On-line detection of P300 and Error Potentials in a BCI speller

Bernardo Dal Seno[a], Matteo Matteucci[a], Luca Mainardi[b]
[a]*Dept. of Electronics and Information, Politecnico di Milano, IIT Unit, Milan, Italy*
[b]*Dept. of Bioengineering, Politecnico di Milano, IIT Unit, Milan, Italy*

Correspondence: B. Dal Seno, Politecnico di Milano, Dipartimento di Elettronica e Informazione. Piazza Leonardo da Vinci, 32. 20133 Milano, Italy. E-mail: bernardo.dalseno@polimi.it, phone +39 0223994031, fax +39 0223993411

Abstract. Error potentials (ErrPs), i.e., deflections/alterations of the EEG traces related to the subject perception of erroneous responses, have been suggested to be an elegant way to recognize misinterpreted commands in brain-computer interface (BCI) systems. In this work, we implemented a P300-based BCI speller similar to the one described in (Donchin et al., 2000) and we used a genetic algorithm (GA) to detect the occurrence of P300 waves. In addition, we added an automatic error-correction system (ECS) based on the single sweep detection of ErrPs. The developed BCI system, with the augmented features of ECS has been tested in on-line applications on three subjects, and here we report preliminary results. In two out of three subjects, the GA provides a good performance in detecting P300 (90% and 60% with 5 repetitions), and it has been possible to detect ErrP with an accuracy (roughly 60%) well above the chance level. To the extent of our knowledge, this is the first time that ErrP detection is performed on-line in a P300-based BCI. Preliminary results are encouraging, but further refinements are needed to improve performances.

Keywords: Brain-Computer Interface; P300; Error Potential; EEG Signal Processing; EEG On-line Pattern Recognition.

1. Introduction

A brain-computer interface (BCI) is an interface that does not entail muscle movements, but it bypasses any muscle or nerve mediation and connects a computer directly with the brain by picking up signals generated by the brain activity. Among the different kinds of brain activity that can be used in a BCI, the *P300* phenomenon has been known [Sutton et al., 1965] and investigated for many years. It is an event-related potential (ERP), visible in an EEG recording as a positive peak at approximately 300 ms from an event. It follows unexpected, rare, or particularly informative stimuli, and it is stronger in the parietal area. The shape of the P300 depends on the characteristics of the stimuli and their presentation.

For BCI applications, the "exact" shape of the P300 is not so important, as to have a way to detect it. Detecting a P300 in a single trial is very difficult and, therefore, repeated stimuli are normally used to facilitate the selection of the one which has generated a P300. The number of repetitions can be predetermined for each user to get the best trade-off between speed and accuracy. In [Donchin et al., 2000], Donchin and colleagues presented the first P300-based BCI, called also P300 speller, which permits to spell words. A grid of letters and symbols is presented to the user, and entire columns or rows are flashed one after the other in random order (see Figure 1 below for an example). When the column/row containing the desired letter is flashed, a P300 is elicited. In Donchin's work, classification is made through stepwise discriminant analysis (SWDA) applied to averages of samples from epochs relative to the same stimulation (same row or same column).

Other BCI interfaces using the P300 protocol have been developed since then. In [Bayliss et al., 2004], a virtual-reality system is presented where subjects operates objects selected through the P300. Classification is made by comparing the correlation of single responses with the averages of all target and non-target responses. In [Piccione et al., 2006], subjects (healthy and impaired ones) control a cursor by choosing among four commands (up, down, left, right) via the P300. In tis case, single-sweep detection is performed; independent component analysis (ICA) is used to decompose the EEG signal, a fuzzy classifier identifies a candidate P300 component among the ones extracted by ICA, and a neural network classifies it as target or non-target. The system is more effective with healthy subjects, though no exact reason could be pinpointed. Finally, in [Vaughan et al., 2006], an initial

attempt at using a BCI in a home environment is reported: a person with amyotrophic lateral sclerosis uses a P300 speller on a daily basis.

Another relevant event-related potential is the *error potential* (ErrP hereafter), which is generated when a subject makes a mistake, and, more interestingly for BCI applications, when the machine behaves differently from the user intent. Known since the late 80s [Falkenstein et al., 1991, Gehring et al., 1993], ErrPs were described as a negative shift in the electric potential over the fronto-central region (from Fz to Cz of the 10-20 system) occurring 50–100 ms after an erroneous response (*error negativity* — Ne — or *error-related negativity* — ERN) and a subsequent positive shift in the parietal region, whose maximum occurs between 200 and 500 ms after the error (*error positivity* — Pe). A high variability in shape, size, and delay of the Ne and Pe components has been observed as the effect of different underlying mechanism, whose nature is not yet certain [Falkenstein, 2004].

