A Direct Optimization Approach to the P300 Speller

Roberto Santana
Computational Intelligence Group
Universidad Politécnica de Madrid, Spain
roberto.santana@upm.es

Santiago Muelas
Universidad Politécnica de Madrid, Spain
smuelas@fi.upm.es

Antonio LaTorre
Universidad Politécnica de Madrid, Spain
atorre@fi.upm.es

Jose M. Peña
Universidad Politécnica de Madrid, Spain
jmpena@datsi.fi.upm.es

ABSTRACT

The P300 component of the brain event-related-potential is one of the most used signals in brain computer interfaces (BCIs). One of the required steps for the application of the P300 paradigm is the identification of this component in the presence of stimuli. In this paper we propose a direct optimization approach to the P300 classification problem. A general formulation of the problem is introduced. Different classes of optimization algorithms are applied to solve the problem and the concepts of k-best and k-worst ensembles of solutions are introduced as a way to improve the accuracy of single solutions. The introduced approaches are able to achieve a classification rate over 80% on test data.

Categories and Subject Descriptors

G.1 [**Optimization**]: Global optimization; G.3 [**Probabilistic methods**]

General Terms

Algorithms

Keywords

brain-computer-interfaces, P300, ensembles, neuroinformatics, classification

1. INTRODUCTION

Brain Computer Interfaces (BCIs) [8, 9, 11, 21] allow to translate the brain signals into commands without the need for motor intervention. The use of BCIs has provided an unprecedented alternative for the communication of individuals with severe communicative impairments. Exemplary uses of BCI technology include games, virtual environments [13], and space applications [15].

GECCO'11, July 12–16, 2011, Dublin, Ireland.

There is a variety of neural signals that have been identified for BCI such as slow cortical potentials, μ and β rhythms, cortical neuronal action potentials, and P300 evoked potentials. They can be recorded using different experimental paradigms and when used for BCI, exhibit different information rates.

One of the neural signals that have been successfully used in the implementation of BCIs is the P300 event-related potential. It is evoked in scalp-recorded electroencephalography (EEG) by external stimuli. One example of the use of this type of signal is the P300-speller. The P300 paradigm presents 36 letters in a 6×6 matrix on the computer screen. Columns and rows of the matrix flash up in a random order and the subject has to be concentrated on the letter it wants to write. Around 300 ms after the corresponding letter is flashing up, the P300 component of the event-related potential (ERP) [3] arises.

To determine when a P300 potential has been evoked and the event that has produced it, classification methods are applied to the EEG signals. Among the classification techniques used with the P300 Speller are Pearson's correlation method [6], stepwise linear discriminant analysis [2], support vector machines [7], and matched filtering [6].

The classification of P300 potentials can be approached as a direct optimization problem in which the optimal solution would correspond to a perfect classifier. However, this is not the approach followed by the most common applied algorithms [6]. This fact might be due to the scarce analysis of the characteristics of the objective function associated to the classification problem. In addition, direct optimization approaches do not employ, to the same extent that other learning algorithms, the available knowledge about the problem. Finally, the solutions given to the P300 classification problem should be robust for unseen data, and it is difficult to guarantee this behavior for solutions achieved by direct optimization techniques.

In this paper, we address the P300 classification problem as a direct optimization problem. Our objective is to determine which are the factors that influence the complexity of the problem for optimizers. We also investigate how different types of optimization algorithms behave for the P300 classification problem.

The paper is organized as follows. In the next section, brain computer interfaces are reviewed and the P300 classi-

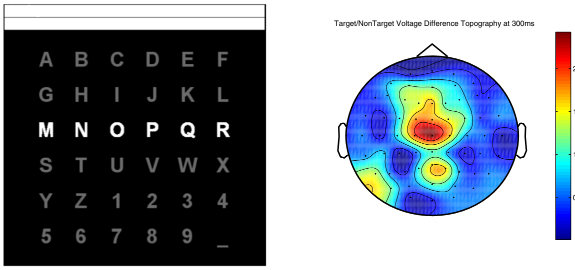

Figure 1: Schematic representation of the P300 speller as seen by the user on the display and channels location and distribution of the P300 response.

fication problem is introduced. In Section 3, the formulation of the P300 classification problem as a direct optimization approach is presented. Section 4 describes the steps in the processing of the brain signals and present the optimization methods used in our experiments. The experimental framework and the numerical results of our experiments are explained in Section 5. Finally, the conclusions of our paper are given in Section 6.

2. BRAIN COMPUTER INTERFACES

A BCI can be divided into a signal acquisition module and a signal processing or transformation module [21]. The signal acquisition is executed using EEG, Magnetoencephalography (MEG), or other techniques for recording brain activity. The transform algorithms usually employ machine learning techniques to convert the neural signals to commands (e.g. cursor or robot arm commands). The user may receive some feedback of this activity (e.g by looking a cursor movement on the screen).

