Empirical Investigation of an Open Conjecture: If A is a subset of the natural numbers such that the sum of the reciprocals of

Agentic NL→Lean 4 Pipeline
Job #21

April 26, 2026

Abstract

This report documents the empirical investigation of an open mathematical conjecture that could not be formally proved or disproved in Lean 4 with Mathlib. Numerical experiments were conducted to gather evidence for or against the conjecture. The empirical verdict is: **Empirically Supported**. The conjecture remains formally open.

1 Conjecture Statement

Conjecture 1.

If A is a subset of the natural numbers such that the sum of the reciprocals of the elements in A is infinite, then A must contain arbitrarily long arithmetic progressions.

2 Status

Formal Status: OPEN — no Lean 4 proof or disproof was found.

Empirical Verdict: Empirically Supported

The pipeline attempted formal verification in Lean 4 with Mathlib but was unable to produce a compiling proof or disproof. Empirical testing was then conducted to gather numerical evidence.

3 Basic Empirical Testing

The following output was produced by the basic numerical experiment:

```
=== EXPERIMENT PLAN ===

Conjecture (őErds conjecture on arithmetic progressions, 1936 - OPEN):
  If A    and sum_{a in A} 1/a = ∞, then A contains arbitrarily long APs.

Partial known results:
  * Szemerédi's theorem: positive upper density   arbitrarily long APs.
  * Green-Tao (2004): the primes contain arbitrarily long APs.
```

```
  * Still OPEN in full generality.

Empirical strategy (computational, not a proof):
  TEST 1 - "Divergent-sum" sets: primes, squarefree numbers, numbers
           with many small prime factors, and random "thinned" sets
           whose harmonic tails diverge.  For each we compute the
           longest AP found up to bound N, and check it grows with N.
  TEST 2 - "Convergent-sum" sets (squares, powers of 2, very sparse
           random sets).  The conjecture is *silent* on these, but
           we verify empirically that long APs are scarce - i.e. the
           divergence condition is doing meaningful work.
  TEST 3 - Counterexample search: random subsets of [1..N] with
           prescribed divergence rate; find longest AP; test whether
           a divergent-sum set can have bounded AP length (that would
           contradict the conjecture).
  TEST 4 - Asymptotic scaling: plot longest-AP length L(N) vs log N
           and vs reciprocal-sum S(N) to see whether L grows without
           bound as the sum diverges.
  TEST 5 - Statistical baseline: compare longest APs in divergent-sum
           sets to those in comparably-sized convergent-sum sets
           (sanity check).

Method for finding longest AP in a finite set S (up to bound N):
  O(|S|^2) dynamic programming on pairs (i,j) giving longest AP
  ending at (a_i, a_j) with common difference a_j - a_i.

=== TEST 1 & 4: Divergent-sum sets, longest AP vs N ===
 N=   500  primes:LΣ=6,1/a=2.10  squarefree:LΣ=24,1/a=4.82  rand_divergent:L
    Σ=6,1/a=2.25  squares:LΣ=3,1/a=1.60  powers_of_2:LΣ=2,1/a=2.00
    rand_convergent:LΣ=2,1/a=0.81
 N=  1000  primes:LΣ=7,1/a=2.20  squarefree:LΣ=33,1/a=5.24  rand_divergent:L
    Σ=6,1/a=2.34  squares:LΣ=3,1/a=1.61  powers_of_2:LΣ=2,1/a=2.00
    rand_convergent:LΣ=2,1/a=0.81
 N=  2000  primes:LΣ=9,1/a=2.29  squarefree:LΣ=44,1/a=5.66  rand_divergent:L
    Σ=6,1/a=2.43  squares:LΣ=3,1/a=1.62  powers_of_2:LΣ=2,1/a=2.00
    rand_convergent:LΣ=2,1/a=0.81
 N=  4000  primes:LΣ=10,1/a=2.38  squarefree:LΣ=48,1/a=6.09  rand_divergent:L
    Σ=7,1/a=2.53  squares:LΣ=3,1/a=1.63  powers_of_2:LΣ=2,1/a=2.00
    rand_convergent:LΣ=2,1/a=0.81
 N=  8000  primes:LΣ=10,1/a=2.46  squarefree:LΣ=48,1/a=6.51  rand_divergent:L
    Σ=7,1/a=2.61  squares:LΣ=3,1/a=1.63  powers_of_2:LΣ=2,1/a=2.00
    rand_convergent:LΣ=2,1/a=0.81
 N= 16000  primes:LΣ=10,1/a=2.53  squarefree:LΣ=48,1/a=6.93  rand_divergent:L
    Σ=7,1/a=2.68  squares:LΣ=3,1/a=1.64  powers_of_2:LΣ=2,1/a=2.00
    rand_convergent:LΣ=2,1/a=0.81

=== TEST 2: Convergent-sum sets (squares, powers of 2) ===
  squares up to 10^6  (#=1000): longest AP = 3  (theorem: 3)
  powers of 2 (all):                longest AP = 2  (theorem: 2)

=== TEST 3: Counterexample search - random divergent-sum sets ===
  trials=40, mean |A |678, mean Σ1/ a2.47, mean longest  AP7.22
  suspicious cases Σ(>5 and  L3): 0
```

```
=== TEST 5: Primes - count of 3-term APs (Green-Tao style) ===
  primes   2000: #3-APs = 4457
  longest AP in primes   5000 = 10
```