In [Schalk et al., 2000] the presence of ErrPs in a BCI paradigm (cursor movement by mu and beta rhythms) was revealed, as a positive peak at Cz 40 ms after the end of erroneous trials. This finding suggests an interesting application: the automatic detection of the errors made by a BCI in recognizing the user's intent and a way to improve its performances. Millán and colleagues [Ferrez and Millán, 2005, Ferrez and Millán, 2008] made experiments with ErrPs found in a motor-imagery BCI. They trained a Gaussian classifier to automatically recognize ErrPs reaching an accuracy of about 80%. In this work we present our experience in detecting P300 and ErrP in a P300-based speller with an integrated automatic error-correction system (ECS) based on the single sweep ErrP detection.

2. Experimental Setting and Data Processing

We developed a classical BCI based on P300, the P300 speller, and integrated the use of ErrP in it. Our P300 speller is very similar in the appearance and in functioning to the paradigm described by Donchin [Donchin et al., 2000]: 36 symbols are disposed on a 6×6 grid, and entire rows and columns of symbols are flashed one after the other in random order. The grid of symbols is visible in Figure 1: There are the letters from the alphabet, some digits, the space, and the *backspace*, represented as BS in the right bottom corner. The intensification of rows and columns lasts for 125 ms and the matrix remains blank for 125 ms between two consecutive flashes. Each row and column is flashed exactly once in the first 12 stimulations; then another round of 12 stimulations is repeated, with flashing of rows and columns done in a new random order, and this procedure is repeated for a total of 5 times. Each block of 12 consecutive stimulations is called a *repetition*, and there is no pause between repetitions.

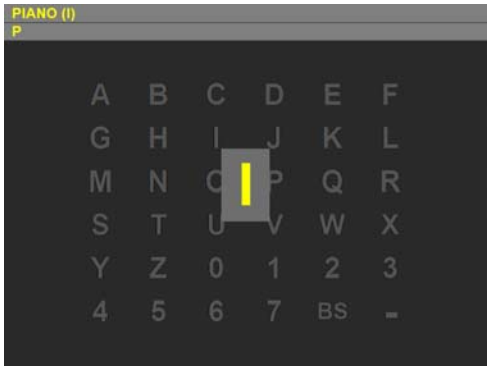

Figure 1. *Graphical interfaces of the P300 spellers used in the experiments, showing the moment of the letter feedback used for ErrP-based confirmation*

After the fifth repetition, the P300 system detects the row and the column that are more likely to have elicited a P300, and selects the letter at their intersection. After a pause of 1 s, the letter is presented to the user in a big rectangle that pops up in front of the grid (see Figure 1). The presentation of the letter should elicit an ErrP if the letter predicted by the P300 system is different from the one the user intended. An ErrP detection system figures out if any ErrP is elicited by the presentation of the selected letter, and in that case it overrides the P300 speller and cancels the last selection; otherwise, the letter is appended to the text at the top. After a 2–3 s pause (this parameter is tuned to each subject's requirements), the speller starts a new series of stimulations for the next letter. A *trial*, in this context, is the whole series of 60 row/column flashes (12 flashes times 5 repetitions) together with the

feedback of the speller selection made for each letter, i.e., a single trial is composed of 60 P300 stimulations and 1 ErrP stimulation (a trial is about 15 s long).

The speller we have implemented is based on BCI2000 [Mellinger and Schalk, 2007], a general-purpose software system developed at the Wadsworth Center of the New York State Department of Health in Albany, New York, USA, for brain-computer interface (BCI) research. We developed three main components: a source module that acquires EEG data from our amplifier, an application derived from the built-in P300 speller, and a dual-classifier processing module to handle both P300 and ErrP classification. The application module implements the P300 speller with ErrP-based error correction, as described above, and a precise synchronization system (fully described in [Dal Seno, 2009]). The processing module splits the EEG signals in epochs synchronized on the stimulation instants, processes the data, and performs the classification of the epochs according to two separate processing chains, one for P300s and one for ErrPs, briefly described below.

EEG data are acquired with an EBNeuro BE Light amplifier at locations Fz, Cz, Pz, and Oz, and at a frequency of 512 Hz. Also, EOG is recorded from the right eye of the subject. For P300 detection, a logistic classifier [le Cessie and van Houwelingen, 1992] is used, trained on features extracted through a genetic algorithm. Genetic algorithms are a class of optimization algorithms that mimic the way natural evolution works; they combine and select sub-optimal solutions until a very good solution is found. In our case, the genetic algorithm optimizes the features specifically for the logistic classifier, which is built inside the algorithm. A feature is obtained by multiplying an EEG channel in an epoch by a weight function, whose exact shape depends on parameters that are encoded in each candidate solution (Figure 2 shows two examples of weight functions). For a complete description of the algorithm, please see [Dal Seno et al., 2008b].