We use the BCI2000's P300 speller paradigm, described originally in [4] and later used in [2]. Data recorded using this paradigm is employed in the section of experiments to compare the applied optimization algorithms.

The P300 Speller presents to the experiment participant a 6×6 matrix of characters. Each row and each column are intensified; the sequential series of intensifications are presented as a random sequence. The user focuses his attention on one of the 36 cells of the matrix. The row and the column containing the character to be communicated constituting the rare set. The other ten intensifications constituting the frequent set. If the participant is attending to the stimulus series, the row and the column containing the target character will elicit a P300 response.

The responses evoked by these infrequent stimuli (i.e., the 2 out of 12 stimuli that did contain the desired character) are different from those evoked by the stimuli that did not contain the desired character and they are similar to the P300 responses previously reported [2, 4]. Figure 1 shows a schematic representation of the P300 speller as seen by the user on the display and the distribution of EEG channels in P300 experiments (spatially distributed dots that cover the scalp) and the P300 evoked signal (different intensity associated to different colors (see color bar)) that is concentrated around the brain central region. EEG channels refer to the EEG electrodes positions. Each channel records brain signals from the scalp area it where is located.

3. P300 CLASSIFICATION AS DIRECT OPTIMIZATION

The presence or absence of a P300 evoked potential from EEG features can be considered a binary classification problem with a discriminant function having a decision hyperplane defined by [6]:

$$w \cdot f(\mathbf{x}) + b = 0 \tag{1}$$

where $\mathbf{x}$ is a feature vector, $f(.)$ is a transformation function, w is a vector of classification weights and b is the bias term.

In [6], it is shown that different classification methods can be posed in terms of alternative strategies for solving w and b. The general problem is simplified because it is assumed that a P300 response is elicited for one of the six row/column intensifications, and that the P300 response is invariant to row/column stimuli. Therefore, the resultant classification is taken as the maximum of the sum of scored feature vectors for the respective rows, as well as for the columns:

$$r^* = argmax_{rows}\left[\sum_{i_{row}} w \cdot f(x_{i_{row}})\right] \tag{2}$$

$$c^* = argmax_{columns}\left[\sum_{i_{column}} w \cdot f(x_{i_{column}})\right] \tag{3}$$

Class labels of $+1$ and -1 are assigned to the target and non-target stimuli, respectively. This design selects the response with the largest positive distance from the trained separating hyper-plane [6]. Since Equations (2) and (3) are invariant to the constant bias term b, it does not need to be computed.

The problem is transformed then in finding a vector of weights that maximizes the predicted accuracy for the training data. From the output of Equations (2) and (3), the predicted character is located at the intersection of the predicted row and column in the matrix.

A very simple example of how to compute the weights for a particular classification algorithm, is the use of the Pearson correlation method. In this case, $f(\mathbf{x}) = \mathbf{x}$, i.e., the features are not transformed. Let y be the class label variable, the weight w_i corresponding to feature x_i is computed as the correlation between x_i and y. In this way, the higher the absolute value of w_i is, the more significant the predictor variable x_i is for the model.

Using the correlation for computing the weights is a very efficient method. However, it does not consider dependencies between the different features. Therefore, investigating a direct optimization approach is an interesting alternative, particularly if the optimization method used is extremely efficient.

There are several steps involved in the preprocessing of the original brain signal and its conversion to the feature vector used in Equation 1. Preprocessing is a sensitive phase since the volume of available information is high. During the experiments, signals are continuously recorded from multiple channels.

The P300 potential is evoked between 300 and 350 ms after the presentation of the stimulus. Therefore, we could expect the brain signals corresponding to this period to be the most informative. Another related issue is the subset of channels from which the processed information is more relevant. Since the P300 has a maximal amplitude over the central and parietal scalp areas, it is expected that channels in this area will be the most informative. However, the exact amount of information each channel delivers is very subject-dependent and recent results suggest that the accuracy of P300-based classifiers can be increased by using a subject-dependent sets of channels [7, 14]. In this paper we start from an initial set of 64 channels.

We summarize these ideas by defining the extended P300 speller optimization problem, in which the variables involved are the following.

- Set of channels ($\{C_1, \dots, C_{nchannels}\}$) from which information is considered: Determining the most informative subset of the total number of available channels is a difficult optimization problem itself.
- Time window (TW): As mentioned above, the time window of the P300 evoked potential is known. However, inter-subject variability and other considerations make convenient to take a longer period of time. Determining the optimal time window in terms of the classification accuracy is also an optimization problem.
- Number of features per time window ($nFeat$): The time window of the signal is usually split in equal parts. From each part, a feature is constructed by averaging or doing some other type of signal transformation. Deciding which is the optimal number of features to take is part of the optimization problem.
- Vector of ($nchannels \times nFeat$) weights. This is the direct optimization problem explained in the previous section.

