Journal of Integer Sequences, Vol. 28 (2025),
Article 25.3.1

On the Sum of a Squarefree Integer and a Power of Two

Christian Hercher
Institut für Mathematik
Europa-Universität Flensburg
Auf dem Campus 1c
24943 Flensburg
Germany
christian.hercher@uni-flensburg.de

Abstract

Erdős conjectured that every odd number greater than one can be expressed as the sum of a squarefree number and a power of two. Subsequently, Odlyzko and McCranie provided numerical verification of this conjecture up to 10^7 and $1.4 \cdot 10^9$. In this paper, we extend the verification to all odd integers up to 2^{50}, thereby improving the previous bound by a factor of more than $8 \cdot 10^5$. Our approach employs a highly parallelized algorithm implemented on a GPU, which significantly accelerates the process. We provide details of the algorithm and present novel heuristic computations and numerical findings, including the smallest odd numbers $< 2^{50}$ that require a higher power of two than all smaller ones in their representation.

1 Introduction

In 1950, Erdős [2] proved the existence of an infinite number of positive integers which cannot be expressed as the sum of a prime number and a power of two. Subsequently, on various occasions [3, 4], he relaxed the condition on the first summand and proposed the conjecture that every odd number greater than one can be expressed as the sum of a squarefree number and a power of two. Formally:

Conjecture 1 (Erdős)**.** Let $n > 1$ be an odd integer. Then there exists a squarefree integer m and an integer k such that

$$n = m + 2^k.$$

This conjecture is listed as Problem #11 on Bloom's list [1].

1.1 Computational verification and contributions from Odlyzko and McCranie

The initial significant computational inquiry into Erdős' conjecture was conducted by Odlyzko, who validated it for all odd integers up to 10^7, as noted by Granville and Soundararajan [5]. Guy [6, sec. A19] reported a further computational validation run up to $1.4 \cdot 10^9$ that was performed by McCranie. These validations furnished empirical corroboration for the conjecture, albeit with limitations due to the available computing resources at the time. Odlyzko's and McCranie's findings prompted subsequent endeavors to examine the conjecture for larger numbers, yet few such studies are documented.

1.2 Connection to Wieferich primes

In addition to computational efforts, the more theoretical work by Granville and Soundararajan [5] yielded a significant implication of Erdős' conjecture. They demonstrated that if the conjecture holds for all odd integers, there is an infinite number of primes which are not Wieferich primes—those primes p for which $2^p - 1 \equiv 1 \pmod{p^2}$. This connection between Erdős' conjecture and the modular properties of primes underscores the relevance of the conjecture to the distribution of primes and to modular arithmetic.

1.3 Contribution of this work

In this paper, we present an extended numerical verification of Erdős' conjecture for all odd integers up to $2^{50} > 10^{15}$, which exceeds the bounds set by Odlyzko and McCranie by a factor of over 10^8 and $8 \cdot 10^5$, respectively. A highly parallelized GPU-based algorithm was employed to efficiently sieve squarefree numbers and test representations for each odd n up to the stated limit. This approach paves the way for further computational studies and provides a more robust empirical foundation for Erdős' conjecture.

1.4 Outline

In Section 2, we present some heuristics to Erdős' conjecture. In particular, we determine the a priori probability that for a randomly selected odd integer $n > 1$, at least one of the integers $n - 2^1$, $n - 2^2$, $n - 2^3$, and so on up to $n - 2^\ell$ is squarefree. In Section 3, we discuss the algorithms used to compute the results presented in Section 4.

2 Heuristic arguments

Let $n > 1$ be a random odd integer, $2 = p_1 < p_2 < \cdots$ be the primes, and $A_{k,i}$ be the event $n \equiv 2^k \pmod{p_{i+1}^2}$. Furthermore, let $A_k = \bigcap_{i=1}^{\infty} A_{k,i}^C$, where A^C denotes the complement of the event A. Then A_k describes the event of $n - 2^k$ being squarefree. The objective is to compute the a priori probabilities $c_\ell := P\left(\bigcup_{k=1}^{\ell} A_k\right)$ of the events that at least one of the integers $n - 2^1, \ldots, n - 2^\ell$ is squarefree.

