

Subset Sum in time $O(2^{0.4877n})$: a complete proof

12 October 2026

This note proves that Subset Sum can be solved in time $O(2^{0.4877n})$, in the model and with the guarantees of OpenAI result #138, “Subset Sum in Time $O(2^{0.49n})$ ” (OpenAI, 4 October 2026). We call that work *the paper*. References such as “Lemma 2.3” or “(4.11)” point to the paper. Appendix B restates, without proof, every result, definition and equation of the paper that this note uses, under the paper’s numbering. We rely on the paper’s proofs of these results and do not repeat them. With Appendix B, this note can be read without the paper.

The algorithm keeps the paper’s framework: isolation, a capped preliminary search, a modular checksum, and the subtraction of aliases. We replace two of its parts. A new alias step (Theorem A) works with a smaller modulus, and a family of 43 checksum filters (Theorem B) uses that smaller modulus. Section 1.2 lists the numerical checks.

Contents.

1. The result
2. Notation, parameters and facts from the paper
3. An averaged small-layer lemma
4. The preliminary search
5. Alias phase sums (Theorem A)
6. The checksum with a filter family (Theorem B)
7. Proof of Theorem 1
8. The certificates

Appendices:

- A. The 43 filters.
- B. Results used from the paper.
- C. The multipliers of the 430 moment certificates.
- D. The two verification scripts.

1. The result

Theorem 1. There is a uniform randomized classical algorithm for Subset Sum with the following properties in the word-RAM model of the paper (word length (1.1)). For every input with $n \geq 2$, it halts on every execution and answers correctly with probability at least $2/3$. For every fixed positive integer c , there are constants $C_c > 0$ and $N_c \geq 2$ such that every input with $n \geq N_c$ and $b \leq n^c$ is processed in at most $C_c 2^{0.4877n}$ word operations, on every outcome of the random choices. The program does not depend on c .

Here b is the largest bit length of the input integers and the target, as in the paper. The intermediate bound is

$$2^{(\tau_0+20\epsilon)n+O(n^{0.7})} \text{poly}(n+b+2), \quad \tau_0 = 0.48769, \quad \epsilon = 10^{-8}.$$

Since $\tau_0 + 20\epsilon = 0.4876902 < 0.4877$, the gap absorbs every factor $2^{O(n^{0.7})} \text{poly}(n+n^c)$.

1.1 Changes relative to the paper

Part	Paper	This note
Work exponent	$\tau = 0.48996$	$\tau_0 = 0.48769$
Bits of the large prime P	ηn with $\eta = 0.51772$	ηn with $\eta = 0.51613$
Preliminary search	Proposition 2.4	the same procedure with cap $2^{\lceil \tau_0 n \rceil}$ (Proposition P)
Checksum	Theorem 4.1, one filter	Theorem B: 43 filters, one chosen for each pattern
Alias step	Theorem 5.1, lists the aliases	Theorem A: returns the phase sum over the aliases
Input guard	$b \leq 2^{0.00001n}$	$10^6 \lceil \log(b+1) \rceil \leq n$

The new alias step has five features.

1. One random partition $[n] = M \cup L \cup R$ with a large block, $|M| \approx 0.473n$. It stays fixed for all trials.
2. The value maps of L and R (distinct values with summed phases) are built once.
3. The trials repeat about $2^{\hat{F}}$ times. Here $\hat{F}$ is a sampled estimate of $-\log \mathbb{E}_V 2^{-d(V)}$ over random $(|M| - 2k)$ -subsets V of M .
4. In each trial, the right masks are grouped into classes with equal reduced and unreduced weights.
5. Coverage uses Corollary D, an averaged form of the small-layer property.

A large block gives each alias many representations, so the alias step works with a large prime of only $0.51613n$ bits. The checksum costs $2^{\tau_0 n}$ with $\tau_0 = \eta - \Delta$. At this η , the filter family certifies the budget $\Delta = 0.02844$, so $\tau_0 = 0.48769$.

1.2 What is machine-checked

Only the numerical inequalities are machine-checked:

Statement	Check
Constraints (C1)–(C8) of the alias step	Arb ball arithmetic: script of Appendix D.1, on the parameters of Section 2.2
The 430 moment inequalities of Proposition B	Arb ball arithmetic: script of Appendix D.2, on the data of Appendices A and C

I re-ran both checks on 12 October 2026 (Section 8). Everything else is a written proof. That includes the parts where an error is most likely: the cost accounting of Theorem A (Section 5.7), the application of Corollary D to a fixed signing (Lemma A5), and the changes to the checksum (Section 6). The proof uses the paper’s lemmas as the paper states them. Where an argument of the paper carries over, the note says which one and what changes.

2. Notation, parameters and facts from the paper

2.1 Notation

- $\log$ means $\log_2$. The binary entropy is $h(x) = -x \log x - (1-x) \log(1-x)$.
- $[n] = \{1, \dots, n\}$. For integer weights $w_1, \dots, w_n$ and $I \subseteq [n]$, $w(I) = \sum_{i \in I} w_i$.

- The transformed weights and targets are $c_i = Ca_i + u_i$ and $t_s = Ct + s$ for $0 \leq s \leq nN_0$, as in (2.1). They have $O(b + \log n)$ bits.
- $\Sigma(V) = \{c(I) : I \subseteq V\}$ and $d(V) = |V| - \log |\Sigma(V)|$, as in (2.2).
- $f(q) = \mathbb{E} d(V_q)$ for a uniform q -subset V_q of $[n]$.
- $e_P(x) = \exp(2\pi i x/P)$. For phases $\varphi_i \in \mathbb{Z}/P\mathbb{Z}$, $\chi(I) = e_P(\sum_{i \in I} \varphi_i)$.
- The paper writes $a = \Delta/\Gamma$ and $v(x) = 2D(x) - A(x) - E(x)$. This note writes a_Γ and $w(x)$, because a is the mask fraction and v is a set size.

2.2 Parameters

Symbol	Value	Meaning
τ_0	0.48769	work exponent
η	0.51613	the large prime P has about ηn bits
Δ	$\eta - \tau_0 = 0.02844$	checksum budget
Γ	$2\tau_0 - \eta = 0.45925$	checksum budget
a_Γ	$\Delta/\Gamma = 0.0619270\dots$	
ϵ	10^{-8}	pattern accuracy, as in the paper
ζ	10^{-6}	slack in the preliminary threshold
μ	0.473	the block M has about μn elements
a	0.0082	the left masks have about $a M $ elements
γ	0.06035	the small prime P' has about $\gamma M $ bits
l	0.48377	the left part L has about ln elements
r	$1 - \mu - l = 0.04323$	the right part R has about rn elements

Alias constraints. The parameters satisfy these strict inequalities. Section 8.1 checks them in Arb.

	Inequality	Value of the left side
(C1)	$\gamma < h(2a)/2$	$\gamma = 0.06035$, $h(2a)/2 = 0.06035987$
(C2)	$l + (h(a) - \gamma)\mu < \tau_0$	0.48767640
(C3)	$r + (1 - \gamma)\mu < \tau_0$	0.48768445
(C4)	$1 - \mu + (h(a) + h(\frac{1}{2} - a) - \gamma)\mu - \eta < \tau_0$	0.48768463
(C5)	$h(\frac{1}{2} - a)\mu + \frac{1}{2} - \tau_0 < \tau_0$	0.48521823
(C6)	$0 < r < \tau_0$ and $l < \tau_0$	$r = 0.04323$, $l = 0.48377$
(C7)	$\mu(1 - 2a) < l$	$\mu(1 - 2a) = 0.46524280$
(C8)	$\frac{\mu}{2} \log 3 < \eta$	0.37484363

Let ε_0 be the least slack in (C1)–(C8). Arb gives $\varepsilon_0 \geq 5.37 \cdot 10^{-6}$; (C4) is the tightest. We use $\zeta + 10^{-6} < \varepsilon_0$.

Sizes. For each n put

$$m = 2\lfloor \mu n/2 \rfloor, \quad \sigma = m/2, \quad k = \lfloor am \rfloor, \quad v = m - 2k, \quad |L| = \lfloor ln \rfloor, \quad |R| = n - m - |L|.$$

For all large n these sizes satisfy:

- (S1) $4k \leq m$, so $2k \leq \sigma$; and $2\sigma = v + 2k$.
- (S2) $v < |L|$, by (C7), and $v \leq \lfloor n/2 \rfloor$.
- (S3) $\binom{\sigma}{k} \binom{v}{k} \leq 2^v$. Indeed the left side is at most $2^{m(h(2a)/2 + h(a))}$, and $h(2a)/2 + h(a) = 0.129 < 1 - 2a$.
- (S4) $\binom{\sigma}{k} \geq 2^{\gamma m}$, by (C1).
- (S5) $\binom{m}{k} \leq 2^{h(a)m}$ and $\binom{m}{\sigma-k} \leq 2^{h(1/2-a)m+1}$.

Input guard. We use the guard $10^6 \lceil \log(b+1) \rceil \leq n$. It implies $b+2 \leq 2^{n/10^6+1}$. For each fixed c , inputs with $b \leq n^c$ satisfy it for all large n .

2.3 Facts from the paper

Appendix B states these results in full. We use them without proof.

- **(F1)** Lemma 2.2. For arbitrary integer weights, $d(\emptyset) = 0$ and $d \geq 0$. Adding an index increases d by at least 0 and at most 1. An exchange of one index changes d by at most 1. Hence $d(U) \leq d(V)$ for $U \subseteq V$.
- **(F2)** Signs. Negating the weights of a set F maps $\Sigma(V)$ to its translate by $-c(F \cap V)$. So d does not change.
- **(F3)** Lemma 2.3. The function f is nondecreasing. Its proof builds the exposure martingale of a uniform q -subset and shows that the increments have absolute value at most 1. The same proof works for a uniform q -subset of any fixed finite ground set. Azuma's inequality then gives, for $q \leq n$,

$$\Pr [|d(V_q) - \mathbb{E} d(V_q)| > n^{0.6}] \leq 2e^{-n^{0.2}/2}.$$

- **(F4)** Lemma 2.5. There are absolute constants K and L_* with this property. Let p be uniform among the primes of $[2^{L'}, 2^{L'+1})$ with $L' \geq L_*$, and let $D \neq 0$ be an integer with $|D| < 2^B$. Then $\Pr[p \mid D] \leq KBL'2^{-L'}$.
- **(F5)** Lemma 2.1 (isolation) for the transformation (2.1).
- **(F6)** Section 6.2: bounded uniform draws, primes by trial division, and finite-precision complex sums. A bounded draw fails with probability at most 2^{-n^3} , and a bounded prime draw with probability at most $e^{-c_0 n^2}$.
- **(F7)** Section 3 and Lemma 3.2. The proof of Lemma 3.2 uses only that r and s are fixed trigonometric polynomials. So its conclusions hold for every such pair, except the numerical moment bound (3.7).
- **(F8)** Section 4: the checksum construction, Lemma 4.2, and the estimates (4.1)–(4.25).

3. An averaged small-layer lemma

Let w be integer weights on a finite index set; equal weights are allowed. For a finite index set V and $j \geq 0$, put

$$\Sigma_j(V) = \{w(T) : T \subseteq V, |T| = j\}, \quad a_j(V) = |\Sigma_j(V)| / \binom{|V|}{j},$$

and $\Sigma(V) = \bigcup_j \Sigma_j(V)$, $d(V) = |V| - \log |\Sigma(V)|$.

Lemma 3.1 (covering). Let $0 \leq k \leq t \leq |V|$, and let W be a uniform $(t-k)$ -subset of V . Then $a_t(V) \leq \mathbb{E}_W a_k(V \setminus W)$.

Proof. For each $x \in \Sigma_t(V)$, fix a t -set $T_x \subseteq V$ with $w(T_x) = x$. Consider the pairs (x, W) with $W \subseteq T_x$ and $|W| = t-k$. The map $(x, W) \mapsto (W, x - w(W))$ is injective, because $x = (x - w(W)) + w(W)$. Its second coordinate is $w(T_x \setminus W) \in \Sigma_k(V \setminus W)$. There are $\binom{t}{k} |\Sigma_t(V)|$ pairs, so

$$\binom{t}{k} |\Sigma_t(V)| \leq \sum_{W \subseteq V, |W|=t-k} |\Sigma_k(V \setminus W)|.$$

Divide by $\binom{|V|}{t} \binom{t}{k} = \binom{|V|}{t-k} \binom{|V|-t+k}{k}$. $\square$

Lemma 3.2 (sampling). Let $k \leq s \leq |V|$, and let S be a uniform s -subset of V . Then $\mathbb{E}_S a_k(S) \geq a_k(V)$.

Proof. For each $x \in \Sigma_k(V)$, fix a k -set K_x with $w(K_x) = x$. Then $x \in \Sigma_k(S)$ whenever $K_x \subseteq S$. This event has probability $\binom{|V|-k}{s-k} / \binom{|V|}{s} = \binom{s}{k} / \binom{|V|}{k}$. Sum over x and divide by $\binom{s}{k}$. $\square$

Lemma 3.3 (a good layer). Some t with $2t \leq |V|$ has $|\Sigma(V)| \leq (|V| + 1) |\Sigma_t(V)|$. In particular $a_t(V) \geq 2^{-d(V)} / (|V| + 1)$.

Proof. The $|V| + 1$ layers cover $\Sigma(V)$, so some layer j has $|\Sigma_j(V)| \geq |\Sigma(V)|/(|V| + 1)$. Complementation gives $\Sigma_{|V|-j}(V) = w(V) - \Sigma_j(V)$, so we may take $t = \min(j, |V| - j)$. Finally $\binom{|V|}{t} \leq 2^{|V|}$. $\square$

Corollary D (averaged small layer). Let M be a finite index set with integer weights. Let k, σ, v be integers with $2k \leq \sigma$, $2\sigma \leq v + 2k$, $v \leq |M|$ and $\binom{\sigma}{k} \binom{v}{k} \leq 2^v$. Let S be a uniform σ -subset and V a uniform v -subset of M . Then

$$\mathbb{E}_S a_k(S) \geq \frac{\mathbb{E}_V 2^{-d(V)}}{v + 1}.$$

Proof. The hypotheses give $k \leq \sigma \leq v$. Fix a v -set $V \subseteq M$, and let S_V be a uniform σ -subset of V . We show that $2^{-d(V)}/(v + 1) \leq \mathbb{E} a_k(S_V)$. Take t from Lemma 3.3.

Case $t \geq k$. Lemma 3.1 gives $a_t(V) \leq \mathbb{E}_W a_k(V \setminus W)$, where $|V \setminus W| = v - t + k \geq v/2 + k \geq \sigma$. Lemma 3.2 for $V \setminus W$ and $s = \sigma$ gives $a_k(V \setminus W) \leq \mathbb{E} a_k(S')$ for a uniform σ -subset S' of $V \setminus W$. When W is uniform, S' is a uniform σ -subset of V . Hence $2^{-d(V)}/(v + 1) \leq a_t(V) \leq \mathbb{E} a_k(S_V)$.

Case $t < k$. Then $t < k \leq v/2$, so $|\Sigma_t(V)| \leq \binom{v}{t} \leq \binom{v}{k}$. Lemma 3.3 gives $2^{v-d(V)} \leq (v + 1) \binom{v}{k}$. With the hypothesis, $2^{-d(V)}/(v + 1) \leq \binom{v}{k} 2^{-v} \leq 1/\binom{\sigma}{k}$. Every σ -set S has $a_k(S) \geq 1/\binom{\sigma}{k}$, because $\Sigma_k(S)$ is not empty.

Finally average over V . A uniform σ -subset of a uniform v -subset of M is a uniform σ -subset of M . $\square$

Remark. Signs do not change d (F2), but they change a_k . Theorem A applies Corollary D to one fixed signing of the block for each alias (Lemma A5, Step 2).

4. The preliminary search

Proposition P. Run the procedure of Proposition 2.4 with the list cap $B_n = 2^{\lceil \tau_0 n \rceil}$ in place of $2^{\lceil 0.489985n \rceil}$. It returns the correct decision or CONTINUE. On every execution it uses at most $2^{\tau_0 n} \text{poly}(n + b + 2)$ word operations. If $\mathbb{E} d(V) > (\frac{1}{2} - \tau_0 + \zeta)n$ for a uniform $\lfloor n/2 \rfloor$ -subset V , it returns CONTINUE with probability $O(1/n)$.

Proof. Correctness and the time bound are those of Proposition 2.4. An answer comes from two complete lists, and every merge stops after $O(B_n)$ steps. Let U be either side of the random split; it has $\lfloor n/2 \rfloor$ or $\lceil n/2 \rceil$ elements. Since f is nondecreasing, $\mathbb{E} d(U) > (\frac{1}{2} - \tau_0 + \zeta)n$. Lemma 2.3 gives $\text{Var } d(U) \leq n$, and Chebyshev's inequality gives

$$\Pr [d(U) < (\tfrac{1}{2} - \tau_0 + \tfrac{\zeta}{2})n] \leq \frac{n}{(\zeta n/2)^2} = \frac{4}{\zeta^2 n}.$$

Otherwise $\log |\Sigma(U)| = |U| - d(U) \leq \frac{n}{2} + \frac{1}{2} - (\frac{1}{2} - \tau_0 + \frac{\zeta}{2})n < \tau_0 n$ for $n > 1/\zeta$. Then both lists fit under the cap, and the search decides. The sampler of the split fails with probability at most $n2^{-n^2}$. $\square$

So, for fixed tie weights, either the preliminary search decides with probability $1 - O(1/n)$, or

$$\mathbb{E}_{V \subseteq [n], |V|=\lfloor n/2 \rfloor} d(V) \leq (\tfrac{1}{2} - \tau_0 + \zeta)n. \quad (\text{LD})$$

Theorem A uses (LD) only to bound the number of trials and to prove coverage. The algorithm never evaluates it.

5. Alias phase sums (Theorem A)

Let P be a prime in $[2^{\lfloor \eta n \rfloor}, 2^{1+\lfloor \eta n \rfloor})$ and $b_i = c_i \bmod P \in \{0, \dots, P - 1\}$. The aliases of the target t_s are

$$\mathcal{A}_s(P) = \{I \subseteq [n] : c(I) \equiv t_s \pmod{P}, c(I) \neq t_s\}.$$

An alias I has $b(I) \equiv t_s \pmod{P}$ and $0 \leq b(I) \leq n(P - 1)$. So it has a unique *carry* $q \in \{0, \dots, n - 1\}$ with

$$b(I) = T_{s,q} := (t_s \bmod P) + qP.$$

Idea. In a trial, an alias I is represented by every split of the set $S = (I \cap M) \triangle F$ into a left mask X of size k and a right mask Y of size $\sigma - k$. The small prime P' keeps a $1/P'$ fraction of the left records. So a trial finds I when the left masks of S take about P' distinct values. Corollary D supplies these values on average over S , and $2^{\hat{F}}$ repeated trials turn the average into coverage of every alias. The deficiencies pay for the repetitions: $d(L)$ pays on the left side, and $d(M)$ pays on the right side, because the right masks are grouped by value.

5.1 Statement

Theorem A. There is a randomized procedure with the following properties for all sufficiently large n . Its input is the transformed weights and targets of (2.1) under the guard of Section 2.2, a prime $P \in [2^{\lfloor \gamma m \rfloor}, 2^{1+\lfloor \gamma m \rfloor}]$, and phases $\varphi \in (\mathbb{Z}/P\mathbb{Z})^n$. It returns FAIL or complex numbers $\hat{A}_s$ for $0 \leq s \leq nN_0$.

1. Assume (LD). Let P be uniform among the primes of the interval, and let φ be arbitrary; it may depend on P . Then, with probability $1 - o(1)$, the procedure does not fail and

$$\left| \hat{A}_s - \sum_{I \in \mathcal{A}_s(P)} \chi(I) \right| \leq \frac{1}{16} \quad \text{for every } s.$$

The $o(1)$ depends only on n .

2. On every execution the procedure uses at most $2^{\tau_0 n + 2n^{0.7}} \text{poly}(n + b + 2)$ word operations.

5.2 The procedure

The procedure makes its random draws with the bounded samplers of Section 6.2 (F6). For $F \subseteq M$, the signed weights on M are

$$c_i^F = (-1)^{[i \in F]} c_i, \quad b_i^F = (-1)^{[i \in F]} b_i \quad (i \in M).$$

For every $Z \subseteq M$ they satisfy

$$c^F(Z) = c(Z \triangle F) - c(F), \quad b^F(Z) = b(Z \triangle F) - b(F). \quad (5.1)$$

Indeed $c(Z \triangle F) - c(F) = c(Z \setminus F) + c(F \setminus Z) - c(F) = c(Z \setminus F) - c(Z \cap F)$.

The *key* of a set A is the pair $\kappa(A) = (c(A), b(A)) \in \mathbb{Z}^2$. For a key $\kappa = (x, y)$ we write $c(\kappa) = x$ and $b(\kappa) = y$.

1. **Partition.** Draw a uniform m -subset $M \subseteq [n]$, then a uniform $|L|$ -subset L of $[n] \setminus M$. Put $R = [n] \setminus (M \cup L)$. The partition stays fixed.
2. **Value maps.** For every $A \subseteq L$, compute $\kappa(A)$ and $\chi(A)$. Sort, and store each distinct key κ once with $\Phi_L(\kappa) = \sum_{A \subseteq L: \kappa(A) = \kappa} \chi(A)$. Do the same for R .
3. **Trial count.** Draw n^2 independent uniform v -subsets $V_1, \dots, V_{n^2}$ of M . Compute each $|\Sigma(V_j)|$ exactly by the merge procedure of Proposition 2.4, without a cap. Put

$$\hat{S} = \frac{1}{n^2} \sum_{j=1}^{n^2} \frac{|\Sigma(V_j)|}{2^v} = \frac{1}{n^2} \sum_{j=1}^{n^2} 2^{-d(V_j)}, \quad \hat{F} = \log \frac{1}{\hat{S}} + e_n, \quad T = n^7 2^{e_n} \left\lceil \frac{1}{\hat{S}} \right\rceil,$$

where e_n is an integer with $2n^{0.6} \leq e_n \leq 2n^{0.6} + 1$. Then $n^7 2^{\hat{F}} \leq T \leq 2n^7 2^{\hat{F}}$.

4. **Trials.** For every target index s and every carry $q \in \{0, \dots, n-1\}$, run T independent trials. A trial for (s, q) does the following.
 - (a) Draw a uniform $F \subseteq M$. Put $T^F = T_{s,q} - b(F)$ and $t^F = t_s - c(F)$.
 - (b) Draw a uniform prime $P' \in [2^{\lfloor \gamma m \rfloor}, 2^{1+\lfloor \gamma m \rfloor}]$ and a uniform $j' \in \mathbb{Z}/P'\mathbb{Z}$.
 - (c) *Left records:* all pairs (κ_L, X) with κ_L a key of L , $X \subseteq M$, $|X| = k$, and $b(\kappa_L) + b^F(X) \equiv j' \pmod{P'}$. To list them, sort the keys of L by $b(\kappa_L) \pmod{P'}$; for each X , find the keys with the needed residue by binary search.

