An Improved Computational Lower Bound for f(3) via Wróblewski's Recursive Pairing of Behrend Sphere Blocks

Darby Bailey McDonough, Ph.D.

Pip Projects, Inc. · Digital Sidewalk Lab · Independent Researcher

ORCID: [0000-0003-4072-6191](0000-0003-4072-6191)

Preprint — April 2026

Keywords: additive combinatorics; arithmetic progressions; harmonic sums; certified computation; Behrend construction; Wróblewski pairing

Abstract

We report a new computational lower bound improving the best published result due to Wróblewski (1984): **f(3) ≥ 3.0084928720**.

The construction is Wróblewski's recursive pairing of Behrend sphere blocks, implemented as a Python script that embeds, compiles, and calls a C kernel at runtime; the C kernel uses `unsigned __int128` arithmetic and OpenMP parallelism for node enumeration, while Python arbitrary-precision integers handle all shift computation. The 3-AP-free property is guaranteed by Wróblewski's pairing theorem and requires no computational verification. A conservative floating-point error analysis, explicitly separating denominator rounding and summation error, confirms that the computed value exceeds Wróblewski's bound by a factor exceeding 10^{15} relative to a conservative total IEEE 754 error bound. Two independent implementations (single-threaded and 12-core OpenMP) produce matching node counts and matching harmonic sums to the reported precision. Full source code, checkpoint data, checksums, and citation metadata are provided as supplementary material in the Zenodo deposit.

1. The Problem

For k ≥ 3, define:

$$f(k) = \sup\left\{\sum_{n\in A}\frac{1}{n} : A \subseteq \mathbb{N}^+, A \textit{ is } k-AP-free\right\}$$

A set A ⊆ $\mathbb{N}^+$ is *k*-AP-free if it contains no *k*-term arithmetic progression. The finiteness of f(3) was proved by Bloom and Sisask (2020) as a consequence of their density bounds on 3-AP-free sets; the explicit value of f(3) remains unknown.

Best published lower bound: f(3) ≥ 3.00849 (Wróblewski 1984, *Mathematics of Computation* 43, 261–262).

This paper: f(3) ≥ 3.0084928720, a strict improvement.

2. Construction

We follow Wróblewski (1984) exactly.

2.1 Behrend Sphere Blocks

For integers $p \geq 2$, $q \geq 1$, $r \geq 0$, define the weight function:

$$T_p(k) = (k - \lfloor(p+1)/2\rfloor)(k - \lfloor(p-1)/2\rfloor) / 2, \quad 0 \leq k < p$$

The Behrend block $B(p, q, r)$ is the set of integers:

$$B(p, q, r) = \{ \Sigma_i d_i(2p-1)^i : 0 \leq d_i < p, \ \Sigma_i T_p(d_i) = r \}$$

Proposition (AP-free by convexity). $B(p, q, r)$ is 3-AP-free. The function T_p is strictly convex; for any $a, c \in B$ with $a \neq c$, the midpoint $(a+c)/2$ satisfies $\Sigma\, T_p(\text{digit}) < r$, so $(a+c)/2 \notin B$.

2.2 Pairing Operation

For AP-free sets Z and T with maxima $z = \max(Z)$ and $t = \max(T)$, set $m = \max(z, t)$ and:

$$s_1 = m + z + 1, \quad s_2 = 3m + 2t + z + 3, \quad s_3 = 3m + 4t + z + 4$$

Define $\text{pair}(Z, T) = Z \cup (T+s_1) \cup (T+s_2) \cup (T+s_3)$.

Theorem (Wróblewski 1984, Theorem 1). If Z and T are 3-AP-free, then $\text{pair}(Z, T)$ is 3-AP-free.

Proof sketch. Each copy $T+s_i$ is AP-free since T is AP-free. The shifts $s_1 < s_2 < s_3$ are chosen so that any AP spanning two distinct components would require its midpoint to exceed $\max(Z)$, contradicting the separation conditions. See Wróblewski (1984) for the full case analysis. □

Consequence. The entire construction is proven AP-free by induction. No computational AP-verification is needed or performed.

2.3 Block Sequence and Harmonic Sum

$H(Z_{n+1}) = H(Z_n) + \Sigma\_\{b \in B_{n+1}\}$ [$1/(b+s_1) + 1/(b+s_2) + 1/(b+s_3)$]

Step	Block	Nodes
Z_0	G_3, elements $\leq$ 21,523,361	65,536
Z_1	B(4,9,5)	64,512
Z_2	B(4,10,5)	258,048
Z_3	B(6,9,12)	1,053,696
Z_4	B(6,10,13)	5,959,680
Z_5	B(6,11,15)	34,174,976
Z_6	B(6,12,16)	197,345,280
Z_7	B(6,13,17)	1,139,294,208
Z_8	B(6,14,18)	6,578,044,928

Z_9	B(6,15,20)	38,260,604,928
Z_{10}	B(6,16,21)	222,697,619,456
Z_{11} ★	B(6,17,22)	1,293,635,551,232

where $G_3 = \{ n \in \mathbb{N}^+ : \text{base-3 digits of } n-1 \in \{0,1\} \}$ is the standard greedy 3-AP-free set, and for $n \geq 3$ the radius follows $r_n = \lfloor 4(n+6)/3 \rfloor$ with $r_5 = 15$ (Wróblewski 1984).