2.1 Computing c_ℓ for $\ell \leq 6$

From the principle of inclusion and exclusion, we can derive the following:

$$\begin{aligned}
c_\ell &= P\left(\bigcup_{k=1}^{\ell} A_k\right) \\
&= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} P\left(\bigcap_{k \in I} A_k\right) \\
&= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} P\left(\bigcap_{k \in I} \bigcap_{i=1}^{\infty} A_{k,i}^C\right) \\
&= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} P\left(\bigcap_{i=1}^{\infty} \left(\bigcap_{k \in I} A_{k,i}^C\right)\right) \\
&= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} P\left(\bigcap_{i=1}^{\infty} \left(\bigcup_{k \in I} A_{k,i}\right)^C\right).
\end{aligned}$$

If we assume, for the purposes of our heuristic, that congruences modulo different primes (and modulo powers of different primes) are stochastically independent, we further obtain the following:

$$\begin{aligned}
c_\ell &= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} \prod_{i=1}^{\infty} P\left(\left(\bigcup_{k \in I} A_{k,i}\right)^C\right) \\
&= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} \prod_{i=1}^{\infty} \left(1 - P\left(\bigcup_{k \in I} A_{k,i}\right)\right). \qquad (1)
\end{aligned}$$

Clearly, we have $P(A_{k,i}) = \frac{1}{p_{i+1}^2}$. Furthermore, if no two indices, $k_1 \neq k_2$, in I satisfy $2^{k_1} \equiv 2^{k_2} \pmod{p_{i+1}^2}$, then $P\left(\bigcup_{k\in I} A_{k,i}\right) = \frac{|I|}{p_{i+1}^2}$. This is true for all values of i, provided that $I \subseteq \{1, \ldots, 6\}$. Therefore, for $\ell \leq 6$, we can further simplify to

$$\begin{aligned} c_\ell &= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} \prod_{i=1}^{\infty} \left(1 - \frac{|I|}{p_{i+1}^2}\right) \\ &= \sum_{m=1}^{\ell} \binom{\ell}{m} \cdot (-1)^{m-1} \cdot \prod_{i=1}^{\infty} \left(1 - \frac{m}{p_{i+1}^2}\right). \end{aligned}$$

Once the products $d_m := \prod_{i=1}^{\infty} \left(1 - \frac{m}{p_{i+1}^2}\right)$ have been computed, numerical expressions for c_1 to c_6 are obtained, as presented in Table 1. These values agree with those presented by Granville and Soundararajan [5].

ℓ	d_ℓ	c_ℓ	$1 - c_\ell$
1	0.810569	0.810569	$1.894 \cdot 10^{-1}$
2	0.645268	0.975870	$2.412 \cdot 10^{-2}$
3	0.501948	0.997851	$2.148 \cdot 10^{-3}$
4	0.378599	0.999860	$1.390 \cdot 10^{-4}$
5	0.273345	0.999993	$6.852 \cdot 10^{-6}$
6	0.184435	0.999999	$2.669 \cdot 10^{-7}$

Table 1: Table of heuristically derived values for a priori probabilities of at least one of the integers $n - 2^1, \ldots, n - 2^\ell$ being squarefree.

2.2 Computing c_ℓ for $7 \leq \ell \leq 20$

For $\ell \geq 6$, this methodology is no longer viable. Given that $2^6 - 1 = 3^2 \cdot 7$, we have $n \equiv 2^{k+6} \pmod{3^2} \iff n \equiv 2^k \pmod{3^2}$; hence, $A_{k+6,1} = A_{k,1}$. With $2^{20} - 1 = 3 \cdot 5^2 \cdot 11 \cdot 31 \cdot 41$ being the next such case, we can conclude that for $7 \leq \ell \leq 20$ for all $i \geq 2$, it is $P\left(\bigcup_{k\in I} A_{k,i}\right) = \frac{|I|}{p_{i+1}^2}$, as previously demonstrated. However, for $i = 1$, we obtain $P\left(\bigcup_{k\in I} A_{k,1}\right) = \frac{|I \bmod 6|}{3^2}$, where $I \bmod 6 := \{k \bmod 6 \mid k \in I\}$. Using the abbreviation $d'_m := \prod_{i=2}^{\infty} \left(1 - \frac{m}{p_{i+1}^2}\right)$ and equation (1) for $\ell \leq 20$, we obtain the following result:

$$\begin{aligned} c_\ell &= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} \prod_{i=1}^{\infty} \left(1 - P\left(\bigcup_{k\in I} A_{k,i}\right)\right) \\ &= \sum_{\emptyset \neq I \subseteq \{1,\ldots,\ell\}} (-1)^{|I|-1} \cdot \left(1 - \frac{|I \bmod 6|}{9}\right) \cdot d'_{|I|}. \end{aligned} \tag{2}$$

