A MEASURE OF THE INFORMATION CONTENT OF NEURAL SPIKE TRAINS

Miguel A. Jiménez-Montaño, Thorsten Pöschel[*] and Paul E. Rapp**
Departamento de Física y Matemáticas, Universidad de las Américas/Puebla Sta. Catarina Mártir, 72820 Puebla, México. e-mail: jimm@udlapvms.pue.udlap.mx
* Humboldt-Universität zu Berlin, Institut für Physik,
Invalindenstrasse 110, D-10115 Berlin, Germany. http://summa.physik.hu-berlin.de:80/˜thorsten
**Department of Physiology, The Medical College of Pennsylvania, Philadelphia,
Pennsylvania 19129, USA. e-mail: rapp@medcolpa.edu

ABSTRACT

After a short review of some informational and grammatical concepts and a former algorithm to evaluate the complexity of neural spike trains, a new algorithm to build a short context-free grammar (also called program or description) that generates a given sequence is introduced. It follows the general lines of the first algorithm but it optimizes the information content, instead of the grammar complexity that was used in the previous work.. It is implemented by means of the program SYNTAX and applied to estimate the information content of neural spikes trains, obtained from a sample of seven neurons, before and after penicillin treatment.. A comparison of the sequences (encoding the inter-spike intervals) according to their information content , grammar complexity , and block-entropies shows that the three context dependent measures of complexity give similar results to categorize the neurons with respect to their structure or randomness , before and after the application of penicillin.

Introduction

The determination of block-entropies is a well established method for the investigation of discrete data, also called symbols (Herzel et al., 1995). In a recent doctoral dissertation, Schmitt (1995) calculated the block-entropies of the digitized neural spike trains previously obtained by Rapp´s group (Rapp et al. , 1994). His results are consistent with the results in the mentioned paper , which deals about the increase of the algorithmic complexity during focal seizures. In that article the measure of algorithmic complexity employed was the grammar complexity (Ebeling & Jimenez-Montaño, 1980). It was estimated by means of the program NVOGRAMM (Quintana-López, 1993), which builds a short context-free grammar (also called program or description) that generates the given sequence; it follows the general lines of a former algorithm (Chavoya et al.,1992). In this paper we propose a new algorithm to build the grammar by minimizing the information-content of the sequence, instead of the grammar complexity. In the new algorithm the block-entropies of the letters of the alphabet (ordinary Shannon entropy) and symbol pairs are used to weight the letters and syntactic categories, respectively, as explained below

The problem with the grammar-complexity based algorithm is that the letters and the syntactic categories (runs of binary symbols in the present application) are treated on the same footing (i.e., with the same weight). Since the self-information of letters and categories is different, this fact should be taken into account in the calculation of the information content. Before one can apply the algorithm to interspike interval data it is necessary to reduce these data to a sequence of symbols. As in the former paper (Rapp et al., 1994) a partition about the median is employed, with the same experimental data, to estimate the information content of neural spike trains.

The main purpose of the present communication is to make a comparison of the ranking of the sequences according to three measures of complexity:

i) The information content ,
ii) the grammar complexity, and
iii) the block-entropies,

to check the consistency of the three approaches. Besides, to have a self-contained presentation, we recall the definitions of the measures employed and their main properties. Last but not least, we recall our former algorithm to estimate the grammar complexity, and give a brief description of the program SYNTAX (Jiménez-Montaño et al., 1995), to estimate the information content of a binary sequence. This last program is written in FORTRAN . It is available upon request.

Materials and Methods

For the sake of completeness, and in order to fix the notation, we recall first some well known concepts from information and formal language theories.

1. Entropy-like Measures of Sequence Structure

Symbol sequences are composed of symbols (letters) from an alphabet of λ letters (e.g. for $\lambda = 4$, {A,C,G,T} is the DNA alphabet ; for $\lambda = 2$, {0,1} is the binary alphabet, etc.). Substrings of n letters are termed n-words. If stationary is assumed, i.e., if any word i can be expected at any arbitrary site to occur with a well-defined probability p_i , then the n-word entropies (block-entropies or higher order entropies) are given by

$$H_n = - \Sigma_i p_i^{(n)} \log_2 p_i^{(n)} \qquad [1]$$

The summation has to be carried out over all words with $p_i > 0$. The maximum number of words is λ^n , so there is a dramatic increase of the number of possible words with respect to n which makes the estimation of higher-order entropies a difficult task (Schmitt, 1995). The entropies H_n measure the average amount of information contained in a word of length n. Defining the *self-information* of a word of length n as

$$I_n = -\log_2 p_i^{(n)} \qquad [2]$$

then $H_n = < I_n >$, is the expected value of I_n [3]

