



Optimized Detection of the Infrequent Response in P300-Based Brain-Computer Interfaces

C. Lindig-León
O. Yáñez-Suárez

Laboratorio de
Neuroimagenología,
Universidad Autónoma
Metropolitana, Iztapalapa,
México.

ABSTRACT

This paper presents an application developed on the BCI2000 platform which reduces the average spelling time per symbol on the Donchin speller. The motivation was to reduce the compromise between spelling rate and spelling accuracy due to the large amount of responses required in order to perform coherent average techniques. The methodology was made under a Bayesian approach which allows calculation of each target's class posterior probability. This result indicates the probability of each response of belonging to the infrequent class. When there is enough evidence to make a decision the system stops the stimulation process and moves on with the next symbol, otherwise it continues stimulating the user until it finds the selected letter. The average spelling rate, after using the proposed methodology with 14 healthy users and a maximum number of 5 stimulation sequences, was of 6.1 ± 0.63 char/min, compared to a constant rate of 3.93 char/min with the standard system.

Keywords: brain-computer interface, oddball paradigm, Bayesian inference.

Correspondencia:
 Ing. Alma Cecilia Lindig León
 Laboratorio de
 Neuroimagenología, UAM
 Iztapalapa Sn Rafael Atlixco
 186, Col. Vicentina 09340 DF
 Edificio T.227
 Correo electrónico:
 cecilindig@yahoo.com.mx

Fecha de recepción:
 13 de Noviembre de 2012
Fecha de aceptación:
 1 de Abril de 2013

RESUMEN

Este trabajo presenta una aplicación desarrollada sobre la plataforma BCI2000 que disminuye el tiempo promedio de selección de los símbolos del deletreador de Donchin. La motivación consistió en reducir el compromiso entre la tasa de deletreo y la precisión correspondiente, la cual surge como consecuencia de la gran cantidad de respuestas necesarias para realizar técnicas de promediación coherente. La metodología propuesta se basa en un enfoque Bayesiano que permite calcular la probabilidad posterior asociada con la clasificación de cada objetivo, resultado que indica la evidencia que presentan las respuestas de pertenecer a la clase infrecuente. Cuando existe evidencia suficiente para tomar una decisión, el sistema detiene el proceso de estimulación y continúa con el siguiente símbolo, de lo contrario permanece estimulando al usuario hasta conseguir identificar la letra seleccionada. Después de utilizar la metodología propuesta sobre los registros de 14 usuarios sanos con un número máximo de 5 series de estimulación, el tiempo promedio de deletreo reportado es de 6.1 ± 0.63 letras/min, el cual es comparado con una tasa constante de 3.93 letras/min obtenido con un sistema convencional.

Palabras clave: interfaz cerebro-computadora, paradigma de evento raro, inferencia Bayesiana.

INTRODUCTION

The brain-computer interface (BCI) known as Donchin's speller [1], allows communicating text using random intensifications of each row and column of a character matrix to evoke EEG event-related potentials (ERPs) [2]. The user focuses attention on the cell containing the letter to be communicated while the rows and the columns of the matrix are intensified, as shown in figure 1. This application is based on an oddball paradigm; the row and the column containing the letter to be communicated are "rare" items which elicit the P300 component and, therefore, can be determined.

This application has the drawback of requiring long periods of time before obtaining a reliable result. One of the reasons behind this inconvenience is that the signal amplitude is too small in comparison with the EEG background



Figure 1. Symbol matrix used in the stimulation process. (This image was taken from [1]).

activity, which hinders the detection of event-related responses. A way to improve the signal

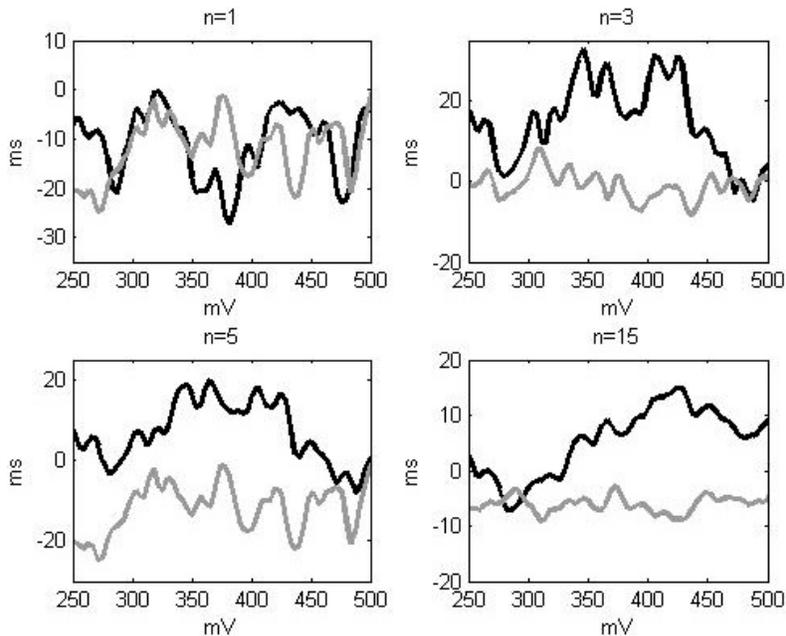


Figure 2. Each plot represents the results of the coherent averaging over 1, 3, 5 and 15 signal segments for infrequent (dark line) and frequent (tenuous line) responses. Observe that as the number of averaged segments increases, the P300 complex in the dark plots gets enhancer.

to noise ratio is using coherent average techniques [3]; which enhance common response features and reduce noise, as shown in figure 2, where the averaging process results over 1; 3; 5 and 15 EEG signal segments are illustrated.

As can be inferred, the more responses available for averaging the better results are obtained; nevertheless, this approach leads to a slow system performance due to the several stimulation sequences that are required. From this fact it can be seen that a compromise between spelling rate and spelling accuracy exists. In most of the cases BCI operators decide to present several stimulation sequences (i.e., one intensification cycle of the entire matrix, six row and six column intensifications) so as to ensure accuracy, nevertheless many of these sequences may result unnecessary because from the third or second, sometimes even from the first, the system has enough information to identify the corresponding target. This fact reveals how inadequate it is to consider the number of stimulation sequences as a static parameter assigning always the same amount to

select the different targets. For conventional systems this parameter which will be, in the present text, denoted by \tilde{N} must be established by the operator during their configuration.

Some of these systems, as the P300_Speller[®], operate using a linear discriminant which defines a hyperplane dividing the feature space into two regions. Each surface side is related to one of the classes (i.e., infrequent and frequent class), so that the classification label of each data can be determined from its relative position with respect to the stated hyperplane; which is determined by the calculated distance sign. Thus, after every stimulation sequence, the relative distances between the mapped EEG epochs to the classification space and the corresponding surface is computed; and once the \tilde{N} sequences have been completed, the sum of the \tilde{N} corresponding results is calculated, so that the information generated for each target from all stimulation sequences get grouped into one single value. The sum results related to the six matrix rows are compared in order to select

