Contributions to Discrete Mathematics

Volume 9, Number 1, Pages 63–69
ISSN 1715-0868

A SHORT NOTE ON INTEGER COMPLEXITY

STEFAN STEINERBERGER

Abstract. Let $f(n)$ be the minimum number of 1's needed in conjunction with arbitrarily many +,* and parentheses to write an integer n (i.e. $f(6) \leq 5$ since $6 = (1 + 1)(1 + 1 + 1)$). Selfridge has shown that $f(n) \geq 3 \log_3 n$ and Guy (using observations of Coppersmith and Isbell) has proven that $f(n) \leq 3.82 \log_3 n$ for 'generic' integers n. We improve on the classical Coppersmith-Guy-Isbell argument by considering a more general discrete dynamical system on $\mathbb{Z}_d$ for some arbitrary fixed $d \in \mathbb{N}$. As n becomes large, the dynamical system approximates a Markov chain whose behavior is explicitely computable. We consider the case when $d = 6$ and use it to prove $f(n) \leq 3.66 \log_3 n$ for generic integers.

1. Introduction

The following problem of Mahler and Popken [6] dates back to 1953: What is the minimum number of 1's necessary to write down an integer n if we can use an arbitrary number of $+, *$ and parentheses? For example, denoting the solution by $f(n)$ and omitting the $*$ whenever it is clear,

$$6 = (1 + 1)(1 + 1 + 1)$$

shows that $f(6) \leq 5$.

The above problem was popularized by Guy [3] who claims in a Monthly article from 1986 to have been periodically reminded of it by Erdős, Isbell and Selfridge. It was included as problem F26 in Guy's *Unsolved Problems in Number Theory* [4] and mentioned in Wolfram's book [9, p. 916]. Furthermore, by writing

$$f(n) = \min_{\substack{d|n \\ m \leq n}} \left\{ f(d) + f\left(\frac{n}{d}\right), f(m) + f(n-m) \right\},$$

the problem can be regarded as an easily stated model problem for studying the difficulties that arise when mixing additive and multiplicative behavior.

Received by the editors May 6, 2012, and in revised form January 12, 2014.
2010 *Mathematics Subject Classification.* 37A45.
Key words and phrases. Integer complexity.

Known results. Mahler and Popken proved a lower bound; a refined version of their argument is due to Selfridge [3], who showed inductively that for any $\theta \in \{-1, 0, 1\}$ the largest number that can be represented using at most $3k + \theta$ 1's is $3^k + \theta 3^{k-1}$. This immediately implies a lower bound of

$$f(n) \geq 3 \log_3 n$$

with equality for $3^n = (1+1+1)^n$. Srinivar and Shankar [8] used this argument to construct an algorithm for computing $f(n)$. Since this lower bound is attained infinitely often, we will write any upper bound as a logarithm in base 3 to simplify comparison.

A very attractive special case is the question whether

$$f\left(2^a 3^b\right) = 2a + 3b$$

is true. Note that, if true, this would immediately imply $f(n) \geq 3.16 \log_3 n$ for $n = 2^k$. The case when a, b are both nonzero and $a \leq 21$, this has been proven by Altman and Zelinsky [1]. However, the conjecture $f(2p) = \min(1 + f(p-1), 2 + f(p))$ for primes p was disproved by Iraids (see [5]) with the smallest counterexample being $p = 10278600694$. Results of a similar type are discussed in [5].

The following explicit construction is attributed to Coppersmith, writing $n = (a_0 a_1 \dots a_n)_2$ in binary and using Horner's scheme, we get

$$n = a_n + (1+1)(a_{n-1} + (1+1)(a_{n-2} + \dots$$

This representation requires the digit 1 at most $3 \log_2 n$ times and thus gives an upper bound of

$$f(n) \leq 3 \log_2 n \sim 4.754 \log_3 n.$$

According to Guy, it was first noted by Isbell that the bound $f(n) \leq 3 \log_2 n$ is rather pessimistic; the bound assumes that the binary representation of n consists of only 1's, which happens only for numbers of the form $n = 2^k - 1$. A 'typical' number will have roughly half of its digits equal to 0 and this shows that

