International Journal of Pure and Applied Mathematics
Volume 97 No. 4 2014, 377-393
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
doi: http://dx.doi.org/10.12732/ijpam.v97i4.1

ENTROPY EFFECT IN QUANTUM COMPUTING AND INFORMATION: AN OPEN-SOURCE ENVIRONMENT SIMULATION

R.H. Moretti[1], M.F. Borges[2] §, J.M. Machado[3], C. Brandão[4]
[1,2,3,4]Department of Computing
São Paulo State University - UNESP
São José do Rio Preto Campus
15054-000 São José do Rio Preto, BRAZIL

Abstract: The technologies are rapidly developing, but some of them present in the computers, as for instance their processing capacity, are reaching their physical limits. It is up to quantum computation offer solutions to these limitations and issues that may arise. In the field of information security, encryption is of paramount importance, being then the development of quantum methods instead of the classics, given the computational power offered by quantum computing. In the quantum world, the physical states are interrelated, thus occurring phenomenon called entanglement. This study presents both a theoretical essay on the merits of quantum mechanics, computing, information, cryptography and quantum entropy, and some simulations, implementing in C language the effects of entropy of entanglement of photons in a data transmission, using Von Neumann entropy and Tsallis entropy.

AMS Subject Classification: 81P45, 62B10, 94A17
Key Words: entanglement, entropy, computing and quantum information

Received: December 13, 2013

§Correspondence author

1. Introduction

The constant search for improved performance and processing boosted technological development that we see today. Following the classical architecture of Von Neumann and Moore's Law, computers are coming to their physical limit, since the higher the processing speed, the smaller the size of the components and the distance between them, where classical physics can't explain correctly the phenomena which eventually occur.

Quantum computing is a possible solution since it is based on the laws of quantum physics that govern the micro-universe and its consequences. With the advent of quantum computing, a new front opens, displaying a wide range of problems to be solved, as for example, in the area of the information security trough the efficiency of the algorithms.

In the area of security we have the encryption of data, since the "classics" methods are based on factoring large prime numbers and so, by the processing power of conventional machines are virtually impossible to be discovered. In the quantum field that is not true, since the processing is alarmingly higher than the conventional one, putting in check the best classical cryptographic methods. Thus, it becomes necessary to develop quantum cryptographic methods to remedy this need as important in electronic media.

Being a relatively new area, the degree of knowledge of quantum phenomena that occurs is still relatively small and therefore the development goes in short steps, beyond the fact that practically there is indeed a quantum computer. Through knowledge of the behavior of photons, the researchers will have a greater mastery of the medium, allowing an improvement of technical proposals to increase security in quantum field.

In the quantum computing, differently from the classics, the physical states are interrelated, through a phenomenon named entanglement. The goal of this work is to present a theoretical foundation of entropy and entanglement, also showing simulations about the effect of entranglement entropy of photons in a data transmission [4]. This will be done using Tsallis entropy [17] and Von Neumann entropy [11].

The results will be analyzed to get an idea of the behavior of photons when interact in different amounts. To perform the simulation, we used the C language to the calculations, compiling programs on open-source Linux environment using GNU-GCC compiler also *open-source*.

1.1. Entanglement

A phenomenon that is not so easy to explain but that is very important in quantum mechanics, is the so named entanglement [8] [12] [13] [18]. It is experimentally illustrated by testing pairs of slits [19], proving that there is a nonlocal correlation between the photon and a detector. This correlation that is created when these elements are together, still remain, even if they are separated by great distances. Thus we can not deal with the separate parts, but with a single system.

Historically, quantum entanglement has been much discussed by the EPR paradox in 1935 [9], which questioned the ability of quantum mechanics to describe completely the reality of physical events. Later on related work have been developed, with textitasis on the work known as Bell inequalities [3] [2]. In 1982 [1], an experiment showed that quantum mechanics can describe so fully reality by performing tests of Bell inequalities and admitting non-local correlations.

