

On the Number of Representations of One as the Sum of Unit Fractions

Matthew Brendan Crawford

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Masters of Science
in
Mathematics

Eyvindur Palsson, Chair
Daniel Orr
Samuel Mendelson

May 10, 2019
Blacksburg, Virginia

Keywords: Egyptian Fractions, Unit Fractions, Combinatorics
Copyright 2019, Matthew Brendan Crawford

On the Number of Representations of One as the Sum of Unit Fractions

Matthew Brendan Crawford

(ABSTRACT)

The Egyptian Fractions of One problem (EFO), asks the following question: Given a positive integer n , how many ways can 1 be expressed as the sum of n non-increasing unit fractions? In this paper, we verify a result concerning the EFO problem for $n = 8$, and show the computational complexity of the problem can be severely lessened by new theorems concerning the structure of solutions to the EFO problem.

On the Number of Representations of One as the Sum of Unit Fractions

Matthew Brendan Crawford

(GENERAL AUDIENCE ABSTRACT)

Expressing numbers as fractions has been the subject of one's education since antiquity. This paper shows how we can write the number 1 as the sum of uniquely behaved fractions called "unit fractions", that is, fractions with 1 in the numerator and some natural counting number in the denominator. Counting the number of ways this can be done reveals certain properties about the prime numbers, and how they interact with each other, as well as pushes the boundaries of computing power.

This Thesis is Dedicated to

My Soon-to-be Wife
My Family
And My Friends,

Who still don't know what I'm talking about,
But have supported, listened to, and **Loved** me just the same.

Acknowledgments

I would like to thank Dr. Eyvindur Palsson, my advisor, and Chair, for all of his help. I am especially thankful for his encouragement to undertake this problem that was not his own, for his patience, for his lighthearted attitude and non-overbearing demeanor, and for making sure I took responsibility. I would also like to thank Dr. Orr for serving on my committee, and Dr. Mendelson who has been one of the few to get down and dirty with me in these fractions, as well as for serving on my committee.

I would also like to recognize all of the other people who helped me finish this thing. Lizette Zietsman and Tia Chung for making sure I was able to begin this journey. Robert “Bob” Strand for humoring my computation questions. Shaun and Guy for having fun and playing with me. James Depret-Guillaume for (almost) never tiring talking about fractions with me, and for the last 3 (10) years together. Christian Kang and Nate Miller, whom(st) helped me code and talked about code and algorithms when I needed it. Parker Griffith and David Falcone whom(st) kept me company while I inattentively said “right...” while working into the wee hours of the morning. My parents who tried to understand what I was doing even when I didn’t feel like explaining. My parents in-law who asked me how it was going once every time I saw them, and knew that once was enough. Sam, whose 2 AM Reddit links were enough to get me to 3 AM. Abi Gopal and Bryan Kaperick, who, whether or not remember it, worked on this with me once freshman year. Thank You all for your support and your love.

And to Elyse, for knowing that the last 2 years would be hard, for always being there, always checking on me, and making sure I “feel good about it”, and getting me out of “the pits”, and knowing that sometimes I have to go do math while I have it in my head, and loving me incessantly ; I Love You 10 and can’t wait to spend the rest of my life with you.

Contents

Chapter 1: Introduction	1
Chapter 2: Historical Background and Previous Work	2
2.1 The “Egypt” in Egyptian Fraction	2
2.2 Greedy Mathematicians	3
2.3 Erdős and Co.	4
2.4 Early Computation	6
2.5 Modern Computation	7
Chapter 3: Structural Theorems	8
3.1 Trimming the Heads	8
3.2 Trimming the Tails	11
Chapter 4: Observations and Conjectures	13
4.1 Run Time and Estimation	13
4.2 Tails Per True Head	16
4.3 Analysis of Head Sums	19
4.3.1 The Sums Themselves	19
4.3.2 Factors of the Denominators of the Head Sums	20
4.4 Paralleliability	22
Chapter 5: Proofs of Theorems	23
Code Appendix	28
Bibliography	32

List of Figures

Figure 2.1	Rhind Papyrus	2
Figure 2.2	James Joseph Sylvester	3
Figure 2.3	The author has an Erdős (above) number of 3	6
Figure 4.1	Matlab Profile Summary of Runtime for $n = 7$	14
Figure 4.2	Distribution of Number of Tails per True Head for $n = 3$	17
Figure 4.3	Distribution of Number of Tails per True Head for $n = 4$	17
Figure 4.4	Distribution of Number of Tails per True Head for $n = 5$	17
Figure 4.5	Distribution of Number of Tails per True Head for $n = 6$	18
Figure 4.6	Distribution of Number of Tails per True Head for $n = 7$	18
Figure 4.7	True Head Sums	20
Figure 4.8	Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 3$	20
Figure 4.9	Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 4$	21
Figure 4.10	Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 5$	21
Figure 4.11	Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 6$	21
Figure 4.12	Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 7$	21

List of Tables

Table 2.1	Example Calculation of Sylvester's Greedy Algorithm	3
Table 3.1	Comparison of solution space sizes for various n between brute force and Theorem 3.1.1	9
Table 3.2	Comparison of solution space sizes for various n between Theorem 3.1.1 and Theorem 3.1.3	10
Table 3.3	Comparison of solution space sizes for various n between Theorem 3.1.3 and Theorem 3.1.6, needing only the possible Heads	10
Table 3.4	Comparison of solution space sizes for various n between Theorem 3.1.6 and Theorem 3.1.8	11
Table 3.5	Comparison of run times (in seconds) for various n between Theorem 3.1.8 and Theorem 3.2.1	12
Table 4.1	Comparison of solutions to 3.1 and number of Heads	15
Table 4.2	Comparison of solutions to 3.1 and number of True Heads	15
Table 4.3	Comparison of number of Heads and number of True Heads	15

Chapter 1

Introduction

An Egyptian Fraction is a sum of positive unit fractions. The Egyptians first did many calculations and kept records using these types of fractions, though the reason as to why is unclear. André Weil once stated their use of these fractions as “a wrong turn” [H].

Leonardo of Pisa, more commonly known as Fibonacci, proved that *any* positive fraction, is an Egyptian Fraction [H]. In 1880, Sylvester began doing work on “proper” Egyptian Fractions (fractions between 0 and 1)[SYL], and Erdős conjectured and proved many theorems regarding Egyptian Fractions [ET, EG, EN] including the famous $4/n$ conjecture with Straus [ES]. Graham and Erdős worked together on bounding the size of the denominators to represent any rational number as an Egyptian Fraction, as well as bounding the number of terms needed to represent any rational number as an Egyptian Fraction [RKG]. Croot proved several of Graham and Erdős’s unproven conjectures in 1994 related to distinct unit fractions [C]. The number of distinct ways (up to reordering) to write 1 as an Egyptian Fraction using n terms is what we will call the EFO problem for n terms. In 1972, Singmaster found the answers to the EFO problem for $n = 1$ through $n = 6$ [SING]. Erdős asked for an asymptotic formula following shortly thereafter.

In 1981, Guy’s famous *Unsolved Problems in Number Theory* stated the “...interest in Egyptian Fractions is as great as it has ever been,” due to the unsolved status of the simply stated problems [RKG]. In 1999, Paulhus confirmed McCranie’s answer to the EFO problem for $n = 7$ [A002966]. Due to the double-exponential growth of the possible denominators for any given solution to $\sum_{i=1}^n \frac{1}{x_i} = 1$ where $x_i \leq x_{i+1}$, the EFO problem and others like it become computationally unfeasible. In this paper, we confirm Dethridge and Le Normand’s result for $n = 8$ [A002966], and give our own conjecture for $n = 9$.

Chapter 2

Historical Background and Previous Work

2.1 The “Egypt” in Egyptian Fraction

The Rhind Papyrus and The Egyptian Leather Roll are two Egyptian artifacts containing 51 and 26 unit fraction series respectively, each of which is the expression of a fraction of the form $\frac{2}{p}$ or $\frac{2}{pq}$ and $\frac{1}{p}$ or $\frac{1}{pq}$ respectively [RP2, EMLR]. As an example of the method they used to determine these Egyptian Fraction representations, we will write $\frac{2}{21}$ as $\frac{2}{A} \frac{A}{21}$. We define $p = 3$ and $q = 7$.

$$\begin{aligned} \frac{2}{21} &= \frac{2}{A} \frac{A}{21} && A = p + 1 \\ &= \frac{2}{3+1} \left(\frac{3+1}{21} \right) = \frac{1}{2} \left(\frac{3}{21} + \frac{1}{21} \right) \\ &= \frac{1}{2} \left(\frac{1}{7} + \frac{1}{21} \right) \\ &= \frac{1}{14} + \frac{1}{42} \end{aligned}$$



Figure 2.1: The Rhind Papyrus, purchased by Alexander Rhind in 1858 is dated to around 1650 BCE [RP1]

2.2 Greedy Mathematicians

Leonardo of Pisa, better known by his pseudonym Fibonacci, proved that *any* rational number could be written as an Egyptian Fraction, using a greedy algorithm [LPF]. Sylvester rediscovered this fact and used the greedy algorithm to write specifically proper fractions as Egyptian Fractions. Here is an excerpt from his paper *On a Point in the Theory of Vulgar Fractions* [SYL].

“Let $\frac{4699}{7320}$ be the fraction to be expanded. The work may be arranged as follows:—

(2)	(8)	(60)	(3660)
4699	2078	1984	1920
7320	14640	117120	7027200
9398	16624	119040	7027200

Table 2.1: Example Calculation of Sylvester’s Greedy Algorithm

(2) is the number one unit greater than $E\frac{7320}{4699}$; 9398 is 2×4699 ; 2078 is $9398 - 7320$; 14640 is 2×7320 .

One element (2) is now determined, and the fraction $\frac{2078}{14640}$ remains to be expanded.

(8) is the number one unit greater than $E\frac{14640}{2078}$; 16624 is 8×2078 ; 1984 is $16624 - 14640$; 117120 is 8×14640 .

