A NOTE ON 3-FREE PERMUTATIONS

Bill Correll, Jr.
MDA Information Systems LLC, Ann Arbor, Michigan
william.correll@mdaus.com

Randy W. Ho
Garmin International, Chandler, Arizona
GooglyPower@gmail.com

Received: 12/1/16, Accepted: 10/31/17, Published: 11/10/17

Abstract

Let $\theta(n)$ denote the number of permutations of $\{1, 2, \ldots, n\}$ that do not contain a 3-term arithmetic progression as a subsequence. Such permutations are known as 3-free permutations. We present a dynamic programming algorithm to count all 3-free permutations of $\{1, 2, \ldots, n\}$. We use the output to extend and correct enumerative results in the literature for $\theta(n)$ from $n = 20$ out to $n = 90$ and use the new values to inductively improve existing bounds on $\theta(n)$.

1. Introduction and Results

Let n be a positive integer and let $[n]$ denote the set $\{1, 2, \ldots, n\}$. Let $\alpha = (a_1, a_2, \ldots, a_n)$ be a permutation of $[n]$. Then α is a *3-free permutation* if and only if, for every index j $(1 \leq j \leq n)$, there do not exist indices $i < j$ and $k > j$ such that $a_i + a_k = 2a_j$. Let $\theta(n)$ be the function that gives the number of 3-free permutations of $[n]$. Of course the value of $\theta(n)$ will be unchanged if we replace $[n]$ with any set of n integers in arithmetic progression so we will hereafter use $[n]$ when referring to $\theta(n)$. In 1973 Entringer and Jackson initiated the study of 3-free permutations by posing

Problem 1 (Entringer and Jackson [5]). Does every permutation of $\{0, 1, \ldots, n\}$ contain an arithmetic progression of at least three terms?

Three solutions (see [7], [8], [12]) to Problem 1 showing that the answer is "No" along with comments [11] containing a table of values of $\theta(n)$ for $1 \leq n \leq 20$ were published. The solutions of Odda [8] and Thomas [12] contained the first constructions for 3-free permutations. Odda describes how to construct one 3-free permutation for each n. Thomas devised a method to generate 2^{n-1} 3-free permutations for each n. Thomas's examples show that the sets of permutations his method generates aren't exhaustive.

The purpose of this note is to present an algorithm that counts the number of 3-free permutations of n consecutive integers for each n. We correct and extend the tables of known values of $\theta(n)$ out to $n = 90$ and improve upper and lower bounds by proving the following four results.

Theorem 2. *For positive integers $n \geq 45$,*

$$\theta(n) \geq \frac{c_1^n}{2}, c_1 = \sqrt[80]{2\theta(80)} = 2.201\ldots. \tag{1}$$

Theorem 3. *For positive integers $n \geq 36$,*

$$\theta(n) \leq \frac{c_2^n}{21}, c_2 = \sqrt[64]{21\theta(64)} = 2.364\ldots. \tag{2}$$

Theorem 4. *For positive integers $k \geq 6$ and $n = 2^k$,*

$$\theta(n) \geq \frac{c_3^n}{2}, c_3 = \sqrt[64]{2\theta(64)} = 2.279\ldots. \tag{3}$$

Theorem 5. *For all positive integers n,*

$$\theta(n) \geq \frac{nc_4^n}{40}, c_4 = \sqrt[40]{\theta(40)} = 2.156\ldots. \tag{4}$$

The existence of $\lim \theta(n)^{1/n}$ as $n \to \infty$ was identified in [9] as a key problem in the study of $\theta(n)$. It remains an open question although Theorems 2, 3 imply that the limit lies within the interval $[c_1, c_2]$ if it exists. The first author explored connections between 3-free permutations and Costas arrays in [2], where slightly weaker versions of Theorems 2 and 3 were stated without proof.

For clarity, we comment here that we are not presenting any results on the related problem of evaluating and bounding the function $r(n)$ giving the longest 3-free subsequence of the sequence $1, 2, \ldots, n$. The latest developments in solving that problem currently appear in [4].