All the previous factors influence the accuracy of a solution and its robustness when applied to unseen data (test set). The general optimization problem is unlikely to be efficiently solved by an optimizer. Notice, for instance, that the number of weights depends on two other variables of the problems, therefore, solutions to the general optimization problem will have different number of components.

Since one of our goals is to understand the complexity of the optimization approach and investigate the suitability of different optimization algorithms, we focus on one scenario in which all the parameters, except the weights, are fixed. We also select a subset of 23 channels from the initial set of 64 channels.

4. OPTIMIZATION METHODS FOR THE P300 SPELLER

In this section we describe the different components of our optimization approach to the P300 classification problem.

4.1 Problem representation fitness function

We present the results for a fixed set of channels that comprises the following 23: P1, Pz, P2, PO7, PO3, POz, PO4, PO8, O1, Oz, O2, Iz, P3, P4, CP3, CP1, CPz, CP2, CP4, C1, Cz, C2 and FCz. The choice of the channels has been motivated because they cover part of the central and parietal scalp areas. Our selection of channels captures most of channels previously used in [7]. The time window size is of $755ms$ and for each time window, we take 5 features. We use a continuous vector representation for $\mathbf{w}$, $w_i \in \mathbb{R}$, $i \in \{1, \dots, n\}$, in which each w_i is represented using a single variable. As the range of values for the weights, we arbitrarily take $x_i \in [-10, 10]$. The number of variables will be equal to the total number of features, i.e. $n = 23 \times 5 = 115$.

To explain the fitness function, it is important to take into account the concepts of intensification and repetition mentioned in Section 2. For each of the 85 characters that are shown to the subject of the experiment, there are 15 repetitions of the 12 intensifications corresponding to the 6 rows and 6 columns. To determine for which of the six columns (respectively rows) a P300 potential has been evoked, the brain signals corresponding to the 15 repetitions are averaged for each of the (85×12) characters.

The averaging step intends to counteract the noise in single repetitions. As a result, there is a vector x^i of average signals for each target character $i \in \{1, \dots, 85\}$. For each target character, the vector is combined with the 12 vectors of signals (6 vectors for column and 6 for row intensifications). The putative character is found as the intersection of r^* and c^*, where r^* and c^* have been found as in Equations 2 and 3. Finally, the objective function is the number of times that the putative character coincides with the target character, i.e. the classification accuracy induced by the weight vector.

4.2 Basic optimization approach

In the basic optimization approach, we employ different direct optimization algorithms and evaluate them in the task of finding the optimal configuration of $\mathbf{w}$. The main steps shared by all the optimization methods are described in Algorithm 1.

Algorithm 1: **Steps for the problem solution**

Load the EEG training data for each channel, each character, and each repetition.
Compute the vector of features as the average signal in the time window, for each channel, each character and each repetition.
for $i = 1 : maxiter$
    Apply the optimization algorithm and output one solution
Evaluate different subsets of the optimal solutions (vector weights) in the test data.

In Algorithm 1, $maxiter$ corresponds to the number of times the optimizer is called within one cycle. The weight vectors found by the optimizers are evaluated on a test data set of brain signals recorded from the same individual in the same experimental conditions. The test data comprises information about 100 characters shown to the subject.

4.3 k-best and k-worst ensembles

We devise two alternative ways to evaluate the set of output optimal solutions. The first, traditional procedure, corresponds to evaluate each single solution independently. The second procedure, that we call, *ensemble method* corresponds to the combination of weight vectors. The rationale behind this idea is that weight vectors may specialize in different types of brain signatures, for example, brain signatures associated to different channels. The combination of these solutions may be more sensitive to the detection of the P300 evoked potential.

When used as a part of a classifier, a simple voting model is used. During a single classification task, first a putative character is proposed by each of the ensemble members. Then, the character that receives the highest number of votes is selected. If there exist more than one candidate with the same number of maximum votes, then the putative character suggested by the member with the highest fitness in the training set, is selected.

To decide which of the solutions will be part of the ensemble, first, solutions are ordered according to the accuracy computed in the training set. Then the k first solutions are selected as part of the ensemble. We call such a parametrized ensemble the k-best ensemble. Notice, that for $k = 1$, we have the case of the traditional approach of evaluating as the best candidate, the solution with the highest score in the training set.

Additionally, we use a different ordering of the solutions to construct the classifiers. In this case, solutions are sorted in ascending value of the training-set classification accuracy. The first solution will correspond to the weight vector with the lowest accuracy. We call the ensembles constructed in this way k-worst ensembles. This type of ensembles are interesting because they can serve to extract information about the gain in classification from the combination of lowest-quality solutions. The gain in k-best ensembles will be less for increasing k, since the best solutions are the first to be added to the ensemble.

