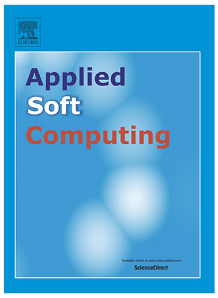
Journal Pre-proof



Brain–computer interface channel selection optimization using meta-heuristics and evolutionary algorithms

Víctor Martínez-Cagigal, Eduardo Santamaría-Vázquez, Roberto Hornero

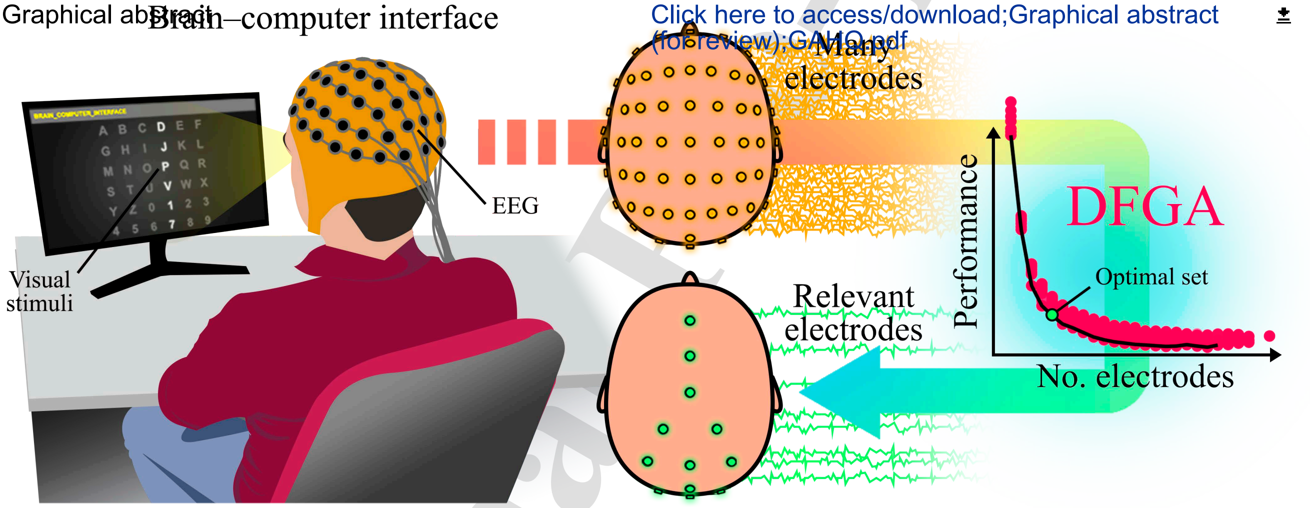
PII: S1568-4946(21)01029-2
DOI: https://doi.org/10.1016/j.asoc.2021.108176
Reference: ASOC 108176

To appear in: *Applied Soft Computing*

Received date : 10 October 2019
Revised date : 23 August 2021
Accepted date : 11 November 2021

Please cite this article as: V. Martínez-Cagigal, E. Santamaría-Vázquez and R. Hornero, Brain–computer interface channel selection optimization using meta-heuristics and evolutionary algorithms, *Applied Soft Computing* (2021), doi: https://doi.org/10.1016/j.asoc.2021.108176.

Graphical abstract



Brain–Computer Interface Channel Selection Optimization using Meta-heuristics and Evolutionary Algorithms

Víctor Martínez-Cagigal[a,b,*], Eduardo Santamaría-Vázquez[a,b], Roberto Hornero[a,b]

[a] *Biomedical Engineering Group, E.T.S.I. Telecomunicación, University of Valladolid, Paseo de Belén 15, 47011, Valladolid, Spain*
[b] *Biomedical Research Networking Center in Bioengineering, Biomaterials and Nanomedicine (CIBER-BBN), Spain*

Abstract

Many brain–computer interface (BCI) studies overlook the channel optimization due to its inherent complexity. However, a careful channel selection increases the performance and users' comfort while reducing the cost of the system. Evolutionary meta-heuristics, which have demonstrated their usefulness in solving complex problems, have not been fully exploited yet in this context. The purpose of the study is two-fold: (1) to propose a novel algorithm to find an optimal channel set for each user and compare it with other existing meta-heuristics; and (2) to establish guidelines for adapting these optimization strategies to this framework. A total of 3 single-objective (GA, BDE, BPSO) and 4 multi-objective (NSGA-II, BMOPSO, SPEA2, PEAIL) existing algorithms have been adapted and tested with 3 public databases: 'BCI competition III–dataset II', 'Center Speller' and 'RSVP Speller'. Dual-Front Sorting Algorithm (DFGA), a novel multi-objective discrete method especially designed to the BCI framework, is proposed as well. Results showed that all meta-heuristics outperformed the full set and the common 8-channel set for P300-based BCIs. DFGA showed a significant improvement of accuracy of 3.9% over the latter using also 8 channels;

*Corresponding author
Email addresses: victor.martinez@gib.tel.uva.es (Víctor Martínez-Cagigal), eduardo.santamaria@gib.tel.uva.es (Eduardo Santamaría-Vázquez), robhor@tel.uva.es (Roberto Hornero)

and obtained similar accuracies using a mean of 4.66 channels. A topographic analysis also reinforced the need to customize a channel set for each user. Thus, the proposed method computes an optimal set of solutions with different number of channels, allowing the user to select the most appropriate distribution for the next BCI sessions.