2.3 Computing d'_k

It is well known that $\prod_p \left(1 - \frac{1}{p^2}\right) = \frac{1}{\zeta(2)} = \frac{6}{\pi^2}$. Therefore, we can conclude that $d_1 = \frac{8}{\pi^2}$ and $d'_1 = \frac{9}{\pi^2}$. For $k \geq 2$, it is necessary to compute the product d'_k numerically. In order to ensure good convergence, the following lemma should be observed.

Lemma 2. *Let $0 < x \leq \frac{1}{9}$ be a real number and $2 \leq k \leq 20$ be an integer with $kx \leq \frac{4}{5}$. Then*

$$\log(1 - kx) = k \cdot \log(1 - x) + \binom{k}{2} \log(1 - x^2) + e(x) \text{ with } |e(x)| < 18x^3.$$

Here and in the rest of the paper, log *denotes the natural logarithm.*

Proof. From the Taylor expansion of $\log(1 - x)$, we get the existence of $0 < \xi < x$ with

$$\log(1 - x) = -x - \frac{1}{2}x^2 - \frac{1}{3(1 - \xi^3)}x^3.$$

Thus,

$$-x - \frac{1}{2}x^2 > \log(1 - x) > -x - \frac{1}{2}x^2 - \frac{243}{728}x^3.$$

In the same way, we get the existence of another $0 < \xi < x$ with

$$\log(1 - x^2) = -x^2 - \frac{1}{2(1 - \xi^2)}x^2.$$

Hence,

$$\begin{aligned} -x^2 > \log(1 - x^2) &> -x^2 - \frac{81}{160}x^4 \\ &\geq -x^2 - \frac{9}{160}x^3. \end{aligned}$$

A third inequality of the same type is derived with

$$\begin{aligned} -kx - \frac{k^2}{2}x^2 > \log(1 - kx) &> -kx - \frac{k^2}{2}x^2 - \frac{125}{183}x^3 \\ &> -kx - \frac{k^2}{2}x^2 - x^3. \end{aligned}$$

By combining these inequalities, we obtain the following result:

$$k \cdot \log(1 - x) + \binom{k}{2} \log(1 - x^2) - x^3 < -kx - \frac{k^2}{2}x^2 - x^3 < \log(1 - kx)$$

and

$$\begin{aligned}
k \cdot \log(1-x) + \binom{k}{2} \log(1-x^2) &> -kx - \frac{k^2}{2}x^2 - \left(\frac{243}{728} \cdot k + \frac{9}{160} \cdot \binom{k}{2}\right) x^3 \\
&> \log(1-kx) - \left(\frac{243}{728} \cdot 20 + \frac{9 \cdot 190}{160}\right) x^3 \\
&> \log(1-kx) - 18x^3.
\end{aligned}$$

The combination of all the inequalities yields the desired result. □

Lemma 3. *Let* $2 \leq k \leq 20$ *and* $m > 5$ *be an integer. Moreover, for* $a \in \{2, 4\}$, *let* $P_{a,m}$ *denote the product* $P_{a,m} := \prod_{5 \leq p \leq m} \left(1 - \frac{1}{p^a}\right)^{-1}$, *where the product runs through all primes* $5 \leq p \leq m$. *Then*

$$d_k' = \left(\frac{9}{\pi^2}\right)^k \cdot \left(\frac{486}{5 \cdot \pi^4}\right)^{\binom{k}{2}} \cdot P_{2,m}^k \cdot P_{4,m}^{\binom{k}{2}} \cdot \prod_{5 \leq p \leq m} \left(1 - \frac{k}{p^2}\right) \cdot f, \text{ where } |1-f| < 4m^{-5}.$$