2. Some Results From the Literature on $\theta(n)$

Davis, Entringer, Graham, and Simmons [3] established a number of bounds on the growth of $\theta(n)$ including the following:

Theorem 6 (Davis, Entringer, Graham, and Simmons, [3]). *For positive integers n,*

$$\theta(2n) \geq 2\theta^2(n), \tag{5}$$

$$\theta(2n+1) \geq 2\theta(n)\theta(n+1). \tag{6}$$

Theorem 7 (Davis, Entringer, Graham, and Simmons, [3]). *For $n = 2^k, k \geq 4$,*

$$\theta(n) \ \geq \ \frac{c^n}{2}, c = \sqrt[16]{2\theta(16)} = 2.248\ldots. \tag{7}$$

Sharma's dissertation [10] is noteworthy in that it established the long-conjectured result that $\theta(n)$ has an exponential upper bound. Sharma used parity arguments to prove

Theorem 8 (Sharma, [9]). *For each $n \geq 3$,*

$$\theta(n) \leq 21\theta\left(\left\lceil\frac{n}{2}\right\rceil\right)\theta\left(\left\lfloor\frac{n}{2}\right\rfloor\right). \tag{8}$$

The key result in [10] (and in the follow-up journal paper [9] as well as the book [13]) he obtains from Theorem 8 is

Theorem 9 (Sharma, [9]). *For $n \geq 11$,*

$$\theta(n) \ \leq \ \frac{2.7^n}{21}. \tag{9}$$

Sharma also improved Thomas's strict, constructive lower bound of 2^{n-1} for $n > 5$ by showing that:

Theorem 10 (Sharma, [9]). *For all positive integers n,*

$$\theta(n) \ \geq \ \frac{n2^n}{10}, \tag{10}$$

but LeSaulnier and Vijay were able to establish

Theorem 11 (LeSaulnier and Vijay [6]). *For $n \geq 8$,*

$$\theta(n) \ \geq \ \frac{1}{2}c^n, \ \textit{where } c \ = \ (2\theta(10))^{\frac{1}{10}} \ = \ 2.152.... \tag{11}$$

In Section 5 we use Theorems 6 and 8 in inductive proofs of Theorems 2, 3, and 4 to improve Theorems 11, 9, and 7, respectively. We also rework Sharma's proof of Corollary 3.2.1 of [9] relying on Theorem 6 using our additional computed values of $\theta(n)$ to improve upon Theorem 10 for $n \geq 19$.

3. Algorithm Descriptions

Recall from Section 1 that we write $\alpha = (a_1, a_2, \ldots, a_n)$ for a permutation of $[n]$. For $j = 1, 2, \ldots, n$, if we define

$$T_j \equiv \{a_k \,|\, j \leq k \leq n\}, \tag{12}$$

then the 3-free property of a permutation can be restated as saying that there do not exist $a_i \notin T_j$ and $a_k \in T_j - \{a_j\}$ such that $a_i + a_k = 2a_j$. Further inspection of the 3-free property allows us to replace $T_j - \{a_j\}$ with T_j, because $a_k = a_j$ would imply $a_i = a_j$ (from $a_i + a_k = 2a_j$), but then a_i would be in T_j. If the 3-free property holds for all $1 \le j \le n$, then α is a 3-free permutation. This suggests the following algorithm to generate 3-free permutations:

Backtracking algorithm to enumerate 3-free permutations

Subroutine Enumerate
Input: A (possibly empty) sequence $\rho = (p_1, p_2, \ldots, p_k)$ of distinct integers $p_k \in [n]$
Output: All 3-free permutations of $[n]$ that begin with ρ

```
1:  procedure Enumerate(ρ)
2:      if |ρ| = n then
3:          print ρ
4:      else
5:          P = {ρ_1, ρ_2, ..., ρ_|ρ|}
6:          for 1 ≤ j ≤ n do
7:              if j ∉ P and ∄i ∈ P, k ∈ [n] \ P such that i + k = 2j then
8:                  Enumerate((ρ, j))
9:              end if
10:         end for
11:     end if
12: end procedure
```

Main Backtracking Algorithm
Input: Positive integer n
Output: List of all 3-free permutations of $[n]$

```
procedure EnumerateMain(n)
    Enumerate([])                                  (▷) Empty sequence
end procedure
```

In the subroutine *Enumerate*, the notation (ρ, j) on line 8 denotes the sequence obtained from appending the integer j to the sequence ρ. The main backtracking algorithm recursively generates the 3-free permutations one by one, so it could be of use in generating data from which new structural properties of 3-free permutations could be deduced. Clearly its running time is bounded below by $\theta(n)$. If we are interested in the number $\theta(n)$ of 3-free permutations and not the permutations themselves, we can speed up the counting process by using dynamic programming. Dynamic programming algorithms (see, for instance, [1]) solve programs by combining solutions to subproblems. The subproblems can be dependent in that they have common subsubproblems.