Keywords: Brain–computer interface (BCI), channel selection, multi-objective optimization, evolutionary algorithms, P300 event-related potentials.

1. Introduction

Brain–Computer Interfaces (BCIs) are communication systems that allow users to control devices and applications using their own brain signals. These systems have been successfully applied in order to improve the quality of life of people with motor disabilities who suffer from a disease that impairs the neural pathways that control muscles or even the muscles themselves [1]. Electroencephalogram (EEG) is commonly used to monitor the brain activity due to its portability, non-invasiveness and low cost. Therefore, electrical potentials are recorded by placing electrodes on the user's scalp [1].

Since decoding users' intentions from the EEG is not straightforward, BCIs rely on control signals to handle the control of the system. In particular, the P300 evoked potentials, which are positive peaks produced in response to infrequent and significant stimuli approximately 300 ms after their onset, are the key aspect of the most well-known BCI-based spelling system [1]. The 'P300 Speller' generates these signals through the odd-ball paradigm in order to spell certain words or commands. The application displays a matrix containing characters or symbols, whose rows and columns are randomly flashing. Users, who have to focus on a desired command, will generate a P300 potential whenever the row or the column that contains the command is highlighted. Hence, the selected command is determined by computing the intersection between the row and the column that produced the potential [2].

Due to the low signal-to-noise ratio and high inter-session variability of these event-related potentials, several repetitions of the same stimulus are required to detect a reliable response. Without a proper processing stage, these high dimensional data can produce over-fitting, resulting in poor performance [3, 4]. The curse of dimensionality can be addressed by means of feature selection and extraction methods [4, 5], regularized classifiers [6] or channel selection procedures [3, 7]. Among them, only channel selection methods are able to reduce the cost of the system, reduce power consumption on EEG caps and increase user comfort [3]. Nevertheless, the selection of the most relevant sensors is not trivial as there are 2^N subset combinations for an N-channel cap, making the exhaustive search intractable in practice [3]. For this reason, most P300-based studies overlook the optimization of the most relevant subset of channels and take a predefined 8-channel set as a general rule of thumb [8]. Notwithstanding its usefulness as a quick solution, an optimization for each user is beneficial owing to the intrinsic inter-subject variability of the BCI systems.

Although there are many feature selection methods that could be applied to this problem, such as step-wise regression [9], fast correlation based filters [10], elastic neural networks [11], or explainable deep learning [12], meta-heuristics have demonstrated high performances solving complex optimization problems [13]. Heuristics refer to problem-specific strategies that iteratively improve a candidate solution, whereas meta-heuristics generalize these strategies to problem-independent frameworks [13, 14]. Swarm intelligence techniques and evolutionary algorithms, families of population-based meta-heuristics, have been previously applied in EEG signals to solve optimization problems [4, 7, 15–19, 19–30]. Despite their popularity, the contribution of meta-heuristics to P300-based BCIs is still scarce. Most of these previous studies are related to motor imagery (MI) BCIs [15–23] or biometric-oriented person identification systems [24, 25], whose signal processing stage is completely different (e.g., neural sources, control signals, paradigms, spatial filtering and feature extraction) and thus, results cannot be generalized to P300-based BCIs. Regarding the P300-based studies, most of them have used single-objective algorithms that optimized the final classification accuracy of the system [4, 26–28, 30]. However, we believe that a channel selection procedure should follow a two-fold objective: (i) to minimize the number of selected channels, and (ii) to maximize the system's performance. Some recent studies used a weighted aggregation approach to combine both objectives into a single one, but the simultaneous optimization was not explored [7, 22, 31].

Traditional multi-objective approaches, which optimize both objectives at the same time, have been explored in MI-based BCIs, such as multi-objective particle swarm optimization (MOPSO) [16–18] or non-sorting genetic algorithm II (NSGA-II) [20, 23]. By contrast, multi-objective algorithms applied to P300-based BCIs are more limited. Kee et al. [19] compared the performance between several single-objective genetic algorithms (GA) and NSGA-II with 2 subjects, whereas Chaurasiya et al. [29] employed a multi-objective binary differential-evolution algorithm with 9 subjects, reaching several subsets of channels that assured suitable classification performances. Nevertheless, the number of subjects was limited, and both databases were recorded using the row-col paradigm (RCP). Nowadays, P300-based BCIs offer a wide range of stimulation paradigms that elicit different event-related responses and thus, the generalization of those results to other setups is unclear. Furthermore, despite their scarce application in P300-based BCI studies, swarm intelligence and evolutionary computation are growing research fields that integrate a large amount of different algorithms that could be adapted to the channel selection problem. In fact, the vast majority of them have yet to be applied to P300-based BCIs. To the best of our knowledge, there are no studies that compare their efficacy in selecting the most appropriate subset of channels or even establishing the key aspects for their adaptation to BCI systems, which is not trivial. Furthermore, none of the previous studies tested any meta-heuristic with paradigms other than RCP, restricting their generalization. Lastly, it is noteworthy that there is also no study aimed at designing any multi-objective algorithm customized for the P300-based BCI channel selection problem.