Proof. As previously indicated, $\prod_p \left(1 - \frac{1}{p^2}\right) = \frac{1}{\zeta(2)} = \frac{6}{\pi^2}$, and therefore, $\prod_{p \geq 5} \left(1 - \frac{1}{p^2}\right) = \frac{9}{\pi^2}$. Similarly, we have $\prod_p \left(1 - \frac{1}{p^4}\right) = \frac{1}{\zeta(4)} = \frac{90}{\pi^4}$ and $\prod_{p \geq 5} \left(1 - \frac{1}{p^4}\right) = \frac{90}{\pi^4} \cdot \frac{16}{15} \cdot \frac{81}{80} = \frac{486}{5 \cdot \pi^4}$. Using these identities and $d_k' = \prod_{p \geq 5} \left(1 - \frac{k}{p^2}\right)$, the equation we want to prove can be equivalently stated as follows:

$$\prod_{p>m} \left(1 - \frac{k}{p^2}\right) = \prod_{p>m} \left(1 - \frac{1}{p^2}\right)^k \cdot \prod_{p>m} \left(1 - \frac{1}{p^4}\right)^{\binom{k}{2}} \cdot f$$

or

$$\sum_{p>m} \log\left(1 - \frac{k}{p^2}\right) = \sum_{p>m} \log\left(k \cdot \left(1 - \frac{1}{p^2}\right) + \binom{k}{2} \cdot \log\left(1 - \frac{1}{p^4}\right)\right) + \log(f).$$

Given that $p > m > 5$, it follows that $0 < \frac{1}{p^2} \leq \frac{1}{25}$ and $k \cdot \frac{1}{p^2} \leq \frac{20}{25} = \frac{4}{5}$. Thus, for all $p > m$ from Lemma 2, we obtain

$$\log\left(1 - \frac{k}{p^2}\right) = k \cdot \log\left(1 - \frac{1}{p^2}\right) + \binom{k}{2} \log\left(1 - \frac{1}{p^4}\right) + e_p \text{ with } |e_p| < 18 \cdot p^{-6}.$$

The sum of these equations for all $p > m$ yields

$$\log(f) = \sum_{p>m} e_p;$$

hence

$$|\log(f)| \le \sum_{p>m} |e_p| < 18 \cdot \sum_{p>m} p^{-6} < \int_{x=m}^{\infty} x^{-6} \mathrm{d}x = \frac{18}{5} m^{-5} < 4m^{-5}.$$

Consequently, this results in the inequality $f > \exp\left(-4m^{-5}\right) > 1 - 4m^{-5}$. However, we observe that for the function $h(x) := \exp(x)$ it is $h(0) = 1$ and $h'(x) < \frac{10}{9}$ for all $0 < x < 0.1$. Thus, for such x we have $h(x) < h(0) + \frac{10}{9} \cdot x$. Since $m > 5$ it is $4m^{-5} < m^{-4} < \frac{4}{3125} < 0.1$. Therefore, $f < \exp\left(\frac{18}{5} m^{-5}\right) < 1 + \frac{10}{9} \cdot \frac{18}{5} m^{-5} = 1 + 4m^{-5}$. □

In order to achieve sufficient computational precision, a Sagemath worksheet was utilized, and the formulas presented in equation (2) and Lemma 3 were implemented. All numerical evaluations were conducted with 40 significant digits and an input value of $m = 10^7$. The results are presented in Table 2. Let $c_0 := 0$. Then, for $\ell \ge 1$, the difference $c_\ell - c_{\ell-1}$ provides the a priori probability that $n - 2^k$ is squarefree, while $n - 2^i$ is not squarefree for all integers $1 \le i < k$. The values of these differences $c_\ell - c_{\ell-1}$ are also presented in Table 2.