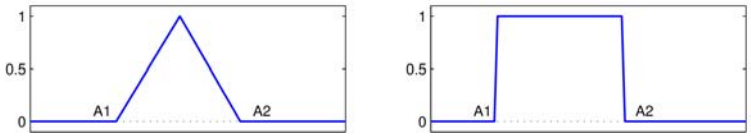

Figure 2. *Weight functions used for P300 feature extraction*

For ErrP detection, a simpler method is used, also because fewer training data are available (there is one ErrP stimulation per letter versus sixty for P300). In the ErrP case, recorded data are segmented in epochs whose extremes are selected with the method fully described in [Visconti et al., 2008]; in short, a statistical analysis is performed to find intervals where ErrP and non-ErrP epochs differ significantly, and some selected intervals are fused together. Features consists just in every fourth sample of the EEG signals, and they are fed into a classifier trained through linear discriminant analysis (LDA).

3. Results

Three subjects participated in a first set of on-line experiments. The P300 speller used 5 repetitions of each stimulation per letter, except for one user, who was tested also with a lower number of repetitions. In these experiments, the users had to select letters indicated to them by the BCI before each trial, so as to simplify the evaluation of the performance. The results are shown in Table 1: They are similar to those obtained off-line (see [Dal Seno et al., 2008b]), and confirm the validity of the classification method. The low performance of Subject B2 seems not to be due to the small training set, but to low concentration; the subject reported problems in focusing on the task, probably because of a failure of the brightness regulation of the computer screen that affected the recordings.

Table 1. *Results of the GA on-line. Training set size is the number of letters spelled in the training set. Performance is given as the number of correctly predicted letters over the total numbers of letters in the on-line usage.*

Subject	*Training set size*	*No. of repetitions*	*On-line performance*
		5	41/44 (93%)
B1	196	4	19/21 (90%)
		3	40/52 (77%)
B2	67	5	57/233 (24%)
B3	108	5	108/181 (60%)

Subjects B1 and B3 also tried to use the BCI to spell words in the so-called *free mode*, where they spelled words of their own choosing and had to correct errors by selecting *backspace*. The results are shown in Table 2 and confirm that the classifier found by the GA can be used to really drive a BCI application. Subject B2 could have tried to use the speller by increasing the number of repetitions, but as the data was recorded also to evaluate error potentials, this would have made the recording sessions much longer.

Table 2. *Results of the GA on-line in free mode. Training set size is the number of letters spelled in the training set. Performance is given as the number of correctly predicted letters over the total numbers of letters in the on-line usage.*

Subject	*Training set size*	*No. of repetitions*	*On-line performance*
B1	196	4	74/109 (68%)
B3	108	5	137/202 (68%)

The experiments with the P300 speller made use also of ErrPs. Classification was applied both off-line and on-line during such experiments; of particular significance are the experiments made with the speller in *free mode*, i.e., when the ErrP classification influences the spelling and the user has a real-time feedback about the classifier decisions and performance. Such experiments were performed only by subject B1 and B3, as already explained earlier.

Results of the on-line experiments are shown in Table 3. The classifiers were tested in sessions different from those used for training, so they are really indicative of a possible on-line use. For both users the classification performance is well above chance level, but this is not enough to say whether ErrP detection has been useful for such users.

Table 3. *Results of the on-line ErrP classification. Training and test set size is the number of selections of each class. Performance is the fraction of correct classification.*

Subject		*Training set size*	*On-line performance*
B1	ErrP	84	23/35 (66%)
	Non ErrP	290	51/74 (69%)
B3	ErrP	64	38/65 (58%)
	Non ErrP	193	91/137 (66%)

4. Discussion

In this paper we have presented an experiment that — to the best of our knowledge — is the first attempt to use a P300 BCI with an integrated error-correction mechanism based on ErrP detection. Although the number of subjects that participated to the on-line study is quite limited, results are encouraging and confirm the feasibility of ErrP single-sweep detection already verified in more populated off-line studies such as [Visconti et al., 2008] or [Dal Seno, 2009].

The use of a genetic algorithm for the definition of features to be used in P300 detection has proven its strength also in the on-line use, after good results in off-line analysis [Dal Seno et al.,

2008b]. In principle, the very same algorithm could be used for the ErrP feature design, but is is somehow prevented by the reduced number of examples that can be gathered in training sessions. A different strategy could be devised for an automatic way of collecting ErrP examples while using our P300 based BCI application, since each back space in *free mode* can be treated as an explicit tagging of an ErrP by the user. With this strategy, data gathering would be still time consuming (we are not changing the odds for ErrP elicitation after all), but it could be more acceptable by the user, and it might enhance her experience with the speller as time passes.