A second element (8) is now found, and $\frac{1984}{117120}$ remains over to be expanded. Proceeding in this manner, and with numerators 4699, 2078, 1984, 1920, necessarily diminishing at each step, we come at last to the element 3660 with a remainder zero. This requires sorites is therefore:

$$\frac{1}{2} + \frac{1}{8} + \frac{1}{60} + \frac{1}{3660}.”$$

We can see that Sylvester’s notation “ $E\frac{a}{b}$ ” means “the whole part of”, or what we would now denote as $\lfloor \frac{a}{b} \rfloor$, since $a > b > 0$. Therefore, he is saying to take the largest unit fraction that is smaller than the current fraction to be expanded — the greedy algorithm.



Figure 2.2: A Portrait of James Joseph Sylvester [JJS]

In the June issue of Scientific American in 1992, Ian Stewart wrote a probing article about the EFO problem, specifically for $n = 4$, but also the article entices the reader to consider $n = 5$, and even past. Stewart tells of a fabled dying war hero from the Middle East, who, on his death bed, recalls with his friend “a mathematical treatise, handed down, it is said, from the great Al-khwarizmi himself.” The treatise is the tale of a dying merchant who splits his property of 17 camels up amongst his sons using ratios, but in such a way that only when a wise man comes, on his own camel, to do the divvying, do the ratios make sense. $1/2$, $1/3$, and $1/9$ are the ratios the merchant decreed to give to his 3 sons by age, but the wise man’s 18th camel is what allows the oldest son to take away 9 camels instead of 8 and $1/2$, and the youngest son to take away 2 camels instead of 1 and $8/9$. The war hero and his friend ponder about whether there are any other such numbers that work in this manner, that is some a, b, c such that

$$\frac{1}{a} + \frac{1}{b} + \frac{1}{c} = \frac{d-1}{d}$$

for some other number d , where all the unknowns are positive integers (they are camels after all). The merchant notices the above equation is what we would now call the EFO problem for $n - 4$, by adding the $1/d$ over to the LHS of the equation. The merchant realizes there is a way to bound the ratios of camels for each of his own 3 sons, using a greedy algorithm; once the bound for the first son is found, the problem splits into different subcases and for each of those, the ratio of camels for the second son can be bounded, and so for the third son. After brute-forcing all of the subcases ($n = 4$ is small enough to brute force by hand), his friend exclaims to the war hero that he too can divvy his 39 camels to his 3 sons, that is if he procures 2 more camels, by using the ratios $1/2$, $1/3$ and $1/7$, leaving the 42nd camel from the wise man in the story. The war hero can die in peace now knowing that his sons will have their shares in such an excellent mathematical way — until a small boy rushes in and proclaims that the war hero has just had another son by his other wife! [IS] The story puts the reader into a situation where the EFO problem is applicable, just as long as one does not have more than a few sons.

2.3 Erdős and Co.

In his second paper, Erdős proves that any finite sum of unit fractions whose denominators have constant differences, cannot be an integer [ERD]. Graham summarizes: “The basic idea of the proof is that some term $a + kd$ is divisible by a higher power of some prime than any other terms. This follows from the analysis of the prime divisors of the expressions $\frac{(a+d)(a+2d)\dots(a+nd)}{n!}$ and $\binom{2n}{n}$,” [RLG]. In a paper with Niven in 1945, Erdős showed that no two partial sums of the harmonic series can be equal [EN]. Perhaps his most famous work dealing with Egyptian Fractions is in his 1950 paper defining $N(a, b)$, which is defined for integers $1 \leq a < b$ to be the least value n such that $\frac{a}{b} = \sum_{i=1}^n \frac{1}{x_i}$ has a solution with $0 < x_1 < \dots < x_n$, that is, $N(a, b)$ is the least such number of distinct unit fractions needed to sum to $\frac{a}{b}$. It is in this paper that the historical $4/n$ -conjecture first was mentioned, as Erdős claims $N(4, b) \leq 3$ for every $b > 2$ [ES]. This conjecture would prove to be one of the most famous of Erdős’,

not only due to its elementary looking Diophantine equation

$$\frac{4}{n} = \frac{1}{a} + \frac{1}{b} + \frac{1}{c}, \text{ for } n, a, b, c \in \mathbb{N},$$

but also its difficulty. Even to this day, there are new contributions to proving the truth of the conjecture [X1]. Extensive computer searches have verified the conjecture to be true for $n \leq 10^{17}$ [X2]. Generalizations have been made for $5/n$ [RKG], k/n [RKG], and $3/n$ using only odd denominators [HAG].

Erdős once conjectured that for distinct denominators $\frac{x_n}{x_1} \geq 3$, and that $\lim_{n \rightarrow \infty} \frac{x_n}{x_1} = \infty$, but this was proved incorrect by Croot in 1999 [C].

Croot showed that for any rational number $r > 0$, and for all $N > 1$, there exist integers x_1, x_2, \dots, x_k , where

$$N < x_1 < x_2 < \dots < x_k \leq \left(e^r + O_r \left(\frac{\log \log N}{\log N} \right) \right) N$$

such that

$$r = \frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_k}.$$

He did this through the conjunctive use of two propositions. He begins by letting $c > 1$ be the smallest real number such that

$$r \leq \sum_{N < n < cN} \frac{1}{n} \leq r + \frac{1}{cN}.$$

And using the fact that $\sum_{1 \leq n \leq t} 1/n = \log t + \gamma + O(1/t)$, where γ is the Euler- Mascheroni Constant, it can be shown that $c = e^r + O_r(1/N)$. He defines

$$\frac{u}{v} = \sum_{N < n < cN} \frac{1}{n}, \text{ where } \gcd(u, v) = 1.$$

The first proposition he uses states that if some fractions can be removed from the sum in the definition of u/v , leaving u'/v' , then all the prime power factors of v' are $\leq \sqrt[5]{N}$, and moreover that

$$\frac{\log \log N}{\log N} \ll_r \sum_{1 \leq i \leq k} \frac{1}{n_i} \ll_r \frac{\log \log N}{\log N},$$

where $u'/v' + \sum_{1 \leq i \leq k} \frac{1}{n_i} = u/v$. His second proposition states that if s is some rational number whose denominator has all prime power factors $\leq M^{1/4-\epsilon}$, and if $s \gg \log \log \log M / \log M$, then there exist integers $M < m_1 < \dots < m_l < e^{(v(\epsilon)+o(1))s} M$, where $v(\epsilon)$ is some constant depending on ϵ , such that

$$s = \sum_{1 \leq i \leq l} \frac{1}{m_i}.$$

He combines these two propositions by letting $M = cN$ and $s = r - u'/v'$. Therefore we have that

$$\frac{\log \log M}{\log M} \ll_r s \ll_r \frac{\log \log M}{\log M},$$

and all of the prime power factors of the denominator of s will be less than or equal to $\sqrt[5]{N}$. This meets the hypothesis of the second proposition and therefore there exist the m_i from above such that

$$r = s + \frac{u'}{v'} = \sum_{\substack{N < n < cN \\ n \neq n_1, \dots, n_k}} \frac{1}{n} + \sum_{1 \leq i \leq l} \frac{1}{m_i}.$$

All of the denominators of the unit fractions above will be between N and $\left(e^r + O_r\left(\frac{\log \log N}{\log N}\right)\right)N$. This shows that the ratio between the last and first denominators in any Egyptian Fraction representation of 1 with distinct denominators (which are a subset of all solutions counted by the EFO problem), is bounded, as are the ratios between any particular denominator and the first denominator.



Figure 2.3: The author has an Erdős (above) number of 3 [PEP, BC]

2.4 Early Computation

Singmaster used a computer in the early 70's to compute $EFO(n)$ for $n = 1$ through $n = 6$, but could not find a formula or asymptotic formula for $EFO(n)$. In his unpublished manuscript on the subject, he also wrote a note at the bottom questioning whether $ENO(n)$ could be related to the Carmichael numbers, and wrote a reminder to himself to write R.K. Guy regarding his knowledge on the subject [SING]. Dr. Robert G. Wilson sent a fax to Dr. Neil J.A. Sloane (the same of OEIS), with exactly the article from Scientific American about the camels, but also in the cover page of the fax, he has listed the first 7 values of $EFO(n)$ as 1, 1, 3, 14, 147, 3462, 294392, which is correct until $n = 7$ which we know is truly 294314. He also includes at the end of his fax, two different codes, each in a different language, to brute force the EFO problem, for $n = 7$ [RW].

2.5 Modern Computation

Other than references on The Online Encyclopedia of Integer Sequences' entry for $EFO(n)$, Paulhus' confirmation of McCranie's value for $n = 7$, and Dethridge and Le Normand's conjectured value for $n = 8$, there are no other citations of these spectacular feats of computation [A002966]. This paper will discuss theoretical and computational improvements on the calculation of $EFO(n)$ specifically for $n = 8$, conjecture a value for $n = 9$, as well as comment on the asymptotic nature of $EFO(n)$ for arbitrarily large n .

Chapter 3

Structural Theorems

The equation describing the topic is the EFO equation for a given n :

$$1 = \sum_{i=1}^n \frac{1}{x_i}, \quad x_i \leq x_{i+1}, \quad x_i \in \mathbb{N} \text{ for } 1 \leq i \leq n. \quad (3.1)$$

The sequence $s_1 = 2$ and $s_n = s_{n-1}^2 - s_{n-1} + 1$, known as Sylvester's Sequence [A000058], gives rise to a sequence of bounds, $b_n = s_n - 1$ for $1 \leq n$, which are attainable upper bounds for the denominators used in 3.1 with n terms. Since we require $x_i \leq x_{i+1}$ for all i , b_n also bounds every denominator in the sum in 3.1. Thus for a given n , a very large solution space for all of the solutions to the EFO equation is $[1, b_n]^n$. Vardi showed that there exists a constant $E \approx 1.2640847353\dots$ [A076393] such that $s_n = \lfloor E^{2^{n+1}} + \frac{1}{2} \rfloor$. Therefore we see that b_n is determined by a double exponential function, and grows extremely quickly.