4.4 Description of the optimization algorithms

As a first step of our experiments we use the Linear Discriminant Analysis (LDA) method [12] to evaluate the complexity of the classification task. LDA finds a linear combination of features which separates two or more classes of objects or events. Using LDA as a classifier produced classification accuracies of 0.42 and 0.59 for the first and second subjects, respectively.

For conducting the experiments, the following algorithms have been selected: the Genetic Algorithm (GA), the Differential Evolution algorithm (DE), the General Opposition-Based Differential Evolution algorithm (GODE) and the Random Hill Climbing search (RHC). GAs are the most used Evolutionary Algorithms (EA) in the literature. They were popularized by the work of John H. Holland in [5]. Since then, they have experienced a deep development and have been applied to solve complex problems in many different domains. The DE algorithm proposed by Storn and Price [17] is one of the recent evolutionary algorithms that, due to its results, has quickly gained popularity on continuous optimization. DE has shown better performance than many other EAs in terms of convergence speed and robustness over a broad spectrum of problems [19]. Opposition-Based Learning (OBL), introduced by Tizhoosh [18], is a machine intelligence strategy, which considers the current estimate and its opposite at the same time in order to achieve a better approximation of the current candidate solution. This idea has been used to enhance population-based algorithms. GODE is one of these successful applications which has shown excellent search abilities in solving both low-dimensional and high-dimensional problems [20]. Finally the RHC is a very simple stochastic search that, at each step, selects randomly a dimension, assigns it a random value of the interval and restores the original value if the fitness of the new solution is worse than the original solution.

All the algorithms were allowed a maximum of $1,000,000$ generations. Based on the parameter values of the literature, the following configurations for each algorithm have been analyzed:

- DE-1 : Population size: 20, CR: 0.5, F: 0.5, selection: random
- DE-2 : Population size: 50, CR: 0.5, F: 0.5, selection: random
- GA-1 : Population size: 50, Crossover operator : BLX-α, Mutation operator: Gaussian, Crossover probability: 0.9, Mutation Probability: 0.01, selection: tournament 2, elitism: 100%
- GA-2 : Population size: 50, Crossover operator : BLX-α, Mutation operator: Gaussian, Crossover probability: 0.9, Mutation Probability: 0.01, selection: tournament 2, elitism: 50%
- GA-3 : Population size: 50, Crossover operator : BLX-α, Mutation operator: Gaussian, Crossover probability: 0.9, Mutation Probability: 0.1, selection: tournament 2, elitism: 100%
- GA-4 : Population size: 50, Crossover operator : BLX-α, Mutation operator: Gaussian, Crossover probability: 0.9, Mutation Probability: 0.1, selection: tournament 2, elitism: 50%
- RHC : The maximum number of random changes is equal to the maximum number of allowed evaluations.
- GODE-1 : Population size: 20, CR: 0.5, F: 0.5, Probability of OBL $p_o = 0.4$,selection: random
- GODE-2 : Population size: 50, CR: 0.5, F: 0.5, Probability of OBL $p_o = 0.4$,selection: random

4.5 Related work

Among the classical methods used for feature extraction are autoregressive parameter estimation, wavelets transform, Karhunen-Loeve transform and other types of transformations [11]. Frequently employed classification methods include linear classifiers (e.g. thresholding, Bayesian, linear discriminant analysis) and also non-linear classifiers such as support vector machines, k-nearest neighbors and neural networks [10].

There is some previous work that combines the use of classification strategies with optimization. In [16], a regularized logistic regression method is combined with multi-objective variable selection using EDAs for the classification of MEG data. The experimental results showed that the proposal was able to improve classification accuracy compared with approaches whose classifiers have the set of channels fixed a priori. The concept of ensemble has been applied with a different interpretation in the context of P300 classification [14]. In this paper, the ensemble is formed by support vector machines (SVMs) which are learned from different subsets of the training dataset.

5. EXPERIMENTS

The main objectives of our experiment are: 1) To test the direct optimization approach as an alternative for the P300-classification problem. 2) To investigate the different factors of the P300 classification problem (direct-optimization) formulation that influence in the behavior of the algorithms and in the quality of the achieved results.

Alg/k	1	3	5	10	15	20	25
$DE-1$	0.69	0.73	0.74	0.77	0.79	**0.80**	0.79
$DE-2$	0.73	0.74	0.78	0.76	0.77	0.77	0.78
$GA-1$	0.64	0.69	0.75	0.75	0.74	0.76	0.77
$GA-2$	**0.75**	**0.78**	0.78	**0.80**	**0.80**	**0.80**	0.79
$GA-3$	0.69	0.76	0.78	0.79	0.78	0.79	0.79
$GA-4$	0.71	0.74	0.76	0.77	0.78	**0.80**	**0.80**
RHC	0.65	0.75	**0.81**	**0.80**	0.79	0.79	0.79
$GODE-1$	0.73	0.76	0.77	0.78	0.79	0.79	**0.80**
$GODE-2$	0.72	0.74	0.78	0.79	0.77	0.78	0.78

Table 1: Accuracy results computed from the test-data and averaged between the two subjects. Results have been achieved with different k-best ensembles.