The objective of this study is two-fold: (1) to propose a novel multi-objective method to find an optimal channel set especially suited for P300-based BCIs and compare its usefulness with 7 additional meta-heuristics; and (2) to establish guidelines for adapting these optimization strategies to the channel selection problem. Although there are many meta-heuristics that could be adapted to this problem, only those that have previously applied in BCIs, that have direct or explicit contribution to our proposed meta-heuristic or that have been recently proposed were included in this comparison: GA, BDE and BPSO as single-objective; and NSGA-II, SPEA2, BMOPSO and PEAIL as multi-objective. We have also tried to maintain diversity in the way they deal to the updating of the population for each iteration. To sum up, the main contributions of this study are the following: proposal of a novel multi-objective algorithm especially designed for this problem, comparison of of 7 meta-heuristics to the P300-based BCI channel selection problem, enumeration of a detailed set of guidelines to adapt any meta-heuristic for the channel selection, and evaluation with three databases that employ different P300-based paradigms.

2. Subjects

In order to improve the generalization of the results, the algorithms have been tested with three public P300-based BCI databases that were recorded using different stimulation paradigms: row-col paradigm (RCP), center speller (CS) and rapid serial visual presentation (RSVP). Examples of the stimulation sequences for these paradigms are depicted in the figure 1.

2.1. BCI competition III: dataset II

The 'BCI competition III: dataset II' [32] was recorded from 2 different healthy subjects (i.e., A and B) that were asked to spell words in 5 RCP sessions. Signals were recorded using a 64-channel EEG cap with a sampling frequency of 240 Hz and band-pass filtered from 0.1 Hz to 60 Hz. Training and testing sets were composed of 85 and 100 trials, respectively [32]. RCP is the most common P300-based spelling paradigm, which consists of displaying a matrix that contains characters or symbols. Users have to stare at the target command

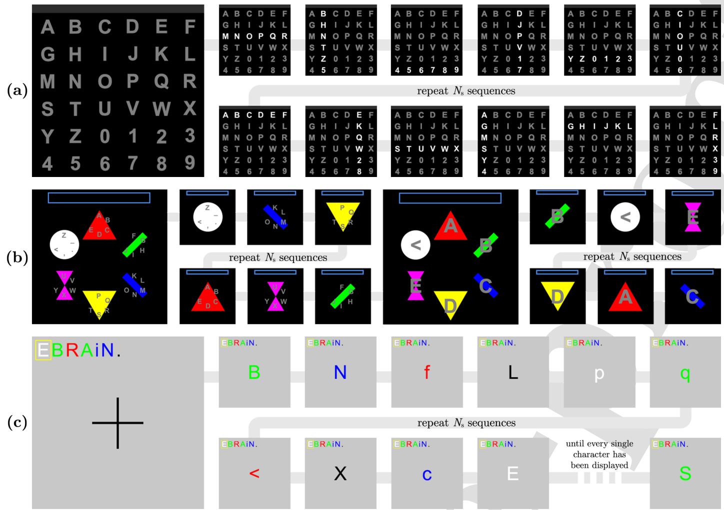


Figure 1: Examples of stimuli intensification sequences for the paradigms: (a) row-col paradigm, (b) center speller, (c) rapid serial visual presentation.

while the matrix's rows and columns are randomly flashing. Whenever the row or column that contains the target is flickered, a P300 potential is generated. Hence, the desired command can be identified by computing the intersection between the row and the column that produced these P300 responses [2]. In this dataset, there are 12 different classes (i.e., rows and columns), and 15 sequences (i.e., repetitions) were used. Therefore, a trial is composed by 180 observations [32].

2.2. Center Speller database

The 'Center Speller (008-2015)' database [33] was recorded from 13 healthy subjects (i.e., C01-C13) that were asked to perform spelling tasks using the CS paradigm. Signals were recorded using a 63-channel EEG cap with a sampling frequency of 250 Hz and band-pass filtered from 0.016 Hz to 250 Hz. Training data was composed of 17 trials, whereas testing data varied between 32-49 trials, depending on the subject [33]. CS was originally designed to avoid eye movements. The paradigm displays groups of commands in the center of the screen, overlaid with colored geometric shapes. The groups are randomly flickered until the user selects one of them. Then, the commands that were included inside the selected group are displayed in the same way, allowing the user to select the final command [33]. In practice, there are 12 different classes (6 groups in 2 levels), and 10 sequences were used. A trial is composed by 120 observations [33].

2.3. RSVP Speller database

The 'RSVP Speller (010-2015)' database [34] was recorded from 12 healthy subjects (i.e., R01-R12) that were asked to perform spelling tasks using the RSVP paradigm. Signals were recorded using a 63-channel EEG cap with a sampling frequency of 1000 Hz, and then down-sampled to 200 Hz [34]. However, since the fifth subject only used 61 channels, electrodes P8 and O2 were excluded from the database for the sake of homogeneity. Training data was composed of 24 trials, whereas testing data (copy and free spelling) varied between 37-50 trials, depending on the subject [34]. RSVP was also developed to exploit the foveal visual field and avoid eye movements by depicting symbols in the center of the screen in a serial manner. The database includes a vocabulary of 30 characters (26 letters and 4 symbols). In order to favor the identification of the shapes, half of the letters were uppercase and the other half lowercase, using 5 different colors. Therefore, there are 30 classes, and 10 sequences were used, resulting in 300 observations per trial [34].