ℓ	$1 - c_\ell$	$c_\ell - c_{\ell-1}$
1	$1.8943 \cdot 10^{-1}$	$8.1056 \cdot 10^{-1}$
2	$2.4129 \cdot 10^{-2}$	$1.6530 \cdot 10^{-1}$
3	$2.1482 \cdot 10^{-3}$	$2.1980 \cdot 10^{-2}$
4	$1.3902 \cdot 10^{-4}$	$2.0092 \cdot 10^{-3}$
5	$6.8527 \cdot 10^{-6}$	$1.3216 \cdot 10^{-4}$
6	$2.6694 \cdot 10^{-7}$	$6.5857 \cdot 10^{-6}$
7	$4.9973 \cdot 10^{-8}$	$2.1696 \cdot 10^{-7}$
8	$2.5127 \cdot 10^{-9}$	$4.7460 \cdot 10^{-8}$
9	$8.5032 \cdot 10^{-11}$	$2.4277 \cdot 10^{-9}$
10	$2.2313 \cdot 10^{-12}$	$8.2801 \cdot 10^{-11}$
11	$4.8076 \cdot 10^{-14}$	$2.1832 \cdot 10^{-12}$
12	$8.7820 \cdot 10^{-16}$	$4.7197 \cdot 10^{-14}$
13	$1.5662 \cdot 10^{-16}$	$7.2190 \cdot 10^{-16}$
14	$4.0728 \cdot 10^{-18}$	$1.5222 \cdot 10^{-16}$
15	$7.3484 \cdot 10^{-20}$	$3.9994 \cdot 10^{-18}$
16	$1.0729 \cdot 10^{-21}$	$7.2411 \cdot 10^{-20}$
17	$1.3401 \cdot 10^{-23}$	$1.0595 \cdot 10^{-21}$
18	$1.4726 \cdot 10^{-25}$	$1.3254 \cdot 10^{-23}$
19	$2.5579 \cdot 10^{-26}$	$1.2168 \cdot 10^{-25}$
20	$4.2068 \cdot 10^{-28}$	$2.5158 \cdot 10^{-26}$

Table 2: Table of heuristically derived values for a priori probabilities of none of the numbers $n - 2^1, \ldots, n - 2^\ell$ being squarefree.

For the sake of simplicity, we may assume that in all cases, $n - 2^k$ is squarefree for a $1 \le k \le 21$. (Only in $< 5 \cdot 10^{-28}$ of cases is an exponent > 20 necessary.) This assumption allows us to conclude that the expected smallest needed exponent of $k = 1.215854247598$. Therefore, the expected sum of the smallest exponents required for all odd integers $1 < n < S$ is given by $E_S := 0.607927123799 \cdot S$. The standard deviation in a single instance is 0.476 and for the interval up to S, the standard deviation is given by $\sigma_S := 0.336 \cdot \sqrt{S}$.

Remark 4. If the congruences $n \equiv 2^k \pmod{p_{k+1}^2}$ for $1 \leq k \leq \ell$ are all true for an odd integer n, then the smallest exponent k for which $n - 2^k$ is squarefree is at least $\ell + 1$. The Chinese remainder theorem ensures the existence of such integers n. Consequently, arbitrarily large exponents are required when scanning through increasingly larger areas of the positive integers for n.

3 Algorithms

In designing and implementing the algorithms, we adhered to two fundamental principles. The first is that a given task should not be recalculated multiple times. Secondly, in order to utilize the parallelism that a modern GPU is capable of, it is necessary to avoid diverging branching as much as possible. This is because parallelism is achieved through a Same Instruction Multiple Data (SIMD) implementation. To adhere to the first principle, the sieving method was employed. This involves the identification of squarefree numbers within large intervals of numbers. To conform to the second principle, a parallel search was made for representations of numbers n as the sum of a squarefree number and a power of two.

3.1 The sieve to identify squarefree numbers

In the PC setup given to the author, the internal memory of the GPU was used to screen odd positive integers for squarefree numbers in intervals of length $2^{30} \approx 10^9$. The search was restricted to odd positive integers, as for odd integers n, the difference $n - 2^k$ is also odd. (The only exception would be for $k = 0$, which was excluded.) The identification of whether an integer in this interval is squarefree or not can be achieved with great efficiency by sieving.

Both the first **forall** loop in lines 1–3 and the inner one in lines 6–8 of Algorithm 1 each can be fully executed in parallel since the instructions are independent of one another. Moreover, there are no diverging branches. Accordingly, this task is optimally suited for the SIMD architecture of modern GPUs. If $\ell = \mathit{end} - \mathit{start}$ is defined as the length of the interval, then the first loop has a time complexity of $\mathcal{O}(\ell)$ and the second has a time complexity of

$$\sum_{3 \leq p < \sqrt{\mathit{end}}} \left(\mathcal{O}(1) + \mathcal{O}\left(\frac{\ell}{p^2}\right) \right) = \mathcal{O}\left(\pi(\sqrt{\mathit{end}})\right) + \mathcal{O}(\ell) = \mathcal{O}\left(\frac{\sqrt{\mathit{end}}}{\log(\mathit{end})} + \ell\right),$$