3. Computational Implementation

3.1 Overflow Hazards at Scale

Three numerical hazards arise at Z_{12} and beyond that do not affect Z_0–Z_{11}.

Shift overflow (Z_{12}+). The shifts $s_2 \approx 1.61\times10^{19}$ and $s_3 \approx 2.17\times10^{19}$ at Z_{12} exceed the 64-bit signed integer maximum (9.22×10^{18}). We use Python arbitrary-precision integers for all shift and z_{max} arithmetic. Shifts are passed to the C kernel as decimal strings and parsed with `strtod()`. These values are not representable exactly once they exceed the exact-integer range of IEEE 754 double precision; the resulting relative rounding error is incorporated into the analysis in Section 3.2.

Node value overflow (Z_{13}+). Node values at Z_{13} reach approximately 3×10^{19}, beyond exact double representation ($2^{53} \approx 9\times10^{15}$) and beyond 64-bit long range. All internal arithmetic in the C recursion uses `unsigned __int128`, with a cast to `double` deferred to the final `1.0/(v+s)` step only.

OpenMP reduction. With 12 threads, each accumulates a thread-local Kahan sum. The final merge uses simple summation (`global_H += local_H`), introducing at most approximately 1.6×10^{-22} absolute error, negligible at 10 decimal places.

3.2 Floating-Point Error Certificate

Let $N = 3 \times 1,293,635,551,232 \approx 3.88\times10^{12}$ be the total number of floating-point additions in the Z_{11} step, let $\varepsilon = 2^{-53} \approx 1.11\times10^{-16}$ be the IEEE 754 double-precision machine epsilon, and let $\max|t_i| \leq 1/s_1$^{min} $\approx 2.20\times10^{-18}$ be the largest possible term magnitude (s_1^{min} is the minimum first shift across all Z_{11} nodes).

Denominator rounding. For each term $1/x$, casting the denominator $x = b+s_i$ to `double` introduces relative error within standard IEEE 754 rounding bounds, giving the conservative bound $\delta_{den} \leq \varepsilon N \max|t_i|$.

Summation error. By standard backward error analysis (Higham, 2002, Theorem 4.1): $\delta_{sum} \leq 2\varepsilon N \max|t_i|$

Combined bound. $\delta_H \leq \delta_{den} + \delta_{sum} \leq 3\varepsilon N \max|t_i| \approx 2.85\times10^{-21}$

Since $\Delta = 3.0084928720 - 3.00849 = 2.872\times10^{-6}$, the margin exceeds the total conservative numerical error bound by a factor exceeding 10^{15}. Therefore $H(Z_{11}) > 3.00849$ is numerically certified under IEEE 754 double-precision arithmetic.

3.3 The C Kernel

Core enumeration (simplified):

```
void go(int pos, int rem, unsigned __int128 val,
        double *H, double *kc, unsigned __int128 *cnt) {
    if (pos == Q) {
        if (rem == 0) {
            (*cnt)++;
            double v = (double)val;    /* sole cast point */
            double x = 1.0/(v+S1)+1.0/(v+S2)+1.0/(v+S3) - *kc;
            double t = *H + x;
            *kc = (t - *H) - x;       /* Kahan compensation */
            *H = t;
        }
        return;
    }
    if (rem > (Q-pos)*MAX_W) return;  /* upper-bound pruning */
    for (int d = 0; d < P; d++) {
        if (Tv[d] > rem) continue;
        go(pos+1, rem-Tv[d],
           val + (unsigned __int128)d * bpow[pos], H, kc, cnt);
    }
}
```

Shifts S1, S2, and S3 are read via `strtod()`. `unsigned __int128` is available on GCC ≥ 4.6 and Clang ≥ 3.1 on standard 64-bit platforms.

3.4 Verification Protocol

At Z_{12}, the first step requiring overflow safeguards, two independent implementations must agree on both metrics before computation continues:

Metric	Required value
Node count	7,563,706,368,000 (exact)
H at Z_{12}	3.0084948386 ± 10^{-8}

Node count agreement rules out branch-dropping in OpenMP. Agreement in H rules out silent overflow. Any deviation halts computation with diagnostics.