3. Methods

3.1. Pre-processing and feature extraction

Before applying any optimization procedure, relevant features of the EEG signals should be extracted for each epoch (i.e., stimulus) and channel. In fact, pre-processing, as well as feature extraction and selection procedures influence the final accuracy in a high extent. Due to the purpose of the study, signal processing stages were composed of a standardized framework, intended to isolate the channel selection procedure. We did not apply any further pre-processing step besides the aforementioned band-pass filtering embedded in each database [32–34]. Epochs were extracted using windows in the range $[0, 800]$ ms from the stimuli onsets, and normalized via z-score over a $[-200, 0]$ ms baseline. As stated in BCI literature, this range is large enough to capture relevant event-related potentials, including the P300 wave [1]. These epochs were then decimated to 25 Hz, keeping a total of 20 features per stimulus and channel. It is noteworthy that the decimation process encompasses a low-pass filtering (to avoid aliasing), followed by a down-sampling procedure [7, 35]. Hereafter, epochs from different databases and sampling rates have the same number of features. Note that, from the point of view of a subsequent classifier, epochs are input observations.

3.2. Defining the optimization problem

The goal of an optimization algorithm is to provide a suitable solution that satisfies the problem constraints and optimizes (either maximizing or minimizing) one or more objective functions to the greatest extent [14]. Since we are considering an N-channel selection problem, a possible solution may be defined as $\boldsymbol{x} = [x_1, x_2, \ldots, x_N], x_i \in \{0, 1\}$, where 1 and 0 represent the selection and rejection of a channel i, respectively. Hence, this combinatorial problem is constrained to a discrete N-dimensional space, whose solutions are restricted to binary positions. When a solution $\boldsymbol{x}$ is evaluated, features associated with the channels that satisfy $x_i = 1$ are concatenated as an input feature vector.

In a BCI channel selection problem, two main objectives must be pursued: (i) maximize system performance, and (ii) minimize the number of channels. Even though the modeling of the latter is straightforward (see equation 1), the system performance can be estimated following several approaches. The most intuitive solution is to use the output training accuracy of the classifier using a certain solution $\boldsymbol{x}$ [19, 26, 29]. However, due to the limited number of trials, this method usually provides a low-resolution score [36]. The resolution can be improved by using stimuli-based, rather than character trial-based. Previous studies used approaches derived from the confusion matrix of the stimuli classification [4, 27, 28]. Nevertheless, the area under ROC curve (AUC) is recommended because it is able to successfully estimate the discriminative ability of a binary classifier using only training data [3, 36]. Therefore, the objectives are modeled as follows:

$$\min F(\boldsymbol{x}) = \begin{cases} f_1(\boldsymbol{x}) = 1 - \overline{AUC}(\boldsymbol{x}) \\ f_2(\boldsymbol{x}) = \sum_{n=1}^{N} x_n \end{cases}, \tag{1}$$

where $f_1(\boldsymbol{x})$ belongs to the first objective (i.e., minimize the system error) and $f_2(\boldsymbol{x})$ to the second objective (i.e., minimize the number of channels). In this study, $\overline{AUC}$ has been derived from a 5-fold cross-validated linear discriminant analysis (LDA) that is applied to the solution $\boldsymbol{x}$ using the training dataset [7, 35, 37]. That is, the features whose channels satisfy $x_i = 0$ are removed from the observations matrix, which is the input of the LDA classifier. Training set is then divided into 5 subsets and a cross-validation procedure is applied (i.e., 4 subsets are used for training and the remaining one for testing), returning a total of 5 AUCs. Finally, $\overline{AUC}$ is computed as the average of all of them. LDA was used as classifier due to its well-known excellent performances in P300-based BCIs and the lack of hyperparameters to optimize [5–7, 19, 20, 29, 33, 38].

3.3. Single-objective meta-heuristics

Meta-heuristics produce acceptable solutions to complex problems in a reasonable computation time [13]. In particular, single-objective meta-heuristics iteratively produce these solutions following a certain objective. However, a BCI channel selection problem should have a two-fold purpose. Thus, the multi-objective problem stated in equation (1) is then combined into a single-objective one [39]:

$$\min F(\boldsymbol{x}) = \omega_1 f_1(\boldsymbol{x}) + \omega_2 \left(\frac{f_2(\boldsymbol{x}) - 1}{N - 1} \right)^3, \tag{2}$$

where $\omega_1 + \omega_2 = 1$, and ω_1 and ω_2 are constants that weigh the importance of each objective. Since we consider that reaching suitable accuracies is more important than drastically reducing the number of required channels, coefficients have been heuristically set to $\omega_1 = 0.7$ and $\omega_2 = 0.3$ [7, 31, 35]. In addition, after mapping the $f_2(\boldsymbol{x})$ from $[1, N] \to [0, 1]$, its output is raised to the third power to empathize the search of lightweight solutions. Note that the polynomial function punishes the search for solutions with a high number of channels more than a simple linear function. This function was heuristically chosen after a preliminary testing [7, 31, 35]. The three single-objective meta-heuristics that have been adapted to BCI framework are described below.

