Factorials and Legendre’s three-square theorem: II

Rob Burns

31st March 2022

Abstract

Let $\bar{S}$ denote the set of integers $n$ such that $n!$ cannot be written as a sum of three squares. Let $\bar{S}(n)$ denote $\bar{S} \cap [1, n]$. We establish an exact formula for $\bar{S}(2^k)$ and show that $\bar{S}(n) = 1/8 * n + \mathcal{O}(\sqrt{n})$. We also list the lengths of gaps appearing in $\bar{S}$. We make use of the software package Walnut to establish these results.

1 Introduction

In a previous paper [1], we constructed an automaton accepting integers $n$ such that $n!$ cannot be written as a sum of three squares. In the current paper we use this automaton and the software package Walnut to explore the sets

$$S := \{ n : n! \text{ is a sum of three squares} \},$$

$$\bar{S} := \{ n : n! \text{ is not a sum of three squares} \},$$

$$S(n) := \{ x \leq n : x! \text{ is a sum of three squares} \}.$$

and

$$\bar{S}(n) := \{ x \leq n : x! \text{ is not a sum of three squares} \}.$$

In particular, we establish a formula for $S(2^k)$, or equivalently, $\bar{S}(2^k)$, and show that $\bar{S}(n) = 1/8 * n + \mathcal{O}(\sqrt{n})$. We also list the lengths of possible gaps in $S$ and its complement $\bar{S}$.

We mention here some similar previous results. In [2], Deshouillers and Luca established the two equivalent bounds

$$S(n) = 7n/8 + \mathcal{O}(n^{2/3}) \quad \text{and} \quad \bar{S}(n) = x/8 + \mathcal{O}(n^{2/3}).$$

This bound was improved by Hajdu and Papp [7] to

$$\bar{S}(n) = x/8 + \mathcal{O}(n^{1/2} \log^2 n).$$
Hajdu and Papp [7] proved that the largest gap between consecutive elements of $\bar{S}$ is 42. In theorem 3.1, we list the lengths of all gaps that occur between consecutive elements of $S$ and $\bar{S}$.

We make use of the software package Walnut. Hamoon Mousavi, who wrote the program, has provided an introductory article [8]. Papers that have used Walnut include [9], [12], [10], [5], [3]. Further resources related to Walnut can be found at Jeffrey Shallit’s page

https://www.cs.uwaterloo.ca/~shallit/papers.html.

The free open-source mathematics software system SageMath [14] was essential for the matrix algebra and factoring of polynomials.

In the next section, we describe the automaton that will be used to examine $\bar{S}$ and $\bar{S}(x)$.

2 The automaton

We first summarise the conditions satisfied by integers in $\bar{S}$. These come from [1] but were previously established by Granville and Zhu in 1990 [6]. Let $n \in \mathbb{N}$, with binary representation given by $n = \sum_{k \geq 0} a_k 2^k$, where all but finitely many $a_i$ are zero. Let $\gamma$ be the highest power of 2 dividing $n!$. For $i \in \{3, 5\}$, define $\alpha_i = \alpha_i(n)$ by

$$\alpha_3 = \alpha_3(n) := \# \left\{ k \geq 0 : \sum_{i=k}^{k+2} a_i 2^{i-k} \in \{3, 4\} \right\} \tag{1}$$

$$\alpha_5 = \alpha_5(n) := \# \left\{ k \geq 0 : \sum_{i=k}^{k+2} a_i 2^{i-k} \in \{5, 6\} \right\}. \tag{2}$$

**Theorem 2.1.** Let $n \in \mathbb{N}$ have binary representation $n = \sum_{k \geq 0} a_k 2^k$, where all but finitely many $a_i$ are zero. If $\gamma$ is the highest power of 2 dividing $n!$, then $n! = 2^\gamma Z$, where $Z$ satisfies

$$Z \equiv 3^{\alpha_3(n)}(-1)^{\alpha_5(n)} \pmod{8}$$

**Corollary 2.2.** If $n \in \mathbb{N}$, then $n!$ cannot be written as a sum of three squares if and only if $\gamma$ and $\alpha_3$ are even and $\alpha_5$ is odd.