4. Results

Step	H	Gain	Nodes	Gain ratio
Z_0	3.0042100	—	65,536	—
Z_1	3.0062834	+0.0020734	64,512	—

Z_2	3.0072915	+0.0010082	258,048	0.4864
Z_3	3.0078115	+0.0005199	1,053,696	0.5156
Z_4	3.0081327	+0.0003212	5,959,680	0.6178
Z_5	3.0083053	+0.0001727	34,174,976	0.5376
Z_6	3.0083962	+0.0000909	197,345,280	0.5261
Z_7	3.0084439	+0.0000477	1,139,294,208	0.5247
Z_8	3.0084689	+0.0000250	6,578,044,928	0.5249
Z_9	3.0084822	+0.0000132	38,260,604,928	0.5296
Z_{10}	3.0084892	+0.0000070	222,697,619,456	0.5306
Z_{11} ★	3.0084928720	+0.0000037	1,293,635,551,232	0.5312
Z_{12}	3.0084948386	+0.0000020	7,563,706,368,000	0.5317
Z_{13}	3.0084958842	+0.0000010	44,239,186,034,688	0.5317

The bound $f(3) \geq 3.0084928720$ is established at Z_{11}. Steps Z_{12} and Z_{13} are reported as supporting computations and indicate that the gain ratio has stabilized near 0.5317. The construction limit extrapolates to $H_\infty \approx H(Z_{11}) + g_{12}/(1 - 0.5317) \approx 3.0084970$. This is a ceiling for the $k = 3$ Wróblewski construction; it is not an upper bound on $f(3)$.

5. Theorem and Proof

Theorem. $f(3) \geq 3.0084928720$.

Proof. By Sections 2.2–2.3, Z_{11} is 3-AP-free by Wróblewski's pairing theorem (Theorem 1), applied inductively. By Section 3.2, the computed harmonic sum 3.0084928720 is certified to exceed 3.00849: the margin $\Delta = 2.872\times10^{-6}$ exceeds the conservative total floating-point error bound $\delta_H \leq 2.85\times10^{-21}$ by a factor exceeding 10^{15}. Two independent implementations with matching node count 1,293,635,551,232 and matching H values provide an independent cross-check against computational error. Since Z_{11} is a 3-AP-free subset of $\mathbb{N}^+$ with computed harmonic sum 3.0084928720, certified to exceed 3.00849, we conclude that $f(3) \geq 3.0084928720$. □

6. Reproducibility and Deposit Contents

The Zenodo deposit contains the following supplementary materials: `f3_lower_bound.py`, `z_state.json`, `README.md`, `SHA256SUMS.txt`, `CITATION.cff`, and `LICENSE`.

Source code. The main executable script is `f3_lower_bound.py`. Requirements: Python 3.8 or later (standard library only) and GCC ≥ 4.6 or Clang ≥ 3.1. OpenMP is optional; single-threaded and parallel implementations produce identical numerical outputs at the stated checkpoints.

Build and run. Running `python f3_lower_bound.py` compiles the C kernel automatically, reconstructs the Z_{11} shift chain, and begins streaming from Z_{12} or resumes from the included checkpoint if `z_state.json` is present.

Checkpoint data. The file `z_state.json` contains the verified Z_{13} state. The field `h` records the last verified harmonic sum, and `last_n = 13` indicates the latest completed step.

Checksums and citation metadata. `SHA256SUMS.txt` records the bundled checksums. `CITATION.cff` provides repository citation metadata for software and dataset reuse.

Representative local validation. In addition to the long production runs used for the reported values, the repository was smoke-tested in the author's local terminal environment: syntax and import checks passed, the certified Z_{13} checkpoint loaded correctly with hash verification, the embedded OpenMP C kernel compiled, and selected small-case comparisons of `B_max_exact` and `stream_step` against brute-force enumeration matched exactly.

Wall times (Apple MacBook Pro, Apple M3, 12-core OpenMP). Z_{11}: 6.0 h (single-threaded C, March 2026); Z_{12}: 2.619 h (12-core OpenMP); Z_{13}: 15.1 h (12-core OpenMP, completed April 5, 2026). Agreement in node count and H at Z_{12} between both implementations was confirmed before Z_{13} was started.