3.3.1. Genetic Algorithm

One of the most well-known meta-heuristics is the genetic algorithm (GA), originally developed by Holland [40]. GAs have been modified to improve their ability to find the global optimum of complex optimization problems in many ways. In short, GAs apply the Darwinian principle of survival of the fittest individuals in a population using recombination, selection and mutation operators [13, 14]. In this study, a GA with elitism, binary tournament selection, single-point crossover and bit string mutation has been employed [13, 14].

3.3.2. Binary Differential Evolution

The differential evolution (DE) algorithm, originally developed by Storn and Price [41] for continuous functions, has some similarities to GAs in terms of its structure, composed by mutation, crossover and selection operator. However, instead of making random mutation and crossover schemes, DE combines the information of three randomly chosen individuals. Binary DE (BDE) applies a discretization of the mutation formula in order to adapt it to binary problems [42]. The mutation of the i-th channel of an individual $\boldsymbol{x}$ is performed as follows:

$$x'_i = \begin{cases} u_i, & \text{if rand} \leq p_c \text{ or } i = r \\ x_i, & \text{otherwise} \end{cases}, \tag{3}$$

where rand $\sim \mathcal{U}(0,1)$, r is a random integer between $[1, N]$, p_c is the crossover rate, and u_i is the mutated channel, computed as:

$$u_i = \begin{cases} 1, & \text{if rand} \leq (1 + e^{\frac{-2b(v_i + F\cdot(y_i - z_i) - 1/2)}{1+2F}})^{-1} \\ 0, & \text{otherwise} \end{cases}, \tag{4}$$

where rand $\sim \mathcal{U}(0,1)$; $\boldsymbol{v}$, $\boldsymbol{y}$ and $\boldsymbol{z}$ are randomly selected individuals of the current population; F is the weighting factor; and $b > 0$ is the bandwidth factor.

3.3.3. Binary Particle Swarm Optimization

Kennedy and Eberhart [43] developed the Particle Swarm Optimization (PSO) algorithm, a nature-inspired meta-heuristic based on the social schooling and flocking behavior of fishes and birds. The optimization relies on adjusting the trajectories and positions of a set of particles (i.e., solutions) that "fly" over the search space, whose movement have both deterministic and stochastic components [13, 14, 43]. In this study, the standard constraint of Clerc and Kennedy [44] is used, leading to:

$$\boldsymbol{v}' = \chi[\boldsymbol{v} + \epsilon_1 C_1(\boldsymbol{l} - \boldsymbol{x}) + \epsilon_2 C_2(\boldsymbol{g} - \boldsymbol{x})], \tag{5}$$

$$\chi = \frac{2}{\phi - 2 + \sqrt{\phi^2 - 4\phi}}, \text{ with } \phi = C_1 + C_2; \tag{6}$$

where $\boldsymbol{v}'$ is the updated velocity of a particle $\boldsymbol{x}$; $\boldsymbol{v}$ is the last velocity; $\epsilon_1, \epsilon_2 \sim \mathcal{U}(0,1)$; χ is the constraint multiplier; C_1 and C_2 are the personal and global confidence constants, respectively; $\boldsymbol{l}$ is the best position found by the particle $\boldsymbol{x}$; and $\boldsymbol{g}$ is the best global position found so far. It is worthy to note that the standard constraint requires that $\phi > 4$ [44, 45]. Since the velocities are continuous, the algorithm should be adapted to binary spaces. Binary PSO (BSPO) is usually achieved using a position transformation via transfer functions [46, 47]. In this study, the adaptation has been performed following the expression:

$$x'_i = \begin{cases} \neg x_i, & \text{if rand} < T(v'_i) \\ x_i, & \text{if rand} \geq T(v'_i) \end{cases}, \tag{7}$$

where rand $\sim \mathcal{U}(0,1)$, and $T(t) = |t/\sqrt{1+t^2}|$ is a v-shaped transfer function [47].

3.4. Multi-objective meta-heuristics

In contrast to the single-objective strategies, multi-objective meta-heuristics involve the simultaneous optimization of two or three objectives [48]. Since these objectives are usually conflicting among themselves, the concept of dominance is introduced for determining the quality of each solution [49]. It is said that a solution $\boldsymbol{y}$ dominates a solution $\boldsymbol{z}$ (i.e., $\boldsymbol{y} \succ \boldsymbol{z}$) if $\forall i : f_i(\boldsymbol{y}) \leq f_i(\boldsymbol{z})$ and $\exists j : f_j(\boldsymbol{y}) < f_j(\boldsymbol{z})$. The Pareto-front, a curve that contains optimal solutions (i.e., those that are not dominated by any other solutions), is estimated by the multi-objective algorithms and depicts the trade-off among the objectives [49]. Regarding the BCI channel selection problem, the Pareto-front returns a set of solutions that have different number of channels, allowing the user to select one of them.