According to the above two results, whether $n!$ can be written as a sum of three squares is determined by the triple $(\gamma \pmod{2}, \alpha_3 \pmod{2}, \alpha_5 \pmod{2})$. For convenience,
we will give this triple a name. For integers $n$, define $\Theta(n)$ by

$$\Theta(n) := (\gamma \pmod{2}, \alpha_3 \pmod{2}, \alpha_5 \pmod{2}).$$

The components of $\Theta(n)$ are

$$\Theta_1(n) := \gamma \pmod{2}, \quad \Theta_2(n) := \alpha_3 \pmod{2}, \quad \Theta_3(n) := \alpha_5 \pmod{2}. \quad (3)$$

Then, $n \in \bar{S}$ if and only if $\Theta(n) = (0, 0, 1)$.

We present here three automata which take as input the binary digits of an integer $n$, starting with the least significant digit. The automata were produced by Walnut based on the instructions provided in [1]. The first automaton, which we call gamma, is pictured in figure 1. It accepts integers $n$ if and only if $\gamma(n)$ is even. The second automaton, which we call a3, is pictured in figure 2. It accepts integers $n$ if and only if $\alpha_3(n)$ is even. The third automaton, which we call a5, is pictured in figure 3. It accepts integers $n$ if and only if the $\alpha_5(n)$ is even.

For any triple of binary digits $(x, y, z)$, we can use Walnut to create an automaton which accepts $n$ if and only if $\Theta(n) = (x, y, z)$. We are particularly interested in determining when $\Theta(n) = (0, 0, 1)$. We create this automaton using the Walnut command:

```
def factauto "?1sd_2 $gamma(n) & $a3(n) & ~$a5(n)";
```
Figure 3: Automaton for the parity of $\alpha_5$

Figure 4: Automaton accepting $n$ if $n!$ cannot be written as a sum of 3 squares
3 Gaps and runs in $S$ and $\bar{S}$

In this section we investigate the gaps and runs in the sets $S$ and $\bar{S}$. A gap $g$ exists in $S$ (or $\bar{S}$) if there are integers $x$ and $y$ which are consecutive elements of $S$ and such that $y - x = g$. A run of length $r$ exists in $S$ (or $\bar{S}$) if $S$ contains $x, x+1, \ldots, x + r - 1$ but $x - 1$ and $x + r$ are not in $S$. Since $S \cup \bar{S} = \mathbb{N}$, gaps in $S$ correspond to runs in $\bar{S}$ and vice versa. For example, if $S$ contains a run of size $r$, then $\bar{S}$ contains a gap of length $r + 1$.

In [7], Hajdu and Papp proved that the largest gap between consecutive elements of $\bar{S}$ is 42.

Theorem 3.1. The gap sizes in $S$ are $\{1, 2, 3, 4\}$. The gap sizes in $\bar{S}$ are $\{1, 2, 3, \ldots, 23, 25, 26, 28, 30, 31, 33, 34, 35, 37, 38, 42\}$.

Proof. We firstly deal with $\bar{S}$. Recall the automaton "factauto" which was defined previously. It accepts integers $n$ such that $n!$ cannot be written as a sum of three squares. We use Walnut to define an automaton, which we call "gaps":

```wsl
def gaps "$\text{lsd}_2 \ & \ \text{factauto}(n) \ & \ \text{factauto}(n+r) \ & \ (A_j \ (j < r-1) \ \Rightarrow \ \neg \ \text{factauto}(n+j+1))$":
```

This automaton takes as input the integer pair $(n,r)$ and reaches an accepting state if and only if $n \in \bar{S}$ and the next integer in $\bar{S}$ is $n + r$, i.e. there is a gap of length $r$ in $\bar{S}$ starting at $n$. Since the automaton has 319 states, we will not display it here. The following Walnut command creates an automaton which accepts an integer $r$ if and only if there is a gap of length $r$ somewhere in $\bar{S}$:

```wsl
eval tmp "$\text{lsd}_2 \ E \ n \ \text{gaps}(n, r)$":
```

This automaton is displayed in figure 5. A patient reader will see that the automaton accepts the integers listed in the theorem.