Nowadays, the entanglement has many applications such as teleportation of quantum states, quantum cryptographic protocols and superdense coding. Briefly, the quantum teleportation is the transmission of a quantum state between two locations without displacement of the distance that separates them; encryption protocols make use of quantum entanglement to ensure the security of communications.

Formally, according to Brandão [5] "the concept of entanglement is defined as a quality of all physical state that can not be represented as a simple tensorial product of the elements of multiplied Hilbert spaces". If the matrix density of an array subsystem is different than the density of a pure state, we say that this subsystem is entangled, with the definition given by entanglement negation, ie:

$$\psi_{ab} \neq |\psi_a > \otimes |\psi_b > \tag{1}$$

$$\rho_{ab} \neq |\psi_{ab} >< \psi_{ab}| \tag{2}$$

In order to quantify the entanglement, we use the entropy as a measure for this, explained in the following section.

1.2. Entropy

Entropy [7] is the term given to a degree of caocity of a system, widely applied in thermodynamics. Its representation is given by the letter S, being a function

of system state.

Its definition and applicability was going through areas from thermodynamics to the Telecommunications. Claude E. Shannon [14] entropy (equation 3) is a measure to help the economy of information transmission and storage. Over time there was a weight gain of entropy in dynamical systems, resulting in the non-extensive entropy of Constantino Tsallis [16] [15] [4].

In this work we will use the entropy of Von Neumann and the entropy of Tsallis in order to quantify the quantum entanglement.

$$S(X) \equiv S(p_1, p_2, ..., p_n) \equiv -1 \sum_{i}^{W} p_i \, log_2 \, p_i \tag{3}$$

Shannon Entropy

1.3. Von Neumann Entropy

Concerned to quantum mechanics, Von Neumann entropy [11] has an analogous concept of Shannon entropy, which is a probability distribution of the measurement uncertainty. Being ρ the density operator, the Von Neumann entropy associated with the state is given by (4).

$$S(\rho) \equiv -Tr(\rho \, log_2 \, \rho) \tag{4}$$

1.4. Tsallis Entropy

The Tsallis entropy possibly presents a generalization of Boltzmann-Gibbs entropy [15], consistent with the second law of thermodynamics and with strong adaptation to different physical systems. It can be an appropriate measure for the quantification of information in dynamic systems that have non-extensive features. It is given by (5).

$$S_q = k \, \frac{1 - \sum_{i}^{W} p_i^q}{q - 1} \tag{5}$$

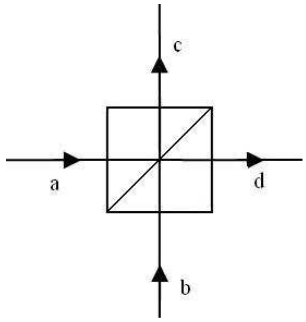

Figure 1: *Beam Splitter: mirror of slim reflexive envoltory, causing some refraction of light due to its structure, which acts as a bulkhead crystal, that have an entry of light beams (a and b) and output of entangled states (c and d).*

1.5. Entanglement in Beam Splitter

The experiments simulated in this work are related on observing what happens in the beam splitter, ilustrated by the figure 1, through which the photons pass and when passing, the respective waves are divided and a portion is transmitted and the other reflected waves being engaged with each other interference (overlap). In these simulations are taken Fock states of the input fields, these states with a definite number of photons.