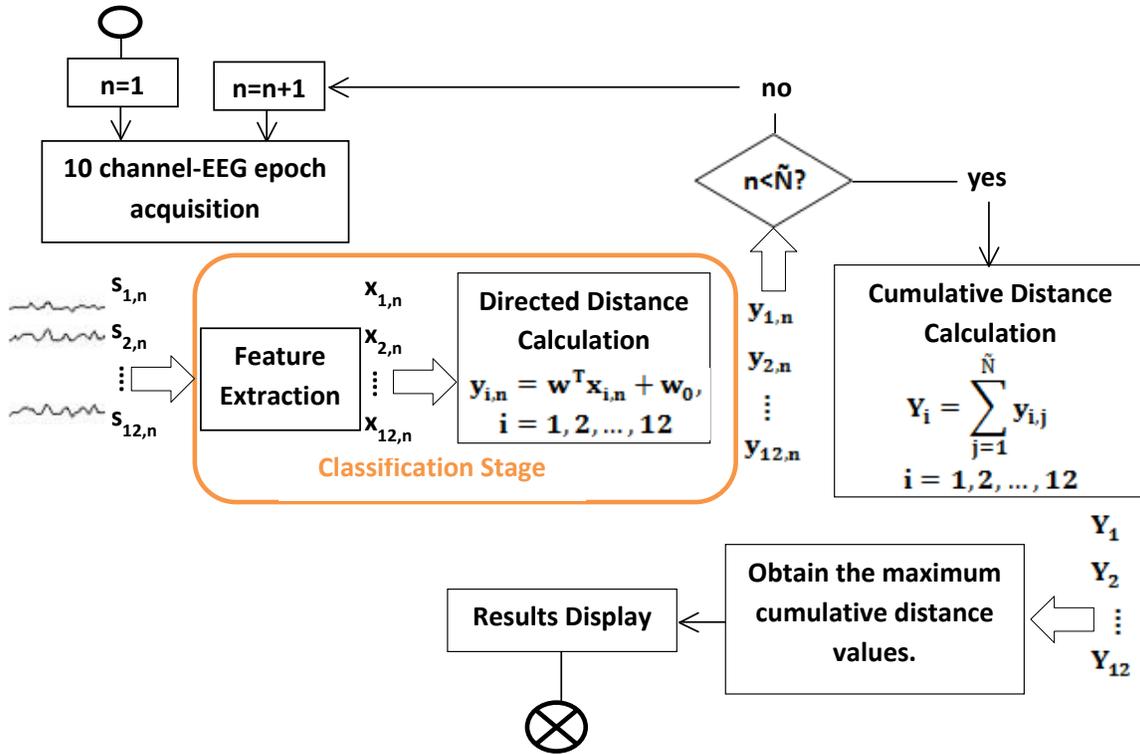


Figure 3. P300_Speller[®] operating diagram. The \tilde{N} parameter indicates the number of stimulation sequences the system emits to select every matrix character.

the target showing the maximum infrequent-signed value; which indicates that the corresponding target caused on average the most alike infrequent responses, and therefore can be selected; the same comparison is made over the sum results related to the six matrix columns. In fact, this is how these methods perform coherent averaging, not over the signal segments but over the relative distances between the mapped EEG epochs to the classification space and the corresponding hyperplane.

Figure 3 shows a diagram of the described above methodology, observe that in order to select any matrix character it is necessary to wait until the \tilde{N} stimulation sequences have finished.

In contrast, the proposed methodology adjusts itself to the data characteristics and presents as many stimulation sequences as necessary to select each matrix symbol; therefore, under this scheme, the \tilde{N} parameter is not considered. This approach determines, based on the evidence found at the end of each stimulation sequence, whether the system must continue the stimulation process, or whether

it can be stopped due to the fact that the user's chosen symbol has been already identified. In order to evaluate this evidence a second classification stage is proposed; which is trained over the cumulative distance vectors constructed from the first linear discriminant results. These vectors are generated at the end of the n th stimulation sequence by setting the sum of the first n target-related directed distances at the n th element position, for $n=1, 2, \dots, N$; where N denotes the maximum number of stimulation sequences the system emits in case it cannot make a decision before; and it corresponds to a new parameter which must be established during configuration.

METHODOLOGY

A. Classification Model

The method described below is based on a two-level classification model which estimates after every stimulation sequence the evidence related to each of the twelve matrix targets of

belonging to the infrequent class; if there is enough information to identify the corresponding symbol, the stimulation process stops and the chosen character is displayed.

The first classification stage consists of a linear discriminant trained on temporal EEG features which defines a hyperplane dividing the feature space into two classes. Feature extraction as well as estimation of the hyperplane which best separates the two classes, is carried out by using a feature selection method which determines from a labeled training dataset which channels and which EEG signal samples give the most information to differentiate both memberships.

In order to evaluate the evidence found at the end of each stimulation sequence a second classification stage is proposed; which is trained over a cumulative distance vector dataset constructed from the first classification stage results. These vectors are generated at the end of the n th stimulation sequence by setting the sum of the first n target-related directed distances (i.e., the relative distances between the mapped EEG epochs to the classification space and the hyperplane defined by the first classification stage) at the n th element position, for $n=1, 2, \dots, N$; where N denotes the maximum number of stimulation sequences the system emits in case it cannot make a decision before; and it corresponds to a new parameter which must be established during configuration.

The second classification stage consists then of N linear discriminants which classify the evidence found until the n th stimulation sequence, for $n=1, 2, \dots, N$; each of these discriminants represents a classification space where a hyperplane separating the corresponding cumulative distance vectors is defined, so that the evidence showed by every unseen data of belonging to the infrequent class is evaluated

through the relative distance between the mapped vector to the classification space and the corresponding classification surface. As it was mentioned, the sign of the resulting directed distances indicates the side of the hyperplane on which the data fall, and the absolute value provides information about how likely the data is to belong to the specified class; the farther away a point is from the hyperplane the more reliable the classification result. Following this idea, the resulting directed distances are mapped onto probability values, which allow making decisions based on the evidence related to the specific data position in the classification space. To this end, a logistic sigmoid function is fitted to each of the N resulting directed distance datasets; which maps the whole real axis into a finite interval between 0 and 1 allowing, after every stimulation sequence, calculation of each target-related posterior probability of belonging to the infrequent class.

The posterior probability results estimated after each stimulation sequence are evaluated in order to determine whether the user's chosen symbol can be identified; if this is the case, the system stops the stimulation process and displays the corresponding character; otherwise the process continues until the character can be determined or when the maximum number of stimulation sequences permitted has been exceeded (i.e., the N stimulation sequences have been completed); if this occurs the system selects the matrix symbol whose row and column position is determined by the targets showing the maximum posterior probability values.

Figure 4 shows a diagram of the proposed methodology, observe that in order to select each matrix symbol the system must determine the number of stimulation sequences required; if it cannot make a decision based on the found evidence it will present a maximum of N .