We can use the same approach to find all gaps in the set $S$. We define the automaton "sgaps" which takes as input the integer pair $(n,r)$ and reaches an accepting state if and only if $n \in S$ and the next integer in $S$ is $n + r$, i.e. there is a gap of length $r$ in $S$ starting at $n$.

```wsl
def sgaps "$\text{lsd}_2 \ \neg \ \text{factauto}(n) \ & \ \neg \ \text{factauto}(n+r) \ & \ (A_j \ (j < r-1) \ \Rightarrow \ \text{factauto}(n+j+1))$":
```

The resulting automaton has 33 states and is pictured in figure 4.
The resulting automaton has 203 states. The following Walnut command creates an automaton which accepts an integer \( r \) if and only if there is a gap of length \( r \) somewhere in \( S \):

\[
\text{eval tmp "?1sd_2 E n $sgaps(n, r)";}
\]

This automaton is displayed in figure 6. It is clear that it only accepts the integers 1, 2, 3, 4.

\[\square\]

Remarks. A gap of 42 in \( \tilde{S} \) commences at \( n = 23268 \) as mentioned by Hajdu and Papp. This is the second last gap to appear in the sequence \( \tilde{S} \). The last gap to appear in \( S \) is 33, which commences at \( n = 153828 = (10010110011100100)_2 \).

Theorem 3.1 is about gaps in the set \( \tilde{S} \) which contains integers \( n \) satisfying \( \Theta(n) = (0, 0, 1) \). The same method can be used for other sets defined in terms of \( \Theta \) (see (9) below). Hajdu and Papp showed that the maximum gap length in any of these sets is 42.

### 4 A formula for \( \tilde{S}(2^k) \)

In this section we establish a formula for \( \tilde{S}(2^k) \). The method we use is taken from the papers [4], [11], [13]. Firstly, we use Walnut to obtain a linear representation of \( \tilde{S}(n) \). Walnut produces matrices \( M_0 \) and \( M_1 \) and vectors \( v \) and \( w \) such that, if \( n \) has binary representation \( (n_0, n_1, \ldots, n_k) \) where \( n_0 \) is the least significant digit, then

\[
\tilde{S}(n) = v \cdot M_{n_0} \cdot M_{n_1} \cdots \cdot M_{n_k} \cdot w
\]
Figure 6: Automaton accepting gap lengths in $S$

In the particular case that $n = 2^k$, we have $\tilde{S}(2^k) = v \ast M_0^k \ast M_1 \ast w$. The theory of linear recurrences says that there are constants $\{c_i\}$ such that $\tilde{S}(2^k)$ can be written as

$$\tilde{S}(2^k) = \sum_i c_i \lambda_i^k$$

where $\{\lambda_i\}$ are the roots of the minimal polynomial of $M_0$. The constants $\{c_i\}$ can be determined by calculating $\tilde{S}(2^k)$ for enough values of $k$ and plugging the values into (5) to obtain a system of linear equations that can be solved.

**Theorem 4.1.** For $k \geq 2$,

$$\tilde{S}(2^k) = \begin{cases} 
2^{k-3} - 2^{(k-4)/2}, & \text{if } k \equiv \{0, 2, 22\} \pmod{24} \\
2^{k-3} - 2^{(k-3)/2}, & \text{if } k \equiv \{1, 3\} \pmod{24} \\
2^{k-3} - 2^{(k-5)/2}, & \text{if } k \equiv 11 \pmod{24} \\
2^{k-3} + 2^{(k-4)/2}, & \text{if } k \equiv \{14, 16, 18\} \pmod{24} \\
2^{k-3} + 2^{(k-3)/2}, & \text{if } k \equiv 17 \pmod{24} \\
2^{k-3} - 3 \ast 2^{(k-5)/2}, & \text{if } k \equiv 23 \pmod{24} \\
2^{k-3} & \text{otherwise}
\end{cases}$$