The entry and exit operators are the pairs a and c to the gate 1 and the pairs b and d to the gate 2, belongs to Hilbert space and have the resulting terms through the coefficients T and R relating to the transmission and reflection, respectively, with norm equal to 1. The phase difference between transmission and reflection is given by ϕ and we have that $T = cos\frac{\theta}{2}$, $R = sen\frac{\theta}{2}$. Have the overall matrix of beam splitter given by (6):

$$B = e^{i\phi_0} \begin{pmatrix} \cos\theta e^{i\phi_T} & \sin\theta e^{i\phi_R} \\ -\sin\theta e^{-i\phi_R} & \cos\theta e^{-i\phi_T} \end{pmatrix} \tag{6}$$

The operators of the output field are given by (7):

$$c = BaB^{\dagger} \quad \text{e} \quad d = BbB^{\dagger} \tag{7}$$

Considering Fock states of independent input, the output state $|\psi>$ is given by:

$$|\psi >= B|n_1 n_2 >= \sum_{N_1 N_2} < N_1 N_2|B|n_1 n_2 > |N_1 N_2 >= \sum_{N_1 N_2} B_{n_1 n_2}^{N_1 N_2}|N_1 N_2 > \tag{8}$$

where

$$B_{n_1 n_2}^{N_1 N_2} = e^{-i\theta(n_1 - N_1)} \sum_{k=0}^{n_1} \sum_{l=0}^{n_2} (-1)^{n_1 - k} R^{n_1 + n_2 - k + l} T^{k+l} \times \times \frac{\sqrt{n_1! n_2! N_1! N_2!}}{k!(n_1 - k)! l! (n_2 - l)!} \times \delta_{N_1, n_2 + k - l} \delta_{N_2, n_1 - k + l} \tag{9}$$

This state is the superposition of input states, where delta is the Kronecker function. The entangled state output has dimension $n_1 + n_2 + 1$, where $n_1 + n_2$ is the sum of the number of photons of input. Thus, using the density operator reduced ρ_c, the Von Neumann entropy ($S(\rho_c)$) and Tsallis entropy ($S_q(\rho_c)$) are given by:

$$S(\rho_c) = - \sum_{N_1 N_2} |B_{n_1 n_2}^{N_1 N_2}|^2 \; ln \; |B_{n_1 n_2}^{N_1 N_2}|^2 \tag{10}$$

$$S_q(\rho_c) = \frac{1}{q-1} \left(1 - \sum_{N_1 N_2} |B_{n_1 n_2}^{N_1 N_2}|^{2q} \right) \tag{11}$$

2. Implementation

Since we have defined Von Neumann and Tsallis entropies (equations 10 e 11), we start to implement the simulation of them. To do so, following an arbitrary adoption and comparing with the results obtained by Brandão [5], were used entropic indices $q = 1$ to Von Neumann entropy, $q = 0.5$ to Tsallis entropy, taking $r = \sqrt{R}$, $t = \sqrt{1 - r^2}$, setting the entry photons as $a = n_1$ e $b = n_2$, and the exit photons as $c = N_1 = n_2 + k - l$ e $d = N_2 = n_1 - k + l$.

The methodology used was to divide and conquer, organizing functions in the component parts of the formulas of entropy, which in composition ultimately determine their values. This approach was chosen to improve the code structure, since these formulas have a large number of operations to be performed.

The philosophy adopted for the choice of programs is based on open source. We adopt the C language to implement the codes because it fits so many different libraries for different purposes. These open source libraries containing

an excellent free compiler GCC, provided with several options for mounting and creation of executables, are also widely used in scientific circles. For the generation of results, the program creates two files named *vetor-von.txt* and *vetor-tsallis.txt*, with a row vector containing the values of the entropies calculated. These values are in scientific format, for example $1.72e320$. The number of existing values corresponds to the precision being defined in the project,with 1000 points to tests with smaller entries and 100 points to tests with bigger entries.

Any numerical handling implementation, with the exception of variable loop control was made using the variables of the MPFR library, and in order to keep the accuracy, no rounding or truncation were made by the language patterns C.

The use of this library has become necessary, once that the work manipulates numbers bigger than the limit of the larger variable of C language, the *long double* $(1.7e \pm 308)$. It also has better implementations of complexity and efficiency that the library *math.h*, language default. For example, the function $mpfrfacui$, which performs operations of factorials in a much shorter time, in comparison with the default implementation, where above a n equal to 120 the trivial algorithm becomes infeasible.