3.1 Trimming the Heads

Theorem 3.1.1. *Let $3 \leq n \in \mathbb{N}$. Let $2 \leq i \leq n - 1$. Then:*

$$2 \leq x_1 \leq n \quad (3.2)$$

$$i + 1 \leq x_i \leq (n - i + 1)b_i \quad (3.3)$$

$$n \leq x_n \leq b_n \quad (3.4)$$

Theorem 3.1.1 greatly decreases the solution space from $[1, b_n]^n$ to $[2, n] \times \times_{i=2}^{n-1} [i + 1, (n - i + 1)b_i] \times [n, b_n]$.

Example 3.1.2. *All of the inequalities in Theorem 3.1.1 are sharp. We will use the case of $n = 5$ and $i = 3$ for the following examples when necessary.*

The first string of inequalities in Theorem 3.1.1 can be shown to be equalities when we take the "Sylvester solution", that is $x = (s_1, s_2, s_3, s_4, b_5) = (2, 3, 7, 42, 1806)$, and when we take the "Constant solution", that is $x = (5, 5, 5, 5, 5)$.

The second string of inequalities can be shown to be equalities when we set the first i terms

to be $i+1$, and distribute the remaining “weight” equally: for example when $x = (4, 4, 4, 8, 8)$, and when we take the first $i-1$ terms to be the respective Sylvester numbers, and split the remaining weight equally: for example when $x = (2, 3, 18, 18, 18)$.

The last string of inequalities can be shown to be equalities when we switch the examples used for the first string of inequalities, that is when $x = (5, 5, 5, 5, 5)$ and when $x = (2, 3, 7, 43, 1806)$.

Let $A_n = |[1, b_n]^n|$ and let $B_n = |[2, n] \times \times_{i=2}^{n-1} [i+1, (n-i+1)b_i] \times [n, b_n]|$.

n	A_n	B_n	B_n/A_n
3	216	16	.0741
4	3.1117×10^6	4.212×10^3	1.354×10^{-3}
5	1.9213×10^{16}	5.1898×10^7	2.7012×10^{-9}
6	1.2080×10^{39}	1.2063×10^{15}	9.9863×10^{-25}
7	1.5540×10^{91}	9.9966×10^{28}	6.4327×10^{-63}
8	2.7392×10^{208}	9.7503×10^{55}	3.5595×10^{-153}

Table 3.1: Comparison of solution space sizes for various n between brute force and Theorem 3.1.1

We may also realize that for a fixed n , and a particular choice of (x_1, \dots, x_{n-1}) , the value of x_n is fixed.

Theorem 3.1.3. *Let $n \in \mathbb{N}$, and let $x = (x_1, \dots, x_n)$ satisfy 3.1. Then*

$$x_n = \frac{\prod_{i=1}^{n-1} x_i}{\prod_{i=1}^{n-1} x_i - \sum_{j=1}^{n-1} x_j^\dagger},$$

where x_j^\dagger is the product of the first $n-1$ x_i 's excluding x_j .

Thus we need only check through the first $n-1$ components, and check whether or not $\prod_{i=1}^{n-1} x_i - \sum_{j=1}^{n-1} x_j^\dagger$ divides $\prod_{i=1}^{n-1} x_i$.

Example 3.1.4. *If we let $n = 4$, and we let $x_1 = 2$, $x_2 = 3$, $x_3 = 11$, then by Theorem 3.1.3 $x_4 = \frac{66}{66-61} = \frac{66}{5}$. But $\frac{66}{5} \notin \mathbb{N}$. So while $(2, 3, 11, \frac{66}{5})$ has components who are within the bounds of Theorem 3.1.1, we see it is not a solution.*

Let B_n be as before, and let $C_n = |[2, n] \times \times_{i=2}^{n-1} [i+1, (n-i+1)b_i]|$.

Definition 3.1.5. *Let $p, q \in \mathbb{N}$. We define $p \perp q$ to mean p is relatively prime to q , that is $\gcd(p, q) = 1$.*

With the following theorem, we can still again severely cut down the solution space.

Theorem 3.1.6. *Let $p, q \in \mathbb{N}$ such that $p \perp q$. Then $\frac{p}{q}$ can be written as:*

$$\frac{p}{q} = \frac{1}{\alpha} + \frac{1}{\beta}, \tag{3.5}$$

for $\alpha, \beta \in \mathbb{N}$ if and only if there exists $a, b \in \mathbb{N}$ where $a|q$ and $b|q$ such that $p|(a+b)$.

n	B_n	C_n	C_n/B_n
3	16	4	.25
4	4.212×10^3	1.08×10^2	2.5641×10^{-2}
5	5.1898×10^7	2.88×10^4	5.54939×10^4
6	1.2063×10^{15}	3.69645×10^8	3.06425×10^{-7}
7	9.9966×10^{28}	9.38647×10^{15}	9.38962×10^{-14}
8	9.7503×10^{55}	8.59639×10^{29}	8.8165×10^{-27}

Table 3.2: Comparison of solution space sizes for various n between Theorem 3.1.1 and Theorem 3.1.3

Theorem 3.1.6 tells us that we need only search through the first $n - 2$ components, and then from those, we may compute how many pairs $(\alpha, \beta) \in \mathbb{N}^2$ satisfy 3.1.6.

Definition 3.1.7. Let $3 \leq n \in \mathbb{N}$, and let $x = (x_i)_{1 \leq i \leq n}$ be a solution to 3.1. We define the Head of x to be the first $n - 2$ components of x . That is $\text{Head}(x) = (x_i)_{1 \leq i \leq n-2}$. Similarly, we define the Tail of x to be the last 2 components of x . That is $\text{Tail}(x) = (x_i)_{n-1 \leq i \leq n}$.

Let C_n be as before and $D_n = \left| [2, n] \times \prod_{i=2}^{n-2} [i+1, (n-i+1)b_i] \right|$.

n	C_n	D_n	D_n/C_n
3	4	2	.125
4	1.08×10^2	1.2×10^1	1.1111×10^{-1}
5	2.88×10^4	3.6×10^2	1.25×10^{-2}
6	3.69645×10^8	1.0248×10^5	2.77239×10^{-4}
7	9.38647×10^{15}	1.4381×10^9	1.53213×10^{-7}
8	8.59639×10^{29}	4.0358×10^{16}	4.69481×10^{-14}

Table 3.3: Comparison of solution space sizes for various n between Theorem 3.1.3 and Theorem 3.1.6, needing only the possible Heads

Using a greedy algorithm, we can yet again cut down the solution space:

Theorem 3.1.8. Let $3 \leq n \in \mathbb{N}$, and let $1 \leq i \leq n - 2$. Let $(x_k)_{1 \leq k \leq i}$ be the first i denominators of a possible solution to 3.1. Let $\frac{p_i}{q_i} = 1 - \sum_{j=1}^i \frac{1}{x_j}$ be such that $p_i \perp q_i$.

Let $w_{i+1} = \left\lfloor \frac{q_i}{p_i} \right\rfloor + 1$. Let $u_{i+1} = \left\lfloor \frac{(n-i)q_i}{p_i} \right\rfloor$. Let $\ell_{i+1} = \begin{cases} w_{i+1} & q_i \geq p_i x_i \\ x_i & q_i < p_i x_i \end{cases}$.

Then $\ell_{i+1} \leq x_{i+1} \leq u_{i+1}$.

Example 3.1.9. Above, w_{i+1} is our modern notation for exactly what Sylvester was doing in 1880, that is, taking the biggest unit fraction less than the remainder that we have to sum to. u_{i+1} is like what we were doing with the upper bounds in Theorem 3.1.1: distributing the

weight of the remaining amount we must sum to across the remaining components. And l_{i+1} must be done in cases because sometimes, the greedy algorithm contradicts the non-decreasing requirement of 3.1. For example, when $n = 4$, and $x_1 = 2$, $x_2 = 5$, we have $\frac{p_i}{q_i} = 3/10$, and $w_{i+1} = 4$, but if we took x_3 to be 4, then we would break the nondecreasing requirement of 3.1, so we must take $x_3 = x_2 = 5$.

Let E_n be the size of solution space using Theorem 3.1.8 while still searching through only the Heads of the possible solutions, and let D_n be as before.

n	D_n	E_n	E_n/D_n
3	2	2	1
4	12	7	5.8333×10^{-1}
5	3.6×10^2	5.1×10^1	1.4167×10^{-1}
6	1.0248×10^5	8.35×10^2	8.1479×10^{-3}
7	1.4381×10^9	6.6162×10^4	4.6006×10^{-5}
8	4.0358×10^{16}	1.5256×10^8	3.7802×10^{-9}

Table 3.4: Comparison of solution space sizes for various n between Theorem 3.1.6 and Theorem 3.1.8

3.2 Trimming the Tails

Theorem 3.2.1. Let $(x_i)_{1 \leq i \leq n-2}$ be a Head of a possible solution to 3.1. Let $p \in \mathbb{N}$ and $q \in \mathbb{N}$ be such that $p \perp q$ and

$$\frac{p}{q} = 1 - \sum_{i=1}^{n-2} \frac{1}{x_i}.$$

Then the only possible pairs of divisors of q that need to be checked are (a, b) that satisfy Theorem 3.1.6 such that

$$\frac{b}{a} \leq \frac{q}{px_{n-2} - q}. \quad (3.6)$$

If $px_{n-2} - q \leq 0$, then all pairs of divisors (a, b) that satisfy 3.1.6 give rise to possible solutions to 3.1.