**Proof.** The Walnut command

\texttt{eval sumfact n "?lsd_2 (j>=1) & (j<=n) & $\$factauto(j)$";}

produces the matrices $M_0$ and $M_1$ and the vectors $v$ and $w$ mentioned above. Since the dimensions of these objects are large, we will not display them here. The minimal
polynomial of $M_0$ is
\[
h(x) = x^{20} - 7 \cdot x^{19} + 20 \cdot x^{18} - 30 \cdot x^{17} + 24 \cdot x^{16} - 40 \cdot x^{14} + 64 \cdot x^{13} - 128 \cdot x^{11} + 160 \cdot x^{10} - 384 \cdot x^8 + 960 \cdot x^7 - 1280 \cdot x^6 + 896 \cdot x^5 - 256 \cdot x^4
\]
\[
= (x - 2) \cdot (x - 1) \cdot (x - i - 1) \cdot (x + i - 1) \cdot (x - i \sqrt{2}) \cdot (x + i \sqrt{2}) \cdot (x + \sqrt{2}) \cdot (x + (-i/2 + 1/2) \cdot \sqrt{3} + i/2 - 1/2) \cdot (x - 1/2 \cdot \sqrt{2} \cdot \sqrt{3} + 1/2 \cdot \sqrt{2}) \cdot (x + 1/2 \cdot \sqrt{2} \cdot \sqrt{3} + 1/2 \cdot \sqrt{2}) \cdot (x + (i/2 + 1/2) \cdot \sqrt{3} - i/2 - 1/2) \cdot (x + (-i/2 - 1/2) \cdot \sqrt{3} - i/2 - 1/2) \cdot (x - 1/2 \cdot \sqrt{2} \cdot \sqrt{3} - 1/2 \cdot \sqrt{2}) \cdot (x + 1/2 \cdot \sqrt{2} \cdot \sqrt{3} - 1/2 \cdot \sqrt{2}) \cdot (x + (i/2 - 1/2) \cdot \sqrt{3} + i/2 - 1/2) \cdot (x - \sqrt{2}) \cdot x^4
\]

All roots of $h$, apart from 2 and 1, are of the form $\lambda_i = \sqrt{2} \cdot (a 24'th root of unity). We then have for $0 \leq s < 24$,
\[
\bar{S}(2^{24k+s}) = \sum_j c_j \lambda_j^{24k+s} = c_0 \cdot 2^{24k+s} + c_1 + \sum_{\lambda_j \neq 1,2} c_j \lambda_j^{24k+s}
\]
\[
=c_0 \cdot 2^{24k+s} + c_1 + 2^{12k} \cdot \sum_{\lambda_j \neq 1,2} c_j \lambda_j^s \tag{6}
\]
The values of the $\{c_j\}$ can be determined by calculating $\bar{S}(2^k)$ for enough $k$ using (4).
We create a matrix $A$ having the same dimension as the number of constants. The $j$’th row of $A$ is
\[
2^{j+2}, 1, \lambda_2^{j+2}, \lambda_3^{j+2}, \ldots
\]
Then the vector of constants is given by
\[
(c_0, c_1, \ldots) = A^{-1} \cdot (\bar{S}(2^2), \bar{S}(2^3), \ldots).
\]
The calculation gives $c_0 = 1/8$ and $c_1 = 0$. Since we are only interested in the values for $c_0$ and $c_1$, an alternative approach is to use (6) and the calculated values of, say, $\bar{S}(2^3) = 0, \bar{S}(2^{27}) = 16773120$ and $\bar{S}(2^{51}) = 281474959933440$ to create three linear equations.

The equations are:
\[
2^3 \cdot c_0 + c_1 + \tau = 0
\]
\[
2^{27} \cdot c_0 + c_1 + 2^{12} \cdot \tau = 16773120
\]
\[
2^{51} \cdot c_0 + c_1 + 2^{24} \cdot \tau = 281474959933440
\]
where $\tau = \sum_{\lambda_j \neq 1,2} c_j \lambda_j^3$. These can be solved to give $c_0 = 1/8$, $c_1 = 0$ and $\tau = -1$.
Plugging the values for $c_0$ and $c_1$ into (6), we see that, for each $0 \leq s < 24$, the terms $\sum_{\lambda_j \neq 1,2} c_j \lambda_j^s$ satisfy
\[
\sum_{\lambda_j \neq 1,2} c_j \lambda_j^s = \bar{S}(2^s) - 2^{s-3}
\]
As an example of the calculation that is required for each $s$, $\bar{S}(2^3) = 0$ so $\sum_{j \neq 1,2} c_j \lambda_j^3 = -1$. Hence, if $n \equiv 3 \pmod{24}$, then $\bar{S}(2^n) = 2^{n-3} - 2^{12s[n/24]} = 2^{n-3} - 2^{(n-3)/2}$. Here $\lfloor x \rfloor$ is the floor function. A similar calculation can be made for each $s$.