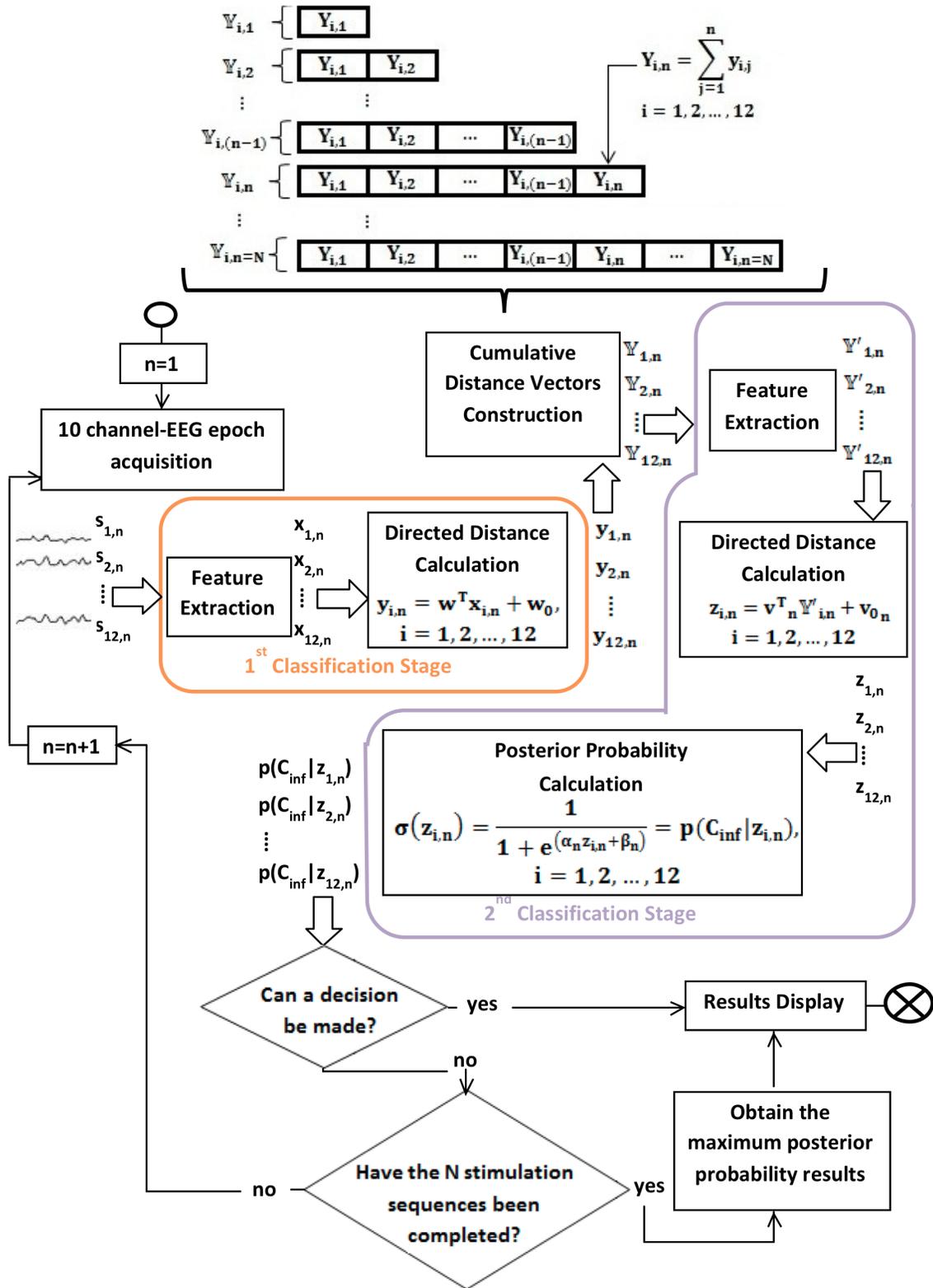


Figure 4. Proposed system's operating diagram.

Table 1. The first two columns show, respectively, the words spelled by one of the users and the corresponding number of stimulation sequences; the last two columns present the data partition from which the training and validation datasets are obtained. All subscripts correspond to an identification number for each character.

Words	# Stimulation Sequences	Validation	Training	
			First Classification stage	Second classification stage
C ₁ A ₂ L ₃ O ₄ R ₅	15	L ₃ C ₆ O ₁₁ I ₁₆ C ₂₂	E ₅₀ S ₅₁ Q ₅₂ U ₅₃ E ₅₄ M ₅₅ A ₅₆ S ₅₇	G ₄₃ A ₄₄ L ₄₅ L ₄₆ E ₄₇ T ₄₈ A ₄₉
C ₆ A ₇ R ₈ I ₉ N ₁₀ O ₁₁	15	O ₃₀ D ₃₂ L ₃₄ C ₃₅ B ₄₀	A ₅₈ L ₅₉ E ₆₀ G ₆₁ R ₆₂ I ₆₃ A ₆₄	
S ₁₂ U ₁₃ S ₁₄ H ₁₅ I ₁₆	15			C ₁ A ₂ O ₄ R ₅ A ₇
S ₁₇ U ₁₈ S ₁₉ H ₂₀ I ₂₁	15			R ₈ I ₉ N ₁₀ S ₁₂ U ₁₃
C ₂₂ E ₂₃ N ₂₄ A ₂₅ R ₂₆	15			S ₁₄ H ₁₅ S ₁₇ U ₁₈ S ₁₉
C ₂₇ O ₂₈ L ₂₉ O ₃₀ R ₃₁	15			H ₂₀ I ₂₁ E ₂₃ N ₂₄ A ₂₅
D ₃₂ U ₃₃ L ₃₄ C ₃₅ E ₃₆ S ₃₇	15			R ₂₆ C ₂₇ O ₂₈ L ₂₉ R ₃₁
N ₃₈ U ₃₉ B ₄₀ E ₄₁ S ₄₂	6			U ₃₃ E ₃₆ S ₃₇ N ₃₈ U ₃₉
G ₄₃ A ₄₄ L ₄₅ L ₄₆ E ₄₇ T ₄₈ A ₄₉	4			E ₄₁ S ₄₂
E ₅₀ S ₅₁ Q ₅₂ U ₅₃ E ₅₄ M ₅₅ A ₅₆ S ₅₇	3			
A ₅₈ L ₅₉ E ₆₀ G ₆₁ R ₆₂ I ₆₃ A ₆₄	3			

B. Subjects

Data were taken from a public domain database [4] which stores records from several users who have participated in various spelling sessions with a system built on the BCI2000 platform [5]; the users were healthy students with an age range of 21 to 25 years. Each session consists of a spelled word with a number of stimulation sequences varying from 1 up to 15. A whole stimulation sequence consists of twelve stimuli: six row-intensifications and six column-intensifications of the symbol matrix. Each target is highlighted for 62.5 ms with an inter-stimuli interval of 125 ms; the pause between stimulation sequences was 4000 ms.

Table 1 presents the spelled words by one of the users; each one corresponds to a spelling session with a number of stimulation sequences varying from 3 to 15. For every character a subscript has been added in order to track it along the partition, which defines the training and validation datasets. The first four words were common to all users, that is, they were asked to participate in four directed spelling sessions in which they have to spell nothing but these words attending to 15 stimulation sequences for each one of the characters. Once each user had finished these directed spelling sessions, she or he was free to decide the words to

spell in the remaining ones, in which the operator decided the number of stimulation sequences the system emitted.

The first stage classification training requires data consisting of a single response, regardless of stimulation sequence in which it was recorded; in the other hand second stage classification training as well as the validation process need data shaped by N successive responses. Thus, all responses of words spelled with less than N stimulation sequences are put together in one group in order to train the first classification stage. The leftover data are divided randomly into two groups: 80% is used to train the second classification stage and the other 20% to validate the system performance. All the remaining responses of data recorded with more than N stimulation sequences belonging to the second stage training dataset are added to the first classification stage training dataset.

In the subsequent sections an example of a system built with a value of N=4 using table 1 data is presented; this partition consists of a randomly generated instance which satisfies the constraint imposed by the corresponding N-value.

EEG signals were recorded using 10 electrodes positioned on the user's scalp according to the 10-20 system, as shown in

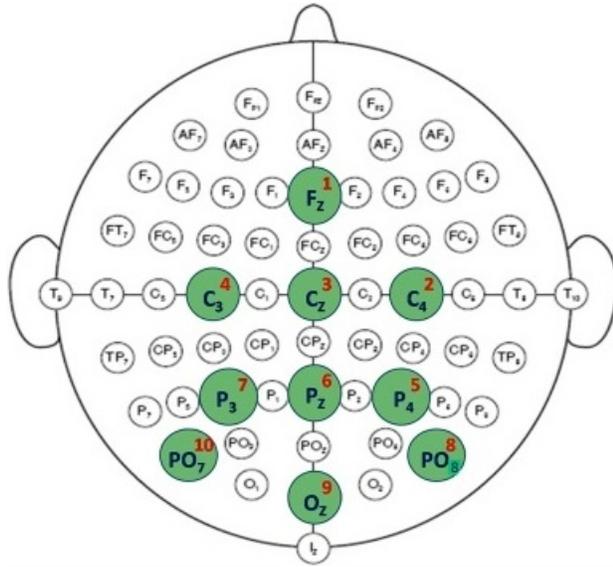


Figure 5. Diagram of the electrode subset configuration during signal acquisition, according to the 10-20 system.