- (d) *Right classes*: list all $Y \subseteq M$ with $|Y| = \sigma - k$, and group them by $\lambda(Y) = (b^F(Y), c^F(Y))$. A *class* is a value $\lambda = (b_\lambda, c_\lambda)$ that occurs, together with the list of its members Y .
 - (e) *Right class records*: all pairs (κ_R, λ) with κ_R a key of R , λ a class, and $b(\kappa_R) + b_\lambda \equiv T^F - j' \pmod{P'}$. List them as in (c).
 - (f) *Matching*: group the right class records by the integer $b(\kappa_R) + b_\lambda$, and sort each group by the integer $c(\kappa_R) + c_\lambda$. For each left record (κ_L, X) , find the group with $b(\kappa_R) + b_\lambda = T^F - b(\kappa_L) - b^F(X)$. In it, find by two binary searches the interval of records with $c(\kappa_R) + c_\lambda = t^F - c(\kappa_L) - c^F(X)$. Skip that interval, and *visit* every other record of the group.
 - (g) *Expansion*: for each visited (κ_R, λ) and each member Y of λ , test $X \cap Y = \emptyset$. If the test passes, put $Z = (X \cup Y) \triangle F$, and insert the key (κ_L, κ_R, Z) into the dictionary $\mathcal{D}_s$ unless it is there.
5. **Output.** For each s , return $\hat{A}_s$, a finite-precision value of $\sum_{(\kappa_L, \kappa_R, Z) \in \mathcal{D}_s} \Phi_L(\kappa_L) \Phi_R(\kappa_R) \chi(Z)$, computed as in Section 6.2.
6. **Cap.** Count the elementary steps of Steps 1–5: subsets enumerated, merge operations, masks listed, keys sorted, records formed, records visited, members expanded, dictionary operations, terms summed, and trial divisions. Return FAIL as soon as the count would exceed $2^{\lceil \tau_0 n \rceil + \lceil n^{0.7} \rceil}$. Also return FAIL if a bounded sampler fails.

Each step costs $\text{poly}(n + b)$ word operations. Keys have $O(b + \log n)$ bits, masks have m bits, and the dictionaries are balanced search trees. The finite-precision sums use n^2 fractional bits (Section 6.2). With the cap, this proves part 2 of Theorem A. The rest of this section proves part 1.

5.3 Keys, classes and soundness

For keys κ_L of L and κ_R of R and a set $Z \subseteq M$, the *class* of (κ_L, κ_R, Z) is

$$\mathcal{C}(\kappa_L, \kappa_R, Z) = \{A \cup Z \cup B : A \subseteq L, B \subseteq R, \kappa(A) = \kappa_L, \kappa(B) = \kappa_R\}.$$

Each $I \subseteq [n]$ lies in exactly one class: the class of $(\kappa(I \cap L), \kappa(I \cap R), I \cap M)$. All members of a class have the same $c(I)$ and the same $b(I)$. So either every member is an alias of t_s or none is. In the first case we call the class an *alias class* of t_s . Since phases add,

$$\sum_{I \in \mathcal{C}(\kappa_L, \kappa_R, Z)} \chi(I) = \Phi_L(\kappa_L) \Phi_R(\kappa_R) \chi(Z). \quad (5.2)$$

Lemma A1 (soundness). Every key inserted into $\mathcal{D}_s$ is an alias class of t_s .

Proof. An insertion comes from a left record (κ_L, X) , a visited record (κ_R, λ) , and a member Y of λ with $X \cap Y = \emptyset$; then $Z = (X \cup Y) \triangle F$. Take A, B with $\kappa(A) = \kappa_L$ and $\kappa(B) = \kappa_R$, and put $I = A \cup Z \cup B$. By (5.1), $b^F(X) + b^F(Y) = b^F(X \cup Y) = b(Z) - b(F)$, and the same holds for c . The group condition gives $b(A) + b(Z) - b(F) + b(B) = T^F$. So $b(I) = T_{s,q}$, and $c(I) \equiv b(I) \equiv t_s \pmod{P}$. The skipped interval gives $c(A) + c(Z) - c(F) + c(B) \neq t^F$, so $c(I) \neq t_s$. Hence I is an alias, and so is every member of its class. $\square$

The alias classes of t_s partition $\mathcal{A}_s(P)$, and the dictionary holds each key once. So, by (5.2) and Lemma A1: if $\mathcal{D}_s$ contains every alias class of t_s , then the exact output sum equals $\sum_{I \in \mathcal{A}_s(P)} \chi(I)$.

Lemma A2 (key and class counts).

1. L has at most $(n + 1) |\Sigma(L)| = (n + 1) 2^{|L| - d(L)}$ keys, and each value $c(\kappa)$ occurs in at most $n + 1$ keys. The same holds for R .
2. Every trial has at most $(2m + 1) 2^{m - d(M)}$ right classes.

Proof. (1) A key (x, y) of L has $y \equiv x \pmod{P}$ and $0 \leq y \leq |L|(P - 1)$. So y takes at most $|L| + 1 \leq n + 1$ values for each x . (2) By (5.1), $c^F(Y) = c(Y \triangle F) - c(F)$ takes at most $|\Sigma(M)| = 2^{m - d(M)}$ values. For each of them, $b^F(Y) \equiv c^F(Y) \pmod{P}$ and $|b^F(Y)| < mP$. This leaves at most $2m + 1$ values of $b^F(Y)$. $\square$

5.4 The trial count

Let V be a uniform v -subset of M , and put

$$g_M = -\log \mathbb{E}_V 2^{-d(V)}, \quad e_M = \mathbb{E}_V d(V).$$

Jensen's inequality gives $g_M \leq e_M$, and (F1) gives $e_M \leq d(M)$.

Lemma A3. With probability $1 - o(1)$ over M , L and Step 3, the following hold.

1. $\hat{F} \geq g_M$.
2. $\hat{F} \leq e_M + 3n^{0.6} + 1 \leq d(M) + 3n^{0.6} + 1$.
3. $\hat{F} \leq d(L) + 5n^{0.6} + 1$.
4. Under (LD): $g_M \leq e_M \leq (\frac{1}{2} - \tau_0 + \zeta)n + n^{0.6}$ and $\hat{F} \leq (\frac{1}{2} - \tau_0 + \zeta)n + 4n^{0.6} + 1$.

Proof. (1) $\hat{S}$ is the mean of n^2 independent samples of $2^{-d(V)}$, so $\mathbb{E} \hat{S} = 2^{-g_M}$. Markov's inequality gives $\hat{S} < 2^{2n^{0.6}} 2^{-g_M}$, except with probability $2^{-2n^{0.6}}$. Then $\hat{F} \geq \log(1/\hat{S}) + 2n^{0.6} > g_M$.

(2) Apply (F3) inside M . For fixed M , every $d(V_j)$ is at most $e_M + n^{0.6}$, except with probability $2n^2 e^{-n^{0.2}/2}$. Then $\hat{S} \geq 2^{-e_M - n^{0.6}}$, so $\hat{F} \leq e_M + n^{0.6} + e_n \leq e_M + 3n^{0.6} + 1$. Also $e_M \leq d(M)$.

Next we show $e_M \leq f(v) + n^{0.6}$ with high probability. Exchange one element $i \in M$ for $j \notin M$. Couple V with $V' = V$ if $i \notin V$, and $V' = (V \setminus \{i\}) \cup \{j\}$ otherwise. Then V' is uniform in the new block, and $|d(V) - d(V')| \leq 1$ by (F1). So e_M changes by at most 1 under an exchange. The exposure martingale of M (as in (F3)) and Azuma's inequality give $e_M \leq \mathbb{E}_M e_M + n^{0.6} = f(v) + n^{0.6}$, except with probability $e^{-n^{0.2}/2}$.

(3) L is a uniform $|L|$ -subset of $[n]$. By (F3), $d(L) \geq f(|L|) - n^{0.6}$, except with probability $2e^{-n^{0.2}/2}$. Since $|L| > v$ (S2) and f is nondecreasing, $\hat{F} \leq e_M + 3n^{0.6} + 1 \leq f(v) + 4n^{0.6} + 1 \leq d(L) + 5n^{0.6} + 1$.

(4) $v \leq \lfloor n/2 \rfloor$ (S2), so $f(v) \leq f(\lfloor n/2 \rfloor) \leq (\frac{1}{2} - \tau_0 + \zeta)n$ under (LD). Combine this with the bounds above. $\square$

Let $\mathcal{E}_3$ be the event of Lemma A3. It depends only on M , L and Step 3, which are independent of P .

5.5 Reduction modulo P on half blocks

Call a σ -subset $S \subseteq M$ *bad* for P if some $\lambda \in \{-1, 0, 1\}^S$ has $\sum_{i \in S} \lambda_i c_i \neq 0$ and $P \mid \sum_{i \in S} \lambda_i c_i$. Put $\delta = (\eta - \frac{\mu}{2} \log 3)/2$. Then $\delta > 0.0706$ by (C8).

Lemma A4. For every fixed M , with probability $1 - o(1)$ over P , at most a $2^{-\delta n}$ fraction of the σ -subsets of M are bad.

Proof. Fix S . There are fewer than 3^σ vectors λ . Each nonzero $\sum \lambda_i c_i$ has absolute value below $2^{b+O(\log n)}$. By (F4) with $L' = \lfloor \eta n \rfloor$, P divides it with probability at most $\text{poly}(n)(b + \log n)2^{-\eta n}$. So S is bad with probability at most $\text{poly}(n)(b + \log n)3^\sigma 2^{-\eta n} \leq \text{poly}(n)(b + \log n)2^{-2\delta n}$, as $\sigma \leq \mu n/2$. The expected bad fraction has the same bound. By Markov's inequality, the fraction exceeds $2^{-\delta n}$ with probability at most $\text{poly}(n)(b + \log n)2^{-\delta n}$. This is $o(1)$ by the guard. $\square$

On a good set, reduction modulo P keeps all distinctions between signed subset sums. Precisely: let S be good, let $\epsilon_i \in \{\pm 1\}$ for $i \in S$, and let $X, X' \subseteq S$. Then

$$\sum_{i \in X} \epsilon_i c_i \neq \sum_{i \in X'} \epsilon_i c_i \implies \sum_{i \in X} \epsilon_i b_i \neq \sum_{i \in X'} \epsilon_i b_i. \quad (5.3)$$

The difference of the left sides is $\sum_{i \in S} \lambda_i c_i$ with $\lambda_i = \epsilon_i([i \in X] - [i \in X'])$. It is not zero, so P does not divide it. The difference of the right sides is congruent to it modulo P , so it is not zero.

5.6 Coverage

Lemma A5 (one trial). Assume (LD), $\mathcal{E}_3$, and the event of Lemma A4. Fix s , an alias $I \in \mathcal{A}_s(P)$ with carry q , and one trial for (s, q) . After the trial, $\mathcal{D}_s$ contains the key of the class of I with probability at least $2^{-g_M} n^{-6}$. The threshold for n does not depend on I , s , P or M .

Proof. Put $A = I \cap L$, $B = I \cap R$ and $S = (I \cap M) \triangle F$.

Step 1 (half of the block). F is a uniform subset of M , so S is too. Hence $|S| = \sigma$ with probability $\binom{m}{\sigma} 2^{-m} \geq 1/(m+1)$. Conditional on this event, S is a uniform σ -subset of M , and $F = S \triangle (I \cap M)$.

Step 2 (the signs on S are fixed). Let $i \in S$. Either $i \in I \setminus F$, and then $c_i^F = c_i$; or $i \in F \setminus I$, and then $c_i^F = -c_i$. So on S the signed weights equal the weights

$$c_i^I = c_i \quad (i \in I \cap M), \quad c_i^I = -c_i \quad (i \in M \setminus I).$$

These depend on I but not on S . The same holds for b^F and b^I .

Step 3 (many layer values). Apply Corollary D to the weights c^I on M , with k, σ, v of Section 2.2; (S1)–(S3) give its hypotheses. By (F2), d is the same for c^I and for c . Hence

$$\mathbb{E}_S a_k(S; c^I) \geq \frac{\mathbb{E}_V 2^{-d(V)}}{v+1} = \frac{2^{-g_M}}{v+1}.$$

Let $W_S = \{b(A) + b^F(X) : X \subseteq S, |X| = k\}$ and $B_S = |W_S|$. If S is good, then (5.3) and Step 2 give $B_S \geq |\Sigma_k(S; c^I)| = a_k(S; c^I) \binom{\sigma}{k}$. Bad sets have probability at most $2^{-\delta n}$. By Lemma A3(4), $2^{-\delta n} \leq 2^{-g_M}/(2(v+1))$, because $\frac{1}{2} - \tau_0 + \zeta < 0.013 < \delta$. Since $a_k \leq 1$,

$$\mathbb{E}_S \left[\mathbf{1}_{S \text{ good}} \frac{B_S}{\binom{\sigma}{k}} \right] \geq \frac{2^{-g_M}}{v+1} - 2^{-\delta n} \geq \frac{2^{-g_M}}{2(v+1)}.$$

Step 4 (the small prime keeps many residues). Fix a good S . The set W_S depends on P, M, L, I and S , but not on P' or j' . Its elements lie in $(-nP, nP)$, so differences have at most $\eta n + \log n + 3$ bits. For distinct $x, x' \in W_S$, (F4) with $L' = \lfloor \gamma m \rfloor$ gives $\Pr[P' \mid x - x'] \leq K_0 n^2 2^{-\gamma m}$, for an absolute constant $K_0 \geq 1$. Let z_y be the number of elements of W_S in the residue class y modulo P' . Then

$$\mathbb{E}_{P'} \sum_y z_y^2 \leq B_S + K_0 n^2 B_S^2 2^{-\gamma m}.$$

If $B_S \leq 2^{\gamma m}$, the right side is at most $2K_0 n^2 B_S$. Otherwise it is at most $2K_0 n^2 B_S^2 2^{-\gamma m}$. By Markov's inequality, with probability at least $1/2$ over P' , $\sum_y z_y^2$ is at most twice that bound. Then the Cauchy–Schwarz inequality $(\sum_y z_y)^2 \leq \#\{y : z_y > 0\} \cdot \sum_y z_y^2$ shows that at least $\min(B_S, 2^{\gamma m})/(4K_0 n^2)$ residues are occupied. Since $P' < 2^{1+\gamma m}$, a uniform j' hits an occupied residue with probability at least

$$\frac{\min(B_S 2^{-\gamma m}, 1)}{8K_0 n^2} \geq \frac{B_S}{8K_0 n^2 \binom{\sigma}{k}}.$$

The last step uses $B_S \leq \binom{\sigma}{k}$ and $2^{\gamma m} \leq \binom{\sigma}{k}$ (S4).

Step 5 (a hit is recovered). Suppose $j' \equiv b(A) + b^F(X) \pmod{P'}$ for some $X \subseteq S$ with $|X| = k$. Then the left record $(\kappa(A), X)$ is listed. Put $Y = S \setminus X$; so $|Y| = \sigma - k$ and $X \cap Y = \emptyset$. Since $S \triangle F = I \cap M$, (5.1) gives

$$b(A) + b^F(X) + b(B) + b^F(Y) = b(A) + b(B) + b(I \cap M) - b(F) = T_{s,q} - b(F) = T^F.$$

So the class record $(\kappa(B), \lambda(Y))$ has the residue $T^F - j'$. It is listed, and it lies in the group of the left record. The same computation with c gives the unreduced sum $c(I) - c(F)$. This differs from $t^F = t_s - c(F)$, because I is an alias. So the record is not in the skipped interval, and it is visited. The expansion reaches its member Y and inserts $(\kappa(A), \kappa(B), (X \cup Y) \triangle F) = (\kappa(A), \kappa(B), I \cap M)$, unless the key is already there.

Step 6 (product). The draws of F, P' and j' are independent. Steps 1–5 give a success probability of at least

$$\frac{1}{m+1} \cdot \mathbb{E}_S \left[\mathbf{1}_{S \text{ good}} \cdot \frac{1}{2} \cdot \frac{B_S}{8K_0 n^2 \binom{\sigma}{k}} \right] \geq \frac{2^{-g_M}}{32K_0 n^2 (m+1)(v+1)} \geq 2^{-g_M} n^{-6}$$

for all large n . $\square$

Corollary A6. Assume (LD), $\mathcal{E}_3$, and the event of Lemma A4. Then, except with probability $o(1)$, every alias class of every target is in its dictionary at the end of Step 4.

Proof. Condition on P, φ, M, L and Step 3. The T trials for a pair (s, q) are independent, and $T \geq n^7 2^{\tilde{F}} \geq n^7 2^{g_M}$ by Lemma A3(1). By Lemma A5, all of them miss a fixed alias with carry q with probability at most $(1 - 2^{-g_M} n^{-6})^T \leq e^{-n}$. There are at most $(nN_0 + 1) 2^n$ pairs of a target and a subset, and $(nN_0 + 1) 2^n e^{-n} = o(1)$. $\square$

5.7 Expected work

Lemma A7. Assume (LD) and $\mathcal{E}_3$. Condition on M , L and Step 3, and average over P and the trials; the phases are arbitrary. Then the uncapped procedure takes in expectation at most

$$(b+2)2^{(\tau_0-\varepsilon_0+\zeta)n+O(n^{0.6})}$$

steps.

Proof. Recall $T \leq 2n^7 2^{\hat{F}}$, and that there are $O(n^3)$ pairs (s, q) . Polynomial factors go into $2^{O(n^{0.6})}$. We bound each kind of step.

1. *Value maps and Step 3:* $2^{|L|} + 2^{|R|} + n^2 2^v$ steps. The exponents l, r and $\mu(1-2a)$ are below $\tau_0 - \varepsilon_0$, by (C6) and (C7).
2. *Masks:* each trial lists $\binom{m}{k} + \binom{m}{\sigma-k} \leq 2^{h(1/2-a)m+2}$ masks and sorts them into classes. By Lemma A3(4) and (S5), the total is $2^{(h(1/2-a)\mu+1/2-\tau_0+\zeta)n+O(n^{0.6})}$. By (C5) the exponent is below $(\tau_0 - \varepsilon_0 + \zeta)n$.
3. *Small primes:* each trial tests at most n^3 candidates, each by at most $2^{\gamma m/2+1}$ divisions. The total is $2^{(1/2-\tau_0+\zeta+\gamma\mu/2)n+O(n^{0.6})}$, far below the bound.
4. *Sorting keys by residue:* each trial sorts at most $(n+1)(2^{|L|-d(L)} + 2^{|R|})$ keys. By Lemma A3(3) and (4), the total is at most $2^{ln+O(n^{0.6})} + 2^{(r+1/2-\tau_0+\zeta)n+O(n^{0.6})}$.
5. *Left records:* for fixed choices other than j' , a pair (κ_L, X) is listed with probability $1/P'$, and $1/P' \leq 2^{1-\gamma m}$. So a trial lists at most $(n+1)2^{|L|-d(L)}\binom{m}{k}2^{1-\gamma m}$ left records in expectation. With Lemma A3(3), the total is $2^{(l+(h(a)-\gamma)\mu)n+O(n^{0.6})}$. Use (C2).
6. *Right class records:* by Lemma A2, a trial handles at most $(2m+1)2^{m-d(M)}$ classes. It lists at most $(n+1)2^{|R|}(2m+1)2^{m-d(M)}2^{1-\gamma m}$ class records in expectation. With Lemma A3(2), the total is at most $2^{(r+(1-\gamma)\mu)n+O(n^{0.6})} + 2^{\mu n+O(n^{0.6})}$. Use (C3), and $\mu < \tau_0 - \varepsilon_0$.
7. *Visits and expansions.* Each expansion step handles one tuple $(\kappa_L, X, \kappa_R, Y)$, where Y is a member of a visited class record. Call such a tuple *visited*. Fix a trial's F and P' . For values $x_L \in \Sigma(L)$ and $x_R \in \Sigma(R)$ and masks X, Y of sizes k and $\sigma - k$, put

$$D = x_L + c^F(X) + x_R + c^F(Y) - t^F.$$

D does not depend on P . A visited tuple with $c(\kappa_L) = x_L$ and $c(\kappa_R) = x_R$ needs three events. First, $D \neq 0$, because the exact interval is skipped. Second, $P \mid D$, because the reduced weights add to $T^F \equiv t^F \pmod{P}$ and $b_i \equiv c_i$. Third, both small-prime conditions hold; given everything else, this happens for at most one residue j' . By (F4), $\Pr_P[P \mid D] \leq \text{poly}(n)(b + \log n)2^{-\eta n}$, as $|D| < 2^{b+O(\log n)}$. By Lemma A2, at most $(n+1)^2$ key pairs share the values x_L and x_R . Hence one trial visits in expectation at most

$$\text{poly}(n)(b + \log n)2^{|L|-d(L)+|R|-d(R)}\binom{m}{k}\binom{m}{\sigma-k}2^{-\eta n-\gamma m}$$

tuples. With Lemma A3(3) and $d(R) \geq 0$, the total over all trials is

$$(b + \log n)2^{(1-\mu+(h(a)+h(1/2-a)-\gamma)\mu-\eta)n+O(n^{0.6})}.$$

By (C4) the exponent is below $(\tau_0 - \varepsilon_0)n$.

8. *Matching, dictionaries and output:* polynomial work for each left record, each class record, each visited tuple and each dictionary key.

Every item is at most $(b+2)2^{(\tau_0-\varepsilon_0+\zeta)n+O(n^{0.6})}$. $\square$

The proof of item 7 averages over P , but it does not condition on the event of Lemma A4. Only coverage uses that event.

5.8 Proof of Theorem A

Part 2 holds by Section 5.2. For part 1, consider these failure events.

- $\mathcal{E}_3$ fails: probability $o(1)$, by Lemma A3.
- The event of Lemma A4 fails: probability $o(1)$. Here M is independent of P .
- Some alias class is missed, given the two events above: probability $o(1)$, by Corollary A6.
- The cap is reached. By Markov's inequality and Lemma A7, this has probability at most

$$o(1) + \frac{(b+2)2^{(\tau_0 - \varepsilon_0 + \zeta)n + O(n^{0.6})}}{2^{\tau_0 n + n^{0.7}}}.$$

The guard gives $b+2 \leq 2^{n/10^6+1}$, and $10^{-6} + \zeta < \varepsilon_0$. So this is $o(1)$.

- A bounded sampler fails. A capped execution makes at most $2^{O(n)}$ draws, so this has probability $2^{O(n)}(2^{-n^3} + e^{-cn^2}) = o(1)$.

