(x) - K\zeta(x)$. The identity holds for any polynomial f of degree less than R ; the choice of f makes this residual vector depend on few basis components. Following (6), define

$$P(T) = (I - K) + \sum_{|S| < R} \gamma_S \prod_{t \in S} \left(1 - \frac{T}{t}\right) \Pi_{[Q] \setminus S}. \quad (7)$$

At the incidence position ω of (e, v) , the output form is $L_{ev}(T, x) = (P(T)\zeta(x))_\omega$.

Measurements. We must determine the residual vector $(I - K)\zeta(x)$ using measurements supported in individual variable groups. There are two steps.

First, call a vector $\mathbf{u} \in \mathcal{H}$ *exceptional* if its number of antisymmetric factors lies outside I_0 . The operator $I - K$ kills every nonexceptional basis vector. Since the basis is orthogonal, the inner products $m_{\mathbf{u}}(x) = \langle \mathbf{u}, \zeta(x) \rangle$ for exceptional $\mathbf{u}$ determine the residual vector:

$$(I - K)\zeta(x) = \sum_{\mathbf{u} \text{ exceptional}} \frac{m_{\mathbf{u}}(x)}{\langle \mathbf{u}, \mathbf{u} \rangle} (I - K)\mathbf{u}.$$

All vectors and denominators on the right are fixed by the construction; only the values $m_{\mathbf{u}}(x)$ depend on the assignment.

Second, each $m_{\mathbf{u}}$ may involve more than one variable group. Fixing its value could couple assignments in those groups, preventing the count from factoring. At each designated point t , write $m_{\mathbf{u}}(x) = \sum_{v=1}^n \alpha_{\mathbf{u},v} x_v$ and split it as

$$m_{\mathbf{u}}(x) = \sum_{g \in [s]} m_{\mathbf{u},t,g}(x), \quad m_{\mathbf{u},t,g}(x) = \sum_{v \in \mathcal{V}_{t,g}} \alpha_{\mathbf{u},v} x_v.$$

Use the nonzero forms $m_{\mathbf{u},t,g}$ as the measurements. Their values determine $m_{\mathbf{u}}$ by addition, while each imposed constraint involves only one group.

This splitting creates at most two measurements per exceptional vector at each designated point. Indeed, the t th tensor factor of $\mathbf{u}$ is $\mathbf{b}_{ii}$ or $\mathbf{b}_{ij} \pm \mathbf{b}_{ji}$, so every position in its support has first-label coordinate i or j . Every variable occupying such a position therefore belongs to group i or j . Supply these measurements and the forms L_{ev} to the recovery algorithm, with factor assignments $g_t(e) = d_e(t)$.

4.3 Separation and parameter bounds

The tensor basis controls both the number and the ranges of the measurements, giving the following bounds.

Lemma 8. *The preceding procedure constructs a separated deformation in polynomial time for fixed b, s , admitting parameter bounds*

$$\begin{aligned} \rho &= R - 1, & Q &= s^6, & w &\leq n/s + O_{b,s}(1), \\ \mu &\leq \frac{128bn}{s^8} + O_{b,s}(1), & B &= 2^Q. \end{aligned}$$

Its size parameter satisfies $N = O_{b,s}(n \log(n + 2))$.

Proof. Since $s \geq 6$, we have $p \in [5/12, 1/2]$, $Q \geq 6R$, and $R \geq 6$. These inequalities imply $I_0 \subseteq [0, Q]$ and that I_0 contains every integer within $R/4$ of Qp .

Each tensor factor has entries in $\{0, \pm 1\}$ and at most two nonzero entries. Thus every $\mathbf{u} \in \mathcal{H}$ has entries in $\{0, \pm 1\}$ and at most 2^Q nonzero entries. Replacing incidence coordinates by variables can merge coefficients, and padding contributes zero; neither operation increases their sum of absolute values. Restricting to one group cannot increase it either. Every measurement therefore has integer coefficients with absolute sum at most 2^Q , so its value on a Boolean assignment lies in $[-2^Q, 2^Q]$. The measurements determine the residual vector by the orthogonal expansion above. Lemma 7 now gives $P(0) = I$, $\deg P \leq R - 1$, and separation at every designated point.

To count the exceptional vectors, use $k \sim \text{Bin}(Q, p)$ (the probability only counts basis vectors; the construction is deterministic). Let Y_t be the indicator that the t th tensor factor is antisymmetric, minus p . The variables Y_t are independent and centered. Since $\mathbb{E}Y_t^4 \leq \mathbb{E}Y_t^2 \leq 1/4$,

$$\begin{aligned} \mathbb{E}(k - Qp)^4 &= \sum_t \mathbb{E}Y_t^4 + 6 \sum_{t < t'} \mathbb{E}Y_t^2 \mathbb{E}Y_{t'}^2 \\ &\leq \frac{Q}{4} + \frac{6}{16} \binom{Q}{2} \leq \frac{Q^2}{4}. \end{aligned}$$

Every exceptional vector has $|k - Qp| \geq R/4$. Applying Markov's inequality to $(k - Qp)^4$ bounds the proportion η of such vectors:

$$\eta \leq \Pr(|k - Qp| \geq R/4) \leq \frac{64Q^2}{R^4} = \frac{64}{s^8}. \quad (8)$$

The two-group support established above gives $\mu \leq 2\eta|\Omega| \leq 128bn/s^8 + O_{b,s}(1)$. The group-size bound follows from Lemma 6.