Let E_n^t be the time it takes to find all solutions using all of the above theorems through Theorem 3.1.8. Let F_n^t be the time it takes to find all solutions using all of the above theorems including Theorem 3.2.1. Note that these run times are the average run time from 100,000 runs, unless marked with an *, which are over 10,000 trials, ** which are over 1,000 trials, and $n = 8$ is over 1 trial which is marked with ***.

n	E_n^t	F_n^t	F_n^t/E_n^t
3	8.02308×10^{-4}	1.75894×10^{-4}	2.19235×10^{-1}
4	2.59997×10^{-3}	1.03656×10^{-3}	3.98682×10^{-1}
5*	2.23508×10^{-2}	1.41102×10^{-2}	6.31306×10^{-1}
6**	1.92233×10^{-1}	1.76185×10^{-1}	9.16518×10^{-1}
7**	2.31190×10^1	2.08543×10^1	9.02042×10^{-1}
8***	9.36100×10^5	8.96622×10^5	9.57827×10^{-1}

Table 3.5: Comparison of run times (in seconds) for various n between Theorem 3.1.8 and Theorem 3.2.1

Chapter 4

Observations and Conjectures

4.1 Run Time and Estimation

The EFO problem, and others similar to it (such as considering strictly increasing denominators, or considering solutions with distinct orderings of the denominators as distinct) are Combinatoric and Number Theoretic, however, at this moment, advances in these problems are highly computational.

Upon running the code used to calculate the solution to the EFO problem for $n = 1$ through $n = 8$, we found the most time consuming and computationally intensive parts of the code were:

1. Computing the prime factorization of q_{n-2} as defined in Theorem 3.1.8, as is necessary to build the divisors used in Theorem 3.1.6. It was found that on average, building the prime factorization of q_{n-2} took approximately slightly more than 25% of the run time.
2. Calculating p_{n-2} and q_{n-2} as defined in Theorem 3.1.8 by calculating the gcd of an equivalent fraction. It was found that on average, calculating this gcd took approximately slightly less than 25% of the run time.
3. Calculating the divisors of q_{n-2} as is necessary in Theorem 3.1.6 from the prime factorization of q_{n-2} . It was found that on average, calculating the divisors from the prime factorization took approximately 14% of the run time.

Profile Summary

Generated 03-Apr-2019 23:57:46 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
tester2	1	24.999 s	0.376 s	
comb	129657	22.992 s	4.340 s	
gcd	159051	6.154 s	6.154 s	
mydivisorsordering	66162	3.512 s	1.387 s	
instances	66162	3.234 s	0.863 s	
factor	66162	3.093 s	1.818 s	
nchoose2	66162	2.660 s	2.660 s	
unique	66162	2.371 s	1.128 s	
kron	201182	2.125 s	2.125 s	
cyclicsum	66162	1.631 s	1.631 s	
primes	66162	1.275 s	1.275 s	
unique>uniqueR2012a	66162	1.243 s	1.243 s	

Self time is the time spent in a function excluding the time spent in its child functions. Self time also includes overhead resulting from the process of profiling.

Figure 4.1: Matlab Profile Summary of Runtime for $n = 7$. Calculating the prime factorization of q_{n-2} is the sum of the times for “instances” and “factor”. Calculating the gcd of p_{n-2} and q_{n-2} is “gcd”. Calculating the divisors of q_{n-2} is “mydivisorsordering”.

With this being said, computation could be sped up by taking advantage of the knowledge that for each $\text{Head}(x)$ searched through, where $x = (x_1, \dots, x_{n-2})$, q_{n-2} is some divisor of the product of the components of x , and hence will have some of the prime factors of the x_i 's, which are usually much smaller than q_{n-2} , and therefore easier to factor historically.

Counting the number of Heads that possibly give rise to a solution to 3.1 itself takes time, and we have calculated the number of Heads for $n = 9$ to be 713,922,875,728,745. For each one of these Heads, p_7 and q_7 must still be calculated, and the divisors of q_7 must be calculated from the prime factorization of q_7 , and pairs of divisors of q_7 must be checked against the criterion of Theorem 3.1.6. As it took 8.85384×10^5 seconds to go through all 1.5256×10^8 Heads for $n = 8$, we see the rate at which each Head is checked to be 5.8035×10^{-3} seconds. Thus, a lower bound (since q_7 for $n = 9$ can be on the order of q_6^2 for $n = 8$) the amount of time necessary to comb through all 7.1382×10^{14} Heads for $n = 9$ is 4.14266×10^{12} seconds, or roughly 131,273 years.

Let $\text{EFO}(n)$ denote the number of solutions to 3.1 for a given $n \in \mathbb{N}$. Assuming $\log(\text{EFO}(n))$ is expressed by an exponential function, that is $\log(\text{EFO}(n)) = a \exp(bn)$ for a and $b \in \mathbb{R}$, using Matlab's built in Curve Fitting Tool, we can estimate $\text{EFO}(9)$ to be 8,511,630,849,266.

We can observe the relationship between the number of Heads for a given n that actually

give rise to a solution to 3.1, and the number of Heads that are actually checked with our algorithm. Let us define a True Head to be a Head that gives rise to a solution to 3.1.

n	EFO(n)	# of Heads	# of Solutions per Head
3	3	2	1.5
4	14	7	2
5	147	51	2.88235
6	3,462	835	4.14611
7	294,314	66,162	4.44838
8	159,330,691	152,562,146	1.04437

Table 4.1: Comparison of solutions to 3.1 and number of Heads

n	EFO(n)	# of True Heads	# of Solutions per True Head
3	3	2	1.5
4	14	7	2
5	147	44	3.34091
6	3,462	561	6.17112
7	294,314	27,853	10.56669
8	159,330,691	13,440,167	11.85481

Table 4.2: Comparison of solutions to 3.1 and number of True Heads

n	# of Heads	# of True Heads	# True Heads per Head
3	2	2	1
4	7	7	1
5	51	44	.86275
6	835	561	.67186
7	66,162	27,853	.42098
8	152,562,146	13,440,167	.088096

Table 4.3: Comparison of number of Heads and number of True Heads

We can see that while our search space and number of possible Heads grows with n , we have proportionally fewer and fewer True Heads actually contributing solutions to EFO(n). Therefore, for the same kind of algorithm used to calculate EFO(8) to be used to calculate EFO(n) for $n \geq 9$, there must be a way to eliminate possible Heads other than actually checking if they are a True Head.

The number of Heads that needs to be checked is more than the number Heads that begin with each component equal to s_i for $1 \leq i \leq n-3$. That is the number of Heads that need to be checked for a specific n is more than the number of Heads that begin with (s_1, \dots, s_{n-3}) . The number of Heads that begin with (s_1, \dots, s_{n-3}) is $2b_{n-2}$. Therefore we know that for an arbitrary n , the number of Heads that need to be checked is greater than $2b_{n-2}$, which is on

the order of $2s_{n-2}$, or $2 \left[E^{2^{n-1}} + \frac{1}{2} \right]$, and hence grows double-exponentially. Konyagin proved that for the EFO problem with distinct denominators, the solution space is also bounded below by a doubly-exponential function, and so we know that the our EFO problem, which does not require distinct denominators, is bounded below by a doubly-exponential function as well, since the requirement of distinct denominators only removes solutions we count from our EFO problem [K].

One could also approximate $EFO(n)$ by noting:

$$EFO(n) = H_n (\text{Proportion of True Heads to Heads}) (\text{Average Number of Tails per True Head}).$$

Specifically for $n = 9$, we can estimate some of these parameters. We know $H_n = 713,922,875,728,745$. We can estimate T_n/H_n by estimating T_n from the other values we have in Table 4.2 or 4.3. We estimate T_9 by assuming T_n has a similar form to that of s_n , that is $T_n = E^{f_n}$ for some function f_n . Then we have that $f_9 \approx 112$, and thus $T_n = 2.50572 \times 10^{11}$, which is a sensible estimate based on H_9 . We also estimate the number of Tails per True Head using a cubic approximation to the last column of table 4.2. We use a cubic approximation because we see that at first the average is increasing, but then it increases by much less going from $n = 7$ to $n = 8$, and from the histograms bellow, we can assume that the number of true heads which produce more than 1 tail or solution begins to diminish compared to the number of true heads that produce only 1 tail. From Matlab's Curve Fitting Tool, assuming a cubic polynomial regression, we obtain a value of 12.11970 for $n = 9$. Thus in all we have another estimate of $EFO(9)$ to be:

$$\begin{aligned} EFO(9) &= (713922875728745)(3.50979 \times 10^{-4})(12.11970) \\ &\approx 3036856704864, \end{aligned}$$

which is of the same order of magnitude as Matlab's Curve Fitting estimate.

4.2 Tails Per True Head

We can take a closer look at the distribution of Tails per True Head. Below are histograms of the distributions of the Number of Tails per True Head.

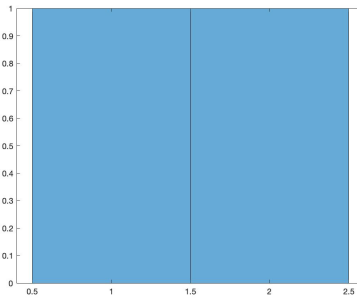


Figure 4.2: Distribution of Number of Tails per True Head for $n = 3$.
 mean= 1.5, median= .15, mode= 1, min= 1, max= 2, std= .70711.

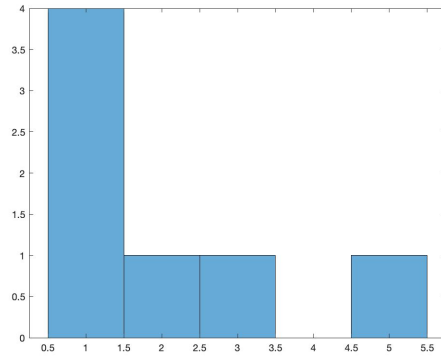


Figure 4.3: Distribution of Number of Tails per True Head for $n = 4$.
 mean= 2, median= 1, mode= 1, min= 1, max= 5, std= 1.52753.

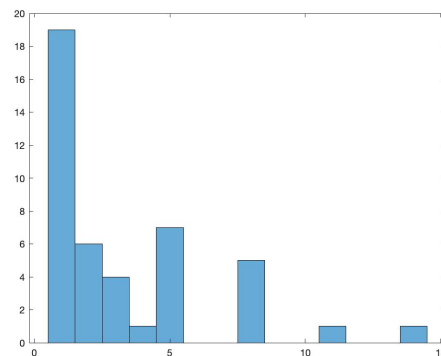


Figure 4.4: Distribution of Number of Tails per True Head for $n = 5$.
 mean= 3.34091, median= 2, mode= 1, min= 1, max= 14, std= 3.10995.