Outside these events, $\mathcal{D}_s$ contains every alias class of t_s (Corollary A6) and nothing else (Lemma A1). Then the exact output sum is $\sum_{I \in \mathcal{A}_s(P)} \chi(I)$. The evaluation of Section 6.2 computes each phase to error 2^{-2n^2} and rounds each term to n^2 fractional bits. The sums Φ_L and Φ_R have at most 2^n terms, each dictionary has at most 2^n keys, and every modulus is at most 2^n . So the total error is $2^{-n^2+O(n)} \leq 1/16$ for large n . $\square$

6. The checksum with a filter family (Theorem B)

6.1 The family

The family has $J = 43$ filters $\mathbf{f}_v = (r_v, s_v, \kappa_v)$, $v = 0, \dots, 42$, with

$$r_v(x) = \sum_{j=1}^8 r_{v,j} \cos jx, \quad s_v(x) = \sum_{j=1}^8 s_{v,j} \cos jx, \quad \kappa_v \in [1.02371, 1.06368].$$

The coefficients are decimal rationals with at most five digits after the point (Appendix A). For each filter, Section 3 of the paper defines $F_v = 2^{r_v}$, $G_v = 2^{s_v}$, the densities f_v and g_v , their normalized powers $f_{*,v}$ and $g_{*,v}$ with exponent κ_v , and the functions D_v, A_v, E_v of (3.2). The paper uses one filter with $\kappa = 1.0315$; here each filter has its own κ_v . Put

$$u_v = A_v + E_v - D_v, \quad w_v = 2D_v - A_v - E_v.$$

These functions exist on the continuous circle $\mathbb{T}$ and, for each P , on the finite circle $\mathbb{Z}/P\mathbb{Z}$ (F7).

Regions. Put $b_0 = 1$, $b_1 = 1.25$, $b_2 = 1.6$, $b_3 = 2$, $b_4 = 2.6$, $b_5 = 3.5$, $b_6 = 5$, $b_7 = 8$, $b_8 = 15$ and $b_9 = \infty$. A real number W lies in the region ρ_0 if $W \leq \Gamma$, and in the region ρ_i ($1 \leq i \leq 9$) if $\Gamma b_{i-1} < W \leq \Gamma b_i$.

Certificate data. For each filter v and region ρ , Appendix C gives a vector $z = z^{(v,\rho)}$ with nonnegative entries and a set $C = C^{(v,\rho)} \subseteq \{0, \dots, 42\} \setminus \{v\}$. They define the row vector $G = G^{(v,\rho)}(x)$ and a condition on z_0 :

- for ρ_0 : $G = (\Delta - u_v, \Gamma - w_v, (u_v - u_l)_{l \in C})$, and $z_0 \geq 1$;
- for ρ_i with $1 \leq i \leq 8$: $G = (a_\Gamma w_v - u_v, w_v - \Gamma b_{i-1}, \Gamma b_i - w_v, (u_v - u_l)_{l \in C})$, and $z_0 b_{i-1} \geq 1$;
- for ρ_9 : $G = (a_\Gamma w_v - u_v, w_v - 15\Gamma, (u_v - u_l)_{l \in C})$, and $15z_0 \geq 1$.

Every certificate has $\sum_t z_t \leq 418.36$.

Proposition B (certified moments). For each of the $43 \times 10 = 430$ pairs (v, ρ) ,

$$\int_{\mathbb{T}} M(x)^2 2^{z \cdot G(x)} dx < 2^{\eta - 0.00025},$$

where the measure on $\mathbb{T} = \mathbb{R}/2\pi\mathbb{Z}$ is normalized and $M(x)^2 = 2 + 2 \cos x$.

Proof. Section 8.2. The least $\log_2$ margin is 0.00025379, at filter 28 in region ρ_0 . $\square$

Lemma B1 (finite circle). There is an absolute P_0 with the following property. For all $P \geq P_0$ and every filter, the finite-circle functions satisfy the conclusions of Lemma 3.2 other than (3.7). Moreover,

$$\frac{1}{P} \sum_{x \in \mathbb{Z}/P\mathbb{Z}} M(x)^2 2^{z \cdot G(x)} < 2^{\eta - 0.0002} \quad \text{for all 430 pairs } (v, \rho).$$

Proof. By (F7), the proof of Lemma 3.2 applies to each filter. It shows that the finite-circle D_v , A_v and E_v converge uniformly to the continuous ones at the grid points. It also gives the uniform bounds, the order (3.3), and the effective approximations. Each row of G is a fixed linear combination of these functions and constants. So the finite averages converge to the integrals of Proposition B, by the same Riemann-sum estimate. There are finitely many pairs, and each has the slack $2^{\eta - 0.0002} - 2^{\eta - 0.00025} > 0$. $\square$

6.2 The construction

The checksum procedure is the one of the paper's Section 4, with $\tau = \tau_0$ and $\eta = 0.51613$, and with three changes.

1. **Patterns.** For each block B_j and each filter v , a frequency's pattern records rational approximations $D_{v,j}, A_{v,j}, E_{v,j}$ of the block averages of D_v, A_v, E_v at its coordinates. They follow the paper's rounding rule, so each has error at most ϵ , and $D_{v,j} \geq 0$, $A_{v,j} \leq D_{v,j}$, $E_{v,j} \leq D_{v,j}$. The collection $\mathcal{P}$ of all such blockwise tuples has $Q_n = 2^{O(\sqrt{n})}$ elements, since J is fixed. For $\pi \in \mathcal{P}$, put $(D_{v,\pi}, A_{v,\pi}, E_{v,\pi}) = \frac{1}{n} \sum_j d_j(D_{v,j}, A_{v,j}, E_{v,j})$, $U_{v,\pi} = A_{v,\pi} + E_{v,\pi} - D_{v,\pi}$ and $W_{v,\pi} = 2D_{v,\pi} - A_{v,\pi} - E_{v,\pi}$.
2. **Filter choice.** For $\pi \in \mathcal{P}$, let $v^*(\pi)$ be the least index v that maximizes $U_{v,\pi}$.
3. **Tables.** For a pattern π with $v = v^*(\pi)$, run the paper's construction with filter v . That is, α_π and H_π come from (4.8) with $(D_\pi, A_\pi, E_\pi, U_\pi, W_\pi) := (D_{v,\pi}, A_{v,\pi}, E_{v,\pi}, U_{v,\pi}, W_{v,\pi})$. The table counts are $R_j = 2^{\lceil (\alpha_\pi D_{v,j} + 4\epsilon) d_j \rceil}$. The scores are $\log f_{*,v}$ and $\log g_{*,v}$, and the thresholds are $\alpha_\pi A_{v,j} - 3\epsilon$ and $\alpha_\pi E_{v,j} - 3\epsilon$. The pattern is summed directly if $\alpha_\pi U_{v,\pi} \geq \Delta$. Otherwise it is sampled with $N = 2^{\lceil \tau_0 n \rceil}$ draws.

Multiplicities, caps, sampling and the finite-precision arithmetic do not change.

Lemma 4.2 holds for this construction with $U_\pi := U_{v^*(\pi), \pi}$. Its proof treats one pattern at a time, with one filter. Part (i) uses only that the pattern of k records the block averages at the $x_i(k)$ for the filter of that pattern. Parts (ii) and (iii) use only (4.5), (4.8), (4.9), the normalization of $f_{*,v}$ and $g_{*,v}$, and the joint moment (4.14). These hold for every filter.

6.3 Domination and the error bound

Let $\mathcal{G}$ be the event of Lemma 4.2(iii). For a frequency k , put $x_i = x_i(k)$, and let $\bar{u}_l$ and $\bar{w}_l$ be the averages of u_l and w_l over the coordinates $x_1, \dots, x_n$. As in (4.21), the pattern π of k satisfies, for every filter l ,

$$|U_{l,\pi} - \bar{u}_l| \leq 3\epsilon, \quad |W_{l,\pi} - \bar{w}_l| \leq 4\epsilon. \quad (6.1)$$

Lemma B2 (domination). Let π be a sampled pattern, $v = v^*(\pi)$, and ρ the region of $W_{v,\pi}$. Let z and G be the certificate data for (v, ρ) . Then every frequency k of pattern π satisfies

$$-\alpha_\pi U_{v,\pi} \leq -\Delta + \frac{1}{n} \sum_{i=1}^n z \cdot G(x_i) + 6\epsilon \sum_t z_t.$$

Proof. Let $\bar{G} = \frac{1}{n} \sum_i G(x_i)$. Let G^π be the same rows with $\bar{u}_v$, $\bar{w}_v$ and $\bar{u}_l$ replaced by $U_{v,\pi}$, $W_{v,\pi}$ and $U_{l,\pi}$. Each row is affine in at most two of these quantities, with coefficients of absolute value at most 1 ($a_\Gamma < 1$). By (6.1), $|\bar{G}_t - G_t^\pi| \leq 6\epsilon$ for every row t . In detail, the rows have errors at most 3ϵ , 4ϵ , 6ϵ , $(4a_\Gamma + 3)\epsilon$, 4ϵ and 4ϵ .

The rows $t \geq 1$ have $G_t^\pi \geq 0$. The rows $\Gamma - W$, $W - \Gamma b_{i-1}$ and $\Gamma b_i - W$ are nonnegative, because $W_{v,\pi}$ lies in the region ρ . The rows $U_{v,\pi} - U_{l,\pi}$ are nonnegative, because v maximizes $U_{\cdot,\pi}$. Since $z \geq 0$,

$$z \cdot \bar{G} \geq z_0 G_0^\pi - 6\epsilon \sum_t z_t.$$

Region ρ_0 . Here $W_{v,\pi} \leq \Gamma$, so $\alpha_\pi = 1$. Sampling means $U_{v,\pi} < \Delta$. Then $G_0^\pi = \Delta - U_{v,\pi} > 0$ and $z_0 \geq 1$ give $-U_{v,\pi} = -\Delta + G_0^\pi \leq -\Delta + z_0 G_0^\pi$.

Region ρ_i , $i \geq 1$. Here $W_{v,\pi} > \Gamma b_{i-1} \geq \Gamma$. So $\alpha_\pi = \Gamma/W_{v,\pi} < 1/b_{i-1} \leq z_0$, and $\alpha_\pi a_\Gamma W_{v,\pi} = \Delta$. Sampling means $\alpha_\pi U_{v,\pi} < \Delta$, that is, $G_0^\pi = a_\Gamma W_{v,\pi} - U_{v,\pi} > 0$. Then $-\alpha_\pi U_{v,\pi} = -\Delta + \alpha_\pi G_0^\pi \leq -\Delta + z_0 G_0^\pi$.

In both cases, combine with the display above. $\square$

Lemma B3 (one sampled pattern). Let π be a sampled pattern and ρ a target residue, and let $E_{\pi,\rho}$ be the error of the ideal estimate for π . Then $\mathbb{E}[\mathbf{1}_G |E_{\pi,\rho}|^2] \leq 2^{-0.00017n+2}$ for all large n .

Proof. Put $c_z = 6 \sum_t z_t \leq 2511$. By (4.11), (4.4) and Lemma B2, every $k < P$ satisfies

$$\mathbf{1}_G \mathbf{1}_{\{\text{pattern}(k)=\pi\}} L_\pi |J_\rho(k)|^2 \leq 2^{(\eta-\Delta+(18+c_z)\epsilon)n} \prod_{i=1}^n M(x_i(k))^2 2^{z \cdot G(x_i(k))}.$$

The right side does not depend on the pattern or on $\mathcal{G}$. For fixed k , the $x_i(k)$ are independent and uniform on $\mathbb{Z}/P\mathbb{Z}$ under the original phase law. So Lemma B1 bounds the expectation of the right side by $2^{(\eta-\Delta+(18+c_z)\epsilon)n} 2^{(\eta-0.0002)n}$. As in (4.25), take the expectation of (4.20) and sum over the P frequencies:

$$\mathbb{E}[\mathbf{1}_G |E_{\pi,\rho}|^2] \leq \frac{2^{(2\eta-\Delta+(18+c_z)\epsilon-0.0002)n}}{NP} \leq 2^{((18+c_z)\epsilon-0.0002)n+2}.$$

The last step uses $N \geq 2^{\tau_0 n}$, $P \geq 2^{\eta n-2}$ and $\eta - \Delta - \tau_0 = 0$. Finally $(18 + c_z)\epsilon < 2.6 \cdot 10^{-5}$. $\square$

6.4 Theorem B

Theorem B. Fix a constant $d > 0$. There is a uniform randomized procedure with the following properties for all sufficiently large n . Its inputs are an integer P with $|\log P - \eta n| \leq 2$, residues $b_1, \dots, b_n$ modulo P , phases $\varphi_1, \dots, \varphi_n$, and a list $\mathcal{T}$ of at most n^d target residues. It returns FAIL or estimates $\hat{V}_\rho$, $\rho \in \mathcal{T}$, of

$$V_\rho = \sum_{I \subseteq [n]: b(I) \equiv \rho \pmod{P}} \chi(I).$$

If the phases are independent and uniform, then with probability $1 - o(1)$ over the phases and the procedure's own draws, it does not fail and $|\hat{V}_\rho - V_\rho| \leq 1/8$ for every $\rho \in \mathcal{T}$. The $o(1)$ is uniform in P , the residues and $\mathcal{T}$. On every execution the procedure uses at most $2^{(\tau_0+20\epsilon)n+O(\sqrt{n})} \text{poly}(n + |\mathcal{T}|)$ word operations.

Proof. This is Theorem 4.1 with new constants, a filter family, and phases given as input. The paper's proof applies with these changes.

- *Sections 4.1–4.3.* Use the construction of Section 6.2 above; Lemma 4.2 holds, as explained there. The paper's procedure draws the phases itself. Its analysis uses only that the phases are uniform, independent, and independent of the tables and the samples. This holds here.
- *Section 4.4.* Direct patterns are exact on $\mathcal{G}$, as before. For sampled patterns, Lemma B3 replaces (4.24)–(4.25). Cauchy–Schwarz over the $Q_n = 2^{O(\sqrt{n})}$ patterns gives $\mathbb{E}[\mathbf{1}_G |\sum_\pi E_{\pi,\rho}|^2] \leq Q_n^2 2^{-0.00017n+2}$. Markov's inequality and a union bound over $\mathcal{T}$ show that, with probability $1 - o(1)$, $\mathcal{G}$ holds and every ideal estimate has error at most $1/16$.
- *Section 4.5.* Resources and precision do not change. The pattern of a frequency needs $3Jq$ approximate block averages; Lemma B1 computes each in polynomial time. The arithmetic error is at most $1/16$. $\square$

7. Proof of Theorem 1

The algorithm.

1. If $n < n_*$ or $10^6 \lceil \log(b+1) \rceil > n$, run the exact meet-in-the-middle fallback of Section 6.3, and stop.
2. Draw the tie weights, and form c_i and t_s by (2.1).
3. Run the preliminary search of Proposition P. If it decides, return its answer.
4. Draw the prime P uniformly from $[2^{\lfloor \eta n \rfloor}, 2^{1+\lfloor \eta n \rfloor})$ (Section 6.2). Draw independent uniform phases $\varphi_1, \dots, \varphi_n \in \mathbb{Z}/P\mathbb{Z}$, and retain them. Put $b_i = c_i \bmod P$.
5. Run the procedure of Theorem A on (c, t, P, φ) . If it fails, return NO.
6. Run the procedure of Theorem B on (P, b, φ) with the target residues $t_s \bmod P$. If it fails, return NO.
7. For each s , put $\hat{B}_s = \hat{V}_{t_s \bmod P} - \hat{A}_s$. Return YES if $|\hat{B}_s| \geq 1/2$ for some s , and NO otherwise. The test compares the sum of the squares of the dyadic real and imaginary parts with $1/4$.

Correctness. Since $b_i \equiv c_i \pmod{P}$, the sum $V_{t_s \bmod P}$ runs over the subsets with $c(I) \equiv t_s \pmod{P}$. So for every s ,

$$V_{t_s \bmod P} - \sum_{I \in \mathcal{A}_s(P)} \chi(I) = B_s := \sum_{I: c(I)=t_s} \chi(I).$$

Fix the tie weights. If (LD) fails, Proposition P decides with probability $1 - O(1/n)$. If (LD) holds, Theorem A gives $|\hat{A}_s - \sum_{I \in \mathcal{A}_s(P)} \chi(I)| \leq 1/16$ for all s with probability $1 - o(1)$. Its success event depends only on P and its own draws, not on the phases. For every fixed P , Theorem B gives $|\hat{V}_\rho - V_\rho| \leq 1/8$ for all targets, with probability $1 - o(1)$ over the phases and its own draws. As in the paper, the main procedures also draw their randomness when the preliminary search decides; then a union bound composes the three statements. So, except with probability $o(1)$, either the preliminary answer is returned, or $|\hat{B}_s - B_s| < 1/4$ for all s .

In the second case the decision is correct, as in Section 6.3. On a NO instance every B_s is 0, by Lemma 2.1. On a YES instance, with probability at least $11/12$ some t_s has exactly one representing subset, and then $|B_s| = 1$. All $o(1)$ terms depend only on n , so a fixed cutoff n_* gives success probability at least $5/6$. Below n_* and when the guard fails, the fallback is exact.

Running time. Take an execution with $n \geq n_*$ that passes the guard.

- Tie weights and the transformation: $\text{poly}(n+b)$.
- Proposition P: $2^{\tau_0 n} \text{poly}(n+b+2)$.
- The prime P : at most n^3 candidates, each tested by trial division in $2^{\eta n/2+O(1)} \text{poly}(n)$.
- Theorem A: $2^{\tau_0 n + 2n^{0.7}} \text{poly}(n+b+2)$.
- Theorem B: $2^{(\tau_0 + 20\epsilon)n + O(\sqrt{n})} \text{poly}(n)$, as there are $O(n^2)$ targets.

So the total is at most $K_0 2^{(\tau_0 + 20\epsilon)n + 2n^{0.7}} (n+b+2)^K$ for fixed constants K_0 and K . Fix c . For large n , every $b \leq n^c$ passes the guard, and

$$K_0 2^{0.4876902n + 2n^{0.7}} (n + n^c + 2)^K = O_c(2^{0.4877n}).$$

The address-space argument of Section 6.3 applies unchanged: the logarithm of the operation count is $0.4877n + O(\log(n+b+2))$, which is below the word length w . The fallback halts on every input and is exact. $\square$

8. The certificates

8.1 The alias constraints

The script of Appendix D.1 reads $\tau_0, \eta, a, \mu, \gamma, l$ (Section 2.2) as exact decimal rationals. It evaluates (C1)–(C8) in Arb ball arithmetic at 200-bit precision, with $h(x) = -(x \ln x + (1-x) \ln(1-x))/\ln 2$. The run on 12 October 2026 printed:

```
h(2a)/2      [0.060359873 +/- 1.01e-10]
left list    [0.48767640 +/- 1.94e-9]
right classes [0.48768445 +/- 3e-13]
pairs        [0.48768463 +/- 6.24e-10]
right masks, all trials [0.48521823 +/- 2.57e-9]
left masks, all trials [0.044761952 +/- 6.27e-11]
(mu/2) log2 3 [0.37484363 +/- 1.43e-9]
v            [0.46524280 +/- 3e-13]
l            [0.48377000 +/- 3e-13]
r            [0.043230000 +/- 1e-14]
least slack eps0 >= [5.3706e-6 +/- 2.34e-11] (lower end of the Arb balls)
ALL ALIAS CONSTRAINTS PASS (strict)
```

The “left masks” line is the left-mask analogue of (C5). It is implied by (C5), because $h(a) < h(\frac{1}{2} - a)$.

8.2 The 430 region integrals

The script of Appendix D.2 follows the method of the paper’s Appendix A (see B.5). It uses $N = 8192$ quadrature nodes, the inner strip half-width $\sigma = 0.45$, the outer strip half-width $2\beta = 0.38$, and 200-bit precision. For each filter (r, s, κ) it does the following.

1. **Node values.** For $x_k = 2\pi k/N$, Arb computes $r(x_k)$, $s(x_k)$, $F(x_k) = 2^{r(x_k)}$ and $G(x_k) = 2^{s(x_k)}$.
2. **Normalizers.** $Z_f = \int F$ lies in $T_N(F) + [-e, e]$ with $e = 2B_0/(e^{N\sigma} - 1)$ and $B_0 = 2^R$, $R = \sum_j |r_j| \cosh(j\sigma)$. This is the analytic trapezoid bound (A.6) of B.5 with N nodes in place of 64, because $|F| \leq 2^R$ on the strip $|\operatorname{Im} z| \leq \sigma$. The same holds for Z_g , $Z_{f,\kappa} = \int F^\kappa$ and $Z_{g,\kappa}$.
3. **Convolutions.** The script encloses $\mathcal{P}(x) = \int F(x+w)G(w)dw$ and $\mathcal{Y}(x) = \int F(x+w)G(w)(r(x+w) + s(w))dw$ at all nodes. The trapezoid sums are exact cyclic correlations, computed by Arb polynomial products. The error bounds use $B_0 = 2^{R_r+R_s}$ and $(R_r + R_s)2^{R_r+R_s}$.
4. **Scores.** Put $L = \log \mathcal{P} - \log(Z_f Z_g)$, $B = \mathcal{Y}/\mathcal{P} - \log(Z_f Z_g)$ and $H = \log Z_{f,\kappa} + \log Z_{g,\kappa} - \kappa \log(Z_f Z_g)$. The identities (A.2) give $D = B - L$, $u = L + (\kappa - 1)B - H$ and $w = (2 - \kappa)B - 2L + H$. So the resulting balls contain the exact values $u(x_k)$ and $w(x_k)$.
5. **Outer strip.** As in the paper’s Appendix A.4 (see B.5), on the strip $|\operatorname{Im} z| \leq 2\beta$: let r_β have the coefficients $r_j \cosh(j\beta)$, and put $A_f = \int 2^{2r_\beta}/Z_f^2$; define A_g in the same way. Then $q_0 = \sqrt{(A_f - 1)(A_g - 1)}$ bounds $|Q - 1|$, and $q_b = \sqrt{A_f A_g} (\sum_j |r_j| \cosh(j\beta) + \sum_j |s_j| \cosh(j\beta) + |\log(Z_f Z_g)|)$ bounds $|QB|$. All 43 filters have $q_0 < 0.677$. Then $|\operatorname{Re} L| \leq -\log(1 - q_0)$ and $|B| \leq q_b/(1 - q_0)$. These give bounds U^* on $|\operatorname{Re} u|$ and W^* on $|\operatorname{Re} w|$.

For each certificate (v, ρ) , the script averages the node values of $M^{2z \cdot G}$. On the outer strip, the real part of each row is at most the matching combination of Δ , Γ , b_i , U^* and W^* . With $|M(z)^2| \leq 2 + 2 \cosh 2\beta$, this bounds the integrand by some B_0 . The outer error is at most $2B_0/(e^{2\beta N} - 1)$, which is below 10^{-300} for every certificate. The script checks that the average plus the error is below 2^η . It also asserts $z \geq 0$ and the condition on z_0 .

The trapezoid bound needs analyticity on a neighborhood of the closed strip. The inner integrands are entire. For the outer integrand, the bounds hold with $q_0 < 1$ strictly; so one can apply the bound on the strip of half-width $0.38 - \delta$ for a small $\delta > 0$. This changes the error term by a negligible amount.