3.4.1. Non-Sorting Genetic Algorithm II

The most popular approach for extending GAs to multi-objective optimization problems is the Non-Sorting Genetic Algorithm II (NSGA-II), proposed by Deb et al. [48]. Crossover and mutation operators are the same as GAs, whereas the selection operator is more complex. Firstly, in order to estimate the quality of each chromosome, the algorithm establishes a hierarchy of Pareto-fronts according to its dominance. The first Pareto-front (i.e., rank = 1) is composed by the non-dominated chromosomes of the current population. Then, the second Pareto-front (i.e., rank = 2) is computed in the same way, but ignoring the chromosomes of the first front. This process is repeated sequentially until there are no chromosomes left [48]. However, the selection of a parent population is not only based on the rank of the chromosomes, but also on their crowding distances. These metrics are included to spread the solutions along the Pareto-front and avoid getting trapped in local minima. The crowding distance of a chromosome is computed as the average distance between its two adjacent solutions with the same rank. Boundary solutions are assigned an infinite distance value. Considering two chromosomes, the solution with lower rank is preferred. Whether both have the same rank, the less crowded solution is preferred (i.e., higher distance value). The parent population is sequentially filled with the firsts Pareto-fronts until the number of included solutions is greater or equal than $m/2$. Then, parent solutions are truncated based on the crowding distances until the number of solutions is exactly $m/2$. Further information can be found in Deb et al. [48].

3.4.2. Binary Multi-Objective PSO

Due to its usefulness to solve complex optimization problems, many authors have tried to adapt the PSO algorithm to multi-objective environments [39]. Here, a Binary Multi-Objective PSO (BMOPSO) approach is applied. Since the conflicting objectives do not allow the establishing of an optimal global solution $\boldsymbol{g}$, the major adaptation must reside in the way to select the leader of each particle. In this study, a repository approach is employed. Non-dominated solutions are stored in an external repository with "unlimited" size. Note that its maximum size would be the maximum number of channels (i.e., the resolution of the BCI problem). A particle's leader is randomly selected from the repository, and it is attached to the particle until the leader is no longer part of the repository. In that case, the leader is substituted by another randomly selected one. In addition, a three-fold bit string mutation is also used, which consists on dividing the swarm in three parts and apply: (1) no mutation; (2) uniform mutation with probability p_m; (3) non-uniform mutation with probability $p_n = (1 - \text{gen}/\text{ngen})^{5N}$ [50].

3.4.3. Strength Pareto Evolutionary Algorithm 2

Zitzler et al. [51] proposed the Strength Pareto Evolutionary Algorithm 2 (SPEA2), a multi-objective algorithm that integrates the concepts of dominance and crowding density in a single metric: the strength. The strength S_i is computed as the number of solutions that the i-th particle dominates. Then, the unified fitness is calculated as follows:

$$F_i = R_i + \frac{1}{\sigma_i^k + 2}, \tag{8}$$

where R_i is the sum of the strengths of the particles that dominates i, and σ_i^k is the distance sought of the particle (i.e., distance to the k-nearest neighbor), where $k = \lfloor\sqrt{m}\rceil$. Note that non-dominated individuals would have $R = 0$ and thus, $F < 1$. SPEA2 also uses a repository with fixed size that is updated following an environmental selection procedure. Solutions are sorted according to their F values, and the repository is filled with them. If the number of solutions of the repository is higher than the maximum size N_r, a truncation process is applied. Then, the algorithm removes solutions from the repository according to their σ^k (i.e., high σ_k values are preferred), in order to preserve Pareto-front spreading [51].

3.4.4. Pareto Evolutionary Algorithm based on Incremental Learning

Recently, Rong-Juan et al. [52] proposed a discrete multi-objective algorithm that introduces the concept of incremental learning to update solutions by exploring probability distributions of promising search regions. The algorithm, known as Pareto Evolutionary Algorithm based on Incremental Learning (PEAIL), also uses non-dominated sorting to keep track of hierarchical Pareto fronts, as NSGA-II does [48]. The incremental learning stage selects an excellent individual, then estimates a probability model and predicts a new children population using that information:

$$\boldsymbol{x}_i \leftarrow (\boldsymbol{x}_i + \boldsymbol{x}_e \cdot L)/(L+1), \tag{9}$$

where $\boldsymbol{x}_i$ is the solution being updated, $\boldsymbol{x}_e$ is a randomly selected solution from the first Pareto front, and L is the learning rate parameter. Check [52] for further information.

3.5. Our proposal: Dual-Front Genetic Algorithm

Even though there is a great variety of meta-heuristics from single to multi-objective algorithms, all of them should be adapted to the channel selection problem. The BCI framework forces the algorithms to work with binary solutions, involving the use of transfer functions in some cases. These functions convert a solution alteration into a probability of change, increasing the stochasticity of the algorithm. Moreover, the conversion can be addressed as a multi-valued function of the type $f : \mathbb{R} \rightarrow \{0,1\}$, which means that there are infinite input values that produce exactly the same output, hindering the local exploitation of new solutions. By extension, there is no point in using operators based on continuous distances. Since $f_2(\boldsymbol{x})$ already restricts the size of multi-objective repositories to N, limitation strategies (e.g., crowding, distance sought) also entail an unnecessary computational cost. In order to overcome these restraints, a novel multi-objective algorithm is proposed: the Dual-Front Genetic Algorithm (DFGA). DFGA is specially designed to the BCI framework by means of five key aspects: (i) deterministic initialization, (ii) dual-front sorting, (iii) genetic operators, (iv) synthetic solutions, and (v) elitism. A detailed flowchart is depicted in the figure 2(a), while the pseudo-code and a complexity analysis are included in the supplementary material.