**Theorem 4.2.** For $k \geq 5$,

$$
\bar{S}(3 \ast 2^k) = \begin{cases} 
3 \ast 2^{k-3} + 1, & \text{if } k \equiv \{7, 9, 13, 18, 19\} \pmod{24} \\
3 \ast 2^{k-3} + 1 - 2^{(k-3)/2}, & \text{if } k \equiv \{1, 3, 5\} \pmod{24} \\
3 \ast 2^{k-3} + 1 - 2^{(k-2)/2}, & \text{if } k \equiv \{0, 4, 10\} \pmod{24} \\
3 \ast 2^{k-3} + 1 - 2^{(k-4)/2}, & \text{if } k \equiv \{2, 6, 8, 12\} \pmod{24} \\
3 \ast 2^{k-3} + 1 - 2^{(k-5)/2}, & \text{if } k \equiv \{11, 23\} \pmod{24} \\
3 \ast 2^{k-3} + 1 + 2^{(k-4)/2}, & \text{if } k \equiv 14 \pmod{24} \\
3 \ast 2^{k-3} + 1 + 2^{(k-1)/2}, & \text{if } k \equiv 15 \pmod{24} \\
3 \ast 2^{k-3} + 1 + 3 \ast 2^{(k-4)/2}, & \text{if } k \equiv 16 \pmod{24} \\
3 \ast 2^{k-3} + 1 + 2^{(k-3)/2}, & \text{if } k \equiv 17 \pmod{24} \\
3 \ast 2^{k-3} + 1 - 3 \ast 2^{(k-4)/2}, & \text{if } k \equiv \{20, 22\} \pmod{24} \\
3 \ast 2^{k-3} + 1 - 2^{(k-1)/2}, & \text{if } k \equiv 21 \pmod{24}
\end{cases}
$$

**Proof.** From (4),

$$
\bar{S}(3 \ast 2^k) = v \ast M_0^k \ast M_1 \ast M_1 \ast w
$$

We can therefore use the same approach as in theorem 4.1. We have,

$$
\bar{S}(3 \ast 2^k) = \sum_i c_i \lambda_i^k
$$

(7)

where $\{c_j\}$ are constants and $\{\lambda_i\}$ are the roots of the minimal polynomial of $M_0$, which we have already calculated. The constants $c_0$ and $c_1$ can be calculated using the values of $\bar{S}(3 \ast 2^3) = 3, \bar{S}(3 \ast 2^{27}) = 50327553$ and $\bar{S}(3 \ast 2^{51}) = 844424913354753$ to create three linear equations. The equations are:

$$
2^3 \ast c_0 + c_1 + \tau = 3
$$

$$
2^{27} \ast c_0 + c_1 + 2^{12} \ast \tau = 50327553
$$

$$
2^{51} \ast c_0 + c_1 + 2^{24} \ast \tau = 844424913354753
$$

where $\tau = \sum_{\lambda_j \neq 1,2} c_j \lambda_j^3$. These can be solved to give $c_0 = 3/8$, $c_1 = 0$ and $\tau = -1$. As before, we have for $0 \leq s < 24$,

$$
\bar{S}(3 \ast 2^{24k+s}) = \sum_j c_j \lambda_j^{24k+s} = 3/8 \ast 2^{24k+s} + 1 + \sum_{\lambda_j \neq 2} c_j \lambda_j^{24k+s}
$$

$$
= 3 \ast 2^{24k+s-3} + 1 + 2^{12k} \sum_{\lambda_j \neq 1,2} c_j \lambda_j^s
$$

(8)
The formula for $\bar{S}(3 \ast 2^k)$ can be deduced once (7) is used to calculate $\sum_{\lambda_j \neq 1, 2} c_j \lambda_j^s$ for each $0 \leq s < 24$.