Result. All 430 inequalities hold. The run on 12 October 2026 reproduced the log of 9 October 2026 line by line. The least $\log_2$ margins, by region:

Region	Least $\log_2$ margin	Filter
ρ_0 : $W \leq \Gamma$	0.00025379	28
ρ_1 : $1 < W/\Gamma \leq 1.25$	0.00035951	34

Region	Least $\log_2$ margin	Filter
ρ_2 : $1.25 < W/\Gamma \leq 1.6$	4.1558	3
ρ_3 : $1.6 < W/\Gamma \leq 2$	10.605	3
ρ_4 to ρ_9 : $W/\Gamma > 2$	at least 17.967	3

I also checked the certificate data directly: every filter has all ten regions, every z is nonnegative, every condition on z_0 holds, and no set $C^{(v,\rho)}$ contains v .

8.3 Repeating the checks

Both scripts need Python with python-flint 0.9.0. Each takes one argument, a JSON file that holds the data printed in this note. For the script of Appendix D.1, the file is one object with the keys `tau0`, `eta`, `a`, `mu`, `gamma` and `l`, set to the values of Section 2.2. For the script of Appendix D.2, it is one object with the keys `tau0` and `eta`, a list `filters` of objects `{r_coef, s_coef, kappa}` from Appendix A, and a list `certs` of objects `{v, case, lo, hi, z, cons}` from Appendix C. Here `case` is "case1" for ρ_0 and "band" otherwise; `lo` and `hi` are b_{i-1} and b_i for ρ_i , and `hi` is null for ρ_9 ; `cons` is the set C . The second script takes a few minutes.

Provenance. A cutting-plane search produced the filters, and linear programming produced the multipliers z . A separate optimization produced the alias parameters. These programs are not part of the proof. The certificates are checked independently of them.

8.4 Limits

The alias constraints (C1)–(C4) are tight to about 10^{-5} , and the checksum certificate has a margin of 0.00025. So this design cannot certify $\tau_0 = 0.4876$. At $\tau_0 = 0.4876$ the cutting-plane search for filters stalls at a $\log_2$ margin of -0.0015 . Further progress needs a new idea.

Appendix A. The 43 filters

$r_v(x) = \sum_{j=1}^8 r_{v,j} \cos jx$ and $s_v(x) = \sum_{j=1}^8 s_{v,j} \cos jx$. Appendix C gives the multipliers $z^{(v,\rho)}$ and the sets $C^{(v,\rho)}$.

v	κ_v	$r_{v,1}, \dots, r_{v,8}$	$s_{v,1}, \dots, s_{v,8}$
0	1.02371	0.99307, 0.47365, 0.24703, -0.26955, 0.07942, -0.1185, -0.00587, 0.06006	1.29217, -0.81333, 0.44354, 0.08221, 0.05022, 0.11128, -0.02479, -0.0816
1	1.03833	1.11312, 0.4803, -0.14032, -0.06862, 0.07166, -0.03251, -0.0345, 0.02296	1.37762, -0.74295, 0.12852, -0.0223, 0.02354, 0.00279, -0.04016, -0.04383
2	1.05826	1.20389, 0.34962, -0.15858, -0.02392, 0.02946, -0.00605, -0.00528, 0.0034	1.41719, -0.61945, 0.04713, -0.00707, 0.00439, 0.00127, -0.00975, -0.00124
3	1.06368	0.6425, 0.54635, 0.73657, -0.56823, -0.21433, 0.03123, -0.00347, 0.01585	0.87763, -0.69266, 0.73949, 0.38951, -0.20075, 0.0364, -0.09908, 0.01145
4	1.04768	0.76631, 0.61145, 0.24893, -0.41359, -0.49022, 0.10407, -0.01777, 0.05843	1.07613, -0.70375, 0.20271, 0.45459, -0.60319, 0.14968, -0.10179, -0.04873
5	1.0433	0.83495, 0.46517, 0.5449, -0.47927, -0.07681, 0.34334, -0.11624, -0.00504	1.08519, -0.70466, 0.6481, 0.22751, -0.03469, -0.23636, -0.12224, 0.0097
6	1.03556	0.87031, 0.62799, 0.1799, -0.35158, -0.09333, 0.32235, -0.36841, 0.01375	1.13882, -0.81379, 0.36137, 0.19229, -0.1388, -0.11502, -0.3531, 0.01982
7	1.03283	0.91928, 0.45906, 0.37137, -0.36271, -0.39634, 0.24445, 0.0778, 0.02384	1.16315, -0.65175, 0.4156, 0.24669, -0.31226, -0.1391, 0.05494, -0.08916
8	1.03767	0.78095, 0.64691, 0.41453, -0.41692, -0.25846, 0.07514, 0.03248, 0.12195	1.12144, -0.82371, 0.47376, 0.28329, -0.23722, 0.02516, -0.05553, -0.09075
9	1.03461	0.86927, 0.53001, 0.41132, -0.44655, -0.2227, 0.32734, -0.04267, 0.01694	1.11897, -0.71831, 0.49366, 0.25437, -0.15994, -0.2237, -0.03326, -0.05703
10	1.03567	0.8321, 0.52819, 0.46889, -0.20175, -0.42154, 0.04074, 0.24988, -0.0618	1.16305, -0.75504, 0.56489, 0.07749, -0.3697, 0.06344, 0.05195, 0.09625
11	1.03551	0.84221, 0.53758, 0.37518, -0.42183, -0.31017, -0.0051, 0.31945, 0.00946	1.13288, -0.69997, 0.35377, 0.34385, -0.34748, 0.10405, 0.1547, -0.03961
12	1.03316	0.87274, 0.48576, 0.45208, -0.38394, -0.35036, 0.1767, 0.08996, 0.03503	1.14932, -0.69434, 0.48876, 0.25489, -0.32142, -0.05422, 0.01517, -0.03626
13	1.03419	0.85174, 0.56596, 0.38584, -0.3491, -0.16892, 0.23733, -0.2596, 0.02587	1.13489, -0.79903, 0.53282, 0.1833, -0.15998, -0.07574, -0.28489, -0.00352
14	1.03247	0.88559, 0.53721, 0.30733, -0.25829, -0.41024, 0.37263, 0.04931, 0.01665	1.14677, -0.72577, 0.40769, 0.13659, -0.29855, -0.24429, 0.06347, -0.0611
15	1.03454	0.84001, 0.57254, 0.38289, -0.34192, -0.38508, 0.27044, 0.04167, 0.01266	1.12958, -0.75004, 0.4546, 0.21648, -0.3285, -0.11539, -0.00938, -0.0199

v	κ_v	$r_{v,1}, \dots, r_{v,8}$	$s_{v,1}, \dots, s_{v,8}$
16	1.03481	0.8362, 0.53334, 0.45578, -0.40199, -0.29198, -0.0118, 0.27142, 0.01226	1.13334, -0.72772, 0.46413, 0.26541, -0.28542, 0.08643, 0.14305, -0.03728
17	1.03175	0.89625, 0.50135, 0.36394, -0.37738, -0.32793, 0.31977, 0.05621, 0.02428	1.15022, -0.68644, 0.42637, 0.24349, -0.26955, -0.19193, 0.04992, -0.06989
18	1.03342	0.86094, 0.53068, 0.4075, -0.35513, -0.35546, 0.27595, 0.10329, -0.00102	1.13535, -0.70827, 0.47222, 0.21607, -0.28977, -0.15573, 0.06196, -0.0245
19	1.03267	0.88294, 0.51847, 0.38488, -0.42188, -0.29302, 0.27218, 0.0419, 0.04304	1.13389, -0.70929, 0.43803, 0.28072, -0.25033, -0.14536, 0.0215, -0.08702
20	1.03545	0.86281, 0.45994, 0.48878, -0.32787, -0.25329, 0.29811, -0.2039, 0.01034	1.13221, -0.72677, 0.62432, 0.16134, -0.20135, -0.14383, -0.23787, 0.06405
21	1.03455	0.88834, 0.49837, 0.38691, -0.38945, -0.28447, 0.25054, 0.31107, -0.25291	1.1201, -0.67256, 0.45506, 0.18534, -0.22116, -0.17539, 0.20721, 0.12574
22	1.03499	0.85001, 0.54566, 0.40484, -0.39481, -0.26303, 0.19392, 0.26159, -0.26262	1.13131, -0.7394, 0.49309, 0.17757, -0.20576, -0.11705, 0.15214, 0.17794
23	1.03534	0.85197, 0.53211, 0.41166, -0.42436, -0.31667, 0.23063, 0.24482, -0.19896	1.10977, -0.6779, 0.45797, 0.24574, -0.27742, -0.11887, 0.13567, 0.12986
24	1.03477	0.84514, 0.54903, 0.42997, -0.37293, -0.30081, 0.27646, -0.08542, 0.02302	1.1227, -0.75372, 0.52559, 0.22297, -0.26354, -0.11398, -0.13015, 0.00362
25	1.03483	0.8658, 0.58243, 0.25803, -0.37795, -0.25582, 0.38649, -0.26534, 0.05029	1.11181, -0.75235, 0.371, 0.25799, -0.26227, -0.15765, -0.25746, -0.00535
26	1.03467	0.85888, 0.55394, 0.36415, -0.42775, -0.20245, 0.27706, -0.21773, 0.06224	1.11769, -0.74839, 0.46479, 0.29293, -0.22279, -0.08262, -0.24819, -0.00591
27	1.03436	0.86478, 0.5278, 0.41777, -0.40473, -0.25776, 0.25915, 0.24767, -0.20953	1.12348, -0.70627, 0.50431, 0.17544, -0.18192, -0.19054, 0.17509, 0.11446
28	1.03394	0.86308, 0.53287, 0.42654, -0.42302, -0.25014, 0.29894, -0.08177, 0.0007	1.1256, -0.73481, 0.52328, 0.24054, -0.20542, -0.15082, -0.11524, -0.00236
29	1.03456	0.86313, 0.55186, 0.34889, -0.36638, -0.304, 0.36213, -0.15893, 0.03608	1.1167, -0.74128, 0.45205, 0.23333, -0.26878, -0.17088, -0.17039, -0.00989
30	1.03415	0.85818, 0.54628, 0.38759, -0.44335, -0.29154, 0.27179, -0.01381, 0.03202	1.11312, -0.71038, 0.43662, 0.31628, -0.28431, -0.11752, -0.05788, -0.04319
31	1.0341	0.85707, 0.54367, 0.37415, -0.43456, -0.26503, 0.33393, -0.07451, 0.00062	1.13404, -0.72594, 0.47385, 0.27029, -0.23227, -0.17966, -0.08723, -0.01861
32	1.03491	0.85072, 0.55482, 0.38188, -0.40032, -0.2798, 0.30596, -0.15929, 0.06331	1.10328, -0.72304, 0.46223, 0.27734, -0.26758, -0.13156, -0.1789, -0.03912
33	1.03398	0.88051, 0.50341, 0.40888, -0.45168, -0.23285, 0.23687, 0.26848, -0.21947	1.12135, -0.67729, 0.46221, 0.22873, -0.18142, -0.17482, 0.19505, 0.11387
34	1.03437	0.86871, 0.51508, 0.42945, -0.43509, -0.25321, 0.2272, 0.23493, -0.22168	1.11865, -0.69451, 0.489, 0.22189, -0.21026, -0.14509, 0.13319, 0.14005
35	1.03391	0.85057, 0.53343, 0.44509, -0.40952, -0.3385, 0.19825, 0.10004, -0.00846	1.12894, -0.70834, 0.48297, 0.26695, -0.30557, -0.06563, 0.01507, -0.00375
36	1.03451	0.86692, 0.50471, 0.42791, -0.43793, -0.27782, 0.25233, 0.2257, -0.18455	1.12258, -0.68146, 0.48267, 0.23354, -0.22862, -0.15992, 0.13775, 0.11201
37	1.03474	0.85647, 0.51996, 0.43532, -0.40925, -0.30279, 0.23825, 0.216, -0.19066	1.1223, -0.69736, 0.50061, 0.20757, -0.24504, -0.1422, 0.12123, 0.12733
38	1.03474	0.86209, 0.51154, 0.43612, -0.44135, -0.27888, 0.22172, 0.23067, -0.19378	1.11689, -0.68778, 0.47777, 0.24619, -0.24013, -0.12521, 0.13192, 0.11959
39	1.03469	0.864, 0.51097, 0.42096, -0.42708, -0.28147, 0.25407, 0.22187, -0.20024	1.11794, -0.68476, 0.47639, 0.23281, -0.23104, -0.16673, 0.13745, 0.11816
40	1.03426	0.87586, 0.50613, 0.42275, -0.447, -0.24132, 0.22823, 0.26037, -0.19339	1.11444, -0.68726, 0.4671, 0.23682, -0.19155, -0.16238, 0.18424, 0.09582
41	1.03452	0.86807, 0.50413, 0.43113, -0.42692, -0.27906, 0.24109, 0.21854, -0.17893	1.11829, -0.68867, 0.49081, 0.23105, -0.23279, -0.15416, 0.13344, 0.10685
42	1.0341	0.85861, 0.53793, 0.39525, -0.43541, -0.24138, 0.34497, -0.06676, 0.00099	1.12556, -0.72213, 0.49236, 0.26456, -0.19709, -0.19857, -0.08363, -0.02398

Appendix B. Results used from the paper

The paper is: OpenAI, *Subset Sum in Time $O(2^{0.49n})$* , 4 October 2026 (OpenAI result #138). This appendix restates, without proof, everything of the paper that this note uses. It keeps the paper’s numbering, so “Lemma 2.3” in the body means Lemma 2.3 below. We rely on the paper’s proofs of these statements. Where a statement depends on the paper’s constants, we state it in the general form that the paper’s proof gives; the note points out each such use.

B.1 Model, isolation and deficiency (paper, Sections 1–2)

Model (1.1). An instance has positive integers $a_1, \dots, a_n$ ($n \geq 2$) and a target $t \geq 0$; equal integers and the empty subset are allowed. Let $b = \max\{1, \lceil \log(1 + \max\{t, a_1, \dots, a_n\}) \rceil\}$. The machine is a word RAM with word length $w = \lceil 4(n + b + \log(n + 2)) \rceil$. Unit-cost operations act on a constant number of words: reads and writes, indirect addressing, comparisons, addition, subtraction, multiplication, quotient and remainder, Boolean operations and shifts. A fresh uniform random word costs one operation. Randomness is one-way: a random word can be reused only if the program stored it. The program is one fixed program for all n and b , with no advice.

Theorem 1.1 (paper). The statement of our Theorem 1 with 0.49 in place of 0.4877.

Transformation (2.1). Let N_0 be the power of two with $12n \leq N_0 < 24n$. Draw independent uniform $u_i \in \{1, \dots, N_0\}$, and put

$$C = 1 + nN_0, \quad c_i = Ca_i + u_i, \quad t_s = Ct + s \quad (0 \leq s \leq nN_0).$$

These integers have $O(b + \log n)$ bits, and there are $O(n^2)$ targets.

Lemma 2.1 (isolation). For every choice of the u_i : some I has $a(I) = t$ if and only if some s and I have $c(I) = t_s$. If the instance has a solution, then with probability at least 11/12 some t_s has exactly one subset I with $c(I) = t_s$.

Deficiency (2.2). For integer weights and $V \subseteq [n]$: $\Sigma(V) = \{c(I) : I \subseteq V\}$ and $d(V) = |V| - \log |\Sigma(V)|$.

Lemma 2.2. For arbitrary integer weights: $d(\emptyset) = 0$ and $d(V) \geq 0$. For $i \notin V$, $0 \leq d(V \cup \{i\}) - d(V) \leq 1$. For disjoint U, V , $d(U \cup V) \geq d(U) + d(V)$. If V, V' have equal size and differ by the exchange of one index, then $|d(V) - d(V')| \leq 1$.

Lemma 2.3. Let V_q be a uniform q -subset of $[n]$. Then $\text{Var } d(V_q) \leq q$, and $\mathbb{E} d(V_q)$ is nondecreasing in q . The proof exposes an ordered sample $X_1, \dots, X_q$ and shows that the Doob martingale $M_r = \mathbb{E}[d(\{X_1, \dots, X_q\}) \mid X_1, \dots, X_r]$ has increments of absolute value at most 1. The same proof applies to a uniform q -subset of any fixed finite ground set, and to any set function that changes by at most 1 under the exchange of one element. (Our (F3) adds Azuma's inequality to this martingale bound.)

Proposition 2.4 (capped preliminary search). The procedure: choose a uniform $\lfloor n/2 \rfloor$ -subset V by bounded rejection sampling. For each of the two sides V and $[n] \setminus V$, build the sorted list of distinct subset sums by inserting one index at a time: merge the current list with its translate by c_i and remove duplicates. Stop with CONTINUE as soon as a list exceeds the cap B_n . If both lists finish, check for every s whether two entries sum to t_s , by a linear scan, and return YES exactly when this happens. The procedure returns the correct decision or CONTINUE, and it runs in $O(B_n)$ poly($n + b + 2$) time on every execution. The paper uses $B_n = 2^{\lceil 0.489985n \rceil}$ and proves: if $\mathbb{E} d(V) > 0.01004n$, then CONTINUE has probability $O(1/n)$. Our Proposition P uses another cap and threshold, with the same proof (given in Section 4).

Lemma 2.5 (random-prime collisions). There are absolute constants K and L_* with this property. Let $L' \geq L_*$, and let p be uniform among the primes in $[2^{L'}, 2^{L'+1})$. If $D \neq 0$ is an integer with $|D| < 2^B$ and $B \geq 1$, then $\Pr[D \equiv 0 \pmod{p}] \leq KBL'2^{-L'}$.

B.2 The filter functions (paper, Section 3)

Let $\mathbb{T} = \mathbb{R}/2\pi\mathbb{Z}$ with normalized measure. Let f, g be strictly positive continuous probability densities on $\mathbb{T}$, and $\kappa > 0$. Put

$$Q(x) = \int_{\mathbb{T}} f(x+w)g(w)dw, \quad K_x(w) = \frac{f(x+w)g(w)}{Q(x)}, \quad f_* = \frac{f^\kappa}{\int f^\kappa}, \quad g_* = \frac{g^\kappa}{\int g^\kappa}. \quad (3.1)$$

$$D(x) = \int K_x \log K_x, \quad A(x) = \int K_x(w) \log f_*(x+w)dw, \quad E(x) = \int K_x(w) \log g_*(w)dw, \\ u = A + E - D, \quad v = 2D - A - E. \quad (3.2)$$

(This note writes w for the paper's v .) These are relative entropies, so

$$D \geq 0, \quad D - A \geq 0, \quad D - E \geq 0. \quad (3.3)$$

The paper's filter takes trigonometric polynomials $r(x) = \sum_j r_j \cos jx$ and $s(x) = \sum_j s_j \cos jx$, puts $F = 2^r$, $G = 2^s$, $f = F/\int F$ and $g = G/\int G$, and uses fixed constants κ, r_j, s_j . Its Proposition 3.1 proves two moment bounds (3.6) for these constants; this note replaces them by Proposition B.

Finite circle. For an integer $P \geq 2$, identify $j \in \mathbb{Z}/P\mathbb{Z}$ with the angle $2\pi j/P$. Replace every integral above, including all normalizers, by the average $P^{-1} \sum_{j=0}^{P-1}$.

Lemma 3.2 (finite filtering data). There is an integer P_0 such that for $P \geq P_0$ the finite-circle functions satisfy (3.3) and the finite form (3.7) of the moment bounds. The absolute values of $D, A, E, u, v, \log f_*, \log g_*$ and $\log K_x(w)$ are bounded independently of P . For each fixed rational $\rho > 0$, one fixed deterministic procedure approximates each of these quantities to error ρ in time polynomial in $\log P$, without a sum over the P points. The proof uses only that r and s are fixed trigonometric polynomials: the finite functions converge uniformly to the continuous ones, and Riemann sums converge for the outer integrals.

B.3 The checksum (paper, Section 4)

Fix τ and η and put $\Delta = \eta - \tau$, $\Gamma = 2\tau - \eta$, $a_\Gamma = \Delta/\Gamma$ and $\epsilon = 10^{-8}$. Let $M(x)^2 = 2 + 2\cos x$.

Theorem 4.1 (modular checksum estimator). Fix $d > 0$. There is a uniform randomized procedure with the following guarantee for large n . Its input is an integer P with $|\log P - \eta n| \leq 2$, residues $b_1, \dots, b_n$ modulo P , and at most n^d target residues $\rho \in \mathcal{T}$. It draws and keeps uniform phases φ_i , and returns FAIL or estimates $\hat{V}_\rho$ of $V_\rho = \sum_{I: b(I) \equiv \rho} e_P(\sum_{i \in I} \varphi_i)$. With probability $1 - o(1)$, uniformly in P , the residues and $\mathcal{T}$, it does not fail and $|\hat{V}_\rho - V_\rho| \leq 1/8$ for all $\rho \in \mathcal{T}$. Every execution takes $2^{(\tau+20\epsilon)n+O(\sqrt{n})} \text{poly}(n + |\mathcal{T}|)$ operations. The paper proves this for its constants; Theorem B is our version.

The construction and the facts we use are these.