A key observation is that the 3-free property of a permutation depends on the *set* of elements that have been used so far in building up that permutation. The exact ordering of those elements is not relevant. The dynamic programming algorithm recursively evaluates $\theta(n)$ using dynamic programming. It uses bitsets to keep track of which integers have not been placed in an effort to build up a 3-free permutation (a *bitset* is a sequence of zeros and ones.)

Dynamic programming algorithm to count 3-free permutations

Subroutine Count
Input: A bitset b of length n
Output: The number $\theta(b)$ of 3-free permutations of $[n]$ that begin with ρ, where ρ is any valid initial sequence that uses exactly the integers that b maps to 0. Note that if there is more than one such sequence, then they must give the same number, due to the 3-free property

```
function COUNT(b)
    if ∃(b, v) ∈ C for some v then
        return v
    else if max{b[1], b[2], ..., b[n]} = 0 then
        return 1
    else
        ans ← 0
        for 1 ≤ j ≤ n do
            if b[j] = 1 and ∄1 ≤ i, k ≤ n such that b[i] = 0 and b[k] = 1 and
i + k = 2j then
                b' ← b
                b'[j] ← 0
                ans ← ans + COUNT(b')
            end if
        end for
        C ← C ∪ {(b, ans)}
        return ans
    end if
end function
```

Main Dynamic Programming Algorithm
Input: Positive integer n
Output: $\theta(n)$

```
function COUNTMAIN(n)
    C ← ∅
    return COUNT((1, 1, ..., 1))                    (▷) Bitset of n ones
end function
```

In the above algorithm, C denotes a set of pairs (b, v), where b is a bitset of length n and v is a non-negative integer. The set C is intended to be implemented by a data structure known as a “map”. In our usage of C, the value of v for each b is $\theta(b)$.

Let $T \subseteq [n]$. It takes $O(n)$ time to check if the 3-free property is violated and it takes $O(n)$ time to iterate over every element t in T. On the surface Algorithm 3 appears to require $O(2^n)$ memory to store $\theta(T)$ for every subset T of $[n]$ and the running time appears to be $O(n^2 2^n)$. However, it turns out that only a small percentage of the subsets of $[n]$ are needed in the recurrence because most of them are not reachable due to a violation of the 3-free property. This helped us to tabulate $\theta(n)$ out to $n = 90$. The value of $\theta(90)$ has 31 digits.

4. Computational Enumerative Results

We pushed a Java implementation of the dynamic programming algorithm out to $n = 90$ and updated entry A003407 of the Online Encyclopedia of Integer Sequences (`http://www.oeis.org/A003407`) with the values in Table 4. For $n = 90$, the fraction of subsets that had to be visited was only

$$254931123/(2^{90}) \approx 2.059(10^{-19}). \tag{13}$$

Our Java implementation ran out of memory for $n = 91$. Algorithm 3 does not lend itself to parallelization due to the way it uses memory. Additional values of $\theta(n)$ can be obtained on computing platforms having additional memory, support for arbitrarily long integers, and adequate processing power.

Before our computations, there were at least 4 published tables of values of $\theta(n)$ for $1 \leq n \leq 20$ although only two of these tables are correct. The very first table to appear is in [11] and claims that 73904 is the value of $\theta(15)$ but the correct value is $\theta(15) = 74904$. For $n = 17$ the table in [9] claims that 360016 is the value of $\theta(17)$ but the correct value is $\theta(17) = 368016$. The first twenty values of $\theta(n)$ listed above do agree with the table in [3]. The first 20 entries in entry A003407 were correct at the time we extended them.

5. Proofs

Theorems 2, 3, and 4 can be proven by induction:

Proof. To prove Theorem 2 it suffices, by Theorem 6, to prove $\theta(n) \geq \frac{c^n}{2}$ for $42 \leq n \leq 83$ and some constant c. Computation shows that the maximal such c is $\min(2\theta(n))^{1/n} = c_1$ and occurs for $n = 42$. □