$$f(n) \lesssim \frac{5}{2} \log_2 n \sim 3.962 \log_3 n$$

should be a more accurate estimate for most numbers. One can extend this thought to other bases and Guy notes that writing numbers in base 24 gives for a 'typical' number n that

$$f(n) \leq 3.819 \log_3 n.$$

This is the place to remark that the notion of 'typical' can be deduced from the above argument. A number is considered 'typical' or 'generic' if the frequency of the digits in base 24 appear to be fully random, i.e. no digit appears with significantly greater frequency than any other digit. If we consider the set

$$A_k = \{n \in \mathbb{N} |\ n \text{ has precisely } k \text{ digits in base } 24\},$$

then there is a direct bijection between A_k and

$$\{1,\dots,23\} \times \{0,\dots,23\}^{k-1}.$$

If we define for $0 \leq i \leq 23$ the function $g_i(n)$ as the number of times the digit i appears in the representation of n in base 24 and μ_k as the normalized counting measure on A_k, then for any $\varepsilon > 0$

$$\lim_{k\rightarrow\infty} \mu_k \left\{ n \in A_k : \sup_{0 \leq i \leq 23} \left| g_i(n) - \frac{k}{24} \right| > \varepsilon \right\} = 0.$$

Or, stated differently, the subset of A_k where our average digit analysis is even slightly off becomes small compared to the full set. This detailed analysis of the Coppersmith-Isbell-Guy argument motivates the language of our result, where we show

$$f(n) \leq 3.66 \log_3 n$$

for 'generic' integers.

Theorem 1.1. *There exists a partition of the nonnegative integers into finite sets, i.e.*

$$\mathbb{N}_{>0} = \bigcup_{k=1}^{\infty} A_k \quad \text{and} \quad A_i \cap A_j = \emptyset$$

for $i \neq j$ with normalized counting measures μ_k such that

$$\lim_{k\rightarrow\infty} \mu_k \left\{ n \in A_k : f(n) > 3.66 \log_3 n \right\} = 0.$$

By elementary measure theory, the theorem is equivalent to the trivial statement that there exist infinitely many numbers for which $f(n) < 3.66 \log_3 n$ (trivial because this is true for the powers of 3). However, the proof is based on an explicit algorithm for computing a representation of n using only 1's and thus provides additional information on the sets A_k. The sets are explicitly given and both their size as well as their largest element grow exponentially in k (with different rates). Furthermore, we believe a stronger result holds and that the set

$$\{n : \text{our algorithm uses the digit 1 fewer than } 3.66 \log_3 n \text{ times}\}$$

has asymptotic density 1.

As previously mentioned, the difficulty stems from mixing additive and multiplicative behavior. Returning to Coppersmith's original argument, we see that writing $n = 2^{2k} - 1$ in binary requires the digit 1 at most $4.754 \log_3 n$ times. However, any number n of this form is a multiple of 3 and we see that

$$\frac{1}{3}(2^{2k} - 1) = (1,0,1,0,\dots,0,1)_2,$$

where the number has $2k-1$ digits in base 2. These numbers satisfy Isbell's average case argument and this immediately gives the improved bound $3.962 \log_3 n$ for numbers of this type. In general, it seems that one can expect that any result of the above type for 'generic' integers should also hold true for all sufficiently large integers because one should always be able to find three 'generic' integers a_1, a_2, a_3 of size $\log a_i \sim \log_3 n - 1$. Proving this rigorously, however, might pose considerable difficulty.

2. Preliminary heuristics

From Coppersmith's original argument, we can regard it as an iteration of the substitution rule

$$n = (n \bmod 2) + 2\left\lfloor \frac{n}{2} \right\rfloor$$

followed by replacing all 2's with (1+1). Heuristically, we would like to have an algorithm yielding a final result in the magnitude of $3\log_3 n$ - applied to substitution rules. This means we would ideally like to use the digit 1 three times if the next iteration step is at most a third of the size. Every additional digit means losing efficiency; applying this to the binary argument, a number needs either 2 or 3 digits (depending on parity) but should ideally only need $3\log_3 2$ digits. Clearly, if a number can be divided by 3, it should also be divided by 3 and this gives rise to the following substitution algorithm.