- **Fourier form (4.3)–(4.4).** $V_\rho = P^{-1} \sum_{k=0}^{P-1} J_\rho(k)$ with $J_\rho(k) = e_P(-k\rho) \prod_{i=1}^n (1 + e_P(x_i(k)))$ and $x_i(k) = \varphi_i + kb_i \bmod P$. So $|J_\rho(k)|^2 = \prod_i M(x_i(k))^2$. For fixed k and uniform independent phases, the $x_i(k)$ are independent and uniform.
- **Blocks and patterns (4.5)–(4.6).** Partition $[n]$ into $q = \Theta(\sqrt{n})$ blocks B_j of sizes $d_j = \Theta(\sqrt{n})$. A frequency's pattern records, for each block, rational approximations D_j, A_j, E_j of the block averages of D, A, E at its coordinates $x_i(k)$, each with error at most ϵ , on a fixed grid, rounded so that $D_j \geq 0$, $A_j \leq D_j$, $E_j \leq D_j$ (4.5). The set $\mathcal{P}$ of all such grid tuples has $Q_n = 2^{O(\sqrt{n})}$ elements (4.6). For $\pi \in \mathcal{P}$: $(D_\pi, A_\pi, E_\pi) = \frac{1}{n} \sum_j d_j (D_j, A_j, E_j)$, $U_\pi = A_\pi + E_\pi - D_\pi$ and $W_\pi = 2D_\pi - A_\pi - E_\pi$.
- **Parameters (4.8)–(4.9).** $\alpha_\pi = 1$ if $W_\pi \leq \Gamma$ and $\alpha_\pi = \Gamma/W_\pi$ otherwise; $H_\pi = \tau - \alpha_\pi(D_\pi - A_\pi)$. Then $\Delta \leq H_\pi \leq \tau$ and $\eta - H_\pi + \alpha_\pi(D_\pi - E_\pi) = \Delta + \alpha_\pi W_\pi \leq \tau$.
- **Digits.** $B_\pi = 2^{\lceil (\eta - H_\pi)n \rceil}$. High digits $0 \leq h < \lceil P/B_\pi \rceil$ give vectors $y_i(h) = \varphi_i + hB_\pi b_i$; low digits $0 \leq \ell < B_\pi$ give $z_i(\ell) = -\ell b_i$. The pair (h, ℓ) represents $k = hB_\pi + \ell$, and $y_i(h) - z_i(\ell) = x_i(k)$.
- **Tables (4.7).** Block B_j gets $R_j = 2^{\lceil (\alpha_\pi D_j + 4\epsilon)d_j \rceil}$ independent tables. Each coordinate of a table has label $*$ with probability $1 - \alpha_\pi$, and otherwise a uniform $\theta \in \mathbb{Z}/P\mathbb{Z}$. A label θ gives the high score $\log f_*(y_i - \theta)$ and the low score $\log g_*(z_i - \theta)$; a label $*$ gives 0. Fixed approximations to error ϵ replace the scores. A vector passes a block table if its average score is at least $\alpha_\pi A_j - 3\epsilon$ (high side) or $\alpha_\pi E_j - 3\epsilon$ (low side). A whole table is a tuple of one block table per block; an entry is a digit with a whole table that it passes; a match is a high entry and a low entry with the same whole table. L_π is the number of matches.
- **Lemma 4.2.** (i) Conditional on any phases, with probability $1 - o(1)$ every $k < P$ has a match in its own pattern. (ii) In each pattern, both entry counts have expectation at most $2^{(\tau+8\epsilon)n+O(\sqrt{n})}$, and $\mathbb{E}L_\pi \leq 2^{(\eta-\alpha_\pi U_\pi+12\epsilon)n+O(\sqrt{n})}$ (4.10). (iii) With probability $1 - o(1)$, all frequencies are covered and, in every pattern, both entry counts are at most $2^{(\tau+14\epsilon)n}$ and

$$L_\pi \leq 2^{(\eta-\alpha_\pi U_\pi+18\epsilon)n}. \quad (4.11)$$

Call this event $\mathcal{G}$. The proof of (ii) uses the joint moment $\mathbb{E}_{\varphi, \text{tables}} 2^{S_h + S_\ell} = 1$ (4.14) for the total scores of a digit pair and a table tuple.

- **Multiplicity (4.16).** $\mu_\pi(k)$ is the number of matches of k in pattern π ; it is computed exactly by retesting the block tables. Entry lists are capped at $C_n = 2^{\lceil (\tau+20\epsilon)n \rceil}$ (4.17).
- **Estimation.** If $\alpha_\pi U_\pi \geq \Delta$, the procedure enumerates all matches and adds $J_\rho(k)/(P\mu_\pi(k))$ for each match with $k < P$ and pattern π (4.18); on $\mathcal{G}$ this is exact. Otherwise it averages $N = 2^{\lceil \tau n \rceil}$ independent uniform matches, each with value $L_\pi J_\rho(k)/(P\mu_\pi(k))$ if $k < P$ has pattern π and 0 otherwise (4.19). Given the phases and tables on $\mathcal{G}$, this is unbiased, and its mean square error is at most

$$\frac{1}{N} \frac{L_\pi}{P^2} \sum_{k < P: \text{pattern}(k) = \pi} |J_\rho(k)|^2. \quad (4.20)$$

- **Pattern errors (4.21).** For a frequency of pattern π , with $\bar{u}, \bar{v}$ the coordinate averages of u, v at the $x_i(k)$: $|U_\pi - \bar{u}| \leq 3\epsilon$ and $|W_\pi - \bar{v}| \leq 4\epsilon$.
- **Error bound (4.24)–(4.25).** The paper proves

$$\mathbf{1}_{\mathcal{G}} \mathbf{1}_{\{\text{pattern}(k) = \pi\}} L_\pi |J_\rho(k)|^2 \leq 2^{(\eta-\Delta+O(\epsilon))n} \prod_i M(x_i(k))^2 2^{R(x_i(k))}$$

with its two functions R . It takes the expectation over the phases, and uses its moment bounds to get $\mathbb{E}[1_{\mathcal{G}}|E_{\pi,\rho}|^2] \leq 2^{-0.0007n}$. Then Cauchy–Schwarz over the Q_n patterns, Markov’s inequality and a union bound over $\mathcal{T}$ give total ideal error at most $1/16$ with probability $1 - o(1)$. Our Lemmas B2 and B3 replace this step.

- **Resources (Section 4.5).** Caps bound the work on every execution; the finite-precision arithmetic of Section 6.2 adds error at most $1/16$.

B.4 Implementation and the main proof (paper, Section 6)

- **Bounded draws (6.2).** A uniform draw from $\{0, \dots, q-1\}$ uses rejection on $\lceil \log q \rceil$ random bits, and fails after n^3 rejections (probability at most 2^{-n^3}). A prime in $[2^h, 2^{h+1})$ is drawn by testing at most n^3 uniform candidates by trial division; conditional on success it is uniform among the primes, and failure has probability at most $e^{-c_0 n^2}$. Coupled with ideal samplers, all capped executions differ from ideal ones with probability $o(1)$.
- **Complex sums (6.2).** Each phase is computed to error 2^{-2n^2} with dyadic parts, each weighted term is rounded to n^2 fractional bits, and the sums are exact. With $2^{O(n)}$ terms of size $2^{O(n)}$, the total error is $2^{-n^2+O(n)}$.
- **Decision and composition (6.1, 6.3).** $\hat{B}_s$ is the checksum estimate minus the alias phase sum; the answer is YES if some $|\hat{B}_s| \geq 1/2$. Any failure of a main procedure gives NO. The main procedures draw their randomness even when the preliminary search decides, so a union bound composes the success events.
- **Fallback (6.3).** Below a fixed cutoff n_* , or when the input guard fails, the program runs exact meet-in-the-middle in $2^{n/2} \text{poly}(n+b+2)$ time.
- **Address space (6.3).** If the operation count is $K_0 2^{cn}(n+b+2)^K$ with $c < 1$, then every address fits in a word of w bits for large n .

B.5 The numerical method (paper, Appendix A)

- **Identities (A.2).** With $\mathcal{P}(x) = \int F(x+w)G(w)dw$, $\mathcal{Y}(x) = \int F(x+w)G(w)(r(x+w)+s(w))dw$, $L = \log Q = \log \mathcal{P} - \log(Z_f Z_g)$, $B = \mathcal{Y}/\mathcal{P} - \log(Z_f Z_g)$ and $H = \log Z_{f,\kappa} + \log Z_{g,\kappa} - \kappa \log(Z_f Z_g)$, where $Z_f = \int F$, $Z_{f,\kappa} = \int F^\kappa$ and so on:

$$D = B - L, \quad u = L + (\kappa - 1)B - H, \quad v = (2 - \kappa)B - 2L + H.$$

- **Trapezoid bound (A.6).** If a 2π -periodic h is analytic on a neighborhood of the strip $|\text{Im } z| \leq \sigma$ and $|h| \leq B_0$ there, then the mean of h at N equally spaced nodes differs from $\int h$ by at most $2B_0/(e^{N\sigma} - 1)$. (The paper states it for $N = 64$; the proof is the same for every N .)
- **Strip bounds (A.4).** $\int |f(w+iy)|^2 dw$ is nondecreasing in $|y|$, and equals $Z_f^{-2} \int 2^{2r_y}$, where r_y has coefficients $r_j \cosh(jy)$. For $|y| \leq 2\beta$, $Q(x+iy) = \int f(x+w+iy/2)g(w-iy/2)dw$, and Cauchy–Schwarz gives $|Q-1| \leq \sqrt{(A_f-1)(A_g-1)}$ with $A_f = \int |f(\cdot + i\beta)|^2$ and A_g likewise. The same split gives $|QB| \leq \sqrt{A_f A_g} (\sup_{|\text{Im}| \leq \beta} |r| + \sup_{|\text{Im}| \leq \beta} |s| + |\log(Z_f Z_g)|)$.

Appendix C. The multipliers of the 430 moment certificates

Each row gives a filter v , a region ρ_i (Section 6.1), the set $C = C^{(v,\rho)}$, and the vector $z = z^{(v,\rho)}$. The entries of z follow the order of the rows of G in Section 6.1: first z_0 , then the region rows, then one entry for each $l \in C$ in the listed order.

v	region	C	z
0	ρ_0	9, 13, 17, 19, 22, 26, 33	9.1679, 0.0, 0.6464, 0.9449, 4.8989, 0.0037, 3.2564, 0.7714, 2.5871
0	ρ_1	9, 13, 14, 22, 24, 27	1.0001, 40.0, 0.0, 1.0117, 8.8167, 2.1195, 14.1535, 7.6113, 4.0057

v	region	C	z
0	ρ_2	9, 13, 14, 22, 24, 27	0.8001, 40.0, 0.0, 0.7875, 8.8962, 1.8576, 13.1238, 8.737, 3.4303
0	ρ_3	9, 13, 14, 22, 24, 27	0.6251, 40.0, 0.0, 0.5848, 8.9607, 1.6289, 12.2357, 9.7076, 2.9428
0	ρ_4	9, 13, 14, 22, 24, 27	0.5001, 40.0, 0.0, 0.4364, 9.004, 1.4658, 11.6087, 10.3927, 2.6034
0	ρ_5	5, 9, 13, 14, 22, 24, 27	0.3848, 40.0, 0.0, 0.009, 0.2951, 9.0455, 1.3235, 11.0516, 11.0089, 2.2703
0	ρ_6	5, 9, 13, 14, 22, 24, 27	0.2859, 40.0, 0.0, 0.1118, 0.1531, 9.1172, 1.2862, 10.753, 11.4234, 1.7128
0	ρ_7	5, 9, 13, 14, 22, 24, 27	0.2001, 40.0, 0.0, 0.1996, 0.029, 9.1779, 1.2529, 10.4943, 11.781, 1.2372
0	ρ_8	5, 13, 14, 22, 24, 27	0.1251, 40.0, 0.0, 0.2565, 9.2408, 1.2155, 10.2707, 12.0537, 0.809
0	ρ_9	5, 13, 14, 22, 24, 27	0.0668, 40.0, 0.2946, 9.2921, 1.1838, 10.0974, 12.2539, 0.4746
1	ρ_0	0, 6, 7, 21	9.9712, 0.0, 15.3809, 0.4218, 4.9982, 0.1638
1	ρ_1	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_2	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_3	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_4	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_5	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_6	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_7	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_8	0, 4, 6, 14	1.5686, 40.0, 0.0, 40.0, 1.5232, 8.2982, 0.663
1	ρ_9	0, 4, 6, 14	1.5686, 40.0, 40.0, 1.5232, 8.2982, 0.663
2	ρ_0	0, 1, 7, 21	10.2703, 0.0, 5.9733, 3.8695, 6.9658, 1.6915
2	ρ_1	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_2	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_3	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_4	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_5	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_6	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_7	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_8	0, 1, 7, 21	1.4183, 40.0, 0.0, 23.4401, 29.9709, 6.8452, 0.4075
2	ρ_9	0, 1, 7, 21	1.4183, 40.0, 23.4401, 29.9709, 6.8452, 0.4075
3	ρ_0	5, 16	17.211, 27.1977, 8.3302, 3.0053
3	ρ_1	5, 8, 16, 35	7.7547, 0.0, 0.0, 4.2699, 2.2635, 0.461, 3.4032
3	ρ_2	4, 25	0.8001, 40.0, 0.0, 10.1405, 0.4377
3	ρ_3	4, 25	0.6251, 40.0, 0.0, 9.9415, 0.5728
3	ρ_4	4, 25	0.5001, 40.0, 0.0, 9.8013, 0.6685
3	ρ_5	4, 25	0.3848, 40.0, 0.0, 9.6733, 0.7561
3	ρ_6	4, 25	0.2859, 40.0, 0.0, 9.5646, 0.8307
3	ρ_7	4, 25	0.2001, 40.0, 0.0, 9.4711, 0.8949
3	ρ_8	4, 25	0.1251, 40.0, 0.0, 9.3899, 0.9508
3	ρ_9	4, 25	0.0668, 40.0, 9.3271, 0.9941
4	ρ_0	7, 11, 15, 25	10.6428, 7.303, 4.0335, 1.1143, 2.334, 3.8337
4	ρ_1	11, 15, 30, 35	7.8954, 0.0, 0.0, 1.9351, 3.5653, 4.1533, 0.954
4	ρ_2	3	0.8001, 40.0, 0.0, 11.481
4	ρ_3	3	0.6251, 40.0, 0.0, 11.3729
4	ρ_4	3	0.5001, 40.0, 0.0, 11.2948
4	ρ_5	3	0.3848, 40.0, 0.0, 11.2222
4	ρ_6	3	0.2859, 40.0, 0.0, 11.1595
4	ρ_7	3	0.2001, 40.0, 0.0, 11.1048
4	ρ_8	3	0.1251, 40.0, 0.0, 11.0567
4	ρ_9	3	0.0668, 40.0, 11.0192
5	ρ_0	9, 28, 33, 36, 38	9.9165, 4.6818, 2.575, 2.3437, 1.6297, 3.3531, 0.1018
5	ρ_1	20, 28, 36	7.9993, 0.0, 0.0, 1.4983, 8.7892, 0.5363
5	ρ_2	3, 14, 21, 25	0.8001, 40.0, 0.0, 15.8577, 1.6266, 7.9554, 12.4666
5	ρ_3	3, 14, 21, 25	0.6251, 40.0, 0.0, 15.5041, 2.2225, 6.9464, 12.1888
5	ρ_4	3, 14, 21, 25	0.5001, 40.0, 0.0, 15.2517, 2.6369, 6.2395, 11.9955
5	ρ_5	3, 14, 21, 25	0.3848, 40.0, 0.0, 15.019, 3.011, 5.5974, 11.8208
5	ρ_6	3, 14, 21, 25	0.2859, 40.0, 0.0, 14.8197, 3.3254, 5.055, 11.6738
5	ρ_7	3, 14, 21, 25	0.2001, 40.0, 0.0, 14.6471, 3.5932, 4.5909, 11.5484
5	ρ_8	3, 14, 21, 25	0.1251, 40.0, 0.0, 14.4963, 3.8239, 4.1894, 11.4402
5	ρ_9	3, 14, 21, 25	0.0668, 40.0, 14.379, 4.001, 3.8801, 11.357
6	ρ_0	13, 25, 26, 33	10.3545, 4.961, 1.1834, 7.1305, 0.7517, 4.2545
6	ρ_1	13, 21, 22, 25, 26	8.4537, 0.0, 0.0, 3.9932, 0.6808, 0.8, 6.5429, 2.8242
6	ρ_2	3, 4, 21	0.8001, 40.0, 0.0, 9.0517, 5.9704, 0.0499
6	ρ_3	3, 4	0.6251, 40.0, 0.0, 8.9388, 5.8095
6	ρ_4	3, 4	0.5001, 40.0, 0.0, 8.8533, 5.6888
6	ρ_5	3, 4	0.3848, 40.0, 0.0, 8.7739, 5.5788
6	ρ_6	3, 4	0.2859, 40.0, 0.0, 8.7056, 5.4855
6	ρ_7	3, 4	0.2001, 40.0, 0.0, 8.6461, 5.4055
6	ρ_8	3, 4	0.1251, 40.0, 0.0, 8.5939, 5.336
6	ρ_9	3, 4	0.0668, 40.0, 8.5532, 5.2823
7	ρ_0	17, 21, 33	10.4052, 5.7867, 4.9545, 0.7827, 3.4876
7	ρ_1	12, 17, 19, 23, 36	8.4306, 0.0, 0.0, 3.0378, 4.2803, 1.4702, 5.61, 0.1052
7	ρ_2	4, 5, 20, 21	0.8001, 40.0, 0.0, 19.085, 18.8588, 0.8077, 13.5458
7	ρ_3	4, 5, 20, 21	0.6251, 40.0, 0.0, 18.8324, 18.5875, 1.0624, 12.8465
7	ρ_4	4, 5, 20, 21	0.5001, 40.0, 0.0, 18.6518, 18.3896, 1.2477, 12.3563
7	ρ_5	4, 5, 20, 21, 23	0.3848, 40.0, 0.0, 18.3187, 18.0314, 1.3823, 11.3486, 1.0118
7	ρ_6	4, 5, 20, 21, 23	0.2859, 40.0, 0.0, 17.9285, 17.6139, 1.4748, 10.1356, 2.5156
7	ρ_7	4, 5, 20, 21, 23	0.2001, 40.0, 0.0, 17.5905, 17.251, 1.556, 9.0877, 3.818
7	ρ_8	4, 5, 20, 21, 23	0.1251, 40.0, 0.0, 17.2948, 16.9327, 1.6279, 8.1735, 4.9568
7	ρ_9	4, 5, 20, 21, 23	0.0668, 40.0, 17.0649, 16.6846, 1.6842, 7.4643, 5.842
8	ρ_0	22, 24, 26, 30, 32	8.8616, 1.223, 1.2823, 2.7043, 2.5334, 1.7898, 2.7054
8	ρ_1	22, 24, 26, 30, 32	8.2246, 0.0, 0.0, 0.2761, 4.2858, 2.4287, 0.9726, 2.596
8	ρ_2	3, 4, 6	0.8001, 40.0, 0.0, 14.5934, 9.3125, 10.3444
8	ρ_3	3, 4, 6	0.6251, 40.0, 0.0, 14.2681, 9.0025, 9.8548
8	ρ_4	3, 4, 6	0.5001, 40.0, 0.0, 14.037, 8.7828, 9.5098
8	ρ_5	3, 4, 6	0.3848, 40.0, 0.0, 13.8246, 8.5813, 9.1948
8	ρ_6	3, 4, 6	0.2859, 40.0, 0.0, 13.6434, 8.4097, 8.9274
8	ρ_7	3, 4, 6	0.2001, 40.0, 0.0, 13.4868, 8.2616, 8.6977
8	ρ_8	3, 4, 6	0.1251, 40.0, 0.0, 13.3503, 8.1327, 8.4981
8	ρ_9	3, 4, 6	0.0668, 40.0, 13.2443, 8.0327, 8.3439
9	ρ_0	26, 28, 31, 34, 42	8.5363, 0.0, 0.4313, 4.8268, 1.0636, 3.0003, 16.624