Proof. To prove Theorem 3, we observe that, for $42 \leq n \leq 83, \max(21\theta(n))^{\frac{1}{n}} = c_2$, and occurs for $n = 64$ so (3) holds for all $n \in [42, 83]$. That inequality (3) holds for

n	$\theta(n)$	n	$\theta(n)$	n	$\theta(n)$
1	1	31	41918682488	61	1612719155955443585092
2	2	32	121728075232	62	4640218386156695178110
3	4	33	207996053184	63	13557444070821420327240
4	10	34	360257593216	64	39911512393313043466768
5	20	35	639536491376	65	67867319248960144994224
6	48	36	1144978334240	66	115643050433241064474672
7	104	37	2362611440576	67	199272038058617170554928
8	282	38	4911144118024	68	344053071167567188894208
9	496	39	10417809568016	69	608578303898604406167840
10	1066	40	22388184630824	70	1080229099508551381463536
11	2460	41	50301508651032	71	1929269192569465070403584
12	6128	42	113605533519568	72	3452997322628833453585008
13	12840	43	265157938869936	73	7096327095079914521075040
14	29380	44	622473467900178	74	14611112240136930804928288
15	74904	45	1527398824248200	75	30235147387260979648843264
16	212728	46	3784420902143392	76	62757445134327428602306464
17	368016	47	9503564310606436	77	132956581436718531491070160
18	659296	48	23991783779046768	78	282272593229156186280461264
19	1371056	49	48820872045382552	79	605672649054377049472147568
20	2937136	50	99986771685259808	80	1302375489530691442230524528
21	6637232	51	209179575852808848	81	2914298247043287576460093712
22	15616616	52	441563057878399888	82	6537258415569149903366841040
23	38431556	53	992063519708141728	83	14713284774210886488265138336
24	96547832	54	2241540566114243168	84	33155372641605493828236640928
25	198410168	55	5185168615770591200	85	77219028670778815210019118736
26	419141312	56	12057653703359308256	86	180104653062631494787580542664
27	941812088	57	31151270610676979624	87	421733920870430143234318231648
28	2181990978	58	81046346414827952010	88	990082990967384066255452324186
29	5624657008	59	213208971281274232760	89	2428249522507620383597702223224
30	14765405996	60	563767895033816986864	90	5963505178650560845887322154368

Table 1: Number of 3-free permutations $\theta(n)$ of $[n]$

all $n \geq 42$ follows from using the fact that it holds for $42 \leq n \leq 83$ as a basis for an inductive argument and from Theorem 8. Straightforward numerical investigation reveals that inequality (3) actually holds for $n \geq 36$ (but not for $n \leq 35$). $\square$

Proof. A proof of Theorem 4 follows by induction on k using Theorem 6. $\square$

To prove Theorem 5, we rework the reasoning of Section 3 of [9] through the proof of Corollary 3.2.1 using an exponential base $\alpha > 2$ and the values of $\theta(n)$ in Table 4. We obtain improved variants of Theorems 3.1 and 3.2 of [9] along the way. First note that:

(a) If for some positive integer $n, \theta(n) \geq \alpha^n$ and $\theta(n+1) \geq \alpha^{n+1}$ then by Theorem 6, we have $\theta(2n) \geq 2\alpha^{2n}$ and $\theta(2n+1) \geq 2\alpha^{2n+1}$.

By computer verification and the data in Table 4 , we see that $\theta(n) \geq \alpha^n$ for $n \in [40, 79]$ for $\alpha = c_4$ and that c_4 is the maximal such value. Thus, by (a), $\theta(n) \geq 2\alpha^n$ for $n \in [80, 159]$. Applying (a) to this inequality yields $\theta(n) \geq 8\alpha^n$ for $n \in [160, 319]$. An inductive argument allows us to prove the following improvement on Theorem 3.1 of [9].

Theorem 12. *For integers $p \geq 2$ and $\alpha = c_4$,*

$$\theta(n) \geq 2^{2^{p-2}-1}\alpha^n \text{ for all } n \in \left[5 \times 2^{p+1}, 5 \times 2^{p+2} - 1\right]. \tag{14}$$

Proof. We know the statement is true for $p = 2$. Suppose the statement holds for all $p \leq l - 1$. Then for $n \in \left[5 \times 2^{l+1}, 5 \times 2^{l+2} - 1\right]$ if n is even, applying the inductive hypothesis to $\frac{n}{2}$ and using Theorem 6 verifies the theorem for n (similarly for n odd applying the induction hypothesis to $\frac{n-1}{2}$ and $\frac{n+1}{2}$). This verifies the statement for $p = l$ as desired. $\square$

Next, we prove the following improvement over Theorem 3.2 of [9].