Every operator acts separately on the D coordinates with a given position number. For fixed s , its coefficients belong to a fixed finite set of rationals. Each input form involves at most D positions, and each measurement arises from at most 2^Q positions. There are $O_{b,s}(n)$ forms and measurements over all points. Sparse encoding, including variable indices, uses $O_{b,s}(n \log(n+2))$ bits; the augmentation in N has the same bound. The construction takes $O_{b,s}(\text{poly}(n))$ time. $\square$

5 The rate and alternative constructions

The bounds just proved make each exponential rate arbitrarily small.

Proof of Theorem 2. Lemma 8 gives

$$\begin{aligned} w &\leq \frac{n}{s} + O_{b,s}(1), \\ \mu \log_2(2B + 1) &\leq \frac{256bn}{s^2} + O_{b,s}(1), \\ \frac{M\rho}{Q} &\leq \frac{bn}{s}. \end{aligned}$$

For the middle bound, use $\log_2(2B + 1) \leq Q + 2 \leq 2s^6$. Given $\delta \in (0, 1]$, choose an integer $s \geq \max\{6, 256b/\delta\}$. All three bounds in (4) follow, so Corollary 5 applies. $\square$

The construction requires both $R/Q \rightarrow 0$, to reduce the Taylor order, and $\eta Q \rightarrow 0$, to reduce the logarithm of the cost of enumerating measurement values relative to n . A second-moment estimate

gives only $\eta Q = O(Q^2/R^2)$, which does not establish the second requirement when $R/Q \rightarrow 0$. The fourth moment gives $\eta Q = O(Q^3/R^4)$, allowing $Q^{3/4} = o(R)$ and $R = o(Q)$. These limits concern $s \rightarrow \infty$; each algorithm fixes s independently of n . With $Q = s^6$ and $R = s^5$, $R/Q = 1/s$ and $Q^3/R^4 = s^{-2}$.

The interpolation of parity used to construct f has a direct antecedent in Alman and Williams [1, Section 2 and Lemma 3.1]. Their interpolation lemma prescribes a symmetric function's values on consecutive Hamming layers, without any distributional assumption. Their Section 2 applies this to a central window containing most inputs under the uniform distribution on the Boolean cube. Here the relevant distribution is instead p -biased: the Boolean coordinates are the eigenvalues $\theta_t \in \{0, 1\}$ of the projections $\Theta_t = (I - \Pi_t)/2$, and parity is $(-1)^{\sum_t \theta_t}$. Alman and Williams also construct probabilistic polynomials with per-input correctness guarantees [1, Theorem 1.2]; here a fixed interpolant suffices because we retain the exceptional components exactly. Their later work on Walsh–Hadamard matrix rigidity uses the same interpolation lemma to represent parity on an interval of inner products [2, Lemmas 3.1–3.2]. Our exceptional set consists of basis vectors rather than matrix entries.

Corollary 5 also in principle permits other constructions. In this tensor basis, a rational polynomial $\psi(\theta_1, \dots, \theta_Q)$ agreeing with parity outside a proportion η of the product distribution with independent coordinates satisfying $\Pr(\theta_t = 1) = p$ can replace $f(\sum_t \theta_t)$. The operator $V\psi(\Theta_1, \dots, \Theta_Q)$ fixes every basis vector on which ψ agrees with parity; the remaining basis vectors supply the measurements. Its expansion in swaps has at most $\deg \psi$ possible locality failures per term. Thus it suffices that $\deg \psi/Q \rightarrow 0$ and $\eta Q \rightarrow 0$, with the stated polynomial construction time and data size. A different representation may replace both the swaps and the basis, but must supply its own group-supported measurements satisfying the number and range bounds of Corollary 5.

References

- [1] J. Alman and R. Williams, *Probabilistic polynomials and Hamming nearest neighbors*, Proc. 56th IEEE Symposium on Foundations of Computer Science (2015), 136–150. Section, theorem, and lemma numbering follows the full version, July 18, 2015: <https://people.csail.mit.edu/rrw/polys-hamming.pdf>.
- [2] J. Alman and R. Williams, *Probabilistic rank and matrix rigidity*, Proc. 49th ACM Symposium on Theory of Computing (2017), 641–652. Lemma numbering follows [arXiv:1611.05558v2](https://arxiv.org/abs/1611.05558v2).
- [3] C. Calabro, R. Impagliazzo, and R. Paturi, *A duality between clause width and clause density for SAT*, Proc. 21st IEEE Conference on Computational Complexity (2006), 252–260.
- [4] R. Impagliazzo and R. Paturi, *On the complexity of k -SAT*, J. Comput. System Sci. **62** (2001), no. 2, 367–375.
- [5] R. Impagliazzo, R. Paturi, and F. Zane, *Which problems have strongly exponential complexity?*, J. Comput. System Sci. **63** (2001), no. 4, 512–530.
- [6] OpenAI, *The Exponential Time Hypothesis is false*, Math Release Preprint, [October 25, 2026 version](#).
- [7] J. von zur Gathen and J. Gerhard, *Modern Computer Algebra*, 3rd ed., Cambridge University Press, 2013.