Expected output at Z_{12} and Z_{13}.

```
Z_12:  7,563,706,368,000 nodes  2.619h   (12-core OpenMP, Apple M3)
       H    = 3.0084948386
       gain = +0.0000019666
✅ Z12 VERIFIED: H and node count match ground truth

Z_13:  44,239,186,034,688 nodes  15.1h   (12-core OpenMP, Apple M3)
       H    = 3.0084958842
       gain = +0.0000010455
✅ Z13 checkpoint saved:
cb371461c83d2a07d6eff58d778c27846dda5f5bda6ac1d8060b80bb08d97da4
       Completed: 2026-04-05 02:49:56
```

7. Prior Work and Future Extension

Wróblewski (1984) established $f(3) \geq 3.00849$ using the identical construction on an ODRA 1305 mainframe. The present work extends the computation through Z_{13} and adds safeguards (`strtod`, `unsigned __int128`) required at Z_{12} and beyond, where a naive implementation using `atoll()` or 64-bit long arithmetic would silently produce incorrect results.

Walker (2022, rev. 2025) conducted an exhaustive branch-and-bound search over Kempner sets in bases $b \leq 120$, finding no improvement to f(3). Wróblewski's construction uses Behrend sphere blocks with level-set constraints on digit sums; Kempner sets restrict digit membership. The two approaches are structurally distinct.

On the upper-bound side, there has been substantial recent progress on 3-term progression-free sets. Bloom and Sisask (2020) proved quantitative density bounds on 3-AP-free sets that imply $f(3) < \infty$. Kelley and Meka (2023) obtained substantially stronger quantitative bounds. Bloom and Sisask (2023) gave an exposition and refinement of the Kelley–Meka method. Raghavan (2026) further improved the exponent to 1/6 up to a log log N factor. However, these works concern upper bounds for $r_3(N)$, not explicit numerical upper bounds for the harmonic sum f(3). To the author's knowledge, no explicit numerical upper bound for f(3) has yet been published. Conversely, the best explicit lower bound remained Wróblewski's construction, which is computationally improved here to 3.0084928720.

Future computational extension. The present lower bound is established at Z_{11}; the Z_{12} and Z_{13} computations are included as supporting data and indicate that the gain ratio has stabilized near 0.5317. A natural next step is to continue the identical certified construction to Z_{14} and beyond using additional compute resources. Such runs would not affect the validity of the present theorem, but could yield a further improved explicit lower bound or a short follow-up computational note.

8. AI Collaboration and Validation Disclosure

This work was developed with the assistance of AI systems from OpenAI, Anthropic, Google, and xAI during March–April 2026, used for editorial feedback, code review discussion, literature cross-checking, and reproducibility review. All long production computations, local reruns, smoke tests, checkpoint validation, and final judgment about all claims were performed by the author in a local Apple MacBook Pro terminal environment. All AI-assisted suggestions were treated as advisory rather than authoritative. The author takes full responsibility for the correctness, wording, and scope of all claims. The recursive pairing construction and the proof of 3-AP-freeness are due to Wróblewski (1984). The computational implementation, overflow safeguards, floating-point error certificate, validation workflow, and numerical results reported here are the contribution of this paper.

Author note. This line of inquiry grew out of the author's prior work in audio signal processing, communications systems, and cybernetics, where questions of structure, resonance, and computational constraint motivated the present investigation.

References

1. Wróblewski, J. (1984). A nonaveraging set of integers with a large sum of reciprocals. *Mathematics of Computation*, 43(167), 261–262. https://doi.org/10.2307/2007580
2. Bloom, T., & Sisask, O. (2020). Breaking the logarithmic barrier in Roth's theorem on arithmetic progressions. arXiv:2007.03528. https://arxiv.org/abs/2007.03528
3. Kelley, Z., & Meka, R. (2023). Strong bounds for 3-progressions. FOCS 2023 Best Paper. arXiv:2302.05537. https://arxiv.org/abs/2302.05537
4. Bloom, T. F., & Sisask, O. (2023). The Kelley–Meka bounds for sets free of three-term arithmetic progressions. *Essential Number Theory*, 2(1), 15–44. https://doi.org/10.2140/ent.2023.2.15 (arXiv:2302.07211)
5. Raghavan, R. (2026). Improved bounds for 3-progressions. arXiv:2603.27045. https://arxiv.org/abs/2603.27045

6. Walker, A. (2022, rev. 2025). Integer sets of large harmonic sum which avoid long arithmetic progressions. arXiv:2203.06045. https://arxiv.org/abs/2203.06045
7. Behrend, F. A. (1946). On sets of integers which contain no three terms in arithmetical progression. *Proceedings of the National Academy of Sciences*, 32(12), 331–332. https://doi.org/10.1073/pnas.32.12.331
8. Higham, N. J. (2002). *Accuracy and Stability of Numerical Algorithms* (2nd ed.). SIAM. https://doi.org/10.1137/1.9780898718027

Correspondence: ORCID 0000-0003-4072-6191

Supplementary material: `f3_lower_bound.py`, `z_state.json`, `README.md`, `SHA256SUMS.txt`, `CITATION.cff`, `LICENSE` (Zenodo deposit)

Zenodo DOI: 10.5281/zenodo.19722202