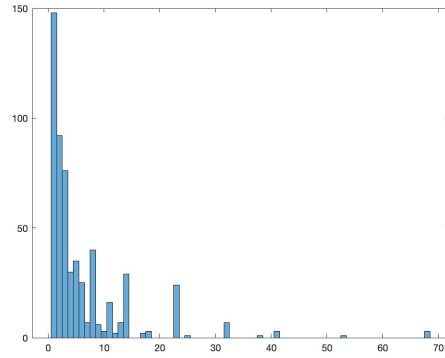


Figure 4.5: Distribution of Number of Tails per True Head for $n = 6$.
 mean= 6.1711, median= 3, mode= 1, min= 1, max= 68, std= 8.45973.

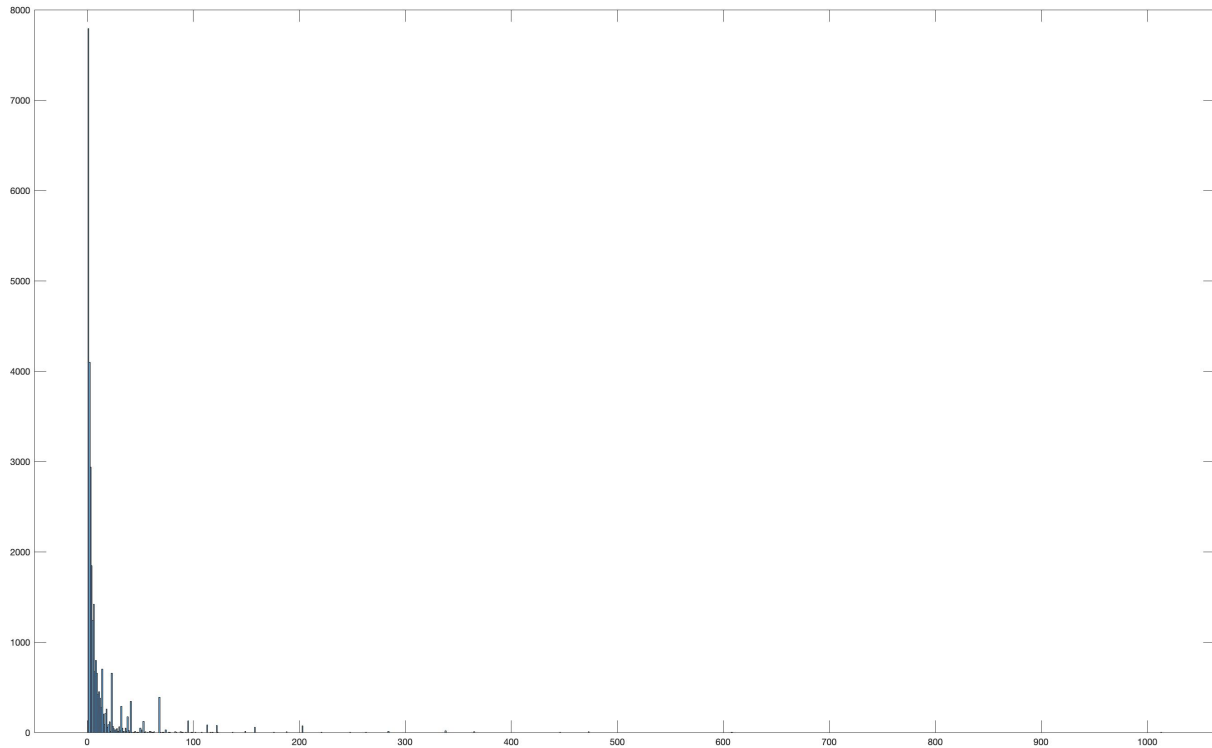


Figure 4.6: Distribution of Number of Tails per True Head for $n = 7$.
 mean= 10.56669, median= 3, mode= 1, min= 1, max= 1013, std= 26.78673.

Alternately, if there was a criterion found to determine which proper fractions $\frac{p}{q}$ could be written as the sum of $n - 2$ unit fractions, then the above criterion used in conjunction with Theorem 3.1.6 applied to $1 - \frac{p}{q} = \frac{q-p}{q}$ would be possible to calculate exactly which Heads are True Heads.

4.3 Analysis of Head Sums

4.3.1 The Sums Themselves

For any given n , the Head whose sum is closest to 1 (in the normal Euclidean sense), is the Head where each component is equal to s_i for $1 \leq i \leq n - 2$. We will call this Head, the Sylvester Head, and denote it by SH_n for any specific $n \in \mathbb{N}$. The Head whose sum is furthest from 1, is the Head where each component is equal to n for $1 \leq i \leq n - 2$. We will call this Head, the Constant Head, and denote it by CH_n . Then it is clear that the range of possible fractions p/q that need to be checked to see if they have an $n - 2$ term Egyptian Fraction expansion, and whose compliment $\frac{q-p}{p}$ has a 2 term Egyptian Fraction expansion is such that

$$\left[\frac{n-2}{n} \leq \frac{p}{q} \leq \frac{b_{n-1}-1}{b_{n-1}} \right].$$

This can be seen by the fact that the sum of the reciprocals of the components of CH_n is $\sum_{i=1}^{n-2} \frac{1}{n} = \frac{n-2}{n}$, and the sum of the reciprocals of the first $n - 2$ Sylvester numbers is

$$\sum_{i=1}^{n-2} \frac{1}{s_i} = \frac{b_{n-1}-1}{b_{n-1}}$$

from the identity

$$\frac{1}{s_i-1} - \frac{1}{s_{i+1}-1} = \frac{1}{s_i},$$

for any $i \in \mathbb{N}$. From this identity, the sum telescopes, and we have

$$\begin{aligned} \sum_{i=1}^{n-2} \frac{1}{s_i} &= \sum_{i=1}^{n-2} \left(\frac{1}{s_i-1} - \frac{1}{s_{i+1}-1} \right) \\ &= \frac{1}{s_1-1} - \frac{1}{s_{n-1}-1} \\ &= 1 - \frac{1}{s_{n-1}-1} \\ &= \frac{s_{n-1}-2}{s_{n-1}-1}. \end{aligned}$$

However, while this bounds the range of the sum of any Head, the lower bound's denominator grows linearly, and the upper bound's denominator growth is doubly exponential. Below is a plot of the density of the sums of the True Heads for $n = 3$ through $n = 7$.

We can see that the sums of each True Head get much more densely packed close to 1 as n increases. We can also see that for a fixed number of terms n , there are fewer fractions close to $\frac{n-2}{n}$ which can be expressed as Egyptian Fractions than fractions close to 1 (or specifically $\frac{b_{n-1}-1}{b_{n-1}}$), which are the lower and upper bounds on the sums of the Heads.

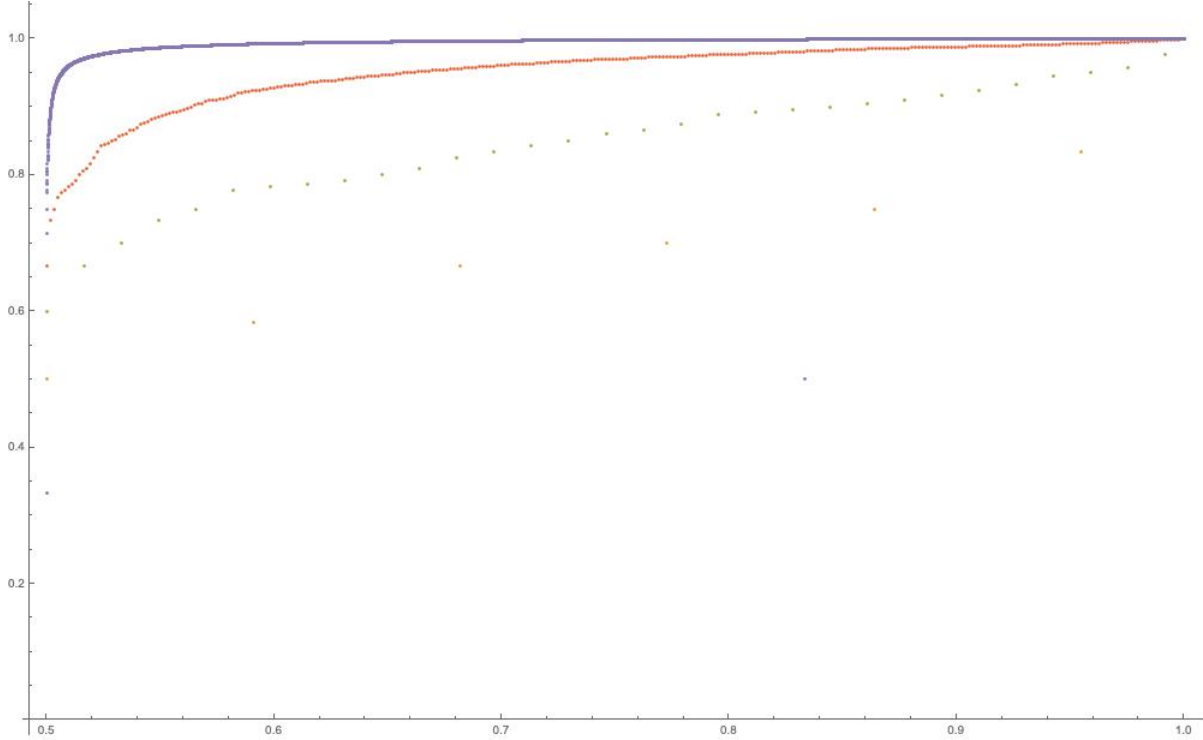


Figure 4.7: True Head Sums
 Purple: $n = 7$, Orange: $n = 6$, Green: $n = 5$, Yellow: $n = 4$, Gray: $n = 3$