The identification of these factors is a required step for the conception of optimization algorithms suitable for this problem. Other issues that are investigated in our experiments are the capacity of the ensemble method to improve the results of single solutions and to compare the behavior of the different optimizers used for our experiments.

The data set used in our experiments represents a complete record of P300 evoked potentials recorded with BCI2001 using the paradigm described in [2], and originally by [4]. Training and test data sets were recorded from two different individuals and used as part of the BCI competition III [1].

5.1 Numerical results

Table 1 summarizes the average results achieved by the optimizers, alone and combined as part of ensembles. The average classification accuraccies are, in general, above the the 0.75 threshold, which is a good value. As a reference, the best results available for this dataset, and winners of the BCI III competition, achieved an average classification rate of 0.965 [14]. However, this approach applies a sophisticated machine learning approach that includes the iterative application of ensembles of SVMs. More importantly, the authors use the information recorded from the set of 64 channels and apply an initial filtering to the data, steps that our algorithm does not require. The ranking of the average classification accuracy achieved by the 10 contributions to the contest were $\{0.965, 0.905, 0.9, 0.895, 0.875, 0.83, 0.785, 0.75, 0.335, 0.075\}$ [1]. Most of these approaches, either apply SVMs or other machine learning procedures such as PCA, boosting, etc. Almost all the methods apply filtering of the initial signal and/or channel selection based on the accuracy computed from the training set.

Another observation that follows from the analysis of Table 1 is that the quality of the results improves when the number of members in the classifier is increased. Best results are achieved for $k \geq 5$. The k-best ensembles clearly outperform the results achieved by the single solutions. In terms of the behavior of the algorithms for the training set,

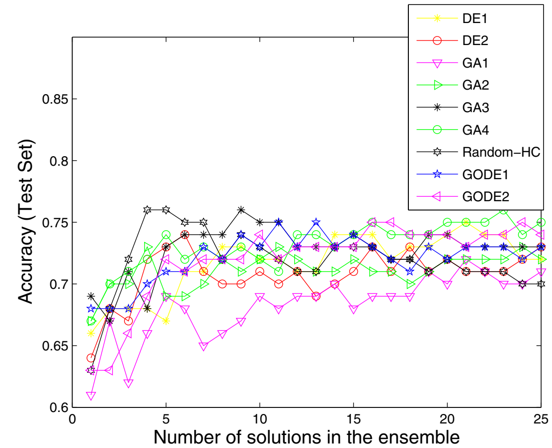

Figure 2: Results of the k-worst ensembles for individual 1 as k increases.

there are only statistical differences between algorithm RHC and the other algorithms (data not shown).

To analyze the differences between the P300 classification problem for the two subjects from a different perspective, we respectively show in Figures 2 and 3 the accuracies achieved by the k-worst ensembles for the different optimizers. For sake of clarity, we have averaged the results of the optimizers which are different only in their parameters values. It can be seen in the figures that, also for the k-worst ensembles there is an increase in the accuracies when k is increased. It is also evident that higher accuracies are achieved for the second individual. For this subject, accuracy values over 0.88 are achieved. Among the algorithms, there seems to be an advantage of the RHC algorithm over the rest. This is a somewhat unexpected result since this method is based on a single point search and uses none information about the search space.

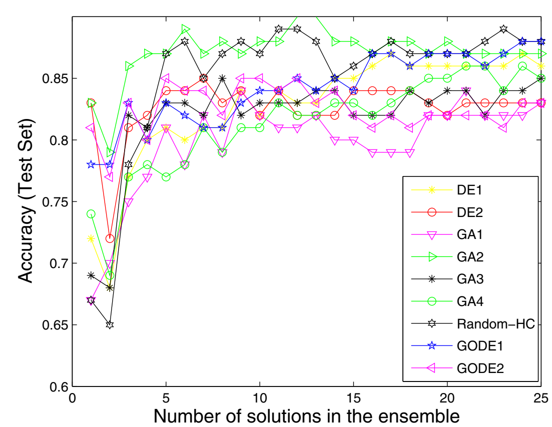

Figure 3: Results of the k-worst ensembles for individual 2 as k increases.

We conclude the analysis of the ensembles by comparing the output of the k-best classifiers, when k is increased, for

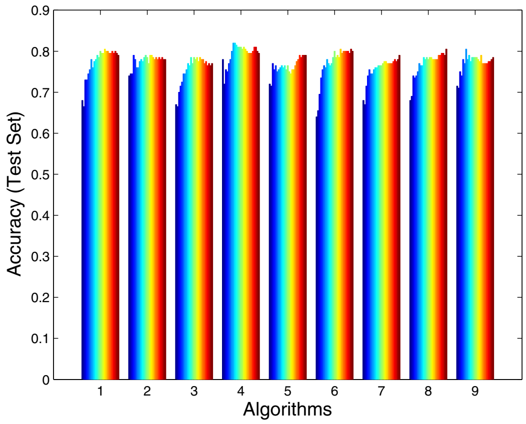

Figure 4: Average results of the k-best ensembles for all the algorithms and k values.