2.1. Linux

The operating system chosen was the Ubuntu GNU / Linux version 10.04, of a free open source. This distribution is based on Debian and is now one of the most popular and disseminated among users. Through the great effort that Canonical Ltd, the sponsor of its development, many institutions and businesses use Ubuntu in their environments, since the system reliability is increased.

Available for multiple platforms, the Ubuntu Linux uses kernel with precompiled packages (which can also compile the package from source code) to provide scientific tools to games from the user, through the many free repositories available.

2.2. GNU MPFR

The MPFR library [10] serves to complement the default libraries of C language, being for use with multiple-precision computations with correct rounding. It has been continuously supported in their development, especially by INRIA, being based on the GMP multiple-precision library. It is distributed free of charge under GNU GPL version 3 or later.

To use this library in a **NIX* environment, just run the following procedures:

I. GMP library installation

[a.] Download of the library lastest version in the official website http://gmplib.org

[b.] Extraction of the container file:

```
$ tar -xvjf gmp-5.1.3.tar.bz2
```

[c.] Access to the folder where the files have been extracted:

```
$ cd gmp-5.1.3
```

[d.] Compilation and installation of the files:

```
$ ./configure && make
# make install
```

II. MPFR Library installation

[a.] Download of the library lastest version in the official website http://www.mpfr.org

[b.] Extraction of the container file:

```
$ tar -xvzf mpfr-3.1.2.tar.gz
```

[c.] Access to the folder where the files have been extracted:

```
$ cd mpfr-3.1.2
```

[d.] Compilation and installation of the files:

```
$ ./configure && make
# make install
```

To the Ubuntu users and systems that uses apt as package manager, just apply the command:

```
$ sudo apt-get install libgmp3-dev libmpfr-dev
```

Generally, to make use of the library in the programs, just inserting the corresponding *header*:

```
#include <mpfr.h>
```

The great advantage of this library is that we can choose the mode of how is done the approximation, being possible to choose between: approximation to the closer, approximation to zero, approximation to positive infinity, rounding to negative infinity or rounding away from zero.

To the compilation of programs which uses the MPFR library, just add the *string*, that makes the *link* of the library to GCC, to the compilation line:

```
$ gcc program.c -o compiled -lmpfr
```

2.3. Code Optimization

To the code optimization, in order to obtain a shorter execution time of the program and that it is shaped to use the processor capabilities where was compiled, the following compilation *cflags* were used:

Cflag -O2: enables the optimizations that are enabled by default with the *cflag* -O and all the optimizations which do not alter the binary file size, neither in the*debug*. This *cflag* was chosen, over others *cflags*, by the fact that preliminary tests (in comparison with *cflags* -O, -O1, -O3, -Os and -Ofast) pointing it as being the more efficient.

Cflag -march=native: this *cflag* instructs the GCC to compile the program to a specific architecture. In the case has been used "native" by the fact that version of GCC be greater than 4.2, where the compilator automatically detects the features of processor architecture in use. The use of this *cflag* makes the program incompatible with others architectures which differ from where it was compiled.

To compile the project using the mentioned *cflags* and the MPFR library, we have the following command line:

```
$ gcc program.c -O2 -march=native -lmpfr -o compiled
```

2.4. Simulations

Starting the simulations, were first injected photons in only one of the entries of the beam splitter, and the other left with vacuum (figures 2 e 3). Note that the behavior for both entropies (Tsallis and Von Neumann) turns out to be a convex function.