4.3.2 Factors of the Denominators of the Head Sums

We may also look at the number of prime factors and distinct prime factors of each of the denominators of the sums of the True Heads:

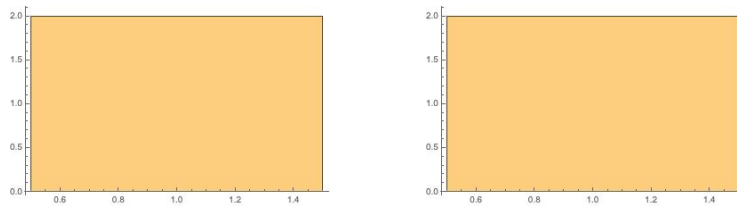


Figure 4.8: Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 3$

There is a weak upper bound on the number of prime factors of the denominator of the sum of any True Head may have, namely the largest power of 2 that is less than or equal to the product of the upper bounds given in Theorem 3.1.6, which is $\frac{n!}{2} \prod_{i=2}^{n-2} b_i$ for a given n , the histograms above would indicate that there is a much lower bound than this in reality. For example, the exponent of the largest power of 2 less than or equal to $\frac{n!}{2} \prod_{i=2}^{n-2} b_i$ for $3 \leq i \leq 7$ are $\{1, 4, 9, 17, 31\}$, while the actual maximum number of prime factors from the histograms above are $\{1, 3, 5, 8, 14\}$. There is also a weak upper bound on the number of distinct prime factors the denominator of the sum of any True Head may have, namely the

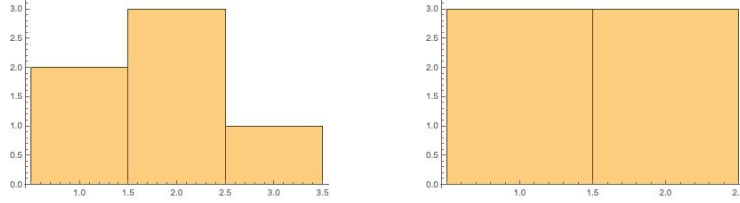


Figure 4.9: Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 4$

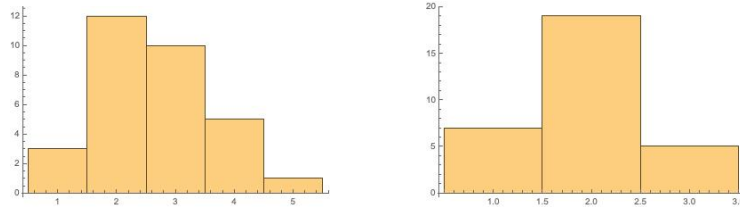


Figure 4.10: Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 5$

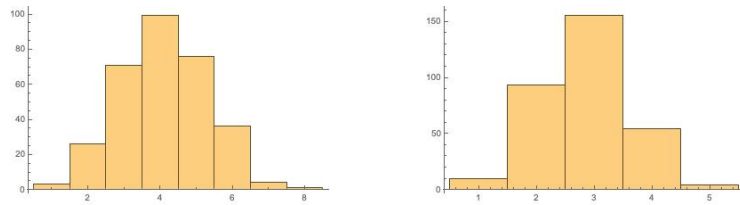


Figure 4.11: Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 6$

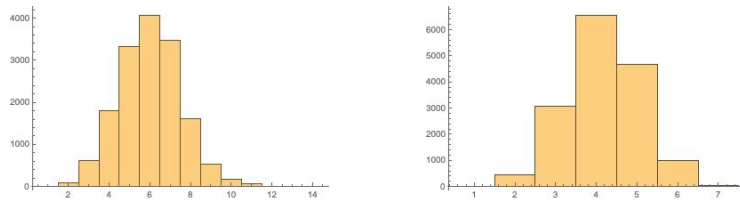


Figure 4.12: Number of Prime Factors (L) and Distinct Prime Factors (R) for $n = 7$

number of factors in the largest primorial (factorial over the primes) less than or equal to $\frac{n!}{2} \prod_{i=2}^{n-2} b_i$ for a given n . For example, the number of prime factors in the primorial less than or equal to $\frac{n!}{2} \prod_{i=2}^{n-2} b_i$ for $3 \leq i \leq 7$ are $\{1, 2, 4, 6, 9, 14\}$, while the actual maximum number of distinct prime factors from the histograms above are $\{1, 2, 3, 5, 7\}$.

4.4 Paralleliability

The algorithms built based off the theorems above is highly parallelizable. For each Head, the computation of the pairs of divisors in 3.1.6 can be done independently, so the computation can be dispersed to multiple processors to greatly reduce run time. The only thing that needs to be done before parallelization is building the Heads themselves, which because no computation is done on the Heads themselves, and they are merely listed, and need to be stored, is quick compared to the computation of the Tails. One could also minimize the amount of data to be stored by computing the number of Tails for each Head as each Head is generated, because the algorithms count solutions, rather than storing solutions.

These same theorems and algorithms could also be used to calculate the number of ways to express 1 as a sum of distinct unit fractions, with minor changes in the algorithms.

If there was found an algorithm to compute the number of ways to write p/q as the sum of 3 unit fractions, rather than just 2 unit fractions as in 3.1.6, then we would be able to bound the search space for the Heads by s_{n-3} rather than s_{n-2} , as above, but also the $4/n$ -conjecture would be solved in full.

Chapter 5

Proofs of Theorems

Theorem 3.1.1

Let $3 \leq n \in \mathbb{N}$. Let $2 \leq i \leq n - 1$. Then:

$$\begin{aligned} 2 &\leq x_1 \leq n \\ i + 1 &\leq x_i \leq (n - i + 1)b_i \\ n &\leq x_n \leq b_n. \end{aligned}$$

Proof. Assume $n \geq 3$.

Let $2 > x_1 \in \mathbb{N}$, then $x_1 = 1$. Therefore $x = (1, x_2, \dots, x_n)$ will never be a solution to 3.1.

Let $n < x_1 \in \mathbb{N}$, then the smallest x_1 could be is $x_1 = n + 1$. The smallest value that the other components of x could take would then be $n + 1$ as well. Then $x = (n + 1, \dots, n + 1)$ has reciprocals which sum to $\frac{n}{n+1} < 1$. Thus $2 \leq x_1 \leq n$.

Let $2 \leq i \leq n - 1$. Let $i + 1 > x_i \in \mathbb{N}$, since the components of x must be non-decreasing, the previous $i - 1$ components could be as large as i . Thus x could be $(i, \dots, i, x_{i+1}, \dots, x_n)$. But then x has reciprocals which sum to $1 + \sum_{j \geq i}^n \frac{1}{x_j} > 1$.

Assume x_j is as small as it can be for all $1 \leq j < i$, that is $x_j = s_j$ for $1 \leq j < i$. Let $x_i = (n - i + 1)b_i + 1$. Assume also that x_j is as small as it can be for $i + 1 \leq j \leq n$, that is $x_j = x_i = (n - i + 1)b_i + 1$ for all $i + 1 \leq j \leq n$. Then $x = (s_1, \dots, s_{i-1}, (n - i + 1)b_i + 1, \dots, (n - i + 1)b_i + 1)$, and the reciprocals of x sum to $1 - \frac{1}{s_{i-1}} + \frac{n-i+1}{(n-i+1)b_i+1} < 1 - \frac{1}{b_i} + \frac{n-i+1}{(n-i+1)b_i} = 1 - \frac{1}{b_i} + \frac{1}{b_i} = 1$. Thus, $i + 1 \leq x_i \leq (n - i + 1)b_i$.

Let $n > x_n \in \mathbb{N}$, and let x_j be as large as is can be for $1 \leq j \leq n - 1$, that is let $x_j = n - 1$ for $1 \leq j \leq n - 1$. Then $x = (n - 1, \dots, n - 1)$ and the reciprocals of x sum to $\frac{n}{n-1} = 1 + \frac{1}{n-1} > 1$.

Let $b_n < x_n \in \mathbb{N}$. Assume x_j is as small as it can be for $1 \leq j \leq n - 1$, that is assume $x_j = s_j$ for $1 \leq j \leq n - 1$. Then $x = (s_1, \dots, s_{n-1}, b_n + 1) = (s_j)_{1 \leq j \leq n}$, and the reciprocals of x sum to $1 - \frac{1}{s_{n+1-1}} < 1$. Thus $n \leq x_n \leq b_n$. \square

Theorem 3.1.3

Let $n \in \mathbb{N}$, and let $x = (x_1, \dots, x_n)$ satisfy 3.1. Then

$$x_n = \frac{\prod_{i=1}^{n-1} x_i - \sum_{j=1}^{n-1} x_j^\dagger}{\prod_{i=1}^{n-1} x_i},$$

where x_j^\dagger is the product of the first $n-1$ x_i 's excluding x_j .

Proof. Assume the hypotheses within the statement of the theorem. Then:

$$\begin{aligned} \sum_{i=1}^n \frac{1}{x_i} &= 1 \\ \sum_{i=1}^{n-1} \frac{1}{x_i} + \frac{1}{x_n} &= 1 \\ \implies \frac{1}{x_n} &= 1 - \sum_{i=1}^{n-1} \frac{1}{x_i} \\ &= \frac{\prod_{i=1}^{n-1} x_i - \sum_{j=1}^{n-1} x_j^\dagger}{\prod_{i=1}^{n-1} x_i} \\ \implies x_n &= \frac{\prod_{i=1}^{n-1} x_i}{\prod_{i=1}^{n-1} x_i - \sum_{j=1}^{n-1} x_j^\dagger} \end{aligned}$$

□

Theorem 3.1.6

Let $p, q \in \mathbb{N}$ such that $p \perp q$. There exists $\alpha, \beta \in \mathbb{N}$ such that:

$$\frac{p}{q} = \frac{1}{\alpha} + \frac{1}{\beta}$$

if and only if there exists $a, b \in \mathbb{N}$ where $a|q$ and $b|q$ such that $p|(a+b)$.