5 A bound for $\bar{S}(n)$

In this section we establish a best possible bound on $\bar{S}(n)$. We borrow an argument from the paper by Hajdu and Papp [7].

We introduce the following sets for any triple $(x, y, z)$ of binary digits:

\[
\bar{S}(x, y, z) := \{ n : \Theta(n) = (x, y, z) \}, \quad \bar{S}(x, y, z)(n) := \{ r \leq n : \Theta(r) = (x, y, z) \}, \\
\bar{S}(x, y, z)(m, n) := \{ m < r \leq n : \Theta(r) = (x, y, z) \}.
\]  

Then $\bar{S}(0, 0, 1) = \bar{S}$ and $\bar{S}(0, 0, 1)(n) = \bar{S}(n)$.

Note that our set $\bar{S}(x, y, z)$ is equivalent to the set $H^{(\alpha, \beta)}$ used by Hajdu and Papp, where $\alpha = x$ and $\beta = 3^y(-1)^z$.

We note that the results from the previous section show that

\[
\bar{S}(2^k) = 1/8 * 2^k + O(\sqrt{2^k}) \quad \text{and} \quad \bar{S}(3 \ast 2^k) = 1/8(3 \ast 2^k) + O(\sqrt{2^k})
\]

The same bound can be shown to apply for other sets of numbers defined by $(\gamma, \alpha_3, \alpha_5)$. In summary, as for the case when $(x, y, z) = (0, 0, 1)$, the set $\bar{S}(x, y, z)(n)$ is accepted by an automaton constructed from the automata "gamma", "a3" and "a5" displayed earlier. For example, the set of $\{ n : \Theta(n) = (1, 1, 1) \}$ is accepted by the automaton defined by the Walnut command:

\[
\text{def factauto111 "?lsd_2 ~$gamma(n) & ~$a3(n) & ~$a5(n)":}
\]

As before, a linear representation of this automaton in terms of the matrices $M_0$ and $M_1$ and the vectors $v$ and $w$ comes from the command:

\[
\text{eval sumfact111 n "?lsd_2 (j>=1) & (j<=n) & $factauto111(j)":}
\]

For all choices of $(x, y, z)$, the matrix $M_0$ that Walnut produces is the same as the matrix $M_0$ that was produced by the automaton "sumfact" that was used in analysing the $\Theta(n) = (0, 0, 1)$ case. All the previous analysis related to the case $\Theta(n) = (0, 0, 1)$ therefore applies to the other choices of $(x, y, z)$. As a result, the same methods show that, for any $(x, y, z)$,

\[
\bar{S}(x, y, z)(2^k) = 1/8 * 2^k + O(\sqrt{2^k}) \quad \text{and} \quad \bar{S}(x, y, z)(3 \ast 2^k) = 1/8(3 \ast 2^k) + O(\sqrt{2^k}).
\]
It then follows that, for any choice of \((x, y, z)\) and \(k\),
\[
\bar{S}(x, y, z)(2^k, 2^{k+1}) = \frac{1}{8} \cdot 2^k + \mathcal{O}(\sqrt{2^k})
\]
\[
\bar{S}(x, y, z)(2^{k+1}, 3 \cdot 2^k) = \frac{1}{8} \cdot 2^k + \mathcal{O}(\sqrt{2^k})
\]
\[
\bar{S}(x, y, z)(3 \cdot 2^{k-2}, 2^k) = \frac{1}{8} \cdot 2^k + \mathcal{O}(\sqrt{2^k}).
\]

**Lemma 5.1** (Hajdu and Papp Lemma 3.1). For any integer \(k\) with \(k \geq 1\), we have \(\gamma(t \cdot 2^k + i) = \gamma(i) + \gamma(t \cdot 2^k) \pmod{2}\) for \(0 \leq i < 2^k\).