Figure 5. Each electrode was connected to a 16-channel differential g.tec[®] gUSBamp amplifier (Graz, Austria) in reference to the right earlobe, with the ground placed on the right mastoid. From each channel a 256 Hz sampled signal is obtained, which is filtered using a 0.1-60 Hz 8th order passband Chebyshev filter and a 60 Hz 4th order Chebyshev notch filter.

The stimulus responses are obtained from these signals using a window size of approximately 600 ms (154 samples); thus each single response consists of a 154×10 array storing the signal segment generated by each of the ten channels starting at the beginning of the corresponding intensification; and N-responses data comprised in the second classification stage training dataset consist of a $154 \times 10 \times N$ array.

It is worth mentioning that the number of first classification stage training data sum a total of 4332 single responses, of which 540 consist of the responses generated from the two table-1-latest words “ESQUEMAS” and “ALEGRIA” which represent 15 letters spelled with 3 stimulation sequences for each of the 12 matrix targets ($15 \times 3 \times 12 = 540$). Furthermore, the 2 remaining responses (i.e., $6 - N = 6 - 4 = 2$) of each of the 4 randomly selected letters

from the word “NUBES” add 96 more single responses ($4 \times 2 \times 12 = 96$) and the 11 remaining responses (i.e., $15 - N = 15 - 4 = 11$) of each of the 28 randomly selected letters from words spelled with 15 stimulation sequences add 3696 more ($28 \times 11 \times 12 = 3696$). Note that 3610 of these 4332 single responses ($540 + 96 + 3696 = 4332$) are frequent type and the remaining 722 are infrequent type.

The number of second classification stage training data sum a total of 468 N-shaped-responses, of which 84 correspond to the data generated from table-1-word “GALLETA”, which consists of 7 letters multiplied by each of the 12 matrix targets ($7 \times 12 = 84$) and, the remaining 384 data, correspond to the 32 randomly selected letters ($32 \times 12 = 384$).

The validation dataset as can be seen from table 1 consists of 10 randomly selected letters which represents, approximately, the 20% of the total spelled characters. The remaining data of letters spelled with more than N stimulation sequences are just discarded, so that the validation dataset consists of 120 N-shaped responses ($10 \times 12 = 120$).

C. First Classification Stage

Consists of a linear discriminant trained on temporal EEG features which defines a hyperplane dividing the feature space into two classes. Feature extraction as well as estimation of the hyperplane which best separates the two classes, is carried out by using a feature selection method. This approach generates four parameters: a vector containing the channels whose signals presented the most relevant information to differentiate data from both classes; a second vector indicating which EEG signal samples recorded by the selected channels give the most information to distinguish both memberships; a third vector, w , whose elements are the coefficients of the normal vector to the hyperplane which best separates the members from both classes; and a fourth scalar parameter, w_0 , which represents the bias of the classification hyperplane.

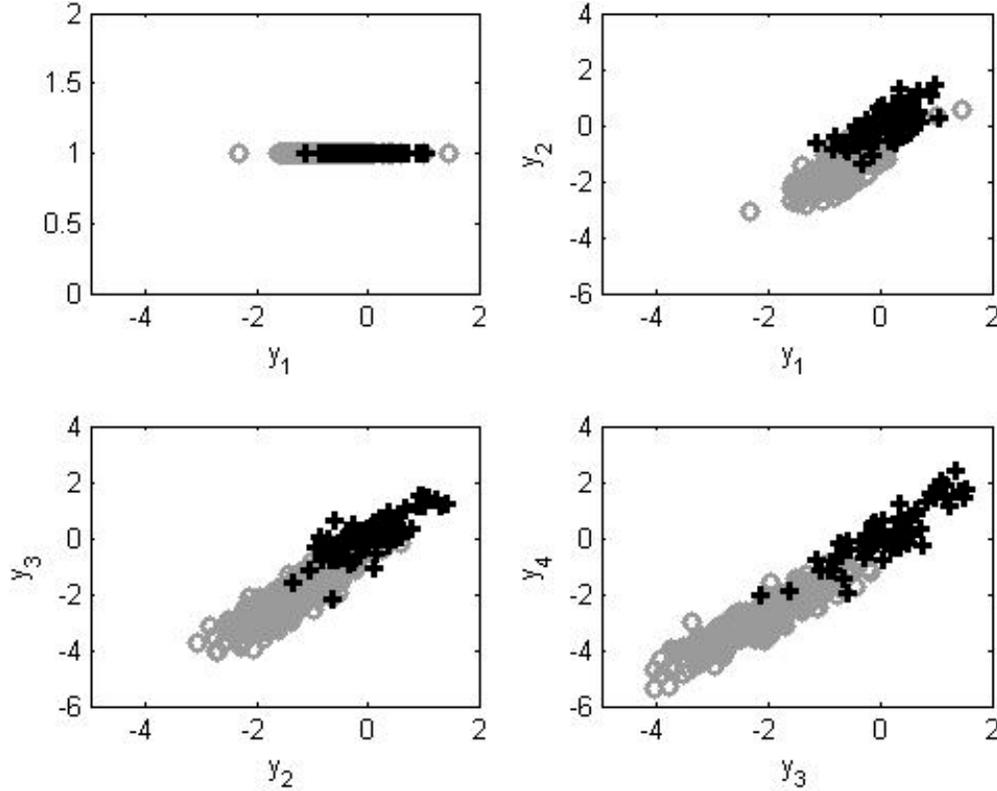


Figure 6. Projections of the cumulative distance vectors for $n=1$; 2; 3 and 4 over the indicated 1-D ($n=1$) or 2-D ($n=2$; 3 and 4) subspaces. The dark “+” data represent directed distances of infrequent vectors and the tenuous “o” directed distances of frequent vectors obtained from the first classification stage. Note that as n -value increases data from both classes get more separated.

C.1. Cumulative Distance Vectors Construction

After feature extraction, data taken from second classification stage training dataset are mapped into the space defined by the first classification stage. The side of the corresponding hyperplane where the resulting data are located determines their class belonging; to know whether a given data is in the infrequent or frequent region, the directed distance, $y_{i,n}$, from it to the classification boundary is computed as it follows:

$$y_{i,n} = w^T x_{i,n} + w_0, \quad (1)$$

where w and w_0 correspond to the parameters determined by the first classification stage and $x_{i,n}$ represents the i -th feature vector in the n -th stimulation sequence for $i=1,2,\dots,12$ (i.e. each of the twelve targets) and for $n=1,2,\dots,N$. As it was mentioned, the sign of this result indicates the side of the hyperplane on which the data fall;

to gather the evidence each target presents at the end of every stimulation sequence a cumulative distance vector, \mathbf{Y}_{in} , is constructed, whose n -th element is computed in the following terms:

$$\mathbf{Y}_{in} = \sum_{j=1}^n y_{i,j} \quad (2)$$

where $y_{i,j}$ represents the directed distance of the i -th target calculated from (1).