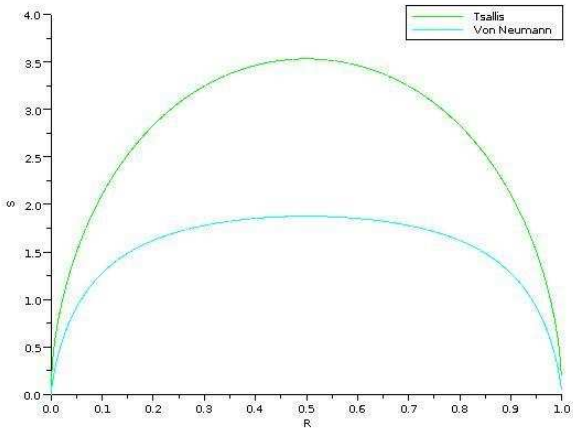

Figure 2: *Entropy to the entries* $a = 10$ *and* $b = 0$ *photons*

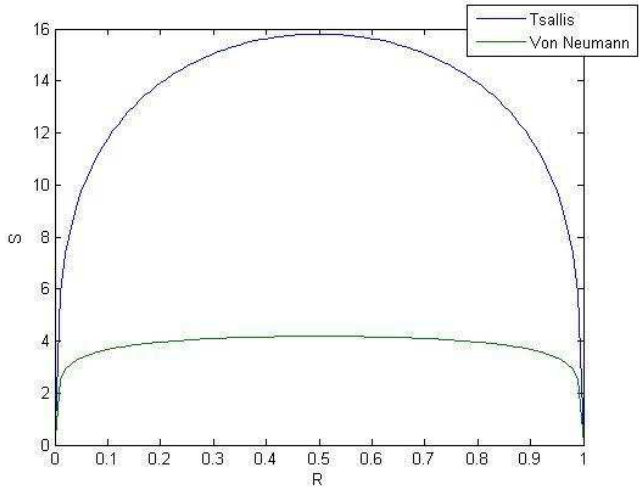

Figure 3: *Entropy to the entries* $a = 1000$ *and* $b = 0$ *photons*

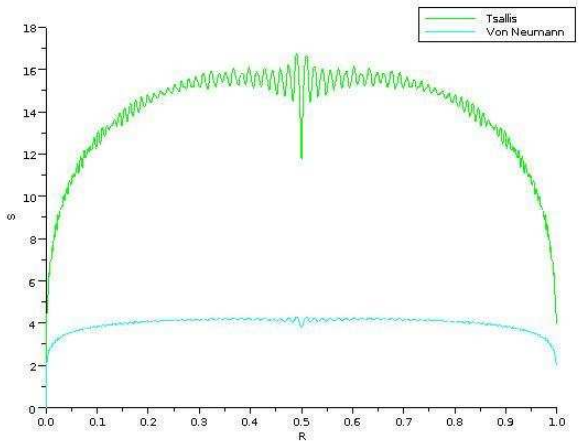

Figure 4: *Entropy to the entries* $a = 50$ *and* $b = 50$ *photons*

Continuing with the tests, passing to inject a same number of photons in both entries of the beam splitter (figures 4, 5 e 6).

In conclusion a long coverage of possibilities of tests is implemented, ends up performing tests with different entries in the beam splitter (figures 7, 8 and 9). It is observed that in these tests, the behavior of the entropies, both Tsallis and Neumann, being turns out to maintain a standard, but with a increment in the number of photons in the entries of the beam splitter, is observed a differentiation in the entropies, as also in the case with only entry and same entries.

2.5. Comparisons

In this section we present the comparisons between the execution times of tests realized in this work with the results obtained in Brandão [5], and Borges and Brandão [6]. Thus we aim to show that the Tsallis and Von Neumann entropies implementation, to entanglement analysis, in the imperative languages (in this paper the C language) are more efficient when compared to the interpreted languages (Mathematica, in this case).

Thus making viable tests with bigger entries in the beam splitter, that are performed in shorter times. The figure 10 shows the runtime gains of each test in the C language versus Mathematica. The table 1 shows the execution times of each test in C language versus Mathematica.

