Article

Not peer-reviewed version

On Representations of Even Integers as a Sum of Two Semiprimes

Ricardo Adonis Caraccioli Abrego *

Posted Date: 31 October 2025

doi: 10.20944/preprints202510.2442.v1

Keywords: semiprimes; circle method; bilinear Bombieri–Vinogradov; singular series; large sieve; dispersion method; minor arcs

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2025 doi:10.20944/preprints202510.2442.v1

Article

On Representations of Even Integers as a Sum of Two Semiprimes

Ricardo Adonis Caracciolo Abrego

Departamento de Ingeniería Eléctrica, Campus Cortés, Universidad Nacional Autónoma de Honduras (UNAH-VS); ricardo.caracciolo@unah.edu.hn

Abstract

Representations of a large even integer N as a sum of two semiprimes (products of two primes, squares allowed) are studied. Using a smooth bilinear weight W localized on $uv \asymp N$ and the Hardy–Littlewood circle method, an asymptotic formula of order N with a power saving in $\log N$ is obtained:

$$R(N) \;=\; \sum_{\substack{s_1+s_2=N\\ s_1,s_2\in S_2}} W(s_1)W(s_2) \;=\; \mathfrak{S}(N)\,N \;+\; O\Big(\frac{N}{(\log N)^{1+\delta}}\Big)$$

for some absolute $\delta > 0$, where the multiplicative singular series $\mathfrak{S}(N)$ is bounded and bounded away from zero along even N. The argument uses a bilinear Bombieri–Vinogradov type input (via the dispersion method and the multiplicative large sieve) together with minor-arc bounds based on Gallagher's lemma. As a direct consequence, every sufficiently large even integer is the sum of two semiprimes (unweighted existence). A computational check (unweighted) finds no counterexamples up to $N = 10^6$ except $2, 4, 6, 22$, in agreement with the asymptotic formula.

Keywords: semiprimes; circle method; bilinear Bombieri–Vinogradov; singular series; large sieve; dispersion method; minor arcs

MSC: 11P32 (primary); 11N36; 11N13; 11N05

Historical note and scope.

Almost-prime Goldbach results have a long history, notably Chen's theorem $N = p + P_2$ for large even N and its expositions. An explicit published statement asserting the unconditional existence of $P_2 + P_2$ representations for all sufficiently large even N in the exact form obtained here (via a weighted asymptotic with positive singular series) has not been located by the author; it may be implicit in the classical literature. The note isolates a transparent circle-method route, with a bilinear Bombieri–Vinogradov input, to a positive singular series and the unweighted existence.

1. Introduction

Let $S_2 = \{n \geq 2 : \ n = pq \text{ with } p \leq q \text{ primes}\}$. Consider the weighted representation function

$$R(N) := \sum_{\substack{s_1+s_2=N\\ s_1,s_2\in S_2}} W(s_1)\,W(s_2), \tag{1}$$

where W is a nonnegative smooth weight localizing $s_i \asymp N$. The main result is an asymptotic of order N with a power saving in $\log N$. The method uses the major/minor arc decomposition and a bilinear Bombieri–Vinogradov type estimate for semiprimes.

2. Statement of Results

Theorem 1. *Let W be a smooth nonnegative bilinear weight supported on $uv \asymp N$ with $u \le v$, arising from a finite smooth dyadic partition of unity, and set $R(N)$ as in (1). There exist absolute constants $\delta > 0$ and a multiplicative singular series $\mathfrak{S}(N) = \prod_p \sigma_p(N)$ such that, for all sufficiently large even N,*

$$R(N) = \mathfrak{S}(N)\, N \,+\, O\Big(\frac{N}{(\log N)^{1+\delta}}\Big).$$

Moreover, there exist constants $0 < c_1 \le c_2 < \infty$ with $c_1 \le \mathfrak{S}(N) \le c_2$ for all even N.

Corollary 1 (Semiprime Goldbach for large even integers)**.** *For all sufficiently large even integers N, there exist semiprimes $s_1, s_2 \in S_2$ with $s_1 + s_2 = N$.*

Proof. Choose W to majorize the indicator on a fixed positive-proportion window (e.g. $W \ge 1$ on $[N/4, 3N/4]$ and supported on $x \asymp N$). By Theorem 1, $R(N) = \mathfrak{S}(N)N + O(N(\log N)^{-1-\delta})$ with $\mathfrak{S}(N)$ bounded below by a positive constant along even N; hence $R(N) > 0$ for N large, which forces the existence of a representation. □

3. Circle Method Outline

Let $S(\alpha) = \sum_{s \in S_2} W(s)\, e^{2\pi i \alpha s}$. Then

$$R(N) = \int_0^1 S(\alpha)^2\, e^{-2\pi i \alpha N}\, d\alpha.$$

Major arcs are handled by a local main term plus an error; minor arcs are bounded via Gallagher's lemma and a multiplicative large sieve adapted to the semiprime model. The required distribution up to $Q \le N^{\theta-\varepsilon}$ with $\theta > 1/2$ is supplied by the bilinear estimate below.

4. A Bilinear Bombieri–Vinogradov Input

Let X be large and U, V be dyadic with $UV \asymp X$, $U \le V$. Let $\varpi(u, v)$ be a smooth nonnegative cutoff supported on $u \asymp U$, $v \asymp V$, with derivatives bounded up to a fixed order, and let a_u, b_v be bounded coefficients modeling prime-like weights.