Figure 6 illustrates the projections of the cumulative distance vectors after the first ($n=1$); second ($n=2$); third ($n=3$) and fourth ($n=N=4$) stimulation sequence. As one can see, the results generated for both classes vary in very different ranges; the more stimulation sequences the system presents, the more distinguishable the vectors become.

From these results N new training datasets are constructed, each one containing the corresponding $\mathbf{Y}_{i,n}$ cumulative distance

vectors, for $n=1,2,\dots,N$. Each dataset contains representative information of the n -th and the previous ($n-1$) target-related responses, which is used to generate the second classification stage corresponding to a bank of N linear discriminants.

C.2. Second Classification Stage

Each one of the N linear discriminants generated in this stage is trained over half of the second classification stage training dataset, as the other half will be used to fit the corresponding logistic sigmoid functions. Each dataset consists of the first n vector components of the training data,

for $n=1, 2, \dots, N$; over which the same feature selection method as the one mentioned in the first classification stage section is performed.

The resulting parameters consist of a set of numbers indicating which elements of the cumulative distance vectors contained the most relevant information to differentiate data from both classes; a vector, v_n , which represents the normal vector to the hyperplane which best separates both memberships; and a third scalar parameter, v_{0n} , which denotes the bias of the classification surface. This means that from each of the N training datasets a classification space is defined; so that, every unseen data classification label can be determined after the n -th stimulation sequence by calculating the directed distance, $z_{i,n}$, between the resultant mapped cumulative distance vector to the classification space, $\mathbf{Y}'_{i,n}$ and the corresponding hyperplane in the following terms:

$$z_{i,n} = v_n^T \mathbf{Y}'_{i,n} + v_{0,n} \quad (3)$$

As it was said, the farther away the mapped vector is from the classification surface the more reliable the classification result; under this scheme, in order to map the whole real axis into a finite interval comprising the range $(0,1)$, a logistic sigmoid function is fitted to the ensuing directed distances. This function can be

presented, in a parametric form, as follows:

$$\sigma(z_{i,n}) = \frac{1}{1 + e^{\alpha z_{i,n} + \beta}} \quad (4)$$

where $z_{i,n}$ denotes the corresponding directed distance and, α & β , represent two adjusted parameters which are calculated in order to enhance the regression result using an improved fitting algorithm [7,8] over the remaining half of the second stage classification data.

As it is assumed that the class-conditional densities for infrequent and frequent classes are Gaussian and that they share the same covariance matrix as well, the results obtained from (4) are equivalent to Bayes' posterior probability, so that:

$$p(C_{inf}|z_{i,n}) = \frac{p(z_{i,n}|C_{inf})p(C_{inf})}{p(z_{i,n}|C_{inf})p(C_{inf}) + p(z_{i,n}|C_{freq})p(C_{freq})} = \sigma(z_{i,n}) \quad (5)$$

This result means that it is possible to compute the posterior probability without having to calculate the class-conditional densities required in order to evaluate Bayes' theorem, since using a logistic sigmoid function represents an equivalent procedure. It is important to emphasize that the second stage classification outputs consist then of probability values which indicate, after every stimulation sequence, all target-related evidence of belonging to the infrequent class.

In figure 7 the classification spaces generated for $n=1$; 2; 3 and 4, as well as the corresponding logistic sigmoid functions, are showed. Observe from the first graphics row that as n increases the elements of both memberships get more separated, so that the classes get well disjoint and are divided by the corresponding decision boundary. Given that the mapped cumulative distance vectors for $n=1$ contain a single element, the resulting classification space is one-dimensional and the corresponding decision boundary is a zero-dimensional hyperplane (i.e. a point). This means that the one-element vectors generated after the first stimulation sequence contain relevant information to differentiate both classes; otherwise it would not have been possible to

define a linear discriminant for these data. The resulting classification spaces for the remaining cases consider the last two elements of each cumulative distance vectors datasets as those presenting the most relevant information to differentiate data from both classes, so that each model consists of a two-dimensional space defining a one-dimensional hyperplane decision boundary (i.e. a line). The second graphics row shows the logistic sigmoid function for $n=1; 2; 3$ and 4 ; the dark data points labeled with “1” consist of the corresponding directed distances from the infrequent data to the decision boundary; and the “0” labeled tenuous data points are the corresponding directed distances calculated for the frequent data. The resulting logistic sigmoid functions fitted to all these points is plotted in black; as it was mentioned, its evaluation returns a value falling between 0 and 1, which indicates the posterior probability

of belonging to the infrequent class. Observe that the intersection of both plots occurs in the middle of this finite interval; where the probability of belonging to either the infrequent or frequent class is 50% and it is related to any data point lying on the hyperplane surface. As the data get farther from this threshold, the certainty of belonging to one of the classes, increases and the system can make a decision based on the found evidence. Because of the mutually exclusive relationship between frequent and infrequent events, the posterior probability of belonging to the frequent class can be computed in terms of the complement of the infrequent class posterior probability, as can be noticed from the symmetry between the dark and the tenuous plots, corresponding respectively to infrequent and frequent class posterior probabilities.

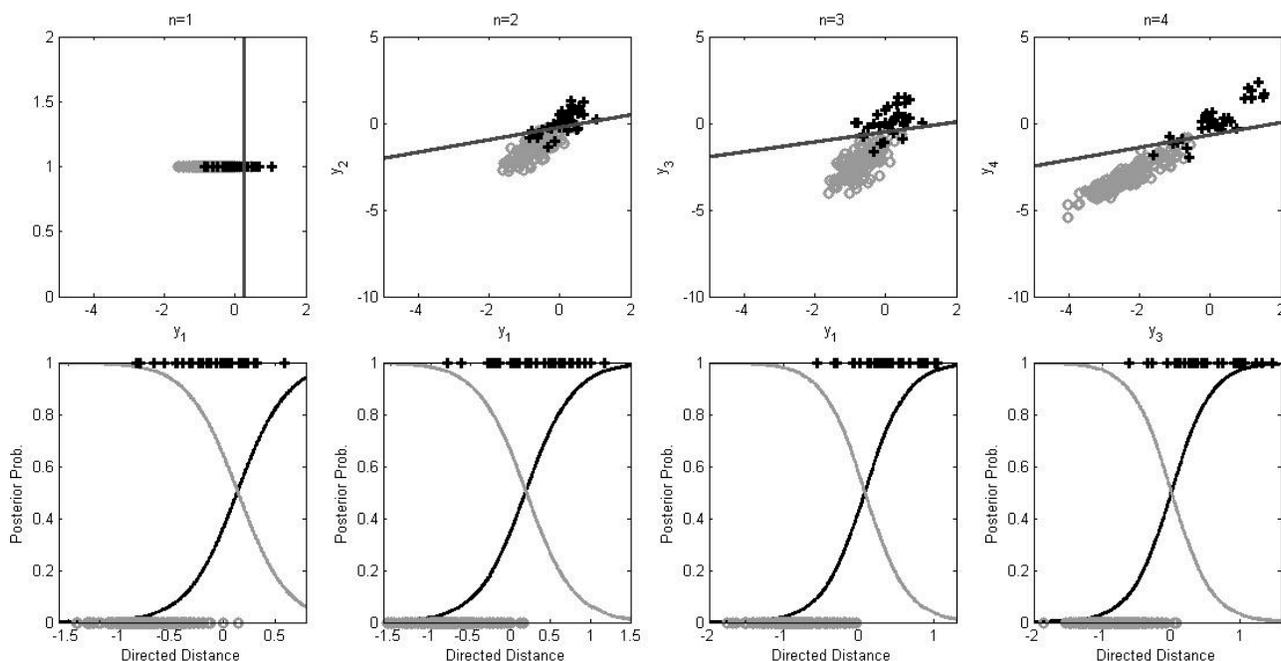


Figure 7. The first graphic row shows the classification space obtained for $n=1; 2; 3$ and 4 . Dark “+” data represent the infrequent mapped vectors and, tenuous “o” data, the frequent mapped vectors; observe that as n increases, data from both classes get more separated. The second graphic row shows the corresponding logistic sigmoid functions fitted to the ensuing directed distances. These logistic sigmoid functions allow compute results in terms of the posterior probability; which will be used in order to make decisions

In these plots an important overlap between data from both classes can be observed; it is evident that several elements near the intersection plots will be misclassified. The classification errors produce false positives and/or false negatives. A false positive implies to label as infrequent an actually frequent response, which leads to select a false coordinate from de symbol matrix. In the other hand, a false negative implies to label as frequent an actually infrequent response. This information, even when it is wrong, does not involve such an unwise decision as the one inferred for false positives; in this case the system just continues the stimulation process until it achieves to classify the infrequent data, without compromising the results precision.