Input: *start* and *end* value of interval to screen
Output: for every odd n in $[start; end[$ the value $true/\ false$, whether n is squarefree or not

```
forall odd integers n in [start; end[ do
    Set is_squarefree(n) ← true;
end
forall primes 3 ≤ p < √end do
    find smallest odd m ≥ n with m ≡ 0 (mod p^2);
    forall 0 ≤ i ≤ ⌊(end−m)/(2·p^2)⌋ do
        Set is_squarefree(m + 2i · p^2) ← false;
    end
end
return is_squarefree;
```

Algorithm 1: Identifying all squarefree numbers by sieving.

where p always denotes a prime and

$$\frac{1}{9} \leq \sum_{3 \leq p < \sqrt{end}} \frac{1}{p^2} < \sum_{k=1}^{\infty} \frac{1}{k^2} = \zeta(2) = \frac{\pi^2}{6} < \infty.$$

This approach allows us to ascertain the status of each n in the specified interval, distinguishing between squarefree and non-squarefree elements in a time complexity of $\mathcal{O}\left(\frac{\sqrt{end}}{\log(end)} + \ell\right)$.

3.2 Finding the representation as the sum of a squarefree number and a power of two

For every odd n in a given interval, the smallest positive integer k can be identified which allows $n - 2^k$ to be squarefree. By restricting k to be positive, it follows that $n - 2^k$ is always odd.

Algorithm 2 employs the results from Algorithm 1, wherein the status of each odd integer is determined to be squarefree or not. Here the status is used of all odd integers within the same interval utilized in the computation of Algorithm 2, or in the preceding interval. This allows for the search for k up to $k_{\max} = \lfloor \log_2(\ell) \rfloor$. Nevertheless, during our computations, the scenario of all numbers $n - 2^k$ with $k \leq k_{\max}$ being non-squarefree did not occur on any occasion.

From the heuristic perspective, it is reasonable to expect that, on average, the **while** loop will be executed $\mathcal{O}(1)$ times for each value of n. This yields an expected time complexity of Algorithm 2 of $\mathcal{O}(\ell)$. Furthermore, as the computations for different values of n are independent of one another, they can be performed in parallel. The sole aspect of the

```
Input: start and end value of interval to screen; k_max as a search limit
Output: for every odd n in [start; end[ the smallest positive integer k with n − 2^k
        being squarefree
forall n in [start; end[ do
    Set k ← 1;
    while k ≤ k_max and not is_squarefree(n − 2^k) do
        Set k ← k + 1;
    end
    if k > k_max then no representation found
        print No representation found for n = s + 2^k with squarefree s and
          1 ≤ k ≤ k_max;
    end
    Set no_of_trials(n) ← k;
end
return no_of_trials;
```

Algorithm 2: Finding representation of $n = s + 2^k$ with squarefree s and smallest positive k.

algorithm that may potentially have diverging branches is the condition of the **while** loop: If some threads of a thread group have completed the loop because they have found a representation, they can only succeed if the other threads have also concluded their computation. Consequently, the computation within a thread group (typically comprising 32 threads in the CUDA architecture) can only proceed after the **while** loop has been completed by all of its threads. This results in the serialization of some parts of the computation. Nevertheless, in the given context, this does not significantly impede the computation, as in the majority of cases the loop is only executed a few times.

3.3 Collecting the results

The application of Algorithms 1 and 2 has enabled the discovery of a representation as $n = s + 2^k$ for every odd n in the current investigated interval, where s is a squarefree integer and k is the smallest positive integer for which $n - 2^k$ is squarefree. However, these data are physically present in the memory on the GPU and, thus, cannot be used directly in further computations on the CPU. But it is not necessary to have all of the details; an aggregate is sufficient. Consequently, this can be computed in parallel as well. Our primary interest lies in the maximal exponent k required for an odd integer n in the interval, as well as the overall sum of these exponents k needed for all odd n in the interval collectively. Both the maximum and the sum can be computed efficiently using the concept of parallel reduction [7]. This approach allows us to just transfer single variables, rather than entire arrays of data, thereby avoiding any potential bottlenecks due to data transfer between the GPU and

CPU.

4 Results

Utilizing the principles and algorithms discussed in Section 3, we developed a C++/CUDA program to identify the smallest positive integer k such that $n = s + 2^k$, where s is a squarefree integer and n is an odd integer with $1 < n < 2^{50}$. As a consequence of the fact that we found that $k \leq 13$ for each of the relevant n, we can conclude that:

Theorem 5. *Let $1 < n < 2^{50}$ be an odd integer. Then there exists a squarefree positive integer s and a positive integer $1 \leq k \leq 13$ with*

$$n = s + 2^k.$$

Consequently, Erdős' conjecture is now numerically verified to a significantly greater extent.