v	region	C	z
9	ρ_1	26, 28, 34, 42	8.1097, 0.766, 0.0, 0.7322, 7.9623, 1.7086, 20.6672
9	ρ_2	3, 4, 5, 8, 25, 29, 31, 42	0.8001, 40.0, 0.0, 0.6502, 5.9345, 40.0, 11.2999, 13.8658, 17.3183, 23.2391, 40.0
9	ρ_3	3, 4, 5, 8, 25, 29, 31, 42	0.6251, 40.0, 0.0, 0.272, 6.0412, 40.0, 11.8999, 13.0328, 16.9991, 21.2834, 40.0
9	ρ_4	3, 4, 5, 8, 25, 29, 31, 42	0.5001, 40.0, 0.0, 0.0059, 6.0947, 40.0, 12.3291, 12.4248, 16.7902, 20.1103, 40.0
9	ρ_5	4, 5, 8, 25, 29, 31, 42	0.3848, 40.0, 0.0, 5.0047, 40.0, 12.8333, 11.337, 17.1158, 30.2913, 40.0
9	ρ_6	4, 5, 8, 25, 29, 31, 42	0.2859, 40.0, 0.0, 4.0465, 40.0, 13.2663, 10.3942, 17.4114, 39.2537, 40.0
9	ρ_7	4, 5, 8, 25, 29, 31, 42	0.2001, 40.0, 0.0, 3.8422, 39.7563, 13.3779, 10.0605, 17.5452, 40.0, 40.0
9	ρ_8	4, 5, 8, 25, 29, 31, 42	0.1251, 40.0, 0.0, 3.6941, 39.3384, 13.4883, 9.6, 17.8184, 40.0, 40.0
9	ρ_9	4, 5, 8, 25, 29, 31, 42	0.0668, 40.0, 3.5789, 39.0141, 13.5734, 9.2426, 18.0318, 40.0, 40.0
10	ρ_0	18, 23, 24, 37	8.745, 0.6735, 1.1431, 2.3284, 3.1588, 4.3415
10	ρ_1	23, 24, 37	8.3077, 0.0, 0.0, 2.427, 4.9925, 3.7514
10	ρ_2	3, 4, 21, 25	0.8001, 40.0, 0.0, 14.079, 5.5689, 3.3718, 13.8544
10	ρ_3	3, 4, 21, 25	0.6251, 40.0, 0.0, 13.9046, 5.3119, 2.5519, 14.1084
10	ρ_4	3, 4, 21, 25	0.5001, 40.0, 0.0, 13.7793, 5.1284, 1.9784, 14.2884
10	ρ_5	3, 4, 21, 25	0.3848, 40.0, 0.0, 13.6629, 4.9591, 1.4579, 14.4535
10	ρ_6	3, 4, 21, 25	0.2859, 40.0, 0.0, 13.5628, 4.8141, 1.0185, 14.5941
10	ρ_7	3, 4, 21, 25	0.2001, 40.0, 0.0, 13.4758, 4.6886, 0.6426, 14.7153
10	ρ_8	3, 4, 21, 25	0.1251, 40.0, 0.0, 13.3994, 4.5788, 0.3175, 14.8208
10	ρ_9	3, 4, 21, 25	0.0668, 40.0, 13.3398, 4.4935, 0.067, 14.9025
11	ρ_0	23, 26, 30, 32	8.5281, 0.0, 5.5848, 2.1883, 0.8096, 3.2527
11	ρ_1	16, 23, 26, 30, 32	8.3201, 0.0, 0.0, 0.2064, 4.8721, 1.843, 0.6515, 4.3175
11	ρ_2	3, 4, 6, 21, 25	0.8001, 40.0, 0.0, 13.7106, 20.5765, 8.9862, 17.4401, 13.3446
11	ρ_3	3, 4, 6, 21, 25	0.6251, 40.0, 0.0, 13.5737, 19.9583, 8.6882, 16.4445, 13.4554
11	ρ_4	3, 4, 6, 21, 25	0.5001, 40.0, 0.0, 13.4747, 19.522, 8.4779, 15.7457, 13.5344
11	ρ_5	3, 4, 6, 21, 25	0.3848, 40.0, 0.0, 13.3825, 19.1233, 8.2857, 15.1098, 13.6069
11	ρ_6	3, 4, 6, 21, 25	0.2859, 40.0, 0.0, 13.3027, 18.7845, 8.1226, 14.5716, 13.6688
11	ρ_7	3, 4, 6, 21, 25	0.2001, 40.0, 0.0, 13.2331, 18.4932, 7.9823, 14.1105, 13.7222
11	ρ_8	3, 4, 6, 21, 25	0.1251, 40.0, 0.0, 13.1718, 18.24, 7.8605, 13.7109, 13.7687
11	ρ_9	3, 4, 6, 21, 25	0.0668, 40.0, 13.1239, 18.0441, 7.7663, 13.4026, 13.8047
12	ρ_0	23, 31, 35, 36, 37	8.5784, 0.0, 0.5944, 1.0368, 6.3616, 4.0629, 2.5742
12	ρ_1	35, 36, 37	8.0772, 0.9811, 0.0, 10.662, 4.5469, 2.7765
12	ρ_2	3, 4, 5, 10, 23	0.8001, 40.0, 0.0, 4.63, 15.0086, 19.4035, 10.5572, 27.1491
12	ρ_3	3, 4, 5, 10, 23	0.6251, 40.0, 0.0, 4.5826, 14.7861, 19.132, 10.4615, 26.2338
12	ρ_4	3, 4, 5, 10, 23	0.5001, 40.0, 0.0, 4.549, 14.6255, 18.9362, 10.3957, 25.5935
12	ρ_5	3, 4, 5, 10, 23	0.3848, 40.0, 0.0, 4.518, 14.476, 18.7541, 10.3368, 25.0123
12	ρ_6	3, 4, 5, 10, 23	0.2859, 40.0, 0.0, 4.4916, 14.3469, 18.5971, 10.2877, 24.5216
12	ρ_7	3, 4, 5, 10, 23, 35	0.2001, 40.0, 0.0, 4.4657, 14.2224, 18.4494, 10.2417, 24.074, 0.1565
12	ρ_8	3, 4, 5, 10, 23, 35	0.1251, 40.0, 0.0, 4.4272, 14.0499, 18.2626, 10.1784, 23.5391, 1.1156
12	ρ_9	3, 4, 5, 10, 23, 35	0.0668, 40.0, 4.3973, 13.9159, 18.1175, 10.1294, 23.125, 1.8588
13	ρ_0	24, 26, 27, 32, 37	8.4877, 0.0, 3.0138, 4.3261, 3.5085, 0.4597, 0.5967
13	ρ_1	24, 26, 27, 37	7.7034, 1.7489, 0.0, 4.0823, 3.8813, 0.7355, 2.1616
13	ρ_2	3, 6, 10, 20, 22	0.8001, 40.0, 0.0, 16.3858, 38.793, 14.471, 2.8242, 14.1504
13	ρ_3	3, 6, 10, 20, 22	0.6251, 40.0, 0.0, 15.9878, 37.2826, 14.3161, 2.2365, 13.5223
13	ρ_4	3, 6, 10, 20, 22	0.5001, 40.0, 0.0, 15.7028, 36.2036, 14.2002, 1.8202, 13.0856
13	ρ_5	3, 6, 10, 20, 22	0.3848, 40.0, 0.0, 15.4393, 35.2077, 14.0896, 1.4385, 12.6911
13	ρ_6	3, 6, 10, 20, 22	0.2859, 40.0, 0.0, 15.2132, 34.3544, 13.992, 1.1134, 12.3594
13	ρ_7	3, 6, 10, 20, 22	0.2001, 40.0, 0.0, 15.017, 33.615, 13.9056, 0.833, 12.0765
13	ρ_8	3, 6, 10, 20, 22	0.1251, 40.0, 0.0, 14.8452, 32.9684, 13.8285, 0.5888, 11.8325
13	ρ_9	3, 6, 10, 20, 22	0.0668, 40.0, 14.7115, 32.4657, 13.7676, 0.3996, 11.6448
14	ρ_0	17, 18, 21, 29, 37	8.4737, 0.0, 1.4952, 4.6461, 1.039, 6.6959, 0.0326
14	ρ_1	18, 21, 29, 39	8.152, 0.61, 0.0, 4.6508, 0.2222, 7.2421, 1.7712
14	ρ_2	3, 4, 25	0.8001, 40.0, 0.0, 9.8357, 4.3953, 23.0982
14	ρ_3	3, 4, 25	0.6251, 40.0, 0.0, 9.7559, 4.1763, 22.5913
14	ρ_4	3, 4, 25	0.5001, 40.0, 0.0, 9.697, 4.0218, 22.2415
14	ρ_5	3, 4, 25	0.3848, 40.0, 0.0, 9.6414, 3.8807, 21.9274
14	ρ_6	3, 4, 25	0.2859, 40.0, 0.0, 9.5928, 3.7607, 21.6648
14	ρ_7	3, 4, 25	0.2001, 40.0, 0.0, 9.55, 3.6576, 21.442
14	ρ_8	3, 4, 25	0.1251, 40.0, 0.0, 9.512, 3.568, 21.2507
14	ρ_9	3, 4, 25	0.0668, 40.0, 9.4822, 3.4987, 21.1041
15	ρ_0	18, 24, 32, 35, 37	8.49, 0.0, 0.8847, 0.6571, 6.9361, 0.4502, 2.1524
15	ρ_1	23, 24, 32, 35	7.71, 1.6389, 0.0, 0.6785, 4.5247, 6.1184, 0.3694
15	ρ_2	3, 4, 8, 10, 14, 25	0.8001, 40.0, 0.0, 14.9333, 16.7966, 23.4241, 20.443, 40.0, 17.4896
15	ρ_3	3, 4, 8, 10, 14, 25	0.6251, 40.0, 0.0, 14.7048, 16.4281, 23.7546, 20.0366, 40.0, 16.5246
15	ρ_4	3, 4, 8, 10, 14, 25	0.5001, 40.0, 0.0, 14.5413, 16.1672, 23.9894, 19.7531, 40.0, 15.8435
15	ρ_5	3, 4, 8, 10, 14, 25	0.3848, 40.0, 0.0, 14.3902, 15.9282, 24.2054, 19.4962, 40.0, 15.221
15	ρ_6	3, 4, 8, 10, 14, 25	0.2859, 40.0, 0.0, 14.2606, 15.7245, 24.3898, 19.2797, 40.0, 14.6919
15	ρ_7	3, 4, 8, 10, 14, 25	0.2001, 40.0, 0.0, 14.1465, 15.5449, 24.5337, 19.0874, 39.9721, 14.2428
15	ρ_8	3, 4, 8, 10, 14, 25	0.1251, 40.0, 0.0, 13.9921, 15.2414, 24.1106, 18.6607, 38.9561, 14.0657
15	ρ_9	3, 4, 8, 10, 14, 25	0.0668, 40.0, 13.8719, 15.0054, 23.7797, 18.3293, 38.1634, 13.9299
16	ρ_0	24, 28, 32, 38	8.4893, 0.0, 0.9621, 0.4743, 4.7923, 4.436
16	ρ_1	24, 28, 32, 38	7.9258, 0.9111, 0.0, 2.1236, 1.0062, 4.188, 2.6765
16	ρ_2	3, 4, 10, 11, 13, 20, 22	0.8001, 40.0, 0.0, 16.0979, 19.1761, 1.8956, 38.375, 36.5682, 25.7049, 21.2476
16	ρ_3	3, 4, 10, 11, 13, 20, 22	0.6251, 40.0, 0.0, 15.6679, 19.0948, 1.3008, 35.7174, 36.0578, 25.2266, 20.3188
16	ρ_4	3, 4, 10, 11, 13, 20, 22	0.5001, 40.0, 0.0, 15.3611, 19.038, 0.8753, 33.8248, 35.6936, 24.8898, 19.6629
16	ρ_5	3, 4, 10, 11, 13, 20, 22	0.3848, 40.0, 0.0, 15.0782, 18.9865, 0.4823, 32.0821, 35.3575, 24.5823, 19.0628
16	ρ_6	3, 4, 10, 11, 13, 20, 22	0.2859, 40.0, 0.0, 14.8359, 18.9431, 0.1452, 30.5915, 35.0697, 24.3213, 18.5524
16	ρ_7	3, 4, 11, 13, 20, 22	0.2001, 40.0, 0.0, 14.6603, 18.8364, 29.5641, 34.8092, 23.9942, 18.142
16	ρ_8	3, 4, 11, 13, 20, 22, 25	0.1251, 40.0, 0.0, 14.5383, 18.6622, 28.8076, 34.4859, 23.5841, 17.7888, 0.0727
16	ρ_9	3, 4, 11, 13, 20, 22, 25	0.0668, 40.0, 14.4499, 18.4561, 27.9295, 33.9437, 23.1421, 17.4429, 0.3763
17	ρ_0	18, 23, 29, 31, 36, 39	8.623, 0.0, 1.4471, 1.1251, 0.7058, 4.1178, 0.1343, 4.9324

v	region	C	z
17	ρ_1	18, 23, 29, 31, 36, 39	8.1251, 1.1251, 0.0, 3.2212, 0.9476, 0.9886, 5.1325, 0.6111, 4.5907
17	ρ_2	4, 5, 14, 21, 36, 42	0.8001, 40.0, 0.0, 10.8709, 14.214, 30.2561, 1.5133, 40.0, 19.8607
17	ρ_3	4, 5, 14, 21, 36, 42	0.6251, 40.0, 0.0, 10.5898, 13.8601, 30.0739, 0.7062, 40.0, 20.0283
17	ρ_4	4, 5, 14, 21, 36, 42	0.5001, 40.0, 0.0, 10.3885, 13.6062, 29.9434, 0.1399, 40.0, 20.1491
17	ρ_5	4, 5, 14, 36, 42	0.3848, 40.0, 0.0, 10.2271, 13.3872, 29.6978, 39.6571, 20.3131
17	ρ_6	4, 5, 14, 36, 42	0.2859, 40.0, 0.0, 10.125, 13.2934, 29.4202, 38.897, 20.5002
17	ρ_7	4, 5, 14, 36, 42	0.2001, 40.0, 0.0, 10.0356, 13.2102, 29.1815, 38.2479, 20.6613
17	ρ_8	4, 5, 14, 36, 42	0.1251, 40.0, 0.0, 9.9568, 13.136, 28.9741, 37.6872, 20.8015
17	ρ_9	4, 5, 14, 36, 42	0.0668, 40.0, 9.8951, 13.0774, 28.8136, 37.2557, 20.9099
18	ρ_0	28, 31, 32, 37	8.528, 0.0, 0.5692, 1.4029, 4.1416, 4.7187
18	ρ_1	24, 32, 35, 37, 42	7.9285, 1.2778, 0.0, 2.637, 5.0418, 0.0354, 4.0037, 0.7809
18	ρ_2	4, 5, 10, 14, 15, 20, 23, 29	0.8001, 40.0, 0.0, 11.2099, 36.9048, 20.9929, 22.9765, 40.0, 1.0497, 24.7907, 27.6067
18	ρ_3	4, 5, 10, 14, 15, 20, 23, 29	0.6251, 40.0, 0.0, 10.8137, 36.2402, 20.6667, 21.7313, 40.0, 0.7695, 23.5879, 27.5377
18	ρ_4	4, 5, 10, 14, 15, 20, 23, 29	0.5001, 40.0, 0.0, 10.5325, 35.763, 20.4315, 20.8478, 40.0, 0.5795, 22.7348, 27.4884
18	ρ_5	4, 5, 10, 14, 15, 20, 23, 29	0.3848, 40.0, 0.0, 10.2741, 35.3209, 20.213, 20.0367, 40.0, 0.4111, 21.9517, 27.4429
18	ρ_6	4, 5, 10, 14, 15, 20, 23, 29	0.2859, 40.0, 0.0, 10.0536, 34.9405, 20.0245, 19.3447, 40.0, 0.2724, 21.2838, 27.4037
18	ρ_7	4, 5, 10, 14, 15, 20, 23, 29	0.2001, 40.0, 0.0, 9.8632, 34.61, 19.8605, 18.7474, 40.0, 0.156, 20.7074, 27.3699
18	ρ_8	4, 5, 10, 14, 15, 20, 23, 29	0.1251, 40.0, 0.0, 9.6969, 34.3203, 19.7166, 18.2264, 40.0, 0.0564, 20.2052, 27.3403
18	ρ_9	4, 5, 10, 14, 15, 23, 29	0.0668, 40.0, 9.571, 34.0869, 19.596, 17.8207, 40.0, 19.8049, 27.3192
19	ρ_0	23, 30, 31, 32, 37, 38	8.6098, 0.0, 0.3903, 0.2715, 4.5217, 3.0355, 1.3507, 3.3631
19	ρ_1	18, 23, 30, 31, 32, 37	6.9298, 6.5593, 0.0, 1.4487, 0.1027, 19.0842, 11.2171, 2.145, 10.5777
19	ρ_2	4, 8, 9, 15, 22, 23, 30, 31, 35, 42	0.8001, 40.0, 0.0, 8.0702, 2.7437, 22.851, 0.4658, 0.7696, 40.0, 40.0, 19.8155, 18.4227, 40.0
19	ρ_3	4, 8, 9, 15, 23, 30, 31, 35, 42	0.6251, 40.0, 0.0, 7.7174, 3.1069, 23.1329, 1.1565, 39.9798, 40.0, 16.4951, 17.8332, 40.0
19	ρ_4	4, 8, 9, 15, 23, 30, 31, 35, 42	0.5001, 40.0, 0.0, 7.4234, 3.1837, 21.9692, 1.4009, 38.9229, 40.0, 16.3843, 18.1183, 40.0
19	ρ_5	4, 8, 9, 15, 23, 30, 31, 35, 42	0.3848, 40.0, 0.0, 7.1497, 3.2541, 20.871, 1.6248, 37.9523, 40.0, 16.3157, 18.3825, 40.0
19	ρ_6	4, 8, 9, 15, 23, 30, 31, 35, 42	0.2859, 40.0, 0.0, 6.9135, 3.3143, 19.9116, 1.8153, 37.1242, 40.0, 16.2823, 18.6097, 40.0
19	ρ_7	4, 8, 9, 15, 23, 30, 31, 35, 42	0.2001, 40.0, 0.0, 6.7076, 3.3662, 19.0673, 1.9796, 36.4093, 40.0, 16.2714, 18.8071, 40.0
19	ρ_8	4, 8, 9, 15, 23, 30, 31, 35, 42	0.1251, 40.0, 0.0, 6.5266, 3.4116, 18.3189, 2.1225, 35.7859, 40.0, 16.2756, 18.9801, 40.0
19	ρ_9	4, 8, 9, 15, 23, 30, 31, 35, 42	0.0668, 40.0, 6.385, 3.4467, 17.7294, 2.2332, 35.3022, 40.0, 16.2892, 19.1154, 40.0
20	ρ_0	27, 28, 32, 37, 41	8.3632, 0.0, 0.7483, 4.6966, 0.1833, 3.4444, 1.3918
20	ρ_1	24, 26, 28, 32, 37, 38	7.917, 0.8422, 0.0, 0.5183, 0.0243, 3.7384, 1.734, 3.0335, 0.768
20	ρ_2	3, 10, 21, 25	0.8001, 40.0, 0.0, 16.4125, 3.418, 32.0545, 4.3774
20	ρ_3	3, 10, 21, 25	0.6251, 40.0, 0.0, 16.0301, 3.5656, 30.4958, 4.6758
20	ρ_4	3, 10, 21, 25	0.5001, 40.0, 0.0, 15.757, 3.6674, 29.394, 4.8867
20	ρ_5	3, 10, 21, 25	0.3848, 40.0, 0.0, 15.505, 3.7589, 28.3856, 5.0798
20	ρ_6	3, 10, 21, 25	0.2859, 40.0, 0.0, 15.2892, 3.8354, 27.5277, 5.2439
20	ρ_7	3, 10, 21, 25	0.2001, 40.0, 0.0, 15.1022, 3.9003, 26.7892, 5.3851
20	ρ_8	3, 10, 21, 25	0.1251, 40.0, 0.0, 14.9386, 3.9561, 26.1466, 5.508
20	ρ_9	3, 10, 21, 25	0.0668, 40.0, 14.8115, 3.9988, 25.6493, 5.603
21	ρ_0	27, 32, 34, 39, 40	8.5072, 0.0, 0.8849, 3.6145, 0.4717, 5.8188, 0.1611
21	ρ_1	23, 27, 32, 39, 40	8.2948, 0.0582, 0.0, 0.9113, 2.5345, 3.6052, 4.7568, 0.1153
21	ρ_2	3, 4, 25	0.8001, 40.0, 0.0, 11.7496, 3.6971, 17.5473
21	ρ_3	3, 4, 25	0.6251, 40.0, 0.0, 11.6497, 3.4713, 18.0065
21	ρ_4	3, 4, 25	0.5001, 40.0, 0.0, 11.5761, 3.3119, 18.3176
21	ρ_5	3, 4, 25	0.3848, 40.0, 0.0, 11.5066, 3.1662, 18.5926
21	ρ_6	3, 4, 25	0.2859, 40.0, 0.0, 11.4458, 3.0424, 18.8193
21	ρ_7	3, 4, 25	0.2001, 40.0, 0.0, 11.3924, 2.936, 19.0093
21	ρ_8	3, 4, 25	0.1251, 40.0, 0.0, 11.345, 2.8435, 19.1707
21	ρ_9	3, 4, 25	0.0668, 40.0, 11.3077, 2.772, 19.2932
22	ρ_0	28, 32, 34, 37	8.457, 0.0, 0.646, 5.0085, 3.1523, 1.22
22	ρ_1	24, 28, 32, 34	7.6971, 1.4924, 0.0, 2.8168, 0.8236, 3.8957, 1.7391
22	ρ_2	3, 4, 6, 25	0.8001, 40.0, 0.0, 12.6692, 4.1182, 0.0957, 19.1734
22	ρ_3	3, 4, 6, 25	0.6251, 40.0, 0.0, 12.5188, 4.1095, 0.6053, 18.2366
22	ρ_4	3, 4, 6, 25	0.5001, 40.0, 0.0, 12.4097, 4.0999, 0.9548, 17.5833
22	ρ_5	3, 4, 6, 25	0.3848, 40.0, 0.0, 12.3078, 4.0887, 1.2672, 16.9919
22	ρ_6	3, 4, 6, 25	0.2859, 40.0, 0.0, 12.2196, 4.0775, 1.5273, 16.4934
22	ρ_7	3, 4, 6, 25	0.2001, 40.0, 0.0, 12.1426, 4.0666, 1.7472, 16.0677
22	ρ_8	3, 4, 6, 25	0.1251, 40.0, 0.0, 12.0747, 4.0562, 1.9355, 15.6999
22	ρ_9	3, 4, 6, 25	0.0668, 40.0, 12.0217, 4.0475, 2.0794, 15.4168
23	ρ_0	32, 37, 38, 39	8.4654, 0.0, 5.1563, 1.6763, 3.4869, 1.464
23	ρ_1	32, 37, 38	7.9483, 0.8114, 0.0, 6.7412, 4.2362, 2.9363
23	ρ_2	3, 4, 5, 21, 22, 25	0.8001, 40.0, 0.0, 14.3944, 16.8813, 6.4856, 40.0, 17.3881, 14.9366
23	ρ_3	3, 4, 5, 21, 22, 25	0.6251, 40.0, 0.0, 13.8561, 17.2811, 7.6465, 40.0, 18.562, 14.3323
23	ρ_4	3, 4, 5, 21, 22, 25	0.5001, 40.0, 0.0, 13.481, 17.5523, 8.4424, 40.0, 19.3795, 13.9093
23	ρ_5	3, 4, 5, 21, 22, 25	0.3848, 40.0, 0.0, 13.1416, 17.7923, 9.1532, 40.0, 20.119, 13.525
23	ρ_6	3, 4, 5, 21, 22, 25	0.2859, 40.0, 0.0, 12.8556, 17.9903, 9.7447, 40.0, 20.7414, 13.2003
23	ρ_7	3, 4, 5, 20, 21, 22, 25	0.2001, 40.0, 0.0, 12.6169, 18.1677, 10.1784, 0.1387, 40.0, 21.231, 12.9555
23	ρ_8	3, 4, 5, 20, 21, 22, 25	0.1251, 40.0, 0.0, 12.4127, 18.3229, 10.5244, 0.3103, 40.0, 21.6383, 12.7562
23	ρ_9	3, 4, 5, 20, 21, 22, 25	0.0668, 40.0, 12.2556, 18.4408, 10.7886, 0.4414, 40.0, 21.9519, 12.6023
24	ρ_0	28, 32, 37	8.4468, 0.0, 2.8607, 4.6434, 2.7779
24	ρ_1	28, 32, 38	7.6081, 1.8327, 0.0, 1.6571, 3.3198, 1.4285
24	ρ_2	3, 5, 8, 10, 15, 22, 25, 29	0.8001, 40.0, 0.0, 15.0424, 18.0017, 21.0568, 24.7704, 33.7546, 10.2306, 9.1279, 40.0