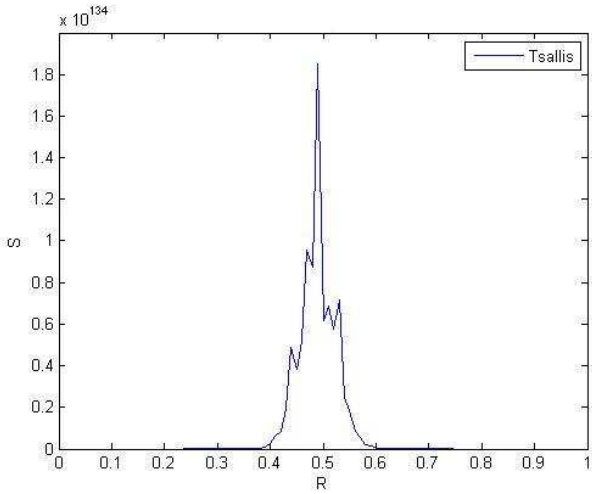

Figure 5: *Tsallis Entropy to the entries* $a = 500$ *and* $b = 500$ *photons*

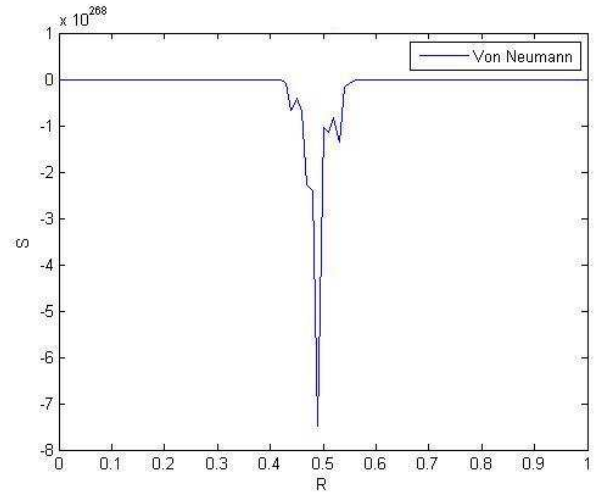

Figure 6: *Von Neumann Entropy to the entries* $a = 500$ *and* $b = 500$ *photons*