Deterministic initialization. Heuristics generally initialize the population by generating random solutions. However, the use of deterministic initialization can reduce the inter-run variability due to stochastic effects and a large amount of computation time. Although deterministic algorithms are unlikely to provide a global optimum, DFGA considers their outputs as intermediate solutions. Regardless of their qualities, we hypothesize that these solutions are equivalent to those that will be eventually reached after several generations of a randomly-initialized algorithm. In this study, backward elimination (BE) is used to initialize the repository. The algorithm begins with the full set of channels and sequentially removes the most irrelevant one [9]. The rejected channel in each step is the one that returns the minimum $f_1(\boldsymbol{x})$ value if removed from the model $\boldsymbol{x}$ (i.e., its inclusion does not contribute to improve the system's performance). The algorithm continues removing channels until the set is empty. Note that this operation will fill the repository $\mathcal{R}$ up with N solutions.

Dual-front sorting. Due to the deterministic initialization, the repository should have a well-defined curve from the very beginning of the algorithm.

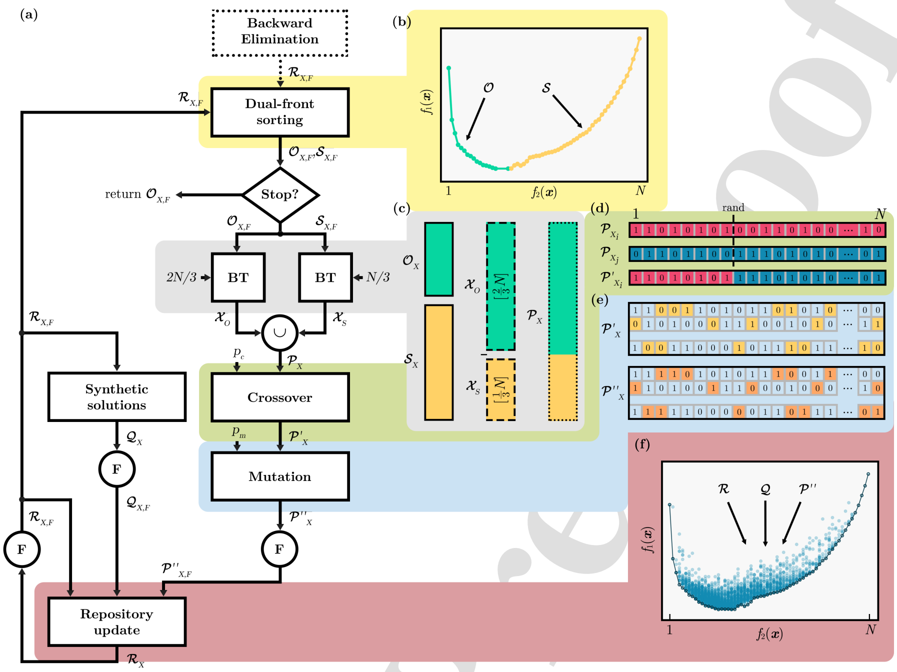


Figure 2: Summary of DFGA with visual aids to clarify operations. (a) Flowchart of the algorithm. Note that the backward elimination step is only performed as an initialization. BT: binary tournament, F: fitness evaluation. (b) Dual-front sorting. (c) Parent selection. (d) Single-point crossover. (e) Bit-string mutation. (f) Elitist repository updating.

This aspect leads to a Pareto-front that is supposed to include solutions with few number of channels. Traditionally, only the Pareto-optimal solutions are considered in the selection stage. Despite their convenience over dominated solutions, considering only the Pareto-front would lead to a local exploitation of solutions with few channels. However, because of the intrinsic fixed size of the repository in BCI problems (i.e., limited to N), the exploitation of solutions with a greater number of channels is no longer an issue, instead it may favor the spreading of the Pareto-front and the global search of DFGA. According to this rationale, DFGA subdivides the repository into two sets: $\mathcal{O}$ (i.e., optimal set), which includes the non-dominated solutions; and $\mathcal{S}$ (i.e., sub-optimal set), which includes the dominated solutions. Dual-front sorting operation is shown in figure 2(b). Then, binary tournament selection is applied in both sets, selecting $^{2N}/_3$ solutions from $\mathcal{O}$, and $^{N}/_3$ solutions from $\mathcal{S}$. Note that a solution may be selected more than once in the new population. Finally, these solutions are combined in the population to suffer recombination (i.e., crossover) and mutation, as shown in the figure 2(c).

Genetic operators. Owing to the binary nature of the search space, we consider that traditional genetic operators are the most convenient approach for generating new solutions from a parent population. First, for each solution $\boldsymbol{x}_i$, single-point crossover is applied with probability p_c. That is, $\boldsymbol{x}_i$ and another randomly picked solution $\boldsymbol{x}_j$ $(i \neq j)$ are combined into $\boldsymbol{x}'_i \leftarrow \boldsymbol{x}_i[1:u] \cup \boldsymbol{x}_j[u+1:N]$, where $u \sim \text{rand} \in [1, N]$. For each solution, bit-string mutation is also computed with probability p_m. In other words, if the n-th bit of a solution $\boldsymbol{x}'_i$ has to be mutated, its value is flipped (i.e., $\boldsymbol{x}''_i[n] \leftarrow \neg \boldsymbol{x}'_i[n]$). The procedure is illustrated in the figure 2(d–e).

