Computational Aspect of Smarandache's Function

Sabin Tabirca **Tatiana Tabirca**

Abstract: *The note presents an algorithm for the Smarandache's function computation. The complexity of algorithm is studied using the main properties of function. An interesting inequality is found giving the complexity of the function on the set {1,2,...,n}.*

1. Introduction

In this section, the main properties of function are reviewed. The Smarandache's function notion reported for the first time in [1]. The main results concerning the function can be found in [1], [2].

The function $S: N \to N$ defined by $S(n) = \min\{k | k! \vdots n\}$ is called Smarandache's function. This concept is connected with the prime number concept, because using the prime numbers an expresion for the function is given. The important properties that are used in this paper, are showed bellow.

1. For all $n \in N$, the inequality $S(n) \le n$ is true. When n is a prime number, the equality is obtained .
2. If $n = p_1^{k_1} \cdot p_2^{k_2} \cdot \ldots \cdot p_m^{k_m}$ is the prime number decomposition then $S(n) = \max\{S(p_1^{k_1}), S(p_2^{k_2}), \ldots, S(p_m^{k_m})\}$. (1)
3. The inequality $S(p^k) \le p \cdot k$ is true, if p is a prime number. (2)

A lot of conjectures or open problems related ot the Smarandache's function appear in the number theory. Several such problems have been studied using computersand reported in relevant literature, e.g.[3], [4]. Using the computer

2. An Algorithm for the Smarandache function

In the following, an algorithms for computing the function S is presented. The main idea of the algorithm is to avoid the calculations of factorial, because the values of $n!$, for $n>10$ are a very big number and cannot be calculated using a computer.

Let $(x_k)_{k\ge 1}$ a sequence of integer numbers defined by $x_k = k! \bmod n, (\forall) n > 0$. Using the definition of sequence, the following equations can be written:

- $x_1 = 1$ (3)
- $x_{k+1} = (k+1)! \bmod n = (k+1)k! \bmod n = (k+1)x_k \bmod n$. (4)

Based on the linear equation (4), S can be calculated as $S(n) = \min\{k | k! \vdots n\} = \min\{k | x_k = 0\}$.

The algorithm for $S(n)$ performs the computation of $x_1, x_2, \ldots, x_k$ until the 0 value is found. The PASCAL description of this algorithm is given bellow.

```
function S(n:integer):integer;
var
 k,x:integer;
begin
 x:=1;k:=1;
 while x<>0 do
   begin
     x:=x*(k+1) mod n;
     k:=k+1;
   end;
 s:=k;
end;
```

An analysis for the complexity of algorithm is presented in the following. The work-case complexity and the average complexity are studied.

Theorem 2.1

If $n = p_1^{k_1} \cdot p_2^{k_2} \cdot \ldots \cdot p_m^{k_m}$ then the complexity of the algorithm for computing S(n) is $O(\max\{p_1 k_1, p_2 k_2, \ldots, p_m k_m\})$.

Proof

The algorithm computes the value S(n) generating the sequence $x_1, x_2, \ldots, x_{S(n)}$. The number of operation has the complexity *O(S(n))*.

Based on (1) and (2), the following inequality holds

$$S(n) = \max\{S(p_1^{k_1}), S(p_2^{k_2}), \ldots, S(p_m^{k_m})\} \leq \max\{p_1 k_1, p_2 k_2, \ldots, p_m k_m\}. \quad (5)$$

Therefore, it can be concluded that the complexity of computing S(n) is $O(\max\{p_1 k_1, p_2 k_2, \ldots, p_m k_m\})$.

Other aspect of complexity is given by the average operations number. Assume the value S(k) is generated, where k is a number between 1 and n. This value can be computed with S(k) operations. Therefore the process takes S(1) operations for the value S(1), S(2) operations for the value S(2), ...,a.s.o. The average of the numbers operations is $\overline{S} = \frac{1}{n}\sum_{i=1}^{n} S(i)$ and gives an other estimation for the complexity of algorithm.

In the following, a possible upper bounds for $\overline{S}$ are sought. Obviously, $\overline{S}$ verifies the simple inequality

$$\overline{S} = \frac{1}{n}\sum_{i=1}^{n} S(i) \leq \frac{1}{n}\sum_{i=1}^{n} i = \frac{n+1}{2} = \frac{1}{2}n + \frac{1}{2}. \quad (6)$$

Inequality (6) can be improved using the strong inequalities for *S(i)*.