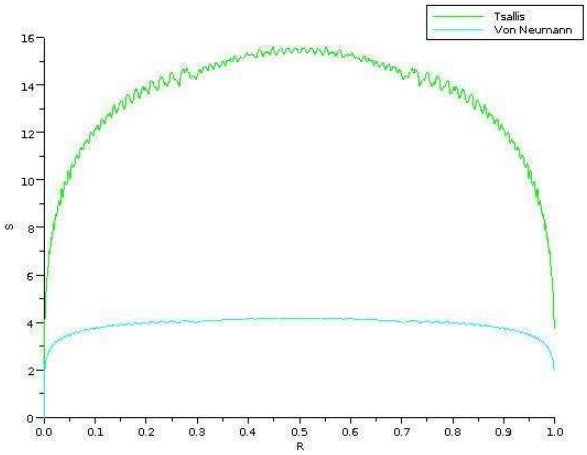

Figure 7: *Entropy to the entries* $a = 30$ *and* $b = 70$ *photons*

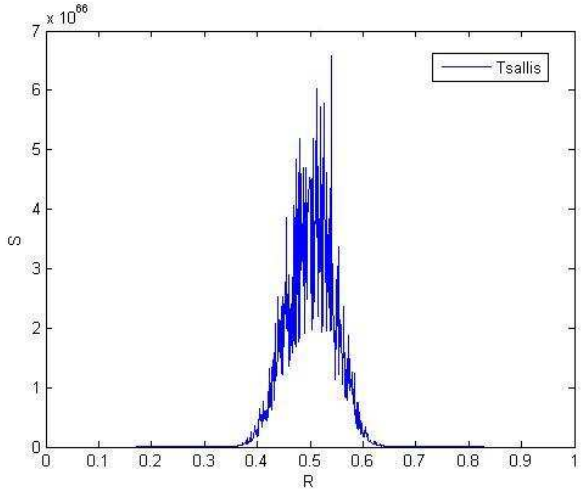

Figure 8: *Tsallis Entropy to the entries* $a = 240$ *and* $b = 320$ *photons*

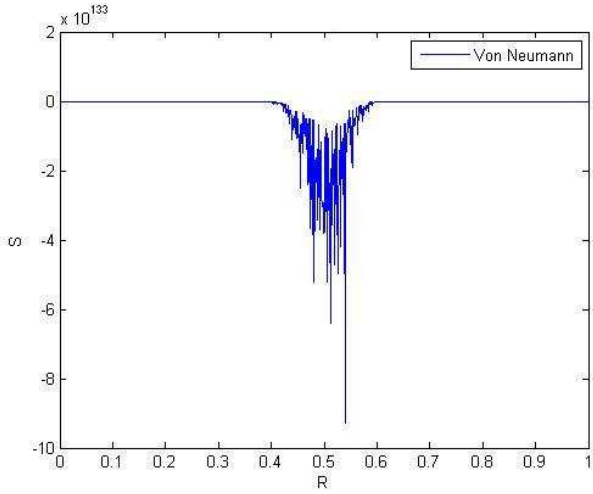

Figure 9: *Von Neumann Entropy to the entries* $a = 240$ *and* $b = 320$ *photons*

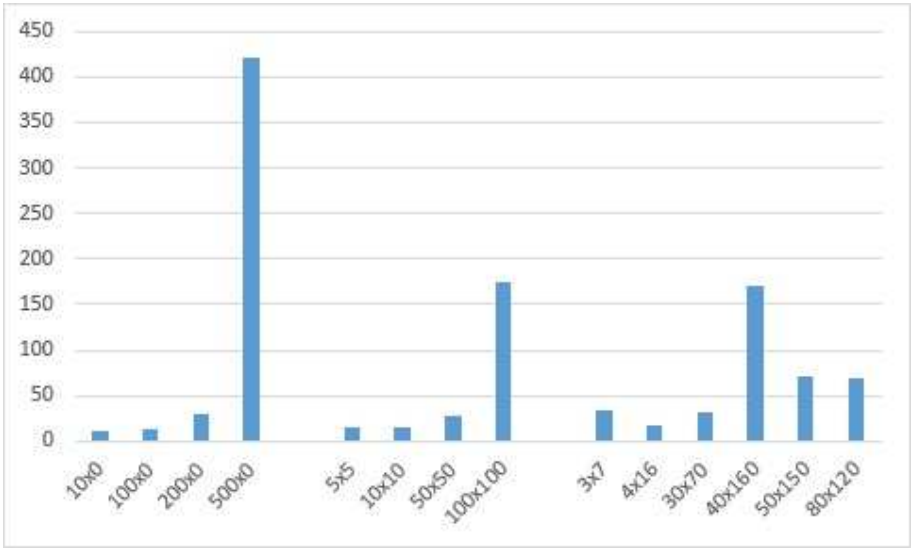

Figure 10: *Graphic Entries x Gain*

Entry	Mathematica	C	Measure	Gain
10x0	0.056394	0.0054	minutes	10.44333333
100x0	6.0701	0.461316667	minutes	13.15820657
200x0	101.5872	3.2975	minutes	30.80733889
500x0	20306.952	48.3007164	minutes	420.42755291
1000x0	-	30.85414	minutes	-
5x5	0.0968751	0.006433333	minutes	15.05830647
10x10	0.412233	0.026916667	minutes	15.31515770
50x50	154.3974	5.477416	minutes	28.18799959
100x100	13017.456	74.49102	minutes	174.75201708
320x320	-	3121.142625	minutes	-
500x500	-	1396.08318	minutes	-
1000x1000	-	7228.955667	minutes	-
3x7	0.1819985	0.005333333	minutes	34.12472088
4x16	0.3539085	0.020483333	minutes	17.27787660
30x70	143.46	4.6113	minutes	31.11053282
40x160	8187.33	47.8793166	minutes	170.99930787
50x150	3998.64	55.9541832	minutes	71.46275347
80x120	4990.35	71.6223	minutes	69.67592495
240x320	-	3307.541816	minutes	-
500x750	-	1161.733333	minutes	-
1000x1500	-	16204.09166	minutes	-

Table 1: Comparative table with execution times

3. Final Considerations

We can affirm that the C implementation is most recommended to the testing, since their execution times are much smaller than the times of same tests performed in Mathematica. The gains are significant, the largest one about reducing a test from 28 days to 1 hour.