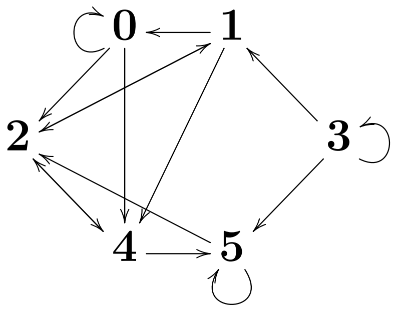
(1) Take an arbitrary number $n \in \mathbb{N}$. If $n \leq 5$, use the representations $2 = (1+1)$, $3 = (1+1+1)$, $4 = (1+1)(1+1)$ and $5 = (1+1)(1+1)+1$, otherwise determine $n \bmod 6$.
(2) If $n \bmod 6 = 0$, write it as $n = 3(n/3)$.
If $n \bmod 6 = 1$, write it as $n = 1 + 3((n-1)/3)$.
If $n \bmod 6 = 2$, write it as $n = 2(n/2)$.
If $n \bmod 6 = 3$, write it as $n = 3(n/3)$.
If $n \bmod 6 = 4$, write it as $n = 2(n/2)$.
If $n \bmod 6 = 5$, write it as $n = 1 + 2((n-1)/2)$.
(3) Repeat step (2) until 1 is reached.

This procedure is a greedy algorithm: Motivated by the heuristics above, in each step the algorithm compares the respective efficency of representations in both mod 2 and mod 3. For example, the case $n \bmod 6 = 4$ can be dealt with by writing either

$$n = 2(n/2) \quad \text{or} \quad n = 1 + 3((n-1)/3),$$

where the measure of wastefulness (digits used minus allowed number of digits given the size of the remainder) is $2 + 3\log_3 1/2 \sim 0.107\ldots$ and $4 + 3\log_3 1/3 = 1$ respectively. The first case outperforms the second and is thus chosen by the algorithm.

It is crucial for the subsequent argument that the sequence of possible states is restricted. Let us denote the six states of the algorithm by $\mathbf{0}, \ldots, \mathbf{5}$. Not every sequence of states is possible: If we are in case **2**, we are dealing with a number of the form $6n + 2$, which will then be reduced to a number $3k + 1$, this number can be either of the form $6n + 1$ or $6n + 4$, i.e. the state **2** can only be followed by either the state **1** or the state **4**. A complete list is given by the following diagram.



It is easily seen that this substitution rule maps the integers bijectively into the space of admissibile sequences, one can iteratively reconstruct the number. Suppose we are given the sequence $(\mathbf{3},\mathbf{5},\mathbf{2},\mathbf{1})$ - it ends in **1** and by following the arrows, one easily sees that it is admissible. We start with the number 1, the penultimate state is **2**. In state **2**, a number is written as $2(n/2)$, where $(n/2)$ is known to be 1. A number in state **5** is written as $1+2((n-1)/2)$, where the number in the parenthesis is known to be 2, thereby giving the number 5. A number in state **3** is written as $3(n/3)$ yielding the resulting number 15.

3. Proof of the Theorem

We begin the proof of Theorem 1.1 by defining the set

$$A_k = \{n \in \mathbb{N}| \text{ the substitution rules traverse } k-1 \text{ states before reaching } 1\}.$$

For example, $A_1 = \{1\}$, $A_2 = \{2,3,5\}$ and $A_3 = \{4,6,7,9,10,15\}$. We are interested in the asymptotic properties of the sequence of states through which a typical element in A_k travels, i.e. in the sequence of probability measures π_k on $\{0,\dots,5\}$ given by

$$\pi_k(i) = \mu_k\left\{n \in A_k | n \bmod 6 = i\right\}.$$

Studying this limit will be done by looking at the inverse time direction $k \to k-1$ for large k. Standard Markovian theory gives that the limit probability distribution π needs to satisfy

$$\pi = \begin{pmatrix} \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{3} & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{3} & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{3} & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \pi,$$

which implies

$$\pi = \left(\frac{1}{13}, \frac{2}{13}, \frac{4}{13}, 0, \frac{3}{13}, \frac{3}{13}\right)^T.$$