The differential entropies,

$$h_n = H_{n+1} - H_n , \qquad [4]$$

give the new information of the n+1th symbol given the preceding n symbols. The entropy of the source

$$h = \lim_{n\to\infty} h_n = \lim_{n\to\infty} H_n/n \qquad [5]$$

quantifies the information content per symbol, and the decay of the h_n measures correlation within the sequence. H_n and h_n are good candidates to detect structure in symbolic sequences since they respond to any deviations from statistical independence. In a random sequence with equidistributed probabilities, $p^{(n)} = 1/\lambda^n$ holds for the probabilities of n-words. Therefore,

$$H_n = n \log_2 \lambda \qquad [6]$$

For binary sequences $\lambda = 2$, and $H_n = n$ bits. H_n exhibits a linear scaling for random non-equidistributed process (Schmitt, 1995),

$$H_n = n H_1 , \qquad [7]$$

the coefficient being $H_1 = - \Sigma^{\lambda}_{i=1} p_i \log_2 p_i$, i.e., the ordinary Shannon entropy.

Mostly, the entropies H_n are estimated from the normalized frequencies of occurrences:

$$H_n^{obs} = - \Sigma_i k_i/N \log_2 k_i/N , \qquad \text{and} \qquad I^{obs} = - \log_2 k_i/N \qquad [8]$$

which are called “observed entropies” (respectively, observed self-information). Here N denotes the total number of words in the sequence, and k_i is the number of occurrences of a certain word i . As it was shown by Ebeling et al. (1987) and Herzel (1988), in general the naive estimation of the probabilities by means of $p_i = k_i/N$ fails, producing a finite sample effect; i.e., a deviation of H_n from its true value as n increases (Pöschel et al.,1995) . However, these effects will not be considered in the present paper. We refer the interested reader to the mentioned works for further details about this point. In contrast to the informational quantities defined before, which are referred to an ensemble of sequences , the observed quantities refer to *individual sequences.* In the following all entropies will be observed entropies (with the superscript “obs.” suppressed for convenience).

2. Grammar Complexity

Grammar Complexity as introduced by Ebeling and Jiménez-Montaño (1980) constitutes an attempt to determine the algorithmic complexity of a sequence. The essence of this concept is to compress a sequence by introducing new variables (syntactic categories). The length of the compressed sequence is then taken as a measure of the complexity of a sequence. However, there are different ways to measure the length of the compressed sequence; in the original paper (Ebeling and Jiménez-Montaño 1980) the number of characters of the compressed sequence was used (counting logarithmically repeated characters). We recall this approach next. Other alternative is the *self-information* employed in the new algorithm introduced below. Further possibilities are discussed in (Schmitt, 1995).

The set of all finite strings (words) formed from the members of the alphabet X is called the free semigroup generated by X, denoted X^{*}. A language over an alphabet X is any subset of X^{*}. If p and q are words from X^{*}, then their concatenated product pq is also a member of X^{*}.

A *context.free grammar* is a quadruple G = { N, T, P, S } where:

(1) N is a finite set of elements called *nonterminals* (syntactic categories), including the start symbol S.

(2) T is a finite set of elements, called *terminal symbols* (letters of the alphabet).

(3) P is a finite set of ordered pairs $A \rightarrow q$, called *production rules* , such that $q \; \varepsilon \; (N \cup T)$ and A is a member of N.

Let us consider a grammar G such that L (G) = w ; i.e., the language generated by G consists of the single sequence w. These grammars are called "programs" or "descriptions" of the word w. The *grammar complexity of w* (Ebeling and Jiménez-Montaño 1980) is defined as follows:

The complexity of a production rule $A \rightarrow q$ is defined by an estimation of the complexity of the word in the right-hand side: $q \rightarrow a_1^{\nu_1} \ldots a_m^{\nu_m}$:

$$K(A \rightarrow q) = \sum_{j=1}^{m} \{[\log \nu_j] + 1\} , \qquad [9]$$

where $a_j \; \varepsilon \; (N \cup T)$, for all j = 1, ...,m .Therefore, in this definition terminals (letters of the alphabet) and non-terminals (syntactic categories; sub-words) are treated on the same footing (i.e., with the same weight). Here [x] denotes the integral part of a real number. The complexity K(G) of a grammar G is obtained by adding the complexities of the individual rules. Finally, the complexity of the original sequence is:

$$K(w) = K(G(w)) = \min \{ K(G) \mid G \rightarrow w \}. \qquad [10]$$

This quantity, which is a particular realization of the algorithmic complexity introduced by Solomonoff (1964), Chaitin (1990) and Kolmogorov (1958), refers to an individual sequence, in contrast to the Shannonian measures which are related to the sequence source.