This last paragraph shows that it is important to select strict thresholds to determine data classification labels and, even though an increase of the number of false negatives, avoid the occurrence of false positives. The decision thresholds are selected analyzing the training results, so that all the non-outliers outputs of the second classification stage get correctly classified. Figure 8 shows the thresholds selected; which corresponds to three variable limits varying according to n 's value due to the fact that, as n increases, the result get more reliable and it is possible to relax the decision criteria.

As the posterior probability results generated from the logic sigmoid functions are very similar for all subjects the decision thresholds selected are the same for all of them; the settled criteria according to these limits are described in the following sections.

D. Decision Criteria

To select any matrix character it is necessary to find its row and column coordinates; the system is unable to make a decision as long as one of them remains unknown. The selection approach is performed in parallel individually for columns and rows; for each group it is expected to obtain five frequent targets and only one infrequent. To make the corresponding inference, four criteria are subsequently applied over the posterior probability results obtained in the previous section as described below.

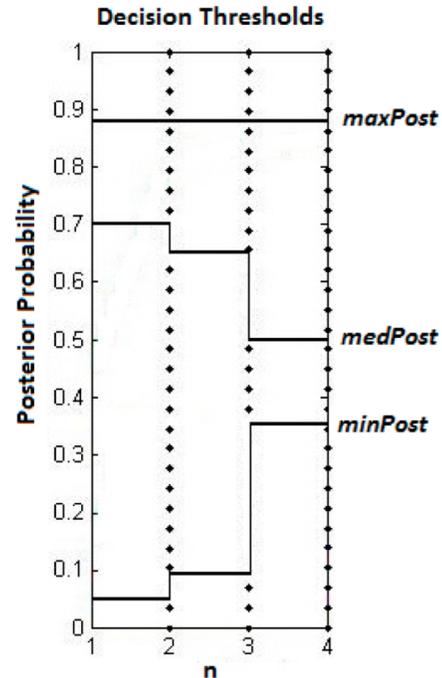


Figure 8. Decision thresholds. Details of their use are described in section II.D

D.1. If there is any value of $p(C_{inf}/z_{i,n}) \geq \text{maxPost}$, for $i=1,2,\dots,6$, select the target which generates the largest posterior probability. . The maxPost value represents a limit above which the certainty of belonging to the infrequent class is enough to make a decision. During the performed tests, this parameter was set to 90%; the first target showing equal or greater evidence is labeled as infrequent. If there is no value satisfying this condition the second criterion is applied.

D.2. If there is a unique value of $p(C_{inf}/z_{i,n}) \geq \text{mediumPost}$, for $i=1,2,\dots,6$, select the corresponding target. The mediumPost parameter corresponds to a less rigid threshold than maxPost , but there is an important constraint that must be satisfied: there must be only one value beyond this limit to be able to make a decision. If there are two or more greater values, and the maximum number of sequences has not been exceeded, the system keeps stimulating the user until it achieves to differentiate the results. The mediumPost initial value is 0.75 and its final value is 0.55; this means that evidence with 75% certainty of belonging to the infrequent class is required to make a decision after the early stimulation sequences;

and a certainty of 55% after the latest ones. This threshold can be decreased because the results become more reliable as more information from subsequent responses is available. If there is no value equal or greater than $mediumPost$, there is still a possibility to make a decision based on the third criterion.

D.3. If there is a unique value of $p(C_{inf}|z_{i,n}) \geq minPost$, for $i=1,2,\dots,6$, select the corresponding target. This criterion is applied when no target shows enough evidence of belonging to the infrequent class. In this case the approach consists in determining the posterior probability of each target of belonging to the frequent class. If there are five targets showing reliable evidence of being of this type, the only discarded one is labeled as infrequent. The frequent class posterior probability is computed in terms of $p(C_{inf}|z_{i,n})$; this means the aim is to find those targets showing little evidence of belonging to the infrequent class. The $minPost$ parameter has an initial value of 0.05 and a final value of 0.3; which implies that there must be

five targets showing evidence with 95% certainty of belonging to the frequent class after the early stimulation sequence to label as infrequent the discarded one. As more information from subsequent responses is available, this tolerance decreases to 70%. If there is no value equal or greater than $minPost$ and the maximum number of sequences has not been exceeded; the systems keeps stimulating the user until it achieves to differentiate the results. If the system was not able to make a decision after the last stimulation sequence, the fourth criterion is applied.

D.4. If none of the preceding criteria are satisfied and the number of stimulation sequences is equal to N , label as infrequent the one which has the maximum value of $p(C_{inf}|z_{i,n})$.

RESULTS AND DISCUSSION

In this section, the system performance is described using Table-1 validation dataset; which consists of an unseen data group from which the efficiency is evaluated.

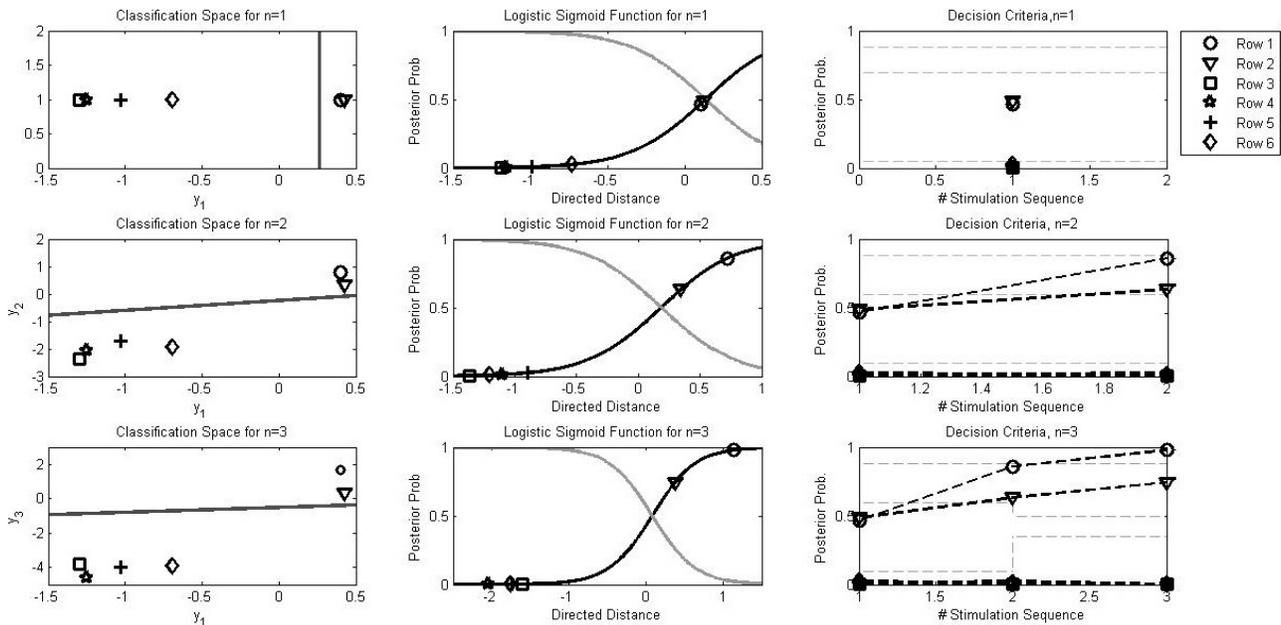


Figure 9. Infrequent target selection process over the row data for one of the validation dataset characters. The first row data meets the first criterion after the third stimulation sequence, so that the system selects as infrequent the first matrix row. Before that because of the nearness between the first and the second data it is no possible to make a decision.