Proof. Assume the forward direction of the hypothesis. Let $d = \gcd(\alpha, \beta)$ and let $\alpha' = \alpha/d$ and $\beta' = \beta/d$. Then:

$$\begin{aligned} \frac{p}{q} &= \frac{1}{\alpha} + \frac{1}{\beta} = \frac{1}{d} \left(\frac{1}{\alpha'} + \frac{1}{\beta'} \right) \\ \implies \frac{pd}{q} &= \frac{1}{\alpha'} + \frac{1}{\beta'}. \end{aligned}$$

Let $r = \gcd(d, q)$. Let $p' = pd/r$ and let $q' = q/r$. Then:

$$\begin{aligned} \frac{pd}{q} &= \frac{\frac{pd}{r}}{\frac{q}{r}} = \frac{p'}{q'} = \frac{1}{\alpha'} + \frac{1}{\beta'} \\ &= \frac{\alpha' + \beta'}{\alpha'\beta'}. \end{aligned}$$

This implies that there exists $\lambda \in \mathbb{Z}$ such that $\alpha' + \beta' = \lambda p'$ and $\alpha' \beta' = \lambda q'$. Therefore we have that $\lambda = \frac{\alpha' \beta'}{q'}$, and substituting into our other equality with λ we have:

$$\begin{aligned}\alpha' + \beta' &= \lambda p' \\ &= \frac{\alpha' \beta'}{q'} p' \\ \implies q'(\alpha' + \beta') &= \alpha' \beta' p' \\ \implies q' \alpha' + q' \beta' &= \alpha' \beta' p'.\end{aligned}$$

Dividing through on both sides by α' , we have that:

$$\begin{aligned}q' \alpha' + q' \beta' &= \alpha' \beta' p' \\ \frac{q' \alpha'}{\alpha'} + \frac{q' \beta'}{\alpha'} &= \frac{\alpha' \beta' p'}{\alpha'} \\ \implies q' + \frac{q' \beta'}{\alpha'} &= \beta' p'.\end{aligned}$$

Because q' and $\beta' p'$ are integers, $q' \beta' / \alpha'$ must also be an integer, but because $\alpha' \perp \beta'$, it must be the case that $\alpha' | q'$. By a similar argument, $\beta' | q'$. Then because $q' | q$, we have that $\alpha' | q$ and $\beta' | q$. We also note that $d/r \in \mathbb{N}$ and since $p' | (\alpha' + \beta')$, $\frac{pd}{r} | (\alpha' + \beta')$ and therefore $p | (\alpha' + \beta')$. Letting $b = \alpha'$ and $a = \beta'$ completes the proof.

Assume the statements of the reverse direction of the hypothesis.

Let

$$\alpha = \frac{q(a+b)}{pb}, \quad \beta = \frac{q(a+b)}{pa}.$$

We note that $\alpha, \beta \in \mathbb{N}$ because $p | (a+b)$ and $a | q$ and $b | q$. Therefore:

$$\frac{1}{\alpha} + \frac{1}{\beta} = \frac{pb}{q(a+b)} + \frac{pa}{q(a+b)} = \frac{pb+pa}{q(a+b)} = \frac{p(b+a)}{q(a+b)} = \frac{p}{q}$$

□

Theorem 3.1.8

Let $3 \leq n \in \mathbb{N}$, and let $1 \leq i \leq n-2$. Let $(x_k)_{1 \leq k \leq i}$ be the first i denominators of a possible solution to 3.1. Let $\frac{p_i}{q_i} = 1 - \sum_{j=1}^i \frac{1}{x_j}$ be such that $p_i \perp q_i$.

Let $w_{i+1} = \left\lfloor \frac{q_i}{p_i} \right\rfloor + 1$. Let $u_{i+1} = \left\lfloor \frac{(n-i)q_i}{p_i} \right\rfloor$. Let $\ell_{i+1} = \begin{cases} w_{i+1} & q_i \geq p_i x_i \\ x_i & q_i < p_i x_i \end{cases}$.

Then $\ell_{i+1} \leq x_{i+1} \leq u_{i+1}$.

Proof. Assume the statements in the hypothesis.

Assume for contradiction that $x_{i+1} < \ell_{i+1}$. Let $q_i < p_i x_i$. Then $x_{i+1} < x_i$, which is not allowed

in 3.1, so $x_{i+1} \geq \ell_{i+1}$. Let $q_i \geq p_i x_i$. Then the largest reciprocal of a natural number that is less than $\frac{p_i}{q_i}$ is $\frac{1}{w_{i+1}}$, but by assumption $x_{i+1} < w_{i+1}$, so $\frac{1}{x_{i+1}} > \frac{1}{w_{i+1}}$. Then it must be that $\frac{1}{x_{i+1}} \geq \frac{p_i}{q_i}$, and since $i \leq n-2$, we will have that $\sum_{j \geq i+1}^{i+1} \frac{1}{x_j} \geq 1$, with $n-i-1$ terms still needed to be used. Thus $x_{i+1} \geq \ell_{i+1}$.

Assume for contradiction that $x_{i+1} > u_{i+1}$. Let $x_j = x_{i+1}$ for $i+2 \leq j \leq n$. Then the difference between 1 and the sum of the reciprocals of x is:

$$1 - \sum_{j=1}^n \frac{1}{x_j} = \frac{p_i}{q_i} - \sum_{j=i+1}^n \frac{1}{x_j} = \frac{p_i}{q_i} - \frac{n-i}{x_{i+1}} > \frac{p_i}{q_i} - \frac{n-i}{u_{i+1}} \geq 0.$$

Therefore if $x_{i+1} > u_{i+1}$, the reciprocals of x could not sum to 1, and so $x_{i+1} \leq u_{i+1}$. \square

Theorem 3.2.1

Let $(x_i)_{1 \leq i \leq n-2}$ be a Head of a possible solution to 3.1. Let $p \in \mathbb{N}$ and $q \in \mathbb{N}$ be such that $p \perp q$ and

$$\frac{p}{q} = 1 - \sum_{i=1}^{n-2} \frac{1}{x_i}.$$

Then the only possible pairs of divisors of q that need to be checked are (a, b) that satisfy Theorem 3.1.6 such that

$$\frac{b}{a} \leq \frac{q}{px_{n-2} - q}.$$

If $px_{n-2} - q \leq 0$, then all pairs of divisors (a, b) that satisfy 3.1.6 give rise to possible solutions to 3.1.

Proof. Assume the statements in the hypothesis, and assume $a \leq b$. In order for a pair of divisors of q , (a, b) , to be valid, we must have that $\alpha \geq x_{n-2}$. Therefore we have:

$$\begin{aligned} & \alpha \geq x_{n-2} \\ \implies & \frac{q(a+b)}{pb} \geq x_{n-2} \\ \implies & \frac{qa+qb}{pb} \geq x_{n-2} \\ \implies & \frac{qa}{pb} + \frac{q}{p} \geq x_{n-2} \\ \implies & \frac{qa}{pb} \geq x_{n-2} - \frac{q}{p} \\ \implies & \frac{qa}{pb} \geq \frac{px_{n-2} - q}{p} \\ \implies & \frac{a}{b} \geq \frac{px_{n-2} - q}{q} \\ \implies & \frac{b}{a} \leq \frac{q}{px_{n-2} - q} \end{aligned}$$

Since $0 < a \geq b$, and $0 < p \leq q$, it could be such that $px_{n-2} - q < 0$. If this is the case, then any pairs of divisors of q , (a, b) , satisfy the penultimate line of the above proof, as the LHS is positive, and the RHS is negative. However, if $px_{n-2} - q = 0$, then again every pair of divisors of q , (a, b) , satisfy the penultimate line of the above proof, as the LHS is positive, and the RHS is 0. \square

Code Appendix

```
1 function[instances]=instances(factorvec)
2 %The function instances(factorvec) takes in the prime factorization ...
3   of a
4   %number and returns an array giving the prime-power decomposition of that
5   %number.
6
7 %factorvec=factor(Number);
8 x = unique(factorvec);
9 N = numel(x);
10 count = zeros(N,1);
11 for k = 1:N
12     count(k) = sum(factorvec==x(k));
13 end
14 instances=horzcat(x',count);
15 %instances=horzcat(x',countmember(x,factorvec)');
16 end
```

```
1 function[newdivsorder]=mydivisorsordering(instancesmatrix)
2 %The function mydivisorsordering(instancesmatrix) takes in the prime-power
3 %factorization of a number in matrix form, and returns a vector of all the
4 %divisors of that number. (Much faster than the built in divisors(n)
5 %function.
6
7 % instancesmatrix=instances(factor(Number));
8 newdivsorder=(instancesmatrix(1,1)).^(0:(instancesmatrix(1,2)));
9 for i=2:size(instancesmatrix,1)
10     newdivsorder=kron((instancesmatrix(i,1)).^(0:(instancesmatrix(i,2)))...
11     ,newdivsorder);
12 end
```

```
1 function [ cycsum ] = cyclicsum(vec)
2 %The function cyclicsum(vec) finds the sum of all of the products
3 %excluding a single term from the vector vec.
4
5 %Example: cyclicsum([2,5,6])=2*5+2*6+5*6=42
6 n=length(vec);
7 cycsum=0;
8 for i=1:n
9     cycsum=cycsum+prod(vec([1:i-1,i+1:end]));
10 end
```