3. Algorithm to Estimate the Grammar Complexity

In former papers (Ebeling & Jiménez-Montaño, 1980; Jiménez-Montaño et al. 1987) an algorithm to estimate the grammar complexity of a sequence was described and applied to the estimation of the complexity of biosequences (DNA, RNA and proteins). Independently, Wolff (1975 & 1982) introduced a similar algorithm and applied it to the discovery of phrase structure in natural language. Briefly, our former procedure is the following:

Algorithm GRAMMAR

INPUT: A sequence q, with characters from alphabet $T = \{a_1, a_2, \ldots a_\lambda \}$.

OUTPUT: A short context-free grammar G such that L (G) = q

PROCEDURE:

```
1) START   $i := 0$ ;
WHILE there are pairs of contiguous characters (in the sequence q) which occur more than two times DO
        $i := i + 1$;
        $\alpha_i := $ the most frequent pair;
SUBSTITUTE in q each occurrence of $\alpha_i$ by a new symbol $A_i$, and introduce the PRODUCTION RULE $A_i \rightarrow \alpha_i$ ;
2) WHILE there are in q two equal sub-words of more than two characters DO
    $i := i + 1$;
    $\alpha_i := $ the longest sub-word;
SUBSTITUTE in q each occurrence of $\alpha_i$ by a new symbol $A_i$, and introduce the PRODUCTION RULE $A_i \rightarrow \alpha_i$ ;
3) The grammar G is:
    $T = ( a_1, a_2 , ..., a_\lambda)$
    $N = (A_1, A_2 ,..., A_r , S)$
    P:   $S \rightarrow q$ ; $A_1 \rightarrow \alpha_1$ ; $A_2 \rightarrow \alpha_2$ ; . . . $A_m \rightarrow \alpha_m$
 with $l(\alpha_m) \leq l(\alpha_k)$ if $k < m$ , where l(x) is the length (number of characters) of x.
K(G(q)) gives a good estimation of the grammar complexity , K(q)
END.
```

This procedure has been implemented with the programs GRAMMAR.C (Chavoya et al, 1992) and NVOGRAMM (Quintana, 1993). In both implementations the heuristic employed was a hill-climbing optimization procedure, which searches to minimize the grammar complexity at each step. Therefore, it is not guaranteed that the grammar found is really the shortest one; that is why one gets only an estimation of K(q).

4. Algorithm to Estimate the Information Content

To estimate the information content of a sequence w, I(w), all one needs to do is to replace in the above optimization procedure K(w) by I(w) . To evaluate I(w) one proceeds as follows:

For each production rule, instead of the complexity of a rule defined in [9] one introduces the *information content of a rule* defined as

$$I\,(A \rightarrow q) = \Sigma_{j=1}^{m} I_j, \qquad [11]$$

where the quantities I_j $(j = 1,...m) = I^{obs} = -\log_2 k_i/N$, (see equation [8]),are the weights of the terminals (for $j = 1$) and non-terminals (for $j > 1$) from which q is composed. The information content of a grammar G, I(G), is obtained by adding the information content of the individual rules. As the number of rules increases I(G) may increase or diminish. If , after the introduction of new rules, I(G) does not diminish the process stops. Therefore, the estimation of I(w) is:

$$I(w) = I(G(w)) = \min \{ I(G) \,|\, G \rightarrow w \}. \qquad [12]$$

END.

This algorithm was implemented with the program SY NTAX (Jiménez-Montaño et al. ,1995).

If S $\rightarrow$ w is the trivial grammar that generates the sequence w. The *average* information content of w, as estimated from this grammar , would be $H_n = n H_1$, with n = l(q), the length of the sequence; a nd where $H_1 = - \Sigma^{\lambda}_{i=1} p_i \log p_i$ is the ordinary Shannon entropy ,estimated from the letter -composition of the sequence (number of zeros and ones, for binary sequences). For example, for binary sequences of length 1000 , with equal number of ze ros and ones, H_n = 1000 bits. In contrast to this quantity, I(w) is the estimation of the information content of an individual sequence w (see Table 1). While the former quantity cannot distinguish among different sequences of the same composition, the later one can.