4 Experiment Code (Basic)

```
import matplotlib
matplotlib.use("Agg")
import numpy as np
import matplotlib.pyplot as plt
import math
import random
from itertools import combinations
from sympy import isprime, primerange

print("===␣EXPERIMENT␣PLAN␣===")
print("""
Conjecture (őErds conjecture on arithmetic progressions, 1936 - OPEN):
  If A    and sum_{a in A} 1/a = ∞, then A contains arbitrarily long APs.

Partial known results:
  * Szemerédi's theorem: positive upper density  arbitrarily long APs.
  * Green-Tao (2004): the primes contain arbitrarily long APs.
  * Still OPEN in full generality.

Empirical strategy (computational, not a proof):
  TEST 1 - "Divergent-sum" sets: primes, squarefree numbers, numbers
           with many small prime factors, and random "thinned" sets
           whose harmonic tails diverge.  For each we compute the
           longest AP found up to bound N, and check it grows with N.
  TEST 2 - "Convergent-sum" sets (squares, powers of 2, very sparse
           random sets).  The conjecture is *silent* on these, but
           we verify empirically that long APs are scarce - i.e. the
           divergence condition is doing meaningful work.
  TEST 3 - Counterexample search: random subsets of [1..N] with
           prescribed divergence rate; find longest AP; test whether
           a divergent-sum set can have bounded AP length (that would
           contradict the conjecture).
  TEST 4 - Asymptotic scaling: plot longest-AP length L(N) vs log N
           and vs reciprocal-sum S(N) to see whether L grows without
           bound as the sum diverges.
  TEST 5 - Statistical baseline: compare longest APs in divergent-sum
           sets to those in comparably-sized convergent-sum sets
           (sanity check).

Method for finding longest AP in a finite set S (up to bound N):
  O(|S|^2) dynamic programming on pairs (i,j) giving longest AP
  ending at (a_i, a_j) with common difference a_j - a_i.
""")

# ----- Longest AP via O(n^2) DP -----
```

```python
def longest_ap(S):
    S = sorted(set(int(x) for x in S))
    n = len(S)
    if n < 3:
        return n
    idx = {v: i for i, v in enumerate(S)}
    dp = [dict() for _ in range(n)]
    best = 2
    for j in range(1, n):
        sj = S[j]
        for i in range(j):
            d = sj - S[i]
            prev = S[i] - d
            if prev in idx:
                ln = dp[i].get(d, 2) + 1
            else:
                ln = 2
            if ln > dp[j].get(d, 0):
                dp[j][d] = ln
            if ln > best:
                best = ln
    return best

# ----- Build test sets -----
def primes_up_to(N):
    return list(primerange(2, N + 1))

def squarefree_up_to(N):
    sieve = np.ones(N + 1, dtype=bool)
    sieve[0] = False
    for p in primerange(2, int(N**0.5) + 1):
        sieve[p*p::p*p] = False
    return [int(i) for i in range(1, N + 1) if sieve[i]]

def squares_up_to(N):
    return [k*k for k in range(1, int(math.isqrt(N)) + 1)]

def powers_of_two_up_to(N):
    out, v = [], 1
    while v <= N:
        out.append(v); v *= 2
    return out

def random_divergent_set(N, seed=0):
    rng = np.random.default_rng(seed)
    ks = np.arange(2, N + 1)
    probs = 1.0 / np.log(ks + 1.0)
    mask = rng.random(ks.shape) < probs
    return [int(k) for k in ks[mask]]

def random_convergent_set(N, seed=0):
    rng = np.random.default_rng(seed)
    ks = np.arange(2, N + 1)
    probs = 1.0 / (ks ** 1.2)
```

```python
    mask = rng.random(ks.shape) < probs
    return [int(k) for k in ks[mask]]

def recip_sum(S):
    return float(np.sum(1.0 / np.array(S, dtype=float)))

# ----- TEST 1 & 4: scaling of longest AP -----
print("===␣TEST␣1␣&␣4:␣Divergent-sum␣sets,␣longest␣AP␣vs␣N␣===")
Ns = [500, 1000, 2000, 4000, 8000, 16000]
results = {name: {"N": [], "S": [], "L": [], "size": []} for name in
           ["primes", "squarefree", "rand_divergent",
            "squares", "powers_of_2", "rand_convergent"]}

builders = {
    "primes":           primes_up_to,
    "squarefree":       squarefree_up_to,
    "rand_divergent":   lambda N: random_divergent_set(N, seed=1),
    "squares":          squares_up_to,
    "powers_of_2":      powers_of_two_up_to,
    "rand_convergent":  lambda N: random_convergent_set(N, seed=2),
}

for N in Ns:
    for name, fn in builders.items():
        S = fn(N)
        if len(S) < 2:
            continue
        rs = recip_sum(S)
        if len(S) > 3500:
            S_use = S[:3500]
        else:
            S_use = S
        L = longest_ap(S_use)
        results[name]["N"].append(N)
        results[name]["S"].append(rs)
        results[name]["L"].append(L)
        results[name]["size"].append(len(S))
    print(f"␣N={N:6d}␣␣"
          + "␣␣".join(f"{n}:L={results[n]['L'][-1]},Sum1/a={results[n]['S
             '][-1]:.2f}"
                      for n in builders if results[n]['L']))

# ----- TEST 2: Convergent-sum known sets cannot have long APs? -----
print("\n===␣TEST␣2:␣Convergent-sum␣sets␣(squares,␣powers␣of␣2)␣===")
sq = squares_up_to(10**6)
p2 = powers_of_two_up_to(10**9)
L_sq = longes
# ... [truncated]
```

5 Conclusion

The conjecture remains formally open. Numerical experiments **support** the conjecture — no counterexamples were found across all tested parameter ranges. Further investigation (both formal and empirical) is warranted.