The results presented are encouraging, but some additional work is still needed to improve the performance. In particular, it is important that ErrP detection reaches a high accuracy, higher than P300 detection. The reason is that ErrP stimulations are generated only once after each letter selection, and this is the only chance to detect an ErrP. An accuracy higher than chance is not sufficient to have a usable interface; the exact impact of error correction on the performance of the speller can be assessed with the *utility* metric introduced in [Dal Seno et al., 2008a] and [Dal Seno, 2009]. We plan to refine our processing methods and presentation interface (to better capture the subject attention) so as to increase the performance of the ErrP classifier; in a more extensive study with more subjects we will assess the on-line performance of the enhancement and also to put the utility metric to the test.

Acknowledgements

This work has been partially supported by the Italian Institute of Technology (IIT), and by the grant “Brain-Computer Interfaces in Everyday Applications” from Politecnico di Milano and Regione Lombardia.

References

Bayliss JD, Inverso SA, and Tentler A. Changing the P300 brain computer interface. *Cyberpsychology & Behavior*, 7(6):694–704, 2004.

Dal Seno B. *Toward An Integrated P300- And ErrP-Based Brain-Computer Interface*. PhD thesis, Politecnico di Milano, 2009.

Dal Seno B, Mainardi L, and Matteucci M. Assessing the performance of a BCI: A task-oriented approach. In *Proceedings of the 4th International Brain-Computer Interface Workshop & Training Course*, 2008a, pages 274–279.

Dal Seno B, Matteucci M, and Mainardi L. A genetic algorithm for automatic feature extraction in P300 detection. In *2008 International Joint Conference on Neural Networks (IJCNN)*, 2008b, pages 3145–3152.

Donchin E, Spencer KM, and Wijesinghe R. The mental prosthesis: Assessing the speed of a P300-based brain-computer interface. *IEEE Transactions on Rehabilitation Engineering*, 8(2):174–179, 2000.

Falkenstein M. ERP correlates of erroneous performance. In Ullsperger M and Falkenstein M, editors, *Errors, Conflicts, and the Brain. Current Opinions on Performance Monitoring*, 2004, pages 5–14.

Falkenstein M, Hohnsbein J, Hoormann J, and Blanke L. Effects of crossmodal divided attention on late ERP components. II. Error processing in choice reaction tasks. *Electroencephalography and Clinical Neurophysiology*, 78(6):447–455, 1991.

Ferrez PW and Millán JdR. You are wrong! —automatic detection of interaction errors from brain waves. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence*, 2005, pages 1413–1418.

Ferrez PW and Millán JdR. Simultaneous real-time detection of motor imagery and error-related potentials for improved BCI accuracy. In *Proceedings of the 4th International Brain-Computer Interface Workshop & Training Course*, 2008, pages 197–202.

Gehring WJ, Goss B, Coles MGH, Meyer DE, and Donchin E. A neural system for error detection and compensation. *Psychological Science*, 4(6):385–390, 1993.

le Cessie S and van Houwelingen JC. Ridge estimators in logistic regression. *Applied Statistics*, 41(1):191–201, 1992.

Mellinger J and Schalk G. BCI2000: A general-purpose software platform for BCI research. In Dornhege G, Millán JdR, Hinterberger T, McFarland DJ, and Müller KR, editors, *Towards Brain-Computer Interfacing*. MIT Press, 2007.

Piccione F, Giorgi F, Tonin P, Priftis K, Giove S, Silvoni S, Palmas G, and Beverina F. P300-based brain computer interface: reliability and performance in healthy and paralysed participants. *Clinical Neurophysiology*, 117(3):531–537, 2006.

Schalk G, Wolpaw JR, McFarland DJ, and Pfurtscheller G. EEG-based communication: presence of an error potential. *Clinical Neurophysiology*, 111(12):2138–2144, 2000.

Sutton S, Braren M, Zubin J, and John ER. Evoked-potential correlates of stimulus uncertainty. *Science*, 150(3700):1187–1188, 1965.

Vaughan TM, Mcfarland DJ, Schalk G, Sarnacki WA, Krusienski DJ, Sellers EW, and Wolpaw JR. The Wadsworth BCI research and development program: At home with BCI. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):229–233, 2006.

Visconti G, Dal Seno B, Matteucci M, and Mainardi L. Automatic recognition of error potentials in a P300-based brain-computer interface. In *Proceedings of the 4th International Brain-Computer Interface Workshop & Training Course*, 2008, pages 238–243.