Results

As mentioned in the introduction , in a former paper (Rapp et al. 1994) it was shown that the algorithmic complexity, as estimated with the help of the program NVOGRAMM (Quintana, 1993), increases during focal seizures. The same experimental results reported in that article, of seven single -unit records obtained from cortical neurons of the rat before and after the application of penicillin ,were used for the present work. Therefore, to save space, we refer the reader to the mentioned paper for the details of the experimental methods employed. The purpose of the present study is to compare the results obtained by means of the information content, calculated with the help of the program SYNTAX, with our previous results (Rap p et al 1994), and with the values of the block -entropies of the same seven sequences in the sample, calculated by Schmitt(1995). As explained in our former paper, before the complexity of neural spike trains can be estimated it is necessary to reduce the neural data to a sequence of symbols. The usefulness of information content calculations depends crucially on the procedure used to partition the data among a finite alphabet of symbols.

Table 1. Average information content of 1000 points data set (N=5)

Distribution	Symbol-sequence formed about the mean	Symbol-sequence formed about the median	Symbol-sequence formed about the midpoint
Uniform	945.74 ± 11.64	943.64 ± 11.96	939.96 ± 9.99
Gaussian	939.41 ± 3.99	939.36 ± 4.36	895.21 ± 38.19
Exponential	923.41 ± 20.74	941.28 ± 3.22	216.62 ± 41.57
Poisson	896.21 ± 53.67	875.53 ± 64.05	449.89 ± 238.53

Calculations with messages obtained by reducing random numbers to binary symbol sequences establish the upper bound of the information content that can be obtained with 1000 elements. The results are reported with SDs obtained from calculations using five data sets for each distribution.

In Table 1 the average information content of 1000 artificial point data sets, with four distributions , are displayed. These results were obtained after the data were reduced to a sequence by partitioning a bout the mean, the median and the midpoint, as explained in (Rapp et al.,1994). The experimental binary symbol sequences employed in this paper were constructed by partitioning inter -spike intervals about the median. This choice entails that $H_1 = 1$, of course. Table 1 is analogous to the corresponding one in (Rapp. et al., 1994) for the grammar complexity. The results of both tables are consistent, except that the information content for the Poisson distribution is slightly different from the other thr ee distributions. On the contrary, the grammar complexity is completely insensitive to the distribution for partitions about the median.

Table 2. Statistical properties of the order sensitive measures C_2 and I, 1000 events in each dataset

	C_2	C_2	I	I	I Shuffled	I Shuffled
Neuron	Spon.	Pen.	Spon.	Pen.	Spon.	Pen.
5	227	269	715.855	941.515	931.033	933.087
1	227	266	736.050	913.542	936.405	936.527
6	223	243	753.431	758.456	932.042	925.033
7	258	274	842.407	932.595	943.858	943.858
3	266	270	890.998	876.362	932.501	937.094
2	276	274	920.899	933.873	932.595	950.370
4	275	275	951.795	924.274	949.690	944.158
Average	**252**	**267**	**830.205**	**897.230**	**936.875**	**937.319**
SD	*23.83*	*11.19*	*95.47*	*64.87*	*7.17*	*8.08*

Significance testing (spontaneous vs penicillin treated)
C_2 : t = 2.277, p = 0.063 (Rapp et al.,1994)
I : t= -1.769, p = 0.0636

The values of the binary complexity ,C_2 , and the information content, I , obtained from 1000 element spike trains partitioned about the median are displayed. In the last two columns the information content of randomly shuffled sequences, for the spontaneous and the penicillin -treated case, are included for control purposes. A paired t test was used to compare the values obtained in the two conditions. The neurons are ordered according to I Spon.

Table 2 contains the main results of the present communication. In this table the values of the grammar complexity (Rapp et al., 1994) and information content of the seven neurons, before and after the application of penicillin, are shown. The corresponding values for random shuffled sequences, of the same composition, are also included for comparison. The results obtained by the three measures of sequence structure; i.e., grammar complexity, information content and block -entropies (the last one calculated by Schmitt (1995), but not reproduced here) are quite consistent: According to the three methods, neurons 1,5 and 6 have a significant structure before the penicillin treatment; and, for the same condition, neurons 2 and 4 have spike train s which are not too different from those of a perfectly random equi -distribution ($H_n = n$). Neurons 3, and 7 are at the border of randomness and penicillin produces no significant effect on these neurons. It is also clear that, with both measures, the average values increase after the penicillin treatment and the SD values decrease. However, these two sets of values still differ from the extremes found for randomly shuffled sequences.