In figure 9 the infrequent target selection process over the row data for one of the validation dataset characters is presented; each graphic line is related, from up to down, to one of the n -values, from $n=1$; 2 and 3; and they represent, from right to left, the mapped cumulative distance vectors, $\mathbf{Y}'_{i,n}$ to the corresponding classification space; the transformation over the logistic sigmoid functions of the directed distances, $z_{i,n}$, to posterior probability results, $p(C_{inf}|z_{i,n})$; and the applied decision criteria. Each one of the six rows as well as each one of the six columns of the symbol matrix is identified in figures 7 and 8. As can be inferred, the system is unable to make a decision until the third stimulation sequence had finished and the first row data meets the first criterion (i.e., its value is greater than *maxPost* threshold) so that the first matrix row is selected as infrequent. Before that because the second and first rows data are very close each other it is necessary in order to distinguish the infrequent-

type one to get more information.

Figure 10 shows the same described process for the corresponding column data. In this case the system would had been able to make a decision after the first stimulation sequence since the third column data meets the third criterion (i.e., is the only value greater than *minPost* threshold); nevertheless, as it is necessary to select both coordinates and the corresponding row had not been yet found, the system must continue the stimulation process. After the third stimulation sequence the row coordinate has been found and therefore it is possible to select the corresponding column coordinate; which is determined by the third row data meeting the third criterion (i.e., its value is greater than *maxPost* threshold); so that the chosen character consists of the one located at coordinates (1, 3), which corresponds to letter “C”. As can be verified from table 1 and the corresponding validation dataset this is a correct result consisting of the 22-subscript symbol.

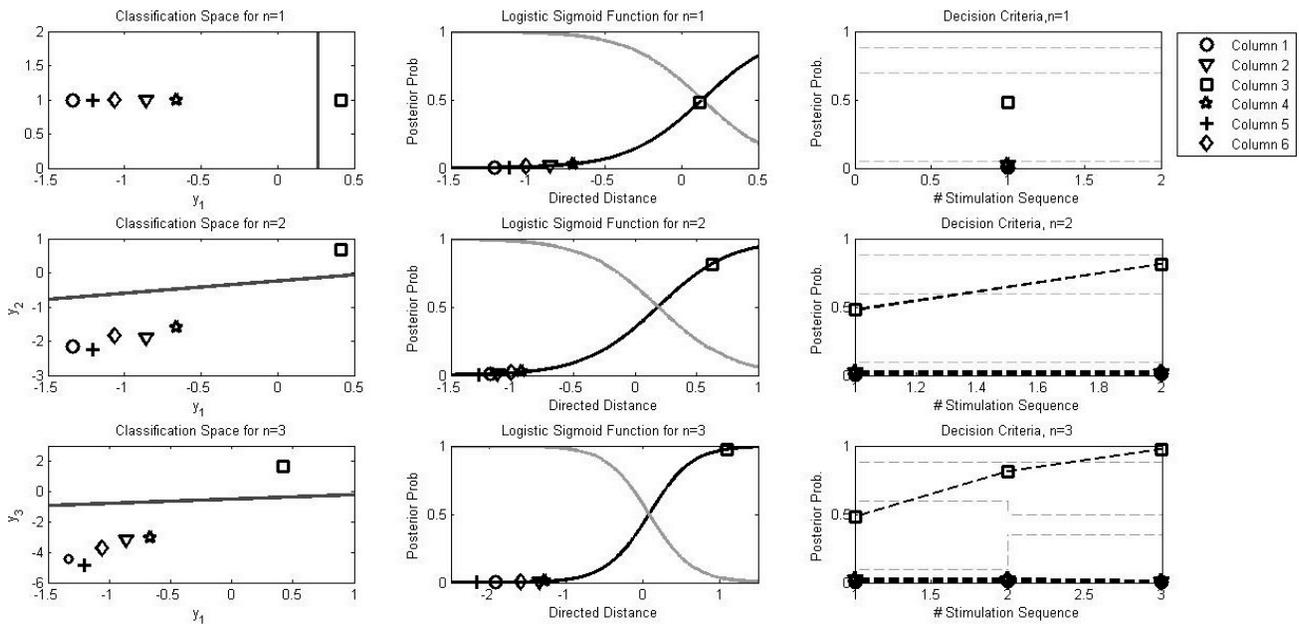


Figure 10. Infrequent target selection process over the column data for one of the validation dataset characters. Even when the third column data could have been selected since the first stimulation sequence based on the third criterion, the system must continue the stimulation process until it had found also the row coordinate.

Table 2. Performance comparison between the conventional method and the proposed methodology for a value of $N=\tilde{N}=4$.

Validation Results for $N = \tilde{N} = 4$					
Proposed Method			P300_Speller		
Accuracy (%)	char/min	seq/char	Accuracy (%)	char/min	seq/char
100	6.67	2.27	100	4.62	4

Table 2 shows the results obtained after using the proposed methodology over the validation dataset presented in table 1 for an N-value of 4; the corresponding results generated by the P300_Speller for $\tilde{N}=4$ are also shown in order to compare both performances. The performance indicators consist of the gained accuracy; the averaged characters spelled per minute; and the averaged stimulation sequences needed in order to select each matrix symbol. Observe that in both cases the gained accuracy is 100% but notice that the new proposal is more than 30% faster; which reveals the convenience of using this approach.

A 10-fold cross validation test is performed in order to guarantee the results independence from

the partition into training and test datasets; to this end table-1-training dataset is partitioned randomly with no repetitions to generate 10 groups. Each one consists of a test dataset comprising approximately 10% of the data; and a training dataset including the remaining information; as it was described in section B, first and second classification stages training datasets are generated from this last group. The results obtained for an N-value of 4 are shown in table 3 together with those generated by the P300_Speller[®] system using the corresponding value of $\tilde{N}=4$. Observe that in both cases the obtained results show little dispersion; which means that they are independent from the partition data.

Table 3. 10-fold cross validation for a value of $N=\tilde{N}=4$; note that the standard deviation obtained for the performance indicators presents little values, which proves the results independence from the partition into training and test datasets.