v	region	C	z
24	ρ_3	3, 5, 8, 10, 15, 22, 25, 29	0.6251, 40.0, 0.0, 14.5812, 17.8577, 21.4365, 24.733, 32.2825, 9.2929, 8.4526, 40.0
24	ρ_4	3, 5, 8, 10, 15, 22, 25, 29	0.5001, 40.0, 0.0, 14.2548, 17.7469, 21.6992, 24.7026, 31.2279, 8.633, 7.9728, 40.0
24	ρ_5	3, 5, 8, 10, 15, 22, 25, 29	0.3848, 40.0, 0.0, 13.9557, 17.6389, 21.9355, 24.6719, 30.2522, 8.0311, 7.5316, 40.0
24	ρ_6	3, 4, 5, 8, 10, 15, 22, 25, 29	0.2859, 40.0, 0.0, 12.4219, 1.027, 21.2866, 25.1056, 26.8688, 27.5949, 6.6037, 6.7714, 40.0
24	ρ_7	3, 4, 5, 8, 10, 15, 22, 25, 29	0.2001, 40.0, 0.0, 10.5081, 2.3871, 26.1589, 29.2102, 29.7897, 24.4573, 4.9504, 5.9374, 40.0
24	ρ_8	3, 4, 5, 8, 10, 15, 22, 25, 29	0.1251, 40.0, 0.0, 8.8393, 3.5732, 30.4046, 32.7873, 32.3352, 21.7171, 3.5097, 5.2094, 40.0
24	ρ_9	3, 4, 5, 8, 10, 15, 22, 25, 29	0.0668, 40.0, 7.5466, 4.4918, 33.6913, 35.5569, 34.3058, 19.5923, 2.3944, 4.6448, 40.0
25	ρ_0	23, 26, 29, 33	8.4155, 0.0, 0.7535, 3.9191, 7.1124, 1.3003
25	ρ_1	21, 22, 26, 29, 32	8.0401, 0.6077, 0.0, 0.8529, 0.0811, 3.249, 4.5745, 4.2571
25	ρ_2	3, 4, 6, 21	0.8001, 40.0, 0.0, 9.9933, 6.8874, 1.2555, 0.3972
25	ρ_3	3, 4, 6, 21	0.6251, 40.0, 0.0, 9.9045, 6.813, 1.6848, 0.2793
25	ρ_4	3, 4, 6, 21	0.5001, 40.0, 0.0, 9.8391, 6.7588, 1.9831, 0.2078
25	ρ_5	3, 4, 6, 21	0.3848, 40.0, 0.0, 9.7773, 6.708, 2.2524, 0.1507
25	ρ_6	3, 4, 6, 21	0.2859, 40.0, 0.0, 9.7233, 6.6639, 2.4788, 0.1084
25	ρ_7	3, 4, 6, 21	0.2001, 40.0, 0.0, 9.6758, 6.6254, 2.6716, 0.0764
25	ρ_8	3, 4, 6, 21	0.1251, 40.0, 0.0, 9.6337, 6.5914, 2.8379, 0.0518
25	ρ_9	3, 4, 6, 21	0.0668, 40.0, 9.6006, 6.5648, 2.9657, 0.0349
26	ρ_0	28, 32, 34, 38	8.4664, 0.0, 1.3666, 5.3536, 1.5467, 1.4069
26	ρ_1	24, 28, 32, 38	7.9562, 0.9097, 0.0, 0.9326, 1.5826, 4.5767, 2.2312
26	ρ_2	3, 4, 6, 11, 22, 25	0.8001, 40.0, 0.0, 16.3611, 1.7703, 5.1, 12.2398, 14.558, 36.6664
26	ρ_3	3, 4, 6, 11, 22, 25	0.6251, 40.0, 0.0, 15.9516, 2.0101, 5.2357, 11.2086, 14.2607, 34.8126
26	ρ_4	3, 4, 6, 11, 22, 25	0.5001, 40.0, 0.0, 15.6603, 2.1785, 5.3306, 10.4795, 14.0565, 33.5006
26	ρ_5	3, 4, 6, 11, 22, 25	0.3848, 40.0, 0.0, 15.3923, 2.3318, 5.4167, 9.8119, 13.8738, 32.2984
26	ρ_6	3, 4, 6, 11, 22, 25	0.2859, 40.0, 0.0, 15.1634, 2.4616, 5.4894, 9.2439, 13.7216, 31.2747
26	ρ_7	3, 4, 6, 11, 22, 25	0.2001, 40.0, 0.0, 14.9655, 2.5728, 5.5516, 8.7546, 13.5929, 30.3925
26	ρ_8	3, 4, 6, 11, 22, 25	0.1251, 40.0, 0.0, 14.7927, 2.6693, 5.6054, 8.3288, 13.4827, 29.6244
26	ρ_9	3, 4, 6, 11, 22, 25	0.0668, 40.0, 14.6587, 2.7437, 5.6468, 7.9992, 13.3985, 29.0294
27	ρ_0	28, 32, 34, 37, 39	8.459, 0.0, 1.5991, 3.9299, 2.0208, 0.9315, 1.5485
27	ρ_1	24, 28, 32, 34, 37	8.0031, 0.6933, 0.0, 0.4752, 2.2035, 4.2074, 0.2738, 1.8622
27	ρ_2	3, 5, 6, 10, 14, 20, 21, 22, 25	0.8001, 40.0, 0.0, 11.83, 11.3211, 2.0953, 2.1662, 5.5224, 25.5822, 40.0, 40.0, 40.0
27	ρ_3	3, 5, 6, 10, 14, 20, 21, 22, 25	0.6251, 40.0, 0.0, 10.5936, 16.044, 2.3876, 5.6925, 3.7508, 20.1871, 40.0, 40.0, 40.0
27	ρ_4	3, 5, 6, 10, 14, 20, 21, 22, 25	0.5001, 40.0, 0.0, 9.7151, 19.3875, 2.5943, 8.1922, 2.4891, 16.3639, 40.0, 40.0, 40.0
27	ρ_5	3, 5, 6, 10, 14, 20, 21, 22, 25	0.3848, 40.0, 0.0, 8.9077, 22.4517, 2.7836, 10.4855, 1.3274, 12.8573, 40.0, 40.0, 40.0
27	ρ_6	3, 5, 6, 10, 14, 20, 21, 22, 25	0.2859, 40.0, 0.0, 8.2182, 25.0615, 2.9447, 12.4404, 0.3339, 9.8687, 40.0, 40.0, 40.0
27	ρ_7	3, 5, 6, 10, 20, 21, 22, 25	0.2001, 40.0, 0.0, 7.8204, 26.5255, 3.0974, 13.4489, 8.5447, 40.0, 40.0, 40.0
27	ρ_8	3, 5, 6, 10, 20, 21, 22, 25	0.1251, 40.0, 0.0, 7.5837, 27.3611, 3.2381, 13.9492, 8.0907, 40.0, 40.0, 40.0
27	ρ_9	3, 5, 6, 10, 20, 21, 22, 25	0.0668, 40.0, 7.4006, 28.005, 3.347, 14.3348, 7.7427, 40.0, 40.0, 40.0
28	ρ_0	32, 34, 37, 38, 39	8.46, 0.0, 3.7252, 0.0692, 0.3588, 1.3805, 1.2413
28	ρ_1	24, 32, 38	7.806, 1.319, 0.0, 1.8426, 3.4987, 1.9034
28	ρ_2	5, 8, 15, 20, 23, 24, 30, 32, 42	0.8001, 40.0, 0.0, 40.0, 19.5051, 2.2287, 10.5766, 19.9014, 40.0, 30.6988, 40.0, 21.7588
28	ρ_3	5, 8, 15, 20, 23, 24, 30, 32, 42	0.6251, 40.0, 0.0, 40.0, 18.4595, 2.6703, 7.7488, 19.9469, 40.0, 28.0138, 40.0, 18.0201
28	ρ_4	5, 8, 15, 20, 23, 24, 30, 32, 42	0.5001, 40.0, 0.0, 40.0, 17.7166, 2.9829, 5.7341, 19.9889, 40.0, 26.0938, 40.0, 15.3643
28	ρ_5	5, 8, 15, 20, 23, 24, 30, 32, 42	0.3848, 40.0, 0.0, 40.0, 17.0335, 3.2693, 3.8781, 20.0347, 40.0, 24.3196, 40.0, 12.9234
28	ρ_6	5, 8, 15, 20, 23, 24, 30, 32, 42	0.2859, 40.0, 0.0, 40.0, 16.4501, 3.5133, 2.2899, 20.0792, 40.0, 22.7974, 40.0, 10.839
28	ρ_7	5, 8, 15, 20, 23, 24, 30, 32, 42	0.2001, 40.0, 0.0, 40.0, 15.9461, 3.7236, 0.9154, 20.1217, 40.0, 21.4769, 40.0, 9.0389
28	ρ_8	5, 8, 15, 23, 24, 30, 32, 42	0.1251, 40.0, 0.0, 40.0, 15.542, 3.9177, 19.955, 40.0, 20.7281, 40.0, 7.6801
28	ρ_9	5, 8, 15, 23, 24, 29, 30, 32, 42	0.0668, 40.0, 40.0, 15.2272, 3.6259, 18.9957, 40.0, 0.8582, 21.6082, 40.0, 5.8662
29	ρ_0	28, 32, 37, 39	8.4836, 0.0, 1.1816, 5.7239, 0.5264, 2.5149
29	ρ_1	24, 27, 32, 37, 42	7.5979, 2.0507, 0.0, 1.8679, 0.2116, 9.2619, 1.3267, 0.0703
29	ρ_2	3, 4, 14, 21, 22, 25	0.8001, 40.0, 0.0, 14.843, 6.3571, 15.556, 5.5528, 5.7576, 40.0
29	ρ_3	3, 4, 14, 21, 22, 25	0.6251, 40.0, 0.0, 14.7655, 6.0631, 16.1838, 4.5794, 6.2574, 40.0
29	ρ_4	3, 4, 14, 21, 22, 25	0.5001, 40.0, 0.0, 14.7077, 5.8556, 16.6223, 3.8966, 6.6157, 40.0
29	ρ_5	3, 4, 14, 21, 22, 25	0.3848, 40.0, 0.0, 14.6526, 5.6658, 17.0199, 3.2757, 6.9473, 40.0
29	ρ_6	3, 4, 14, 21, 22, 25	0.2859, 40.0, 0.0, 14.6041, 5.5044, 17.3554, 2.7506, 7.2321, 40.0
29	ρ_7	3, 4, 14, 21, 22, 25	0.2001, 40.0, 0.0, 14.5611, 5.3656, 17.6423, 2.3007, 7.4793, 40.0
29	ρ_8	3, 4, 14, 21, 22, 25	0.1251, 40.0, 0.0, 14.5228, 5.2449, 17.8904, 1.9112, 7.6958, 40.0
29	ρ_9	3, 4, 14, 21, 22, 25	0.0668, 40.0, 14.4926, 5.1515, 18.0814, 1.6108, 7.8643, 40.0
30	ρ_0	28, 31, 32, 38	8.4753, 0.0, 1.3556, 1.1162, 4.6796, 3.0421
30	ρ_1	28, 32, 35, 38	7.9935, 0.7745, 0.0, 2.5639, 6.0856, 0.8719, 1.229
30	ρ_2	4, 5, 8, 22, 26, 42	0.8001, 40.0, 0.0, 27.6138, 32.195, 11.8912, 1.8721, 22.3933, 16.7236
30	ρ_3	4, 5, 8, 22, 25, 26, 42	0.6251, 40.0, 0.0, 26.389, 32.4625, 12.2138, 1.5335, 1.6203, 19.6765, 12.0937
30	ρ_4	4, 5, 8, 22, 25, 26	0.5001, 40.0, 0.0, 24.3495, 34.4861, 12.5199, 1.112, 6.4365, 13.4941
30	ρ_5	4, 5, 8, 22, 25, 26	0.3848, 40.0, 0.0, 23.9452, 34.0414, 12.71, 0.951, 6.2881, 13.1063
30	ρ_6	4, 5, 8, 22, 25, 26	0.2859, 40.0, 0.0, 23.5999, 33.6606, 12.8699, 0.8176, 6.1596, 12.7776
30	ρ_7	4, 5, 8, 22, 25, 26	0.2001, 40.0, 0.0, 23.3017, 33.3309, 13.0061, 0.7055, 6.047, 12.4957
30	ρ_8	4, 5, 8, 22, 25, 26	0.1251, 40.0, 0.0, 23.0416, 33.0428, 13.1235, 0.61, 5.9477, 12.251
30	ρ_9	4, 5, 8, 22, 25, 26	0.0668, 40.0, 22.8397, 32.8188, 13.2137, 0.5373, 5.87, 12.062
31	ρ_0	28, 32, 38, 39	8.4669, 0.0, 2.0055, 4.1501, 0.9035, 2.1279
31	ρ_1	23, 28, 32, 37, 38	7.7427, 1.5111, 0.0, 0.2012, 4.3219, 6.8835, 0.1089, 0.8989
31	ρ_2	4, 5, 8, 22, 25, 42	0.8001, 40.0, 0.0, 21.8583, 38.6821, 4.2934, 4.3617, 26.2854, 40.0

v	region	C	z
31	ρ_3	4, 5, 8, 22, 25, 42	0.6251, 40.0, 0.0, 21.3883, 37.7995, 4.6644, 4.1145, 25.4473, 40.0
31	ρ_4	4, 5, 8, 22, 25, 42	0.5001, 40.0, 0.0, 21.054, 37.1712, 4.924, 3.9471, 24.8502, 40.0
31	ρ_5	4, 5, 8, 22, 25, 42	0.3848, 40.0, 0.0, 20.7466, 36.5927, 5.1597, 3.7993, 24.3002, 40.0
31	ρ_6	4, 5, 8, 22, 25, 42	0.2859, 40.0, 0.0, 20.4839, 36.0981, 5.3587, 3.6775, 23.8297, 40.0
31	ρ_7	4, 5, 8, 22, 25, 42	0.2001, 40.0, 0.0, 20.2568, 35.6703, 5.529, 3.5756, 23.4227, 40.0
31	ρ_8	4, 5, 8, 22, 25, 42	0.1251, 40.0, 0.0, 20.0586, 35.2966, 5.6763, 3.4893, 23.0672, 40.0
31	ρ_9	4, 5, 8, 22, 25, 42	0.0668, 40.0, 19.9047, 35.0064, 5.7898, 3.4239, 22.791, 40.0
32	ρ_0	28, 34, 37, 38, 39	8.46, 0.0, 1.685, 0.0692, 0.3588, 1.3805, 1.2413
32	ρ_1	24, 28, 38	7.7141, 1.6056, 0.0, 2.0404, 1.2528, 2.2207
32	ρ_2	3, 4, 5, 20, 22, 25	0.8001, 40.0, 0.0, 9.355, 16.7155, 20.5904, 2.8685, 28.1601, 39.0966
32	ρ_3	3, 4, 5, 20, 22, 24, 25	0.6251, 40.0, 0.0, 9.3693, 15.9842, 19.1977, 2.972, 26.9122, 0.4184, 37.7918
32	ρ_4	3, 4, 5, 20, 22, 24, 25	0.5001, 40.0, 0.0, 9.3444, 15.5059, 18.3538, 2.9442, 26.0789, 1.0339, 36.9474
32	ρ_5	3, 4, 5, 20, 22, 24, 25	0.3848, 40.0, 0.0, 9.3209, 15.0671, 17.5798, 2.919, 25.3174, 1.5965, 36.1709
32	ρ_6	3, 4, 5, 20, 22, 24, 25	0.2859, 40.0, 0.0, 9.3003, 14.6929, 16.9199, 2.8977, 24.6702, 2.0747, 35.5077
32	ρ_7	3, 4, 5, 20, 22, 24, 25	0.2001, 40.0, 0.0, 9.282, 14.37, 16.3509, 2.8793, 24.1137, 2.4861, 34.9345
32	ρ_8	3, 4, 5, 20, 22, 24, 25	0.1251, 40.0, 0.0, 9.2659, 14.0886, 15.8548, 2.8637, 23.6296, 2.8436, 34.4343
32	ρ_9	3, 4, 5, 20, 22, 24, 25	0.0668, 40.0, 9.2533, 13.8703, 15.4701, 2.8517, 23.2551, 3.1203, 34.0462
33	ρ_0	31, 32, 34, 36, 40, 42	8.4858, 0.0, 0.6407, 3.6349, 2.9376, 2.8963, 2.7034, 0.0095
33	ρ_1	27, 32, 34, 39, 40, 42	7.8569, 1.3641, 0.0, 0.1874, 3.0897, 1.8933, 4.1996, 12.6495, 0.4099
33	ρ_2	3, 5, 21, 22, 23, 25	0.8001, 40.0, 0.0, 14.2149, 0.9381, 40.0, 0.3832, 14.2852, 29.2081
33	ρ_3	3, 5, 21, 22, 23, 25	0.6251, 40.0, 0.0, 13.868, 1.501, 40.0, 0.8903, 14.057, 28.9425
33	ρ_4	3, 5, 21, 22, 23, 25	0.5001, 40.0, 0.0, 13.6242, 1.8876, 40.0, 1.2549, 13.8824, 28.7492
33	ρ_5	3, 5, 21, 22, 23, 25	0.3848, 40.0, 0.0, 13.4021, 2.2334, 40.0, 1.5929, 13.7132, 28.5683
33	ρ_6	3, 5, 21, 22, 23, 25	0.2859, 40.0, 0.0, 13.2139, 2.5218, 40.0, 1.8836, 13.5621, 28.4114
33	ρ_7	3, 5, 21, 22, 23, 25	0.2001, 40.0, 0.0, 13.0524, 2.7657, 40.0, 2.1362, 13.427, 28.2742
33	ρ_8	3, 5, 21, 22, 23, 25	0.1251, 40.0, 0.0, 12.9123, 2.9748, 40.0, 2.3576, 13.3057, 28.1531
33	ρ_9	3, 5, 21, 22, 23, 25	0.0668, 40.0, 12.8041, 3.1347, 40.0, 2.53, 13.2093, 28.0584
34	ρ_0	28, 32, 37, 38, 39	8.46, 0.0, 1.685, 3.7252, 0.3588, 1.3805, 1.2413
34	ρ_1	28, 32, 37	8.0578, 0.5422, 0.0, 2.4521, 4.3254, 2.2871
34	ρ_2	3, 5, 21, 22, 23, 25, 27, 39	0.8001, 40.0, 0.0, 13.7208, 5.0245, 19.6336, 40.0, 40.0, 13.1679, 26.5879, 34.2393
34	ρ_3	3, 5, 21, 22, 23, 25, 27, 39	0.6251, 40.0, 0.0, 13.2916, 5.1613, 17.1356, 40.0, 40.0, 12.7892, 27.9057, 34.0223
34	ρ_4	3, 5, 21, 22, 23, 25, 27, 39	0.5001, 40.0, 0.0, 12.9866, 5.256, 15.3696, 40.0, 40.0, 12.5154, 28.849, 33.8607
34	ρ_5	3, 5, 21, 22, 23, 25, 27, 39	0.3848, 40.0, 0.0, 12.7062, 5.3411, 13.753, 40.0, 40.0, 12.2606, 29.7206, 33.7066
34	ρ_6	3, 5, 21, 22, 23, 25, 27, 39	0.2859, 40.0, 0.0, 12.4667, 5.4125, 12.3776, 40.0, 40.0, 12.0406, 30.4681, 33.5706
34	ρ_7	3, 5, 21, 22, 23, 25, 27, 39	0.2001, 40.0, 0.0, 12.2598, 5.4733, 11.1932, 40.0, 40.0, 11.8489, 31.116, 33.4496
34	ρ_8	3, 5, 21, 22, 23, 25, 27, 39	0.1251, 40.0, 0.0, 12.0792, 5.5257, 10.1625, 40.0, 40.0, 11.6803, 31.683, 33.342
34	ρ_9	3, 5, 21, 22, 23, 25, 27, 39	0.0668, 40.0, 11.9391, 5.5658, 9.3646, 40.0, 40.0, 11.5487, 32.1242, 33.2564
35	ρ_0	28, 32, 37, 38	8.4785, 0.0, 1.8096, 4.0834, 1.3459, 2.2544
35	ρ_1	24, 28, 32, 38	7.967, 0.8534, 0.0, 1.7215, 1.7706, 3.6903, 2.3521
35	ρ_2	3, 4, 5, 10, 23, 24	0.8001, 40.0, 0.0, 8.2122, 16.0399, 11.1008, 3.2521, 27.8334, 38.2439
35	ρ_3	3, 4, 5, 10, 23, 24	0.6251, 40.0, 0.0, 8.1294, 15.6306, 10.6351, 3.0732, 26.5334, 38.3422
35	ρ_4	3, 4, 5, 10, 23, 24	0.5001, 40.0, 0.0, 8.0704, 15.3392, 10.3038, 2.9468, 25.6154, 38.4109
35	ρ_5	3, 4, 5, 10, 16, 23, 24	0.3848, 40.0, 0.0, 7.997, 15.0939, 10.0609, 2.8402, 0.1118, 24.6759, 38.4192
35	ρ_6	3, 4, 5, 10, 16, 23, 24	0.2859, 40.0, 0.0, 7.8888, 14.9387, 10.0017, 2.7713, 0.4747, 23.6377, 38.2964
35	ρ_7	3, 4, 5, 10, 16, 23, 24	0.2001, 40.0, 0.0, 7.7955, 14.8042, 9.9496, 2.7119, 0.7869, 22.7439, 38.1904
35	ρ_8	3, 4, 5, 10, 16, 23, 24	0.1251, 40.0, 0.0, 7.7141, 14.6864, 9.9035, 2.6603, 1.0585, 21.9662, 38.0979
35	ρ_9	3, 4, 5, 10, 16, 23, 24	0.0668, 40.0, 7.651, 14.5948, 9.8672, 2.6203, 1.2687, 21.3642, 38.0261
36	ρ_0	28, 32, 38, 39, 41	8.4679, 0.0, 1.8976, 3.2031, 2.0689, 3.9792, 0.0146
36	ρ_1	28, 32, 38, 39, 41, 42	7.8724, 1.103, 0.0, 1.0529, 3.8946, 1.3251, 2.0595, 12.8877, 0.7308
36	ρ_2	3, 4, 5, 21, 22, 23, 25, 27, 33, 39	0.8001, 40.0, 0.0, 4.6584, 17.1524, 40.0, 40.0, 2.7476, 40.0, 6.553, 13.5827, 31.1995, 40.0
36	ρ_3	3, 4, 5, 21, 22, 23, 25, 27, 33, 39	0.6251, 40.0, 0.0, 4.4093, 16.1698, 40.0, 40.0, 5.1801, 40.0, 7.8025, 7.6136, 25.4956, 40.0
36	ρ_4	3, 4, 5, 21, 22, 23, 25, 27, 33, 39	0.5001, 40.0, 0.0, 4.2314, 15.4779, 40.0, 40.0, 6.8994, 40.0, 8.6747, 3.4137, 21.4945, 40.0
36	ρ_5	3, 4, 5, 21, 22, 23, 25, 33, 39	0.3848, 40.0, 0.0, 4.069, 14.8953, 40.0, 40.0, 8.3079, 40.0, 9.3834, 18.0789, 40.0
36	ρ_6	3, 4, 5, 21, 22, 23, 25, 33, 39	0.2859, 40.0, 0.0, 3.9424, 14.7409, 40.0, 40.0, 8.3598, 40.0, 9.4125, 16.7566, 40.0
36	ρ_7	3, 4, 5, 21, 22, 23, 25, 33, 39	0.2001, 40.0, 0.0, 3.833, 14.609, 40.0, 40.0, 8.4079, 40.0, 9.4341, 15.6363, 40.0
36	ρ_8	3, 4, 5, 21, 22, 23, 25, 33, 39	0.1251, 40.0, 0.0, 3.7374, 14.4952, 40.0, 40.0, 8.4523, 40.0, 9.4502, 14.6754, 40.0
36	ρ_9	3, 4, 5, 21, 22, 23, 25, 33, 39	0.0668, 40.0, 3.6633, 14.4076, 40.0, 40.0, 8.4883, 40.0, 9.461, 13.9404, 40.0
37	ρ_0	28, 32, 34, 38, 39	8.46, 0.0, 1.685, 3.7252, 0.0692, 1.3805, 1.2413
37	ρ_1	24, 28, 32, 38	7.9204, 0.8944, 0.0, 0.7723, 2.1886, 4.1463, 0.2466
37	ρ_2	3, 4, 5, 10, 21, 22, 23, 27	0.8001, 40.0, 0.0, 6.0626, 23.9406, 39.0678, 15.5292, 40.0, 40.0, 4.3521, 39.3821
37	ρ_3	3, 4, 5, 10, 21, 22, 27	0.6251, 40.0, 0.0, 5.905, 23.9094, 38.7627, 15.2526, 40.0, 40.0, 37.9198
37	ρ_4	3, 4, 5, 10, 21, 22, 27	0.5001, 40.0, 0.0, 5.7412, 23.7059, 38.8547, 15.5102, 40.0, 40.0, 36.6537
37	ρ_5	3, 4, 5, 10, 21, 22, 27	0.3848, 40.0, 0.0, 5.5664, 23.4987, 38.8482, 15.6429, 39.4074, 40.0, 35.909
37	ρ_6	3, 4, 5, 10, 21, 22, 27	0.2859, 40.0, 0.0, 5.384, 23.2971, 38.7325, 15.626, 38.1328, 40.0, 35.8074
37	ρ_7	3, 4, 5, 10, 21, 22, 27	0.2001, 40.0, 0.0, 5.2269, 23.1222, 38.63, 15.6104, 37.0339, 40.0, 35.7196
37	ρ_8	3, 4, 5, 10, 21, 22, 27	0.1251, 40.0, 0.0, 5.0902, 22.9691, 38.5387, 15.5959, 36.0766, 40.0, 35.6428
37	ρ_9	3, 4, 5, 10, 21, 22, 27	0.0668, 40.0, 4.9842, 22.8498, 38.4667, 15.5843, 35.3349, 40.0, 35.5832
38	ρ_0	28, 32, 34, 37, 39	8.46, 0.0, 1.685, 3.7252, 0.0692, 0.3588, 1.2413
38	ρ_1	24, 28, 32, 37	8.0134, 0.6396, 0.0, 0.3063, 2.3275, 4.2612, 1.7726
38	ρ_2	3, 4, 5, 21, 22, 23, 25, 33, 39	0.8001, 40.0, 0.0, 12.9905, 14.3216, 18.8796, 22.9238, 38.1451, 40.0, 4.1588, 24.292, 40.0
38	ρ_3	3, 4, 5, 21, 22, 23, 25, 33, 39	0.6251, 40.0, 0.0, 12.9515, 13.0823, 17.4601, 22.4132, 36.0402, 40.0, 5.4192, 18.5827, 40.0