Table 3. Statistical properties of the order sensitive rates RC_2 and RI, 1000 events in each dataset

Neuron	RC_2 Spon.	RC_2 Pen.	RI Spon.	RI Pen.
5	0.899	1.671	2.835	5.848
1	1.622	2.454	5.259	8.427
6	2.367	3.160	8.005	9.864
7	1.167	3.506	3.810	11.933
3	2.952	3.667	9.888	11.902
2	0.320	0.454	1.067	1.547
4	1.991	3.614	6.885	10.832
Average	**1.617**	**2.647**	**5.392**	**8.621**
SD	*0.902*	*1.209*	*3.083*	*3.764*

The values of the rates of increase of the binary complexity ,C_2 , (Rapp el al., 1994) and the information content, I , obtained from 1000 element spike trains partitioned about the median are displayed. As in Table 2, the neurons are ordered according to I Spon.

As mentioned in our former publication, because the mean firing rate differs from neuron to neuron, the records employed cover very different time intervals. Since, as argued in (Rapp et al.,1994) timely responsiveness is an essential property of any successful biological system, the information content generation per unit time may give additional information for the problem at hand. It is calculated by dividing the information content by the corresponding time required by that neuron to fire the 1000 action potentials. It represents the rate at which the neuron is increasing its information content and, therefore, loosing its structure. In Table 3 these rates are displayed, together with the corresponding values for the grammar complexity rates obtained in the former publication (Rapp et al.,1994). From the displayed values it is clear that, with both measures, the different neurons are becoming disorganized at higher rates after the treatment. However, the increase in firing rate in the untreated condition seems to be uncorrelated with the degree of structure. Neurons 5,7 and 2 have low rates, while the others have somewhat greater rates of disorganization. For the treated condition only neuron 2 present an anomalous low rate.

Final Remarks

In this communication we have shown that three different measures of complexity produce consistent results about the degree of structure of neural spike trains. It is important to notice, as we noticed in the former publication (Rapp et al.,1994), that the classification of neurons made in this and the former publication could not have been made on the basis of distribution -determined measures. The patterns observed in spike trains appear to be genuine and not due to chance variations.

Acknowledgement. The authors thank Miguel A. Hernández Morales for his help with the computer calculations.

References:

Chaitin G.J. (1990) Information, Ramdomness & Incompleteness (Papers on Algorithmic Information Theory), World Scientific, Singapore.

Chavoya-Aceves O., García F., Jiménez -Montaño M.A. (1992) Programa para estimar la complejidad gramatical de una secuencia., Memorias IX Reunión Nacional de Inteligencia Artificial. pp 243-254 Megabyte, Grupo Noriega Editores, Mex.

Ebeling W., Feistel R., and Herzel H., Phys. Scr. (1987) **35**: 761.

Ebeling W. and Jiménez-Montaño M.A. (1980) Math. Biosc. **52**: 53-71.

Herzel H., (1988) Sys. Anal. Mod. Sim. **5** : 435.

Herzel H., Ebeling W., Schmitt A.O., Jiménez -Montaño M.A. (1995) Entropies and Lexicographic Analysis of Biosequences, in From Simplicity to Complexity in Chemistry, and Beyond. Müller A, Dress A & Vögtle F. (Eds.) Vieweg, Braunschweig, 1995, pp 7 -26.

Jiménez-Montaño M.A, Ebeling W., Pöschel T. (1995), SYNTAX: A Computer Program to Compress a Seque nce and to Estimate Its Information Content, presented in Complex Systems and Binary Networks Conference, Guanajuato, Mexico, Jan. 16 -22 1995. No proceedings were published.

Jiménez-Montaño M.A., Zamora L., Trejo J., (1987) Aportaciones Matemáticas, Comuni caciones **5**: 31-52..

Kolmogorov A.K. (1958)., Dolk. Akad. Sci USSR **119**: 861.

Pöschel T., Ebeling W., Rose H. (1995) J. Stat. Phys., **80** :1443-1452

Quintana-López M., (1993)., Análisis Sintáctico de Biosecuencias., M. Sc. Thesis. Instituto Tecnológico de Estudios Superiores de Monterrey campus Edo. de Mex. México.

Rapp P.E., Zimmerman I.D., Vining E.P., Cohen N., Albano A.M., Jiménez -Montaño M.A. (1994) The Journal of Neuroscience, **14** (8): 4731-4739.

Schmitt A. (1995) Structural Analysis of DNA Sequences. Bi oinformatik Band 1, Berlin: Verlag Dr. Köster

Solomonoff R. J. (1964) Inform. & Control **7**: 1-22.

Wolff J.G. (1975) Br. J Psychol. **66**:79-90.

Wolff J.G. (1982) Language & Communication. **2** (No.1): 57-89.