The computations were performed on a desktop PC with an Intel i9 CPU of the 11th generation and a Geforce RTX 3070 GPU from NVIDIA. The entire computation required approximately 120 hours.

During this process, we gathered a wealth of data, which led us to identify the smallest odd integers n for which $n - 2^k$ is not squarefree for all $1 \leq k \leq m$ with $m \leq 12$. These values are presented in Table 3 as well as in the OEIS sequence A377587.

m	smallest odd n
1	11
2	29
3	533
4	849
5	434977
6	10329791
7	28819433
8	129747557
9	6915752957
10	2569472629649
11	23373845739407
12	60690478781437

Table 3: Table of smallest odd n with $n - 2^1, \ldots, n - 2^m$ all not squarefree.

The sum of the smallest positive exponents k required for $n - 2^k$ to be squarefree for all odd $1 < n < 2^{50}$ is

$$k_{\text{sum}} = 684465092067182.$$

From Section 2, we would expect

$$\begin{aligned} E_{2^{50}} &= 0.607927123799 \cdot 2^{50} \\ &\approx 684465092052491. \end{aligned}$$

Accordingly, the difference between the observed and the heuristically derived values is merely

$$k_{\text{sum}} - E_{2^{50}} \approx 14690,$$

or about $1.3 \cdot 10^{-3}$ standard deviations. Consequently, the heuristic offers an exceptionally precise approximation.

For the interval $1 < n < 2^{30}$, each k was recorded as the smallest positive exponent with $n - 2^k$ being squarefree. The results are accompanied by the expected values of occurrences as calculated by the heuristics from Section 2, the standard deviations of these random experiments, and the difference between the real and expected values in absolute terms and relative to the standard deviations. These are presented in Table 4. In general, it can be observed that for small values of k the actual findings on the frequency of occurrence of each value of k are better described by the expected values derived from the heuristic approach than by the results of random experiments with the specified probabilities.

k	# n with smallest exp. k	expected value	σ	$\lvert\Delta\rvert$	$\lvert\Delta\rvert/\sigma$
1	435171212	435171170.1	9079.3	41.8	0.0046
2	88745588	88745444.2	8606.7	143.7	0.0167
3	11800589	11800957.9	3397.3	368.9	0.1085
4	1078868	1078702.6	1037.5	165.3	0.1593
5	71032	70958.0	266.3	73.9	0.2777
6	3473	3535.7	59.4	62.7	1.0546
7	122	116.4	10.7	5.5	0.5111
8	24	25.4	5.0	1.4	0.2932
9	3	1.3	1.1	1.7	1.4860
10	0	0.0	0.2	0.0	0.2108

Table 4: Table of statistics for the smallest exponents k with $n - 2^k$ being squarefree for all odd $1 < n < 2^{30}$.

References

[1] T. Bloom, https://www.erdosproblems.com/go_to/11 (2024).

[2] P. Erdős, On integers of the form $2^k + p$ and some related problems, *Summa. Brasil. Math.* **2** (1950), 113–123.

[3] P. Erdős and R. Graham, *Old and New Problems and Results in Combinatorial Number Theory*, Monographies de L'Enseignement Mathematique, 1980.

[4] P. Erdős, Problems in number theory, *New Zealand J. Math.* (1997), 155–160.

[5] A. Granville and K. Soundararajan, A binary additive problem of Erdős and the order of 2 mod p^2, *Ramanujan J.* **2** (1998), 283–298.

[6] R. K. Guy, *Unsolved Problems in Number Theory*, 3rd edition, Springer, 2004.

[7] M. Harris, Optimizing parallel reduction in CUDA, *Nvidia developer technology*, **4** (2007), 70. Available at `https://developer.download.nvidia.com/compute/DevZone/C/html/C/src/reduction/doc/reduction.pdf`.

2020 *Mathematics Subject Classification*: Primary 11A67; Secondary 11Y99.
Keywords: squarefree number, power of two, additive number theory.

(Concerned with sequence A377587.)

Received March 17 2025; revised versions received April 13 2025; April 14 2025. Published in *Journal of Integer Sequences*, April 14 2025.

Return to https://cs.uwaterloo.ca/journals/JIS/.