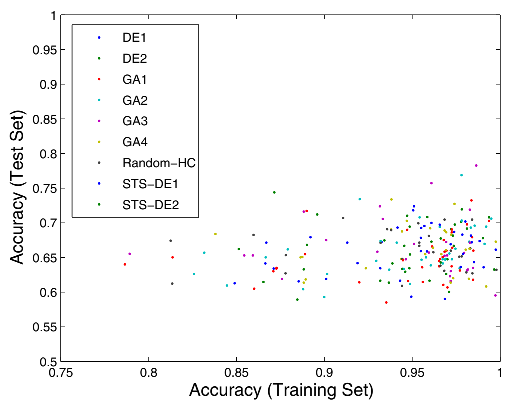

Figure 5: Mapping between the accuracy values obtained for training and test sets by the best found solutions for individual 1.

all the algorithms. These results are shown in Figure 4, where the numbers associated to the algorithms correspond to their order in Table 1. It can be seen from Figure 4 that, while some algorithms show an almost flat shape, e.g. Algorithm 2, corresponding to situations in which the addition of members to the ensemble does not produce an increase of the classification accuracy, other algorithms exhibit a characteristic saw shape, which correspond to situations in which new added ensemble members contribute to improve the accuracy. One of the issues that research on direct optimization approaches to the P300 classification problem reveals, is that the optimization algorithms could be conceived with the goal of finding a complementary set of solutions instead of a single optimal solution. The solutions could serve as an enhanced, more robust, ensemble. We did not modify our optimization algorithms to deal with this question. It is open for future research.

In the next step, we investigate the mapping between the training and test accuracy values for the two subjects. This is a critical point since, in the direct optimization formulation, we can expect good results of the methods, only if the information provided by the objective function in the training set is also informative (or predictive) about the quality of the solutions in the test set.

In Figures 5 and 6, the mapping between training and test objective values for the 225 optimal solutions, is shown. Since many solutions are assigned the same position in the graph (there are only 101 possible accuracy values), we have added a small random noise to the points coordinates to ease the visualization.

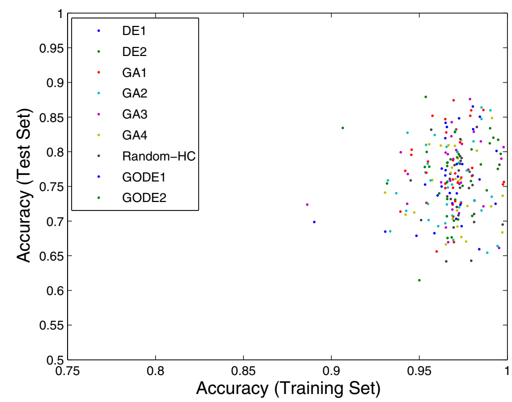

Figure 6: Mapping between the accuracy values obtained for training and test sets by the best found solutions for individual 2.

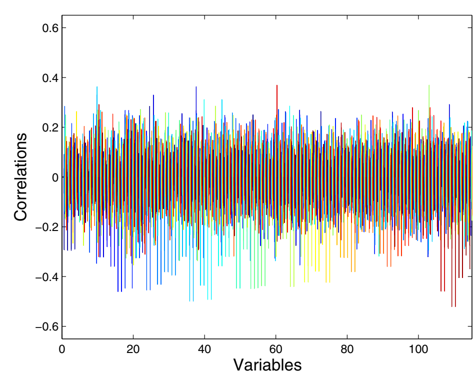

Figure 7: Correlations between the problem variables for individual 1.

The comparison between the two figures confirms that the second fitness landscape is easier for all the algorithms. For this function, most of the solutions are located above the 0.95 value in the X axis. Solutions shown in Figure 5 are more scattered in the X axis. The correlations between the accuracy values of the training and test sets were 0.1763 and 0.0972 for the first and the second individual, respectively. The low correlations illustrate one common problem in BCI experiments: intersession variability. Brain signals can strongly vary between sessions and classifiers should be able to adapt to these changes.

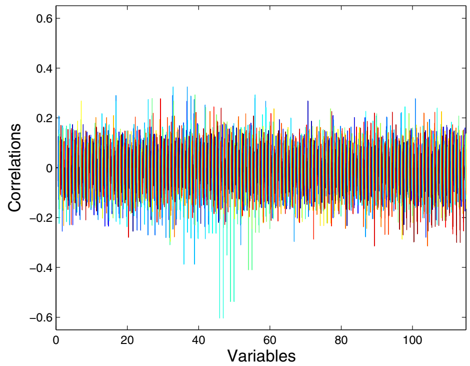

Figure 8: Correlations between the problem variables for individual 2.

We also analyzed the correlations between the problem variables from the sets of solutions found by the algorithms. The analysis of the correlations has two goals. The first one, is to identify complex patterns of correlations that may explain the difficulties of the optimizers in solving the problems. If correlations are common and strong, then some optimization methods, specially suited to deal with interactions between the variables could be recommended for this problem.

The second use of the analysis of correlations is to detect features that interact in the solution of the problem. If there is a couple of weights with a significant correlation, the corresponding features they map could be candidates of brain signatures. We are thinking of situations in which the solutions obtained by direct optimization are preprocessed and analyzed by statistical techniques to create more accurate classifiers. The correlations computed from the two complete sets of 225 solutions are shown in Figures 7 and 8. There are different patterns of correlations for the two problems. The first problem, has a larger number of correlations below the -0.3 threshold. The second problem has a few very strong negative correlations between the variables. This is the type of correlations that could serve as a clue for the identification of brain signatures associated to the P300 evoked potential.

We also evaluated the distribution of the optimal solutions. How similar are the optimal solutions between them? How similar are the solutions found by each algorithm or between algorithms? Figures 9 and 10 respectively show the matrices of similarities between the 225 solutions using the Euclidean and Correlation measures. Higher values in the colormap shown in the figures represent that solutions are more similar between them. Solutions have been ordered according to the order of the algorithms presented in Table 1. The first 25 solutions correspond to the $DE-1$ and the last 25 to $GODE-2$ algorithms. The most dissimilar set of solutions produced by an algorithm is that output by the $GA-2$ algorithm. In general, the solutions produced by the GAs are more dissimilar between them than to the other solutions. In Figure 10, some red stripes denote the existence of more similarities between some pairs of algorithms. The most important conclusion from this analysis is that other criteria, in addition to optimality, could be included in the

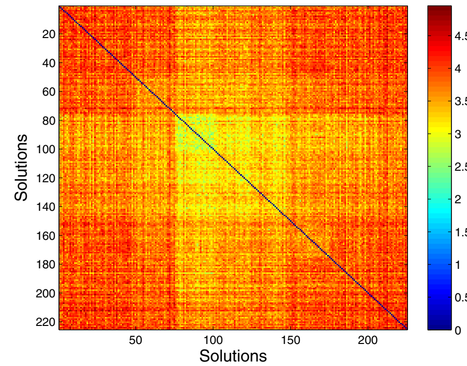

Figure 9: Similarity values between the genotypes of the best solutions found for individual 1. The similarity measure is based on the Euclidean distance.

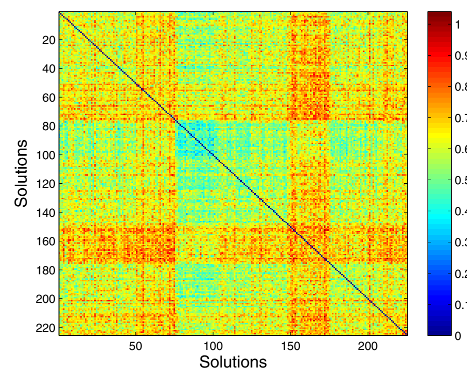

Figure 10: Similarity values between the genotypes of the best solutions found for individual 1. The similarity measure is based on the correlation.

search for solutions to the P300 classification problem. Diversity, or similarity between the selected optimal solutions could be used as a way to improve the k-best or k-worst ensembles.

6. CONCLUSIONS

In this paper we have proposed a framework for the solution of the P300 classification problem as a direct optimization problem. We have made a dissection of the factors that influence the complexity of the general problem and focused on the optimization of weights.

The k-best and k-worst ensembles have been proposed as feasible ways to combine the information contained in different solutions. The ensemble approach has shown to improve the results reached by single solutions. For the data sets considered, we have achieved average classifications accuracies over 0.8 using only a subsets of the 64 channels available. Considering the simplicity and generality of the approach proposed, these are satisfactory results.

Moreover, we have further evaluated different facets related with the direct optimization approach and illustrated the benefits of obtaining a set of possible solutions. These benefits include the extractions of common patterns shared by the solutions that can indicate regularities of the P300 classification problem for the given individual. Since the P300 paradigm is used for a broad spectrum of BCI applications [7, 11, 14], our results could be applied in these areas.

There are several ways in which our results can be extended. One possibility is to analyze the problem for a wider sets of channels. However, the inclusion of all the channels may provoke, on the one hand, the addition of redundant or noisy information and, on the other hand, make more difficult the optimization problem. Also, a preprocessing of the signals should be added to our algorithm. Finally, it would be possible to add more information to the fitness function. This information could be related with the robustness of the fitness values when the number of repetitions is decreased.

The successful application of machine learning techniques, able to generalize to unseen data, and in some cases are also able to deliver interpretable information about the brain processes involved in the experiments, has limited the application of direct optimization approaches to BCI problems. There are many opportunities in the application of optimization techniques in BCI research. We see the results presented in this paper as an initial step in this direction.

7. ACKNOWLEDGMENTS

This work was financed by the Spanish Ministry of Science (TIN2010-21289-C02-02, TIN2010-20900-C04-04, and Consolider Ingenio 2010 CSD2007-00018 projects), and supported by the Cajal Blue Brain Project. The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the Centro de Supercomputación y Visualización de Madrid (CeSViMa) and the Spanish Supercomputing Network.

8. REFERENCES

[1] B. Blankertz, K. Muller, D. Krusienski, G. Schalk, J. Wolpaw, A. Schlogl, G. Pfurtscheller, J. Millan, M. Schroder, and N. Birbaumer. The BCI competition III: Validating alternative approaches to actual BCI problems. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(2):153–159, 2006.

[2] E. Donchin, K. Spencer, and R. Wijesinghe. The mental prosthesis: assessing the speed of a P300-based brain-computer interface. *Rehabilitation Engineering, IEEE Transactions on*, 8(2):174–179, 2002.

[3] M. Fabiani, G. Gratton, D. Karis, and E. Donchin. Definition, identification, and reliability of measurement of the P300 component of the event-related brain potential. *Advances in psychophysiology*, 2(S 1):78, 1987.

[4] L. Farwell and E. Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 70(6):510–523, 1988.

[5] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

[6] D. Krusienski, E. Sellers, F. Cabestaing, S. Bayoudh, D. McFarland, T. Vaughan, and J. Wolpaw. A comparison of classification techniques for the P300 speller. *Journal of neural engineering*, 3:299, 2006.

[7] D. Krusienski, E. Sellers, D. McFarland, T. Vaughan, and J. Wolpaw. Toward enhanced P300 speller performance. *Journal of neuroscience methods*, 167(1):15–21, 2008.

[8] M. Lebedev and M. Nicolelis. Brain-machine interfaces: Past, present and future. *TRENDS in Neurosciences*, 29(9):536–546, 2006.

[9] C. Lin, L. Ko, M. Chang, J. Duann, J. Chen, T. Su, and T. Jung. Review of wireless and wearable electroencephalogram systems and brain-computer interfaces–A mini-review. *Gerontology*, 56(1):112–119, 2009.

[10] F. Lotte, M. Congedo, A. Lecuyer, and F. Lamarche. A review of classification algorithms for EEG-based brain–computer interfaces. *Journal of Neural Engineering*, 4:R1–R13, 2007.

[11] S. Mason, A. Bashashati, M. Fatourechi, K. Navarro, and G. Birch. A comprehensive survey of brain interface technology designs. *Annals of Biomedical Engineering*, 35(2):137–169, 2007.

[12] G. McLachlan and J. Wiley. *Discriminant analysis and statistical pattern recognition*. Wiley Online Library, 1992.

[13] A. Nijholt. BCI for games: A state of the art survey. pages 225–228. Springer, 2009.

[14] A. Rakotomamonjy and V. Guigue. BCI competition III: Dataset II-ensemble of SVMs for BCI P300 speller. *Biomedical Engineering, IEEE Transactions on*, 55(3):1147–1154, 2008.

[15] L. Rossini, D. Izzo, and L. Summerer. Brain-machine interfaces for space applications. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, volume 1, pages 520–523, 2009.

[16] R. Santana, C. Bielza, and P. Larrañaga. Regularized logistic regression and multi-objective variable selection for classifying MEG data. 2010. Submmitted for publication.

[17] R. Storn and K. Price. Differential evolution- a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report, International Computer Science Institute, 1995.

[18] H. R. Tizhoosh. Opposition-Based Learning: A New Scheme for Machine Intelligence. In *International Conference on Computational Intelligence for Modelling, Control and Automation, 2005*, volume 1, pages 695–701, 2005.

[19] J. Vesterstrom and R. Thomsen. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In *Proceedings of the IEEE Congress on Evolutionary Computation*, volume 2, pages 1980 – 1987 Vol.2, 2004.

[20] H. Wang, Z. Wu, S. Rahnamayan, and L. Kang. A Scalability Test for Accelerated DE Using Generalized Opposition-Based Learning. In *Ninth International Conference on Intelligent Systems Design and Applications, 2009. ISDA '09.*, pages 1090–1095, 2009.

[21] J. Wolpaw, N. Birbaumer, D. McFarland, G. Pfurtscheller, and T. Vaughan. Brain-computer interfaces for communication and control. *Clinical Neurophysiology*, 113(6):767–791, 2002.