Theorem 13. *For any fixed integer $p \geq 5$ and $\alpha = c_4$,*

$$\lim_{n \to \infty} \frac{\theta(n)}{n^p \times \alpha^n} = \infty. \tag{15}$$

Proof. Consider the sequence $a_n = \frac{\theta(n)}{n^{p+1} \times \alpha^n}$ for $n \geq 5 \times 2^{p+1}$. Note that $a_{2n} = \frac{\theta(2n)}{(2n)^{p+1} \times \alpha^{2n}} \geq \frac{2 \times [\theta(n)]^2}{(2n)^{p+1} \times \alpha^{2n}} \geq \frac{\theta(n)}{n^{p+1} \times \alpha^n} \times \frac{\theta(n)}{\alpha^{n+p}} = a_n \times \frac{\theta(n)}{\alpha^{n+p}} \geq a_n$ (as $\theta(n) \geq 2^{2^{p-2}-1}\alpha^n$ and $2^{p-2} - 1 \geq p \log_2 \alpha$ for the intervals of n and p values). Similarly $a_{2n+1} \geq a_{n+1}$ for all such n (proof is identical with the additional step of noting that $(2n + 2)^{p+1} \geq (2n + 1)^{p+1}$). Let $\gamma = \min a_n$ for $n \in \left[5 \times 2^{p+1}, 5 \times 2^{p+2} - 1\right]$. Using the statements $a_{2n} \geq a_n$ and $a_{2n+1} \geq a_{n+1}$ recursively implies $a_n \geq \gamma$ for all $n \geq 5 \times 2^{p+1}$. Therefore $\frac{\theta(n)}{n^p \times \alpha^n} = n \times a_n \geq n \times \gamma$ for all $n \geq 5 \times 2^{p+1}$ and $\frac{\theta(n)}{n^p \times \alpha^n}$ clearly tends to ∞ as $n \to \infty$ as desired. $\square$

We now prove Theorem 5.

Proof. Let $a_n = \frac{\theta(n)}{n \times \alpha^n}$. From the values of $\theta(n)$ in Table 4 we note that $a_n \geq \frac{1}{40}$ for all $n \in [40, 79]$. Since $\theta(n) \geq \alpha^n$ for all $n \geq 40$ by Theorem 12, reasoning as in the proof of Theorem 13 lets us prove that $a_{2n} \geq a_n$ and $a_{2n+1} \geq a_{n+1}$ for all $n \geq 40$. This proves that $a_n \geq \frac{1}{40}$ for all positive integers n. □

6. Conjecture

Define the function $h(n) = log(\theta(n+1)) - log(\theta(n))$. Examining a plot of $h(n)$ suggests

Conjecture 1. *The function* $h(n)$ *is increasing on the intervals* $\left[2^k, 2^k + 2^{k-1} - 1\right]$ *and* $\left[2^k + 2^{k-1}, 2^{k+1} - 1\right]$ *but is decreasing on* $\left[2^k + 2^{k-1} - 1, 2^k + 2^{k-1}\right]$ *and the interval* $\left[2^{k+1} - 1, 2^{k+1}\right]$ *for* $k \geq 2$.

References

[1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, MIT Press, Cambridge, 2009.

[2] B. Correll Jr., The density of Costas arrays and three-free permutations, *Proc. IEEE Statist. Signal Process. Workshop* (2012), 492-495.

[3] J. Davis, R. Entringer, R. Graham, and G. Simmons, On permutations containing no long arithmetic progressions, *Acta Arith.* **34** (1977), 81-90.

[4] J. Dybizbański, Sequences containing no 3-term arithmetic progressions, *Electron. J. Combin.* **19** (2012), 1-5.

[5] R. C. Entringer and D. E. Jackson, Elementary problem 2440, *Amer. Math. Monthly* **82** (1973), 76-77.

[6] T. D. LeSaulnier and S. Vijay, On permutations avoiding arithmetic progressions, *Discrete Math.* **311** (2011), 205-207.

[7] R. C. Lyndon, Solution to problem E2440, *Amer. Math. Monthly* **82** (1975), 74-75.

[8] T. Odda, Solution to problem E2440, *Amer. Math. Monthly* **82** (1975), 74.

[9] A. Sharma, Enumerating permutations that avoid 3-free permutations, *Electron. J. Combin.* **16** (2009), 1-15.

[10] A. Sharma, *The structure of 3-free permutations*, PhD thesis, U. C. Berkeley, 2009.

[11] G. J. Simmons, Solution to problem E2440, *Amer. Math. Monthly* **82** (1975), 76-77.

[12] H. E. Thomas, Solution to problem E2440, *Amer. Math. Monthly* **82** (1975), 75-76.

[13] A. Sharma, *The Structure of 3-Free Permutations*, BiblioBazaar, Charleston, 2011.