For the following lemma, we divide the region \([0, 2^s]\) into four equal sized subregions given by:

\[
\begin{align*}
I_1 : & \{i : 0 \leq i < 2^{s-2}\} \\
I_2 : & \{i : 2^{s-2} \leq i < 2^{s-1}\} \\
I_3 : & \{i : 2^{s-1} \leq i < 3 \cdot 2^{s-2}\} \\
I_4 : & \{i : 3 \cdot 2^{s-2} \leq i < 2^s\}
\end{align*}
\]

**Lemma 5.2** (Hajdu and Papp Lemma 3.3). Let \(k\) be an integer and \(t\) an odd integer with \(t \geq 1\). Then, for each \(j \in \{1, 2, 3, 4\}\), there exist numbers \(c(t, j) \in \{1, 3, 5, 7\}\) such that
\[
3^{\alpha_3(t \cdot 2^k + i)} \cdot (-1)^{\alpha_5(t \cdot 2^k + i)} = c(t, j) \cdot 3^{\alpha_3(i)} \cdot (-1)^{\alpha_5(i)} \pmod{8}
\]
for all \(i \in I_j\).

**Lemma 5.3.** For integers \(r, s\) and \(t\) satisfying \(0 \leq r < s\), \(0 \leq t < 2^s\), \(2 \nmid t\),
\[
\bar{S}(t \cdot 2^s, t \cdot 2^s + 2^r) = \frac{1}{8} \cdot 2^r + \mathcal{O}(\sqrt{2^r})
\]

*Proof.*
\[
\begin{align*}
\bar{S}(t \cdot 2^s, t \cdot 2^s + 2^r) &= \# \{i : 0 \leq i < 2^r, \Theta(t \cdot 2^s + i) = (0, 0, 1)\} \\
&= \sum_{j=1}^{4} \{i \in I_j : \Theta(t \cdot 2^s + i) = (0, 0, 1)\}
\end{align*}
\]
where \(I_j\) are the regions defined in (12). From lemma 5.1, if \(0 \leq i < 2^s\),
\[
\Theta_1(t \cdot 2^s + i) = 0 \iff \Theta_1(i) = -\Theta_1(t \cdot 2^s)
\]
From lemma 5.2, if \( i \in I_j \),
\[
(\Theta_2(t * 2^s + i), \Theta_3(t * 2^s + i)) = (0, 1)
\]
\[
\iff c(t, j) * 3^{\alpha_3(i)} * (-1)^{\alpha_5(i)} \equiv -1 \pmod{8}
\]
\[
\iff (\Theta_2(i), \Theta_3(i)) = (d_2(t, j), d_3(t, j))
\]
for some \( d_1, d_2 \in \{0, 1\} \). Hence,
\[
\tilde{S}(t * 2^s, t * 2^s + 2^r) = \sum_{j=1}^{4} \#\{i \in I_j : \Theta(i) = (-\Theta_1(t * 2^k), d_2(t, j), d_3(t, j))\}
\]
\[
= \tilde{S}_{(x,y,z)}(2^{r-2}, 2^r) + \tilde{S}_{(x,y,z)}(2^{r-2}, 2^{r-1}) + \tilde{S}_{(x,y,z)}(2^{r-1}, 3 * 2^{r-2}) + \tilde{S}_{(x,y,z)}(3 * 2^{r-2}, 2^r)
\]
\[
= 4 * (1/8 * 2^{r-2} + \mathcal{O}(\sqrt{2^r}))
\]
\[
= 1/8 * 2^r + \mathcal{O}(\sqrt{2^r}).
\]

where \((x, y, z)_j = (-\Theta_1(t * 2^k), d_2(t, j), d_3(t, j))\) for \( j \in \{1, 2, 3, 4\} \) and we have used (11).

The argument in the following theorem comes from theorem 2.3 in the paper by Hajdu and Papp.

**Theorem 5.4.** \( \tilde{S}(n) = 1/8 * n + \mathcal{O}(\sqrt{n}) \)

**Proof.** We first note that (10) implies that for any \((x, y, z)\),
\[
\tilde{S}_{(x,y,z)}(2^{k-2}, 2^{k-1}) = 1/8 * 2^{k-2} + \mathcal{O}(\sqrt{2^k})
\]
\[
\tilde{S}_{(x,y,z)}(2^{k-1}, 3 * 2^{k-2}) = 1/8 * 2^{k-2} + \mathcal{O}(\sqrt{2^k})
\]
\[
\tilde{S}_{(x,y,z)}(3 * 2^{k-2}, 2^k) = 1/8 * 2^{k-2} + \mathcal{O}(\sqrt{2^k}).
\]