Synthetic solutions. When the values of p_c or p_m are too high, the mutated population tends to exploit the middle part of the repository. In other words, solutions with few channels tend to add more channels, whereas crowded solutions tend to decrease their number of channels. In order to maintain a similar exploitation across the entire repository spectrum, synthetic solutions are generated apart from the mutated population. However, a random generation of solutions across this spectrum will unnecessarily increase the number of evaluations, slowing down the algorithm. DFGA generates synthetic solutions trying maintain the most relevant channels of the current repository. The rank of the i-th channel is defined as the number of times that the channel i is present in the repository (i.e., $r_i = |i \in \mathcal{R}|$). DFGA iteratively creates solutions that have from 1 to $N-1$ channels by means of a roulette wheel selection (i.e., fitness proportionate selection) based on the rank values. It is worthy to mention that DFGA generates a total of $N-1$ solutions, since the N-th solution that contains all the channels is already part of the repository.

Elitism. In each generation, the repository is updated following an elitist approach. As depicted in the figure 2(f), for each unique value of $f_2(\boldsymbol{x})$ (i.e.,

Table 1: Method-specific hyperparameters.

Prm.	**Value**	**Description**	**Algorithm**
m	20	No. individuals	All
p_m	$1/N$	[a]Mutation rate	GA, NSGA-II, BMOPSO, SPEA2, PEAIL, DFGA
p_c	0.90	[a]Crossover rate	GA, NSGA-II, SPEA2, PEAIL DFGA
F	0.80	[b]Weighting factor	BDE
b	N	[b]Bandwidth factor	BDE
p_{de}	0.20	[b]BDE crossover rate	BDE
C_1	2.05	[c]Personal confidence	BPSO, BMOPSO
C_2	2.05	[c]Global confidence	BPSO, BMOPSO
V_{max}	1.00	[c]Maximum velocity	BPSO, BMOPSO
L	0.06	[d] Learning rate	PEAIL

[a]Deb et al. [48], [b]Wang et al. [42], [c]Clerc and Kennedy [44], [c] Rong-Juan et al. [52].

for each number of channels), the repository solution that minimizes $f_1(\boldsymbol{x})$ is selected. Note that this operation is applied in the repository, which includes both non-dominated and dominated solutions, creating a balance between local and global exploitation.

4. Results

Hyperparameters, detailed in table 1, were set following the recommendations of the literature [42, 44, 48, 52]. In order to assure a fair comparison among the algorithms, the number of generations varied in function of the amount of evaluations that were performed in a single iteration, while the number of individuals of every single meta-heuristic was fixed to $m = 20$ [4, 7, 25, 30]. Table 2 details the computational cost, including the number of evaluations per generation and the number of generations for each method. In total, 4000 evaluations were performed. Furthermore, all the algorithms were computed 20 times in order to avoid local minima. The experiments were executed in an Intel Core i7-7700 CPU @ 3.60 GHz, 32GB RAM, Windows 10 Pro, using MATLAB® 2018b.

A convergence analysis for single-objective meta-heuristics is depicted in the figure 3. These averaged convergence curves show the evolution of the

Table 2: Approximate computational costs of single and multi-objective meta-heuristics.

	Mtd.	No. eval.	Eval. time	No. gen.
Single	GA	20 eval./gen.	785 ms/eval.	200 gen.
	BDE	20 eval./gen.	810 ms/eval.	200 gen.
	BPSO	20 eval./gen.	858 ms/eval.	200 gen.
Multi	NSGA-II	40 eval./gen.	331 ms/eval.	100 gen.
	SPEA2	20 eval./gen.	835 ms/eval.	200 gen.
	BMOPSO	20 eval./gen.	852 ms/eval.	200 gen.
	PEAIL	40 eval./gen.	415 ms/eval.	100 gen.
	DFGA	123 eval./gen.	591 ms/eval.	32 gen.

Mtd.: method, gen.: generation, eval.: evaluation.

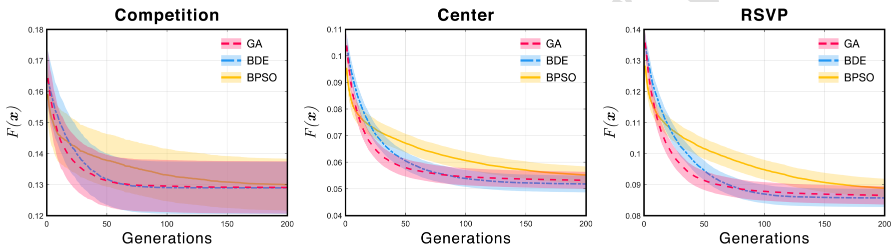


Figure 3: Averaged convergence curves of single-objective meta-heuristics (GA, BDE and BPSO) for each database in function of the $F(\boