Computing Numbers in Section I of the Totient Iteration

T. D. Noe

November 18, 2008

1 Introduction

Let $\phi(x)$ denote Euler's totient function. Defining $\phi^0(x) = x$, the iterated totient function is defined recursively for $n > 0$ by $\phi^n(x) = \phi(\phi^{n-1}(x))$. For $x > 1$, $\phi(x) < x$. Hence, for some n we will have $\phi^n(x) = 2$. That x is said to be in class n, and we define the function $C(x) = n$. We define $C(1) = 0$.

New results about the totient iteration are in the paper [2]. Recently, Theorem 4 of that paper was disproved, which put many of the paper's computational results in question. The purpose of this paper is to introduce a new method for computing the smallest numbers in each class of the totient iteration. We use this method to calculate the least number in each class up to 1000, enabling us to check the results in the original paper.

2 Section I Numbers

The numbers in class n between 2^n and 2^{n+1} are called section I numbers. Because the smallest numbers in a class are all section I numbers, we will concentrate on those numbers. Many properties of numbers in classes of the totient iteration are listed in Noe [2, Section 2]. Two theorems are key to generating section I numbers:

Theorem 1 (Noe [2, Theorem 1])**.** *Suppose p is an odd prime and $p = 1 + 2^k m$, with $k > 0$ and m odd. Then p is in section I of its class if and only if m is in section I of its class.*

Theorem 2. *If integers x and y are in section I of their respective classes $C(x)$ and $C(y)$, and if $xy < 2^{C(x)+C(y)+1}$, then xy is in section I of class $C(x) + C(y)$.*

Proof. Because x and y are in section I, they are odd numbers. Hence $C(xy) = C(x)+C(y)$. By definition, any number in class n and less than 2^{n+1} is in section I. Hence, xy is in section I. $\square$

By Shapiro [4, Theorem 15], the factors of every section I number are section I numbers.

3 Section I Algorithm

These two theorems enable us to construct an algorithm for generating section I numbers. Let's stipulate that the number 1 is in section I of class 0. Also suppose that the sets S_i contain the section I numbers in class i for $0 \le i < n$. Given a set S, we denote by $P(S)$ the set of primes in S and by $L(S,x)$ the set numbers in S less than x. Now consider how the prime and composite numbers in section I of class n can be generated.

By theorem 1, the prime numbers in section I of class n are

$$L\Big(P\Big(\bigcup_{k=1}^{n}\Big(1+2^k S_{n-k}\Big)\Big), 2^{n+1}\Big),$$

where the notation $1+2^k S_{n-k}$ means that each element of the set S_{n-k} is multiplied by 2^k and then incremented by 1. By theorem 2, the composite numbers in section I of class n are

$$L\Big(\bigcup_{k=1}^{n}\Big(S_k\ S_{n-k}\Big), 2^{n+1}\Big),$$

where $S_k\ S_{n-k}$ is the set of all products of two numbers, one from each set. Hence, the set of section I numbers in class n is the union, which can be written

$$S_n = L\Big(P\Big(\bigcup_{k=1}^{n}\Big(1+2^k S_{n-k}\Big)\Big) \bigcup \Big(\bigcup_{k=1}^{n}\Big(S_k\ S_{n-k}\Big)\Big), 2^{n+1}\Big).$$

In the implementation of this algorithm, for efficiency, the L and P functions would be applied to each element of $1+2^k S_{n-k}$ and $S_k\ S_{n-k}$ when the element is computed.

This algorithm works well for about $n \le 80$. It is certainly much faster than the naive approach of computing $\phi(x)$ for every x between 2^n and 2^{n+1}. However, as shown by sequence A092878 in Sloane [5], the size of the set S_n grows almost exponentially. Therefore, to have any chance of computing the smallest 100 numbers in S_{1000}, we need a different algorithm.

4 Section α Numbers

For a positive number $\alpha \le 1$, let's define the numbers in class n between 2^n and $2^{n+\alpha}$ to be section α numbers. For $\alpha = 1$, we obtain the usual section I numbers. Indeed, for a given class n, every section α number is a section I number. Hence, all the properties of section I numbers can be extended to section α numbers. For example, all section α numbers are odd. Theorems 1 and 2 are easy to extend to section α:

Theorem 3. *Suppose p is an odd prime and $p = 1 + 2^k m$, with $k > 0$ and m odd. Then p is in section α of its class if and only if m is in section α of its class.*

Theorem 4. *If integers x and y are in section α of their respective classes $C(x)$ and $C(y)$, and if $xy < 2^{C(x)+C(y)+\alpha}$, then xy is in section α of class $C(x) + C(y)$.*

The proofs of these two theorems is an easy exercise. Below we state and prove the analogue of Shapiro's Theorem 15. The proof is essentially the same as Shapiro's, but with 1 replaced by α.

Theorem 5. *If integer x is in section α of its class, then every divisor of x is in section α of its class.*

Proof. Since x is in section α of its class, it is odd. The theorem is obviously true for prime x, so we need consider only composite x. Let d be a proper divisor of x so that $x > d$ and $x = ds$. The fact that x is in section α implies

$$2^{C(x)+\alpha} = 2^{C(ds)+\alpha} > x > 2^{C(ds)} = 2^{C(x)}.$$

Since both d and s are odd, we have

$$ds < 2^{C(ds)+\alpha} = 2^{C(d)+C(s)+\alpha}.$$

Suppose that d is not in section α of its class. Then by definition, we would have $d > 2^{C(d)+\alpha}$. However, since $s > 2^{C(s)}$, we would have $ds > 2^{C(s)+C(s)+\alpha}$, which is a contradiction. $\square$

5 Section α Algorithm

These new theorems enable us to construct an algorithm for generating section α numbers. Again, the number 1 is in section α of class 0. Also suppose that the sets S_i^α contain the section α numbers in class i for $0 \le i < n$. As before, given a set S, we denote by $P(S)$ the set of primes in S and by $L(S, x)$ the set numbers in S less than x. The prime and composite numbers in section α of class n can be generated in a similar way as before.

By theorem 3, the prime numbers in section α of class n are

$$L\Big(P\Big(\bigcup_{k=1}^{n}\Big(1 + 2^k S_{n-k}^\alpha\Big)\Big), 2^{n+\alpha}\Big).$$

By theorem 4, the composite numbers in section α of class n are

$$L\Big(\bigcup_{k=1}^{n}\Big(S_k^\alpha\ S_{n-k}^\alpha\Big), 2^{n+\alpha}\Big).$$

The set of section α numbers in class n is the union

$$S_n = L\Big(P\Big(\bigcup_{k=1}^{n}\Big(1 + 2^k S_{n-k}^\alpha\Big)\Big) \bigcup \Big(\bigcup_{k=1}^{n}\Big(S_k^\alpha\ S_{n-k}^\alpha\Big)\Big), 2^{n+\alpha}\Big).$$

For a fixed n, as we make α smaller, this algorithm produces smaller and smaller sets S_n^α. In fact, it is easy to see that if $\alpha < \beta$, then $S_n^\alpha \subseteq S_n^\beta$. However, for a fixed value of α, we eventually see the same sort of exponential-like growth of the size of the sets S_n^α as n increases. Note that if 65537 is the last Fermat prime, any $\alpha < \log_2 65537 - 16 \approx 0.000022$. will generate only empty sets.

6 Using Nonincreasing Values of α

Suppose that we want to compute only the M smallest numbers in S_n, the numbers in section I of class n. How can we accomplish this task? We propose the following algorithm:

1. Start with $n = 0$ and $\alpha = 1$.

2. Increment n and compute S_n^α.

3. If the size of S_n^α is greater than M, decrease α.

4. Go to step 2.

Note that this algorithm requires that successive values of α be nonincreasing (otherwise, for example, we could switch to $\alpha = 1$ at any time and compute all the section I numbers). The key to the successful operation of this algorithm is step 3. However, if the choice of α generates a set S_{n+1}^α with too few elements, α can be adjusted upwards and S_{n+1}^α can be recomputed.

This algorithm was implemented in *Mathematica*. We chose $M = 400$ in order to find the smallest prime number in all classes $n \leq 1000$. The final value of α was 0.00047. The computation required several hours on a Macintosh G5 computer. In addition, all primes were proved prime by either *Mathematica's* ProvablePrimeQ function or the PRIMO software [3], a task requiring several more hours. The results were added to OEIS sequences A007755, A060611, A092873, A098196, A136040, A145443 as b-files. Sequences A092873 and A136040 were corrected and sequences A092878 and A135833 were extended.

References

[1] P. A. Catlin, Concerning the iterated ϕ function, *Amer. Math. Monthly*, **77** (1970), 60–61.

[2] Tony D. Noe, Primes in classes of the iterated totient function, *J. Integer Sequences*, **11** (2008), Article 08.1.2.

[3] Marcel Martin, Primo 3.0.5 prime proving software, (2008).

[4] Harold Shapiro, An arithmetic function arising from the ϕ function, *Amer. Math. Monthly* **50** (1943), 18–30.

[5] N. J. A. Sloane, *The On-Line Encyclopedia of Integer Sequences*, published electronically at www.research.att.com/~njas/sequences.