Lemma 2.1.

If $i>2$ is an integer number the following inequalities hold

1. $S(2i)+S(2i+1)\leq 3i+1$. (7)
2. $S(2i-1)+S(2i)\leq 3i-1$. (8)

Proof

Assuming $i>2$ folow $S(2i) \le i$. Applying this result both inequality are true.

Based on lemma 2.1, we can found an upper bound for the $\overline{S}$ better than in (6).

Theorem 2.2

If $n>5$ is a integer number then the inequality $\overline{S} = \frac{1}{n}\sum_{i=1}^{n} S(i) \le \frac{3}{8}n + \frac{1}{4} + \frac{2}{n}$ is true.

Proof

Two cases are considered to prove the inequality.

Case 1: n=2m

$$n\overline{S} = \sum_{i=1}^{n} S(i) = \sum_{i=1}^{n/2}[S(2i-1)+S(2i)] = S(1)+S(2)+S(3)+S(4)+\sum_{i=3}^{n/2}[S(2i-1)+S(2i)]$$

Based on (8) it can be derived that

$$n\overline{S} = 9+\sum_{i=3}^{n/2}[S(2i-1)+S(2i)] \le 9+\sum_{i=3}^{n/2}(3i-1) = 2+\sum_{i=1}^{n/2}(3i-1) = 2+\frac{3}{2}\frac{n}{2}(\frac{n}{2}+1)-\frac{n}{2} =$$

$$= 2+\frac{3}{2}\frac{n}{2}(\frac{n}{2}+1)-\frac{n}{2} = \frac{3}{8}n^2+\frac{1}{4}n+2.$$

Dividing by n, we obtain the inequality $\overline{S} \le \frac{3}{8}n+\frac{1}{4}+\frac{2}{n}$. (9)

Case 2: n=2m+1

$$n\overline{S} = \sum_{i=1}^{n} S(i) = S(1)+\sum_{i=1}^{(n-1)/2}[S(2i)+S(2i+1)] = S(2)+S(3)+S(4)+S(5)+\sum_{i=3}^{(n-1)/2}[S(2i)+S(2i+1)]$$

Using (7), it is found

$$n\overline{S} = 14+\sum_{i=3}^{(n-1)/2}[S(2i)+S(2i+1)] \le 14+\sum_{i=3}^{(n-1)/2}(3i+1) = 3+\sum_{i=1}^{(n-1)/2}(3i+1) = 3+\frac{3}{2}\frac{n-1}{2}\frac{n+1}{2}+\frac{n-1}{2} =$$

$$= 3+\frac{3}{8}(n^2-1)+\frac{n-1}{2} = \frac{3}{8}n^2+\frac{1}{2}n+\frac{17}{8}.$$ Thus, $\overline{S} \le \frac{3}{8}n+\frac{1}{2}+\frac{17}{8n}$. (10)

From (9) and (10), it is found $\overline{S} \le \frac{3}{8}n+\min\{\frac{1}{4}+\frac{2}{n},\frac{1}{2}+\frac{17}{8n}\} \le \frac{3}{8}n+\frac{1}{4}+\frac{2}{n}$.

3. Final Remarks

1. Based on theorem 2.2 we can say that the average operations number for computing the Smarandache's function is less than $\frac{3}{8}n+\frac{1}{4}+\frac{2}{n}$.
2. The upper bound $\frac{3}{8}n+\frac{1}{4}+\frac{2}{n}$ improves the previous bound $\frac{1}{2}n+\frac{1}{2}$.
3. The improving process can be extended using other sort of inequalities give the prime numbers 2 and 3. A lemma as similar as lemma 2.1 finds the upper bounds the sum of sixth consecutive terms of Smarandache's function.
4. Using the algorithm for computing the function S, the Smarandache's function can be tabulated. The values S(1),...,S(n) for all n <5000 can be found. The algorithm should be reviewed to be able to compute the Smarandache function for the big numbers.

References

1. F. Smarandache, *A Function in the Number Theory*, An.Univ. Timisoara, Vol XVIII, 1980.

2. F Smarandache, *An Infinity of Unsolved Problems Concerning a Function in the Number Theory*, Presented at The 14th American Romanian Academy Annual Convention, Los Angeles, 1989.

3. H.Ibstedt, *Surfing on the Ocean of numbers - a few Smarandache Notions and Similar Topics*, Erhus University Press, 1997.

4. Personal Computer Word, January 1996.