Let \( n \) be an integer with binary representation \( n = \sum_{i} 2^{f_i} \), where \( f_1 > \cdots > f_j \geq 0 \). Then,
\[
\tilde{S}(n) = \tilde{S}(2^{f_1}) + \tilde{S}(2^{f_1}, 2^{f_1 + 2^{f_2}}) + \cdots + \tilde{S}(2^{f_1} + \cdots + 2^{f_j-1}, 2^{f_1} + \cdots + 2^{f_j})
\]

Any of the terms above can be written as
\[
\tilde{S}(2^{f_1} + \cdots + 2^{f_j-1}, 2^{f_1} + \cdots + 2^{f_j}) = \tilde{S}(t * 2^l, t * 2^h + 2^{l+1})
\]
where \( t < 2^l \) is odd. Since,
\[
\tilde{S}(t * 2^l, t * 2^h + 2^{l+1}) = 1/8 * 2^{h+1} + \mathcal{O}(\sqrt{2^{h+1}})
\]
by lemma 5.3, we have

\[ \bar{S}(n) = \sum_{i=1}^{j} 1/8 \ast 2^{f_i} + O(\sqrt{2}^{f_i}) \]

\[ = 1/8 \ast n + O((\sqrt{2}^{f_1+1} - 1)/(\sqrt{2} - 1)) \]

\[ = 1/8 \ast n + O(\sqrt{2}^{f_1}). \]

Since \( f_1 \leq \log_2(n) \) we have

\[ \bar{S}(n) = 1/8 \ast n + O(\sqrt{n}). \]

References

[1] Rob Burns. Factorials and Legendre’s three-square theorem. arXiv, 2021. 1, 2, 3

[2] Jean-Marc Deshouillers and Florian Luca. How often is \( n! \) a sum of three squares? In The Legacy of Alladi Ramakrishnan in the Mathematical Sciences, pages 243–251. Springer New York, 2010. 1

[3] Chen Fei Du, Hamoon Mousavi, Eric Rowland, Luke Schaeffer, and Jeffrey Shallit. Decision algorithms for fibonacci-automatic words, ii: Related sequences and avoidability. Theoretical Computer Science, 657:146–162, 2017. 2

[4] Chen Fei Du, Hamoon Mousavi, Luke Schaeffer, and Jeffrey Shallit. Decision algorithms for fibonacci-automatic words, III: Enumeration and abelian properties. International Journal of Foundations of Computer Science, 27(08):943–963, dec 2016. 6

[5] Daniel Goč, Hamoon Mousavi, and Jeffrey Shallit. On the number of unbordered factors. In Language and Automata Theory and Applications, pages 299–310. Springer Berlin Heidelberg, 2013. 2

[6] Andrew Granville and Yiliang Zhu. Representing binomial coefficients as sums of squares. The American Mathematical Monthly, 97(6):486–493, 1990. 2

[7] L. Hajdu and A. Papp. On asymptotic density properties of the sequence \( n! \). Acta Arithmetica, 184(4):317–340, 2018. 1, 2, 5, 10

[8] Hamoon Mousavi. Automatic theorem proving in Walnut. arXiv, 2016. 2

[9] Hamoon Mousavi, Luke Schaeffer, and Jeffrey Shallit. Decision algorithms for fibonacci-automatic words, i: Basic results. RAIRO Theor. Informatics Appl., 50:39–66, 2016. 2
[10] Narad Rampersad and Jeffrey Shallit. Congruence properties of combinatorial sequences via walnut and the rowland-yassawi-zeilberger automaton. arXiv, 2021. 2

[11] Jeffrey Shallit. Sumsets of wythoff sequences, fibonacci representation, and beyond. arXiv, 2020. 6

[12] Jeffrey Shallit. Frobenius numbers and automatic sequences. arXiv, 2021. 2

[13] Jeffrey Shallit. Intertwining of complementary thue-morse factors. arXiv, 2022. 6

[14] The Sage Developers. SageMath, the Sage Mathematics Software System (Version 8.2), 2018. https://www.sagemath.org. 2