The production and the code optimization met the expectations, showing that the imperative language have better performance than the interpreted ones, and turning the tests with bigger entries much faster.

This work shows that with entries bigger than 60, the Von Neumann entropy looks like can't stablish the approprieted correlationships with the real world, once that all of his entropics values end up to be negatives, differently of the Tsallis entropy that keeps the behavior pattern being, perhaps, more able to demonstrate the interactions that occur in the microscope world.

With this work we expect tor contribute to one more step in the development of quantum computing, presenting new simulations about the effect of the

entropy and bringing a bigger knowledge of the behavior of quantum systems of photons transmission.

References

[1] A. Aspect, J. Dalibard and G. Roger, Experimental test of bell's inequalities using time–varying analyzers, *Phys. Rev. Lett.*, **49**(1982), 1804–1807.

[2] J. S. Bell, *Speakable and Unspeakable in Quantum Mechanics*, Cambridge University Press (1993).

[3] J. S. Bell, On the Einstein Podolsky Rosen paradox, *Physics*, **1**(1964), 195–200.

[4] M. F. Borges, R. Godoy and G. A. Pratavieira, Towards a quantum information process using tsallis entropy, *International Journal of Applied Mathematics*, **21**(2008), 645–655.

[5] C. Brandão, *Ensaios sobre computação e informação quânticas: fundamentação e simulações sobre o efeito da entropia*, UNESP(2010).

[6] C. Brandão and M. F. Borges, Quantum entanglement and information: the effect of entropy revisited, *International Journal of Pure and Applied Mathematics*, **68**(2011), 25–35.

[7] M. Cunha, *Entropia: introdução à Teoria Matemática da (des)Informação*, UFMG(2004).

[8] M. Cunha, *Emaranhamento: caracterização, manipulação e consequências*, UFMG(2005).

[9] A. Einstein, B. Podolsky and N. Rosen, Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?, *Phys. Rev.*, **47**(1935), 777–780.

[10] L. Fousse, G. Hanrot, V. Lefèvre, P. Pélissier and P. Zimmermann, Mpfr: A multiple-precision binary floating-point library with correct rounding, *ACM Trans. Math. Softw.*, **33**, 2(2007), 13.

[11] J. Neumann, *The Mathematical Foundations of Quantum Mechanics*, Princeton University Press (1955).

[12] A. Salles, *Emaranhamento quântico: Detecção, dinâmica e não-localidade*, UFRJ (2009).

[13] D. C. Santos, *Em busca de um entendimento completo acerca do emaranhamento*, UFMG (2006).

[14] C. E. Shannon, A mathematical theory of communication, *Bell Sys. Tech. J.*, **27** (1948), 379–423 and 623–656.

[15] C. Tsallis, Possible Generalizations of Boltzmann - Gibbs statistics, *J. Stat. Phys.*, **52**, 1–2(1988) ,479–487.

[16] C. Tsallis, M. Gell-Mann and Y. Sato, Extensivity and Entropy Production, *Europhysics News*, **36**, 6 (2005), 186–189.

[17] C. Tsallis, F. C. S Barreto and E. D. Loh, Generalization of the planck radiation law and application to the cosmic microwave background radiation, *Phys. Rev. E*, **52**(1955), 1447–1451.

[18] A. Vidiella, Entanglement and nonextensive statisics, *Physics Letters A*, **260**, 5(1999), 335–339.

[19] T. Young, On the theory of light and colours, *Philosophical Transactions of the Royal Society of London*, **92**(1802), 12–48.