10-Fold Cross Validation Test for $N = \tilde{N} = 4$						
Fold	Proposed Method			P300_Speller		
	Accuracy (%)	char/min	seq/char	Accuracy (%)	char/min	seq/char
1	100	5.58	3	100	4.62	4
2	100	5.58	3	100	4.62	4
3	33.33	4.9	3.67	33.33	4.62	4
4	100	5.22	3.33	100	4.62	4
5	100	7.74	1.67	100	4.62	4
6	66.67	5.22	3.33	100	4.62	4
7	100	5.58	3	100	4.62	4
8	100	6.49	2.33	100	4.62	4
9	100	6.49	2.33	100	4.62	4
10	100	7.74	1.67	100	4.62	4
average	90 ± 22.5	6.05 ± 1.03	2.73 ± 0.7	93.33 ± 21.08	4.62 ± 0	4 ± 0

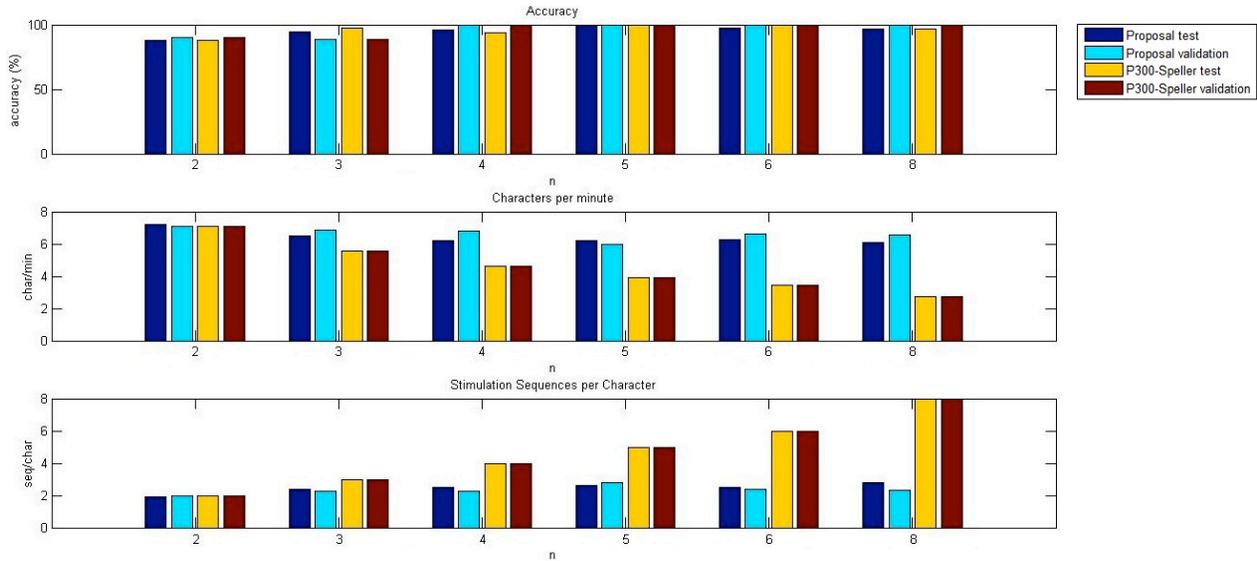


Figure 11. Results obtained from table-1 data using the proposed methodology and the P300_Speller[®] system for different values of $N=\tilde{N}$.

Table 4. Averaged results from 14 subjects using the proposed methodology and the P300_Speller[®] for different values of $N=\tilde{N}$.

Validation						
$N = \tilde{N}$	Proposed Method			P300_Speller		
	Accuracy (%)	char/min	seq/char	Accuracy (%)	char/min	seq/char
2	75.34±19.69	7.27±0.27	1.92±0.11	74.58±19.76	7,06±0	2±0
3	86.61±12.61	6.62±0.44	2.37±0.25	85.41±14.37	5.58±0	3±0
4	94.49±6.20	6.55±0.58	2.48±0.40	94.49±6.20	4.62±0	4±0
5	94.95±6.54	6.10±0.63	2.86±0.52	92.55±6.29	3.93±0	5±0
6	98.7±3.30	6.14±0.78	2.77±0.63	96.47±6.11	3.42±0	6±0
8	99.29±2.67	6.04±0.93	2.99±0.76	96.68±7.03	2.73±0	8±0

Figure 11 shows the results obtained for different N-values using table-1 data; the partition into first and second stage training datasets was performed for each particular N-value as it was described in section B; the corresponding \tilde{N} -value results generated by the P300_Speller[®] system are also shown to compare both performances. Additionally, the averaged 10-fold cross validation test outcomes obtained with each of the systems are plotted in order to verify the results independence from the partition into training and test datasets.

Observe that the gained accuracy with both systems is satisfactory and that it remained consistent at 100% from a value of $N=\tilde{N}=4$. As regards the spelling rate, it was much more superior for the proposed system since a value of $N=\tilde{N}=3$ and it remains consistent for greater values; on the contrary, for the P300_Speller[®] the spelling rate starts falling dramatically.

Finally in table 4 the averaged results obtained from all 14 users are shown; observe that the gained accuracy remains similar for both systems and that it is satisfactory from

a value of $N=\tilde{N}=4$. Once again, the resulting spelling rate for the proposed methodology is much more superior than the one achieved by the P300_Speller[®]. Observe that the indicated standard deviation shows little dispersion; which demonstrates that both methods are suitable for different users.

The averaged outcomes of the 10-fold cross validation obtained for each user also showed little dispersion, so that it can be stated the independence of the results from the partition data.

CONCLUSIONS

The results obtained in this work are promising; using a dynamic parameter to determine the number of stimulation sequences for each matrix symbol seems to be a very adequate approach to control this kind of applications. The compromise between spelling rate and spelling accuracy has been considerably reduced and the system presents as many stimuli as necessary to determine each symbol matrix; which diminish the average spelling rate. This improvement is due to the dynamic determination of the number of stimulation sequences; which allows selecting very quickly those targets showing important evidence of belonging to the infrequent class and using the saved time to find those difficult ones.

From the obtained results it can be observed that the new proposal gets for all N -values a little bit better accuracy as the gained for the corresponding \tilde{N} values by the P300_Speller[®]; nevertheless, as regards the spelling rate, an important improvement has been achieved by the new methodology; which enhanced the operating speed for an N -value of $N=\tilde{N}=2; 3; 4; 5; 6$ and 8 over 2.89%; 15.71%; 29.47%; 35.57%; 44.14% and 54.8%, respectively.

Acknowledgment

C. Lindig-León was awarded with a scholarship from the National Council of Science and

Technology (CONACyT) for the realization of this research project.

REFERENCES

1. Farwell L.A., Donchin E., "Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials", *Electroencephal Clin Neurophysiol*, 70(6):510-23, 1988.
2. Lee T.W., Yu Y.W., Wu H.C., Chen T.J., "Do resting brain dynamics predict oddball evoked-potential?", *BMC Neurosci*, 24:12:121, 2011.
3. Rompelman O, Ros HH., "Coherent averaging technique: a tutorial review. Part 1: Noise reduction and the equivalent filter", *J Biomed Eng*, 1986.
4. Ledesma C., Bojorges E.R., Gentiletti G., Bougrain L., Saavedra C., Yanez O, "P300-Speller Public-Domain Database", <http://akimpech.izt.uam.mx/p300db>
5. Schalk G., McFarland D., Hinterberger T., Birbaumer N., Wolpaw, J., "BCI2000: A General-Purpose Brain-computer Interface System", *IEEE Trans. Biomed. Eng.*, 51:1034-1043, 2004.
6. Bishop C.M., *Pattern Recognition and Machine Learning*, 1st ed, Springer, 2006.
7. Platt J., "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods", in *Advances in Large Margin Classifiers*, A. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, Eds. Cambridge, MA: MIT Press, 2000.
8. Lin H.T., Lin C.J., Weng R.C., "A Note on Platt's Probabilistic Outputs for Support Vector Machines", *Machine Learning*, Volume 68, No. 3, 2007.