However, this immediately implies the average decay rates (state $\mathbf{0}$ is essentially multiplication with $1/2$, state $\mathbf{1}$ is essentially multiplication with $1/3$ and so forth) and this implies

$$\lim_{k \to \infty} \frac{1}{\# A_k} \sum_{n \in A_k} \log n = 2^{10/13} 3^{3/13} \sim 2.1961 \dots .$$

At the same time, the stationary distribution also implies the average cost of the algorithm (state $\mathbf{0}$ needs 3 digits, $\mathbf{1}$ needs 4 digits, ...) and thus the average number of digits used in each step is

$$2\left(\pi(\mathbf{2}) + \pi(\mathbf{4})\right) + 3\left(\pi(\mathbf{0}) + \pi(\mathbf{3}) + \pi(\mathbf{5})\right) + 4\pi(\mathbf{1}) = \frac{34}{13}.$$

Summarizing, the average cost is given by

$$f(n) \sim \frac{34}{13} \frac{\log 3}{\frac{10}{13}\log 2 + \frac{3}{13}\log 3} \log_3 n \sim 3.6522 \dots \log_3 n.$$

Showing that the average case is typical follows immediately from standard deviation theory. □

4. Concluding Remarks

The described method can be extended to any base. For instance, the original Coppersmith-Isbell-Guy argument of writing a number in base 24 can be understood as considering 24 states and using the trivial transition rule: Every state i is mapped to all other states with equal probability (because knowledge of a digit does not allow us to conclude anything about the preceding digit). However, as demonstrated by the fact that our refined dynamical system in base 6 improves on the trivial method in base 24 and since there is reason for the trivial method to be particularly effective in this larger framework, it is natural to assume that our method will always allow us to get better results.

The real remaining challenge is of a computational nature; specifically, how should one define the transition rule? Our approach was greedy and attempts to make every step as small and using as few digits as possible. There is no reason to assume that this approach is optimal. indeed, it might be better to accept a local loss by mapping the states less efficiently or by using more 1's than necessary to favorably change the global dynamics of the flow in the Markov chain for an overall improvement. Due to the nontrivial amount of computation, we do not know whether this phenomenon is observable and consider it to be an interesting problem.

We remark that since this is the runtime analysis of a $\mathcal{O}(\log n)$ algorithm, one can very well see it in practice. Let us conclude the paper by noting that the lower bound has not been improved in the slightest. It has been suspected and seems reasonable, but it is unknown whether

$$f(n) \sim 3 \log_3 n$$

is false.

Acknowledgements

I am grateful to Richard Guy for pointing out [5].

References

1. H. Altman and J. Zelinsky, *Numbers with integer complexity close to the lower bound*, Integers **12** (2012), no. 6, 1093–1125.
2. J. Arias de Reyna, *Complexity of the natural numbers. (Spanish)*, Gac. R. Soc. Mat. Esp. **3** (2000), no. 2, 230–250.
3. R. K. Guy, *Unsolved problems: Some suspiciously simple sequences*, Amer. Math. Monthly **93** (1986), no. 3, 186–190.
4. ———, *Unsolved problems in number theory*, 3rd ed., Springer, 2004.
5. J. Iraids, K. Balodis, J.Cerenoks, M. Opmanis, R. Opmanis, and K. Podnieks, *Interger complexity: Experimental and analytical results*, Scientific Papers University of Latvia, Computer Science and Information Technologies **787** (2012), 153–179.
6. K. Mahler and J. Popken, *On a maximum problem in arithmetic. (Dutch)*, Nieuw Arch. Wiskd. **3** (1953), no. 1, 1–15.
7. D. Rawsthorne, *How many 1's are needed?*, Fibonacci Quart. **27** (1989), no. 1, 14–17.
8. V. V. Srinivas and B. R. Shankar, *Integer complexity: Breaking the $\theta(n^2)$ barrier*, World Academy of Science, Engineering and Technology **2** (2008), no. 5, 454 – 455.
9. S. Wolfram, *A new kind of science*, 1st ed., Wolfram Media, 2002.

Department of Mathematics, Yale University, 10 Hillhouse Avenue, New Haven, CT 06520, USA
E-mail address: stefan.steinerberger@yale.edu