The code below makes use of the function `nchoose2.m` [NC2].

```

1 function[numpairs,pairs]=comb(num,den,head,lastdenom)
2 %This function comb(num,den,head,lastdenom) first checks to see if num/den
3 %is in simplest form, and then builds all possible pairs of divisors of
4 %den and combs through the pairs and pulls out the pairs that satisfy
5 %Theorem 3.1.4.
6
7 g=gcd(num,den);
8 if g==1
9     primefactorization=instances(factor(den));
10    %primefactorization=instances(oaasoufheadfactor(den,head));
11    %Number Theory Easy Answer
12    divs=mydivisorsordering(primefactorization);
13    testerdivisorpairs=vertcat([1,1],nchoose2(divs));
14    ms=testerdivisorpairs(:,1);
15    ns=testerdivisorpairs(:,2);
16    %First Check: Divisor Pairs Satisfy  $ns./ms \geq den/(num*lastdenom-den)$ 
17    bottom=num*lastdenom-den;
18    if bottom>0
19        trues=ns./ms<=den/(bottom);
20        ms=ms(trues);
21        ns=ns(trues);
22    end
23    %Second Check: Sum of Divisors is congruent mod num
24    sums=ms+ns;
25    trues=mod(sums,num)==0;
26    if any(trues)==0
27        numpairs=0;
28        pairs=[];
29    else
30        ms=ms(trues);
31        ns=ns(trues);
32        sums=sums(trues);
33        %Third Check: Divisors are Relatively Prime
34        trues=gcd(ms,ns)==1;
35        if any(trues)==0
36            numpairs=0;
37            pairs=[];
38        else
39            ms=ms(trues);
40            ns=ns(trues);
41            sums=sums(trues);
42            %Fourth Check: Smallest New Denominator is Less Than or
43            %Equal to last denominator used in head:"lastdenom"
44            trues=((den./max(ms,ns)).*(sums./num)>=lastdenom)==1;
45            numpairs=sum(trues);
46            if nargout > 1
47                ms=ms(trues);
48                ns=ns(trues);
49                sums=sums(trues);
50                pairs=[(den./max(ms,ns)).*(sums./num),(den./min(ms,ns))...
51                    .* (sums./num)];

```

```

52         end
53     end
54 end
55 else
56     %[numpairs,pairs]=comb(num/g,den/g,lastdenom,head);
57     %head(end)=head(end)/gcd(head(end),prod(head(1:end-1)));
58     [numpairs,pairs]=comb(num/g,den/g,head,lastdenom);
59 end

```

```

1 function [EFOS,TrueHeads]=Tails(mat)
2 %The function Tails(mat) takes in a matrix of Heads called mat, and returns
3 %the number of Tails from the Heads in mat, as well as the True Heads from
4 %mat.
5
6 EFOS=0;
7 Tracker=zeros(size(mat,1),1);
8 for i=1:size(mat,1)
9     den=prod(mat(i,:));
10    num=den-cyclicsum(mat(i,:));
11    tails=comb(num,den,mat(i,:),mat(i,end));
12    if tails>0
13        EFOS=EFOS+tails;
14        Tracker(i)=1;
15    end
16 end
17 Tracker=logical(Tracker);
18 TrueHeads=mat(Tracker,:);

```

```

1 function [nextterm]=nextgreedyufoterm(vec)
2 %The function nextgreedyufoterm(vec) returns what would be the next term
3 %using a greedy algorithm, given an initial vector of denominators in the
4 %UFO problem. nextgreedyufoterm also assumes the final vector of terms is
5 %infinite (like the infinite Sylvester Sequence).
6
7 %Example: nextgreedyufoterm([2])=3
8
9 %Example: nextgreedyufoterm([2,5,6])=8
10
11 %Example: nextgreedyufoterm([2,4,6])=13
12
13 %Example: nextgreedyufoterm([5,5,5])=5
14
15 b=vec(end);
16 a=floor(prod(vec)/(prod(vec)-cyclicsum(vec)))+1;
17 if a<=b
18     nextterm=b;
19 else
20     nextterm=a;
21 end
22 end

```

```

1 function [sn] = Sylvester(n)
2 %Sylvester - returns the nth term of Sylvester's Sequence, where
3 %a(n)=[s(n-1)]^2-s(n-1)+1.
4
5 if n==1
6     sn=2;
7 else
8     sn=(Sylvester(n-1))^2-Sylvester(n-1)+1;
9 end
10 end

```

This is an example of a brute force way to find the Heads, this is for $n = 7$.

```

1 heads7=[];
2 for i1=2:7
3     for i2=nextgreedyufoterm(i1):floor(prod(i1)/(prod(i1)-cyclicsum(i1))*6)
4         for i3=nextgreedyufoterm([i1,i2]):floor(prod([i1,i2])/...
5             (prod([i1,i2])-cyclicsum([i1,i2]))*5)
6             for i4=nextgreedyufoterm([i1,i2,i3]):floor(prod([i1,i2,i3])/...
7                 (prod([i1,i2,i3])-cyclicsum([i1,i2,i3]))*4)
8                 for i5=nextgreedyufoterm([i1,i2,i3,i4]):...
9                     floor(prod([i1,i2,i3,i4])/(prod([i1,i2,i3,i4])...
10                         -cyclicsum([i1,i2,i3,i4]))*3)
11                     heads7=vertcat(heads7,[i1,i2,i3,i4,i5]);
12                 end
13             end
14         end
15     end
16 end

```

Bibliography

- [A000058] OEIS Foundation Inc. (2019), The On-Line Encyclopedia of Integer Sequences, <http://oeis.org/A000058>.
- [A002966] OEIS Foundation Inc. (2019), The On-Line Encyclopedia of Integer Sequences, <http://oeis.org/A002966>.
- [A076393] OEIS Foundation Inc. (2019), The On-Line Encyclopedia of Integer Sequences, <http://oeis.org/A076393>.
- [BC] Ezra Brown & Matthew Crawford (2018) Five Families Around a Well: A New Look at an Old Problem, *The College Mathematics Journal*, 49:3, 162-168, DOI: 10.1080/07468342.2018.1447203
- [C] E. S. Croot III, *On some questions of Erdős and Graham about Egyptian fractions*, *Mathematika* **46** (1999), 359-372.
- [EG] P. Erdős & R.L. Graham, *Old and New Problems and Results in Combinatorial Number Theory*, Mono. no. 28 de L'Enseignement Math., Univ. Geneva (1980), 128.
- [EMLR] Gardner, Milo. "Egyptian Mathematical Leather Roll." From MathWorld—A Wolfram Web Resource, created by Eric W. Weisstein. <http://mathworld.wolfram.com/EgyptianMathematicalLeatherRoll.html>
- [EN] P. Erdős & I. Niven, *On certain variations of the harmonic series*, *Bull. Amer. Math. Soc.*, **51** (1945), 433-436
- [ERD] P. Erdős, Egy Kürschák-féle elemi számelméleti tétel általánosítása (Generalization of an elementary number-theoretic theorem of Kürschák, in Hungarian), *Mat. Fiz, Lapok* **39** (1932), 17-24.
- [ES] P. Erdős, *On a Diophantine equation (Hungarian)*, *Mat. Lapok 1* (1950), 192-210.
- [ET] P. Erdős & P. Turán, *On Some Sequences of Integers*, *The Journal of the London Mathematical Society*, **s1-11** (1936), no.4, 261-264.
- [H] P. Hoffman, *The Man Who Loved Only Numbers: The Story of Paul Erdős and the Search for Mathematical Truth*, New York: Hyperion (1998), 153-157.

- [HAG] T.R. Hagedorn, *A Proof of a Conjecture on Egyptian Fractions*, The Amer. Math. Monthly, **107**, no.1, (2000), 62-63
- [IS] I. Stewart, *The Riddle of the Vanishing Camel*, Scientific American, **266** (1992), no. 6, 122-125
- [JJS] James Joseph Sylvester Picture Wikipedia - https://commons.wikimedia.org/wiki/File:James_Joseph_Sylvester.jpg
- [K] Konyagin, S.V. Math Notes (2014) 95: 277. <https://doi.org/10.1134/S0001434614010295>
- [LPF] Sigler, L. (2002). *Fibonacci's Liber Abaci*, New York, NY: Springer New York. 119-126.
- [NC2] Jos (10584) (2019). nchoose2 (<https://www.mathworks.com/matlabcentral/fileexchange/20144-nchoose2>), MATLAB Central File Exchange. Retrieved May 3, 2019.
- [PEP] Paul Erdos Picture Wikipedia - By Kmhkmh - Own work, CC BY 3.0, <https://commons.wikimedia.org/w/index.php?curid=38087162>
License: <https://creativecommons.org/licenses/by/3.0/deed.en>
- [RKG] R. K. Guy, *Unsolved problems in Number Theory*, Third Edition, Springer, New York, 2004, 437.
- [RLG] Graham R.L. (2013) Paul Erdős and Egyptian Fractions. In: Lovász L., Ruzsa I.Z., Sós V.T. (eds) Erdős Centennial. Bolyai Society Mathematical Studies, vol 25. Springer, Berlin, Heidelberg
- [RP1] Rhind Papyrus Image Wikipedia - Paul James Cowie (Pjamescowie) - https://commons.wikimedia.org/wiki/File:Rhind_Mathematical_Papyrus.jpg
- [RP2] Gardner, Milo. "Rhind Papyrus." From MathWorld—A Wolfram Web Resource, created by Eric W. Weisstein. <http://mathworld.wolfram.com/RhindPapyrus.html>
- [RW] R.G. Wilson, "A2966." Message to Neil J.A. Sloane. 9 September 1994. Facsimile. <https://oeis.org/A002966/a002966.1.pdf>
- [SING] D. Singmaster, *The number of representations of one as a sum of unit fractions*, unpublished manuscript, 1972.
- [STEW] Stewart, Ian. "MATHEMATICAL RECREATIONS: The Riddle of the Vanishing Camel." Scientific American, vol. 266, no. 6, 1992, pp. 122–125. JSTOR, www.jstor.org/stable/24939113.
- [SYL] J.J. Sylvester, *On a Point in the Theory of Vulgar Fractions*, Amer. Jour. of Math. **3** (1880) no. 4, 332-335.

- [X1] D.J. Negash, *Solutions to Dophantine Equation of Erdos-Straus Conjecture*,
<https://arxiv.org/pdf/1812.05684.pdf>
- [X2] S.E. Salez, *The Erdős-Straus conjecture New modular equations and checking up to $N = 10^{17}$* ,
<https://arxiv.org/pdf/1406.6307.pdf>