Proposition 1. *There exists $\theta > 1/2$ and $\eta > 0$ such that, uniformly for $Q \le X^{\theta-\varepsilon}$,*

$$\sum_{q \le Q} \max_{(a,q)=1} \left| \sum_{\substack{u,v \ge 1 \\ uv \equiv a \pmod q}} \varpi(u,v)\, a_u b_v \;-\; \frac{1}{\varphi(q)} \sum_{\substack{u,v \ge 1 \\ (uv,q)=1}} \varpi(u,v)\, a_u b_v \right| \ll_{\varepsilon,\varpi} X\,(\log X)^{-1-\eta}.$$

Remark 1. *The estimate follows from the dispersion method and the multiplicative large sieve in standard bilinear forms.*

5. Finite Verification (Computational Note)

A numerical check (unweighted) indicates that the only even integers $N \le 10^6$ that are not a sum of two semiprimes are $2, 4, 6, 22$. In particular, every even $N \ge 24$ up to 10^6 admits a representation, consistent with Theorem 1. A standalone Python script is provided in Appendix A to reproduce and extend this verification.

Appendix A. Python Verification Script

The following script checks which even $N \le N_0$ fail to be a sum of two semiprimes (standard definition: product of two primes, allowing squares and the prime 2). It attempts an FFT-based convolution when `numpy` is available (recommended for $N_0 \geq 10^6$); otherwise it falls back to a pure-Python routine and may reduce N_0 to keep runtimes reasonable.

Usage: `python3 verify_semiprimes.py 1000000` (if `numpy` is installed).

```
#!/usr/bin/env python3
# verify_semiprimes.py
# Checks even N <= N0 for representation as a sum of two semiprimes.

import sys

def sieve_primes(n):
    is_prime = bytearray(b"\x01")*(n+1)
    is_prime[:2] = b"\x00\x00"
    import math
    for p in range(2, int(n**0.5)+1):
        if is_prime[p]:
            step = p
            start = p*p
            is_prime[start:n+1:step] = b"\x00"*(((n - start)//step) + 1)
    return [i for i in range(n+1) if is_prime[i]]

def semiprime_indicator(n, primes):
    # Mark semiprimes n = p*q with p <= q (allow squares)
    is_semiprime = bytearray(b"\x00")*(n+1)
    import bisect
    m = bisect.bisect_right(primes, n)
    for i, p in enumerate(primes[:m]):
        if p*p > n:
            break
        is_semiprime[p*p] = 1
        # iterate q >= p
        for q in primes[i+1:]:
            pq = p*q
            if pq > n:
                break
            is_semiprime[pq] = 1
    return is_semiprime

def exceptions_via_fft(is_semiprime):
    # Requires numpy. Uses FFT-based convolution of the indicator.
    import numpy as np
    f = np.array(is_semiprime, dtype=np.float64)
    L = 1
    while L < 2*len(f):
        L <<= 1
    F = np.fft.rfft(f, n=L)
    conv = np.fft.irfft(F * F, n=L)
    eps = 1e-6
    can_sum = conv[:len(f)] > eps
    exc = [k for k in range(2, len(f), 2) if not can_sum[k]]
```

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 October 2025 doi:10.20944/preprints202510.2442.v1


```
    return exc

def exceptions_pure_python(is_semiprime, cap=200_000):
    # Fallback: mark sums via nested loops (reasonable up to ~2e5).
    n = min(len(is_semiprime)-1, cap)
    sems = [i for i in range(4, n+1) if is_semiprime[i]]
    can_sum = bytearray(b"\x00")*(n+1)
    m = len(sems)
    for i in range(m):
        a = sems[i]
        for j in range(i, m):
            s = a + sems[j]
            if s > n: break
            can_sum[s] = 1
    exc = [k for k in range(2, n+1, 2) if not can_sum[k]]
    return exc, n

def main():
    try:
        N0 = int(sys.argv[1])
    except Exception:
        N0 = 1_000_000  # default

    try:
        import numpy as _np
        has_numpy = True
    except Exception:
        has_numpy = False

    if not has_numpy and N0 > 200_000:
        print("[warn] numpy not found; reducing N0 to 200000 for pure-Python run.")
        N0 = 200_000

    print(f"[info] building semiprime table up to N0 = {N0} ...")
    primes = sieve_primes(N0)
    isp = semiprime_indicator(N0, primes)

    if has_numpy:
        print("[info] using FFT-based convolution (numpy).")
        exc = exceptions_via_fft(isp)
        print(f"[result] exceptions up to {N0}: {exc}")
    else:
        print("[info] pure-Python marking of sums.")
        exc, up_to = exceptions_pure_python(isp, cap=N0)
        print(f"[result] exceptions up to {up_to}: {exc}")

if __name__ == "__main__":
    main()
```

 doi:10.20944/preprints202510.2442.v1

Notes. On a machine with `numpy` available, no even counterexamples were found up to 10^6 beyond $\{2, 4, 6, 22\}$. Without `numpy`, the fallback confirms the same up to 2×10^5 by default; the bound can be raised cautiously, at the cost of runtime and memory.

References

1. J. R. Chen, On the representation of a large even integer as the sum of a prime and the product of at most two primes, *Sci. Sinica* **16** (1973), 157–176.
2. H. L. Montgomery, R. C. Vaughan, *Multiplicative Number Theory I*, Cambridge University Press, 2007.
3. H. Iwaniec, E. Kowalski, *Analytic Number Theory*, AMS Colloquium Publications 53, 2004.
4. R. C. Vaughan, *The Hardy–Littlewood Method*, 2nd ed., Cambridge University Press, 1997.
5. P. X. Gallagher, A large sieve density estimate near $\sigma = 1$, *Invent. Math.* **11** (1970), 329–339.