v	region	C	z
38	ρ_4	3, 4, 5, 21, 22, 23, 25, 33, 39	0.5001, 40.0, 0.0, 12.9215, 12.2062, 16.4581, 22.0462, 34.559, 40.0, 6.306, 14.5508, 40.0
38	ρ_5	3, 4, 5, 21, 22, 23, 25, 33, 39	0.3848, 40.0, 0.0, 12.8923, 11.4041, 15.5416, 21.7058, 33.208, 40.0, 7.115, 10.8627, 40.0
38	ρ_6	3, 4, 5, 21, 22, 23, 25, 33, 39	0.2859, 40.0, 0.0, 12.8663, 10.7215, 14.7625, 21.4129, 32.0618, 40.0, 7.8012, 7.7263, 40.0
38	ρ_7	3, 4, 5, 21, 22, 23, 25, 33, 39	0.2001, 40.0, 0.0, 12.8429, 10.1336, 14.0919, 21.1581, 31.0774, 40.0, 8.3905, 5.0269, 40.0
38	ρ_8	3, 4, 5, 21, 22, 23, 25, 33, 39	0.1251, 40.0, 0.0, 12.8218, 9.6218, 13.5086, 20.9347, 30.2226, 40.0, 8.9023, 2.6783, 40.0
38	ρ_9	3, 4, 5, 21, 22, 23, 25, 33, 39	0.0668, 40.0, 12.8051, 9.2254, 13.0571, 20.7606, 29.5619, 40.0, 9.2978, 0.8606, 40.0
39	ρ_0	28, 32, 34, 37, 38	8.46, 0.0, 1.685, 3.7252, 0.0692, 0.3588, 1.3805
39	ρ_1	28, 32, 34, 37, 38	7.881, 0.9904, 0.0, 2.3541, 5.0639, 0.6916, 3.7929, 0.7231
39	ρ_2	3, 4, 5, 21, 22, 23, 25	0.8001, 40.0, 0.0, 4.4531, 16.6514, 40.0, 40.0, 34.6448, 40.0, 15.4614
39	ρ_3	3, 4, 5, 21, 22, 23, 25	0.6251, 40.0, 0.0, 4.2896, 16.5103, 40.0, 40.0, 34.7025, 40.0, 15.2344
39	ρ_4	3, 4, 5, 21, 22, 23, 25	0.5001, 40.0, 0.0, 4.1719, 16.4111, 40.0, 40.0, 34.7546, 40.0, 15.0698
39	ρ_5	3, 4, 5, 21, 22, 23, 25	0.3848, 40.0, 0.0, 4.0627, 16.3206, 40.0, 40.0, 34.8104, 40.0, 14.9162
39	ρ_6	3, 4, 5, 21, 22, 23, 25	0.2859, 40.0, 0.0, 3.9686, 16.2439, 40.0, 40.0, 34.8637, 40.0, 14.7834
39	ρ_7	3, 4, 5, 21, 22, 23, 25	0.2001, 40.0, 0.0, 3.8868, 16.1781, 40.0, 40.0, 34.9139, 40.0, 14.6675
39	ρ_8	3, 4, 5, 21, 22, 23, 25	0.1251, 40.0, 0.0, 3.815, 16.1209, 40.0, 40.0, 34.9607, 40.0, 14.5655
39	ρ_9	3, 4, 5, 21, 22, 23, 25	0.0668, 40.0, 3.759, 16.0767, 40.0, 40.0, 34.9988, 40.0, 14.4857
40	ρ_0	28, 32, 34, 36, 38	8.4641, 0.0, 1.1682, 4.4575, 2.2769, 1.9541, 0.8421
40	ρ_1	27, 28, 32, 34, 38	8.0576, 0.5755, 0.0, 1.0797, 2.0556, 4.8491, 1.2892, 2.7233
40	ρ_2	3, 5, 6, 21, 22, 23, 25, 33	0.8001, 40.0, 0.0, 16.2802, 10.0071, 0.8967, 40.0, 13.2398, 29.6258, 40.0, 40.0
40	ρ_3	3, 5, 6, 21, 22, 23, 25, 33	0.6251, 40.0, 0.0, 16.1565, 9.8913, 0.6795, 40.0, 13.4879, 27.8043, 40.0, 40.0
40	ρ_4	3, 5, 6, 21, 22, 23, 25, 33	0.5001, 40.0, 0.0, 16.0669, 9.8071, 0.5206, 40.0, 13.6789, 26.5124, 40.0, 40.0
40	ρ_5	3, 5, 6, 21, 22, 23, 25, 33	0.3848, 40.0, 0.0, 15.9587, 9.6872, 0.3073, 40.0, 13.8761, 25.1584, 40.0, 39.1112
40	ρ_6	3, 5, 21, 22, 23, 25, 33	0.2859, 40.0, 0.0, 15.7173, 9.4906, 40.0, 13.9328, 23.6801, 39.7836, 33.8486
40	ρ_7	3, 5, 21, 22, 23, 25, 33	0.2001, 40.0, 0.0, 15.4874, 9.4508, 40.0, 13.7933, 23.0911, 39.2752, 29.5496
40	ρ_8	3, 5, 21, 22, 23, 25, 33	0.1251, 40.0, 0.0, 15.2864, 9.4166, 40.0, 13.6738, 22.5771, 38.8308, 25.8031
40	ρ_9	3, 5, 21, 22, 23, 25, 33	0.0668, 40.0, 15.1301, 9.3901, 40.0, 13.5824, 22.1784, 38.4854, 22.8975
41	ρ_0	28, 32, 37, 38, 39	8.4617, 0.0, 2.048, 3.4139, 0.7549, 2.1776, 2.5777
41	ρ_1	28, 32, 37, 38	7.9215, 0.8957, 0.0, 3.274, 4.1968, 4.3058, 2.4521
41	ρ_2	3, 4, 5, 20, 21, 22, 23, 25, 29, 32, 36, 37, 38, 39	0.8001, 40.0, 0.0, 4.2353, 15.2321, 40.0, 16.3957, 40.0, 24.1538, 40.0, 5.2109, 0.2405, 27.3646, 40.0, 40.0, 40.0
41	ρ_3	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.6251, 40.0, 0.0, 3.8072, 14.9576, 40.0, 16.5738, 40.0, 23.9801, 40.0, 3.3489, 32.333, 40.0, 40.0, 40.0
41	ρ_4	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.5001, 40.0, 0.0, 3.5157, 14.7537, 40.0, 16.6401, 40.0, 23.8734, 40.0, 1.9975, 35.6723, 40.0, 40.0, 40.0
41	ρ_5	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.3848, 40.0, 0.0, 3.2474, 14.5665, 40.0, 16.701, 40.0, 23.7793, 40.0, 0.7564, 38.7337, 40.0, 40.0, 40.0
41	ρ_6	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.2859, 40.0, 0.0, 3.0923, 14.4424, 40.0, 16.6592, 40.0, 23.7056, 40.0, 0.1685, 40.0, 40.0, 40.0, 40.0
41	ρ_7	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.2001, 40.0, 0.0, 2.9118, 14.3698, 40.0, 16.7854, 39.1052, 23.81, 40.0, 0.1101, 40.0, 40.0, 40.0, 40.0
41	ρ_8	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.1251, 40.0, 0.0, 2.7218, 14.3081, 40.0, 16.9674, 38.0552, 23.9516, 40.0, 0.0784, 40.0, 40.0, 40.0, 40.0
41	ρ_9	3, 4, 5, 20, 21, 22, 23, 25, 32, 36, 37, 38, 39	0.0668, 40.0, 2.5742, 14.2601, 40.0, 17.1084, 37.2414, 24.0626, 40.0, 0.0532, 40.0, 40.0, 40.0, 40.0
42	ρ_0	28, 31, 32, 39	8.4489, 0.0, 2.9557, 0.5842, 3.6518, 2.8956
42	ρ_1	28, 32, 37	7.7615, 1.3738, 0.0, 5.2569, 5.1678, 1.1169
42	ρ_2	3, 4, 5, 8, 25, 29	0.8001, 40.0, 0.0, 2.7474, 11.5296, 40.0, 3.1096, 25.1785, 20.4246
42	ρ_3	3, 4, 5, 8, 25, 29	0.6251, 40.0, 0.0, 2.3752, 11.4155, 40.0, 3.8174, 24.0895, 20.2282
42	ρ_4	3, 4, 5, 8, 25, 29	0.5001, 40.0, 0.0, 2.1107, 11.3338, 40.0, 4.3177, 23.3171, 20.0931
42	ρ_5	3, 4, 5, 8, 25, 29	0.3848, 40.0, 0.0, 1.8676, 11.2583, 40.0, 4.7757, 22.6083, 19.9723
42	ρ_6	3, 4, 5, 8, 25, 29	0.2859, 40.0, 0.0, 1.66, 11.1935, 40.0, 5.1653, 22.0037, 19.8716
42	ρ_7	3, 4, 5, 8, 25, 29	0.2001, 40.0, 0.0, 1.4806, 11.1373, 40.0, 5.5009, 21.482, 19.7864
42	ρ_8	3, 4, 5, 8, 25, 29	0.1251, 40.0, 0.0, 1.324, 11.088, 40.0, 5.7929, 21.0271, 19.7135
42	ρ_9	3, 4, 5, 8, 25, 29	0.0668, 40.0, 1.2025, 11.0496, 40.0, 6.0191, 20.6745, 19.6578

Appendix D. The verification scripts

D.1 The alias constraints

```

"""Rigorous check (Arb) of the alias-step constraints for the big-block design with value classes of right masks.
Exponents per n; worst case over all deficiency profiles with f(1/2) <= 1/2 - tau0 (else the preliminary search wins).
- 2^F repeated trials, F = -log2 E_V 2^{-d(V)} over (m-2k)-subsets V of M, so F <= d(M) and F <= 1/2 - tau0.
- Right masks Y are grouped by (b^F(Y), c^F(Y)); at most (2m+1) 2^{-m-d(M)} classes, so the right list over all trials
  is at most r + (1 - gamma) mu. The left side pays with d(L) >= F, which needs l >= mu(1-2a).
Usage: python verify_alias_bigclass.py params.json"""
from flint import arb, ctx
import json, sys
ctx.prec = 200
P = json.load(open(sys.argv[1]))
Q = lambda x: arb(str(x))
tau0, eta, a, mu, gam, l = Q(P["tau0"]), Q(P["eta"]), Q(P["a"]), Q(P["mu"]), Q(P["gamma"]), Q(P["l"])

```

```

r = 1 - mu - 1
def h(x): return -(x*x.log() + (1-x)*(1-x).log())/arb(2).log()
half = Q("0.5"); v = mu*(1 - 2*a); Fmax = half - tau0
log2_3 = arb(3).log()/arb(2).log()
vals = {
    "h(2a)/2": h(2*a)/2,
    "left list": 1 + (h(a) - gam)*mu,
    "right classes": r + (1 - gam)*mu,
    "pairs": 1 - mu + (h(a) + h(half - a) - gam)*mu - eta,
    "right masks, all trials": h(half - a)*mu + Fmax,
    "left masks, all trials": h(a)*mu + Fmax,
    "(mu/2) log2 3": (mu/2)*log2_3, "v": v, "l": 1, "r": r}
checks = {
    "coverage: gamma < h(2a)/2": gam < h(2*a)/2,
    "left list: 1 + (h(a) - gamma) mu < tau0": vals["left list"] < tau0,
    "right classes: r + (1 - gamma) mu < tau0": vals["right classes"] < tau0,
    "pairs: 1 - mu + (h(a)+h(1/2-a)-gamma) mu - eta < tau0": vals["pairs"] < tau0,
    "right masks in every trial: h(1/2-a) mu + (1/2 - tau0) < tau0": vals["right masks, all trials"] < tau0,
    "left masks in every trial: h(a) mu + (1/2 - tau0) < tau0": vals["left masks, all trials"] < tau0,
    "value maps of L and R: l, r < tau0": (l < tau0) and (r < tau0) and (r > 0),
    "left side pays the trials: l > mu(1-2a)": l > v,
    "reduction mod P on half blocks: (mu/2) log2 3 < eta": vals["(mu/2) log2 3"] < eta,
    "sampled set fits in half the input: mu(1-2a) <= 1/2": v <= half,
}
for k, x in vals.items(): print(f"{k:24s} {x.str(8)}")
ok = True
for k, c in checks.items():
    print(("PASS " if c else "FAIL ") + k); ok &= bool(c)
slacks = [h(2*a)/2 - gam, tau0 - vals["left list"], tau0 - vals["right classes"], tau0 - vals["pairs"],
    tau0 - vals["right masks, all trials"], 1 - v, eta - vals["(mu/2) log2 3"]]
eps0 = min(x.lower() for x in slacks)
print(f"least slack eps0 >= {eps0.str(5)} (lower end of the Arb balls)")
print("ALL ALIAS CONSTRAINTS PASS (strict)" if ok else "SOME CONSTRAINT FAILS")

```

D.2 The 430 region integrals

""Rigorous certificate (Arb ball arithmetic) for a FAMILY of checksum filters chosen per pattern.

For every filter v and every region of patterns assigned to v (case 1, or a band of W in case 2), the JSON lists multipliers $z \geq 0$ ($z[0]$ is the Chernoff exponent c) and the rows G of the domination inequality:

case 1: $G = [\Delta - u_v, \Gamma - w_v, u_v - u_l \text{ (1 in cons)}]$
band $[lo, hi)$: $G = [a_{w_v} - u_v, w_v - \Gamma_{lo}, \Gamma_{hi} - w_v \text{ (if hi)}, u_v - u_l \text{ (1 in cons)}]$

We prove $\int M(x)^2 2^{-z} \cdot G(x) dx < 2^{-\eta}$ (normalised circle integral) for every listed certificate.

Method: the paper's Appendix A, as in `verify_moments_gen.py`. Inner convolutions are evaluated exactly at N nodes with Arb polynomial products; trapezoid errors use analytic-strip bounds. On the outer strip $|Im x| \leq 2\beta$, $|Re u| \leq U^*$ and $|Re w| \leq W^*$ follow from the Cauchy-Schwarz bounds on Q and QB , which bounds $Re(z \cdot G)$.""

```

from flint import arb, arb_poly, ctx
import json, sys, time
ctx.prec = 200
A = arb
P = json.load(open(sys.argv[1]))
tau, eta = A(str(P["tau0"])), A(str(P["eta"]))
Delta = eta - tau; Gamma = 2*tau - eta; aG = Delta/Gamma
N = int(P.get("N", 1024)); sig, beta = A(str(P.get("sig", "0.45"))), A(str(P.get("beta", "0.19")))
pi, ln2 = A.pi(), A(2).log()
pow2 = lambda t: (t*ln2).exp()
log2 = lambda t: t.log()/ln2
xs = [2*pi*k/N for k in range(N)]
def trap_err(B0, width): return 2*B0/((A(N)*width).exp() - 1)
def ball(x, err): return x + A(0, err.upper())
def abs_sum(coef, y): return sum(abs(coef[j])*(A(j+1)*y).cosh() for j in range(len(coef)))
mean = lambda v: sum(v)/N
COS = {}
def cosj(j):
    if j not in COS: COS[j] = [(A(j)*x).cos() for x in xs]
    return COS[j]
def trig_nodes(coef): return [sum(coef[j]*cosj(j+1)[k] for j in range(len(coef))) for k in range(N)]
def corr(Fv, Gv):
    ""c[j] = (1/N) sum_i F[(i+j) mod N] G[i], via one polynomial product.""
    pa = arb_poly(Fv)
    b = [A(0)]*(N + 1)
    for i in range(N): b[N - i] = Gv[i]
    pr = pa*arb_poly(b)
    co = [pr[t] for t in range(2*N)]
    return [(co[N + j] + co[j])/N for j in range(N)]

def filter_data(fp):
    r = [A(str(x)) for x in fp["r_coef"]]; s = [A(str(x)) for x in fp["s_coef"]]; kappa = A(str(fp["kappa"]))
    rk, sk = trig_nodes(r), trig_nodes(s)
    F = [pow2(t) for t in rk]; G = [pow2(t) for t in sk]
    Rs, Ss = abs_sum(r, sig), abs_sum(s, sig)
    Zf = ball(mean(F), trap_err(pow2(Rs), sig)); Zg = ball(mean(G), trap_err(pow2(Ss), sig))
    Zfk = ball(mean([pow2(kappa*t) for t in rk]), trap_err(pow2(kappa*Rs), sig))
    Zgk = ball(mean([pow2(kappa*t) for t in sk]), trap_err(pow2(kappa*Ss), sig))
    LZ = log2(Zf*Zg); H = log2(Zfk) + log2(Zgk) - kappa*LZ

```

```

eP = trap_err(pow2(Rs + Ss), sig); eY = trap_err(pow2(Rs + Ss)*(Rs + Ss), sig)
Pc = corr(F, G)
Yc1 = corr([F[k]*rk[k] for k in range(N)], G); Yc2 = corr(F, [G[k]*sk[k] for k in range(N)])
u, w = [], []
for j in range(N):
    Pj = ball(Pc[j], eP); Yj = ball(Yc1[j] + Yc2[j], eY)
    L = log2(Pj) - LZ; B = Yj/Pj - LZ
    u.append(L + (kappa - 1)*B - H); w.append((2 - kappa)*B - 2*L + H)
# outer strip bounds (|Im x| <= 2 beta)
rb = [r[j]*(A(j+1)*beta).cosh() for j in range(len(r))]; sb = [s[j]*(A(j+1)*beta).cosh() for j in range(len(s))]
Ir = ball(mean([pow2(2*t) for t in trig_nodes(rb)]), trap_err(pow2(2*abs_sum(rb, sig)), sig))
Is = ball(mean([pow2(2*t) for t in trig_nodes(sb)]), trap_err(pow2(2*abs_sum(sb, sig)), sig))
Af, Ag = Ir/(Zf*Zf), Is/(Zg*Zg)
q0 = ((Af - 1)*(Ag - 1)).sqrt()
qb = (Af*Ag).sqrt()*(abs_sum(r, beta) + abs_sum(s, beta) + abs(LZ))
assert q0 < 1, "q0 >= 1: outer strip too wide"
Bmax = qb/(1 - q0); lneg = -log2(1 - q0)
Ustar = lneg + abs(kappa - 1)*Bmax + abs(H)
Wstar = 2*lneg + abs(2 - kappa)*Bmax + abs(H)
return dict(u=u, w=w, Ustar=Ustar, Wstar=Wstar, q0=q0)

t0 = time.time()
FD = [filter_data(fp) for fp in P["filters"]]
print(f"{len(FD)} filters, N = {N}, prepared in {time.time() - t0:.0f} s; max q0 = "
      f"{max(float(d['q0'].upper()) for d in FD):.4f}", flush=True)
for i, d in enumerate(FD):
    print(f"  filter {i}: q0 {float(d['q0'].upper()):.4f}  U* {float(d['Ustar'].upper()):.3f}  W* {float(d['Wstar'].upper()):.3f}")
STRIP = [dict(U=float(d['Ustar'].upper()), W=float(d['Wstar'].upper())) for d in FD]
if len(sys.argv) > 2 and sys.argv[2] == "strip":
    json.dump(STRIP, open(sys.argv[1] + ".strip.json", "w")); sys.exit(0)
m2 = [2 + 2*x.cos() for x in xs]; M2s = 2 + 2*(2*beta).cosh(); target = pow2(eta)
worst = None; ok = True
for c in P["certs"]:
    v, z = c["v"], [A(str(t)) for t in c["z"]]; d = FD[v]; cons = c["cons"]
    if c["case"] == "case1":
        rows = [[Delta - d["u"][k], Gamma - d["w"][k] for k in range(N)]
        sup = [Delta + d["Ustar"], Gamma + d["Wstar"]]
        assert z[0] >= 1
    else:
        lo, hi = A(str(c["lo"])), (A(str(c["hi"])) if c["hi"] is not None else None)
        rows = [[aG*d["w"][k] - d["u"][k], d["w"][k] - Gamma*lo + ((Gamma*hi - d["w"][k]) if hi is not None else [])
                 for k in range(N)]
        sup = [aG*d["Wstar"] + d["Ustar"], d["Wstar"] - Gamma*lo + ((Gamma*hi + d["Wstar"]) if hi is not None else [])
        assert z[0]*lo >= 1, "Chernoff exponent must be >= alpha <= 1/lo"
    for l in cons:
        for k in range(N): rows[k].append(d["u"][k] - FD[l]["u"][k])
        sup.append(d["Ustar"] + FD[l]["Ustar"])
    assert len(z) == len(rows[0]) and all(t >= 0 for t in z)
    T = mean([m2[k]*pow2(sum(z[i]*rows[k][i] for i in range(len(z)))) for k in range(N)])
    B0exp = sum(z[i]*sup[i] for i in range(len(z)))
    err = trap_err(M2s*pow2(B0exp), 2*beta)
    I = T + A(0, err.upper())
    good = bool(I < target); ok &= good
    marg = eta - log2(I)
    if worst is None or float(marg.lower()) < float(worst[0].lower()): worst = (marg, c)
    print(f"filter {v} {c['case']}: {' ' if c['case'] == 'case1' else ' ' + str(c['lo'])}: I <= {I.upper().str(10)}"
          f"  log2 margin {marg.str(5)}  (outer err {err.str(3)}) {'PASS' if good else 'FAIL'}", flush=True)
print(f"worst log2 margin {worst[0].str(5)} at filter {worst[1]['v']} {worst[1]['case']}")
print("ALL CERTIFICATES PASS" if ok else "SOME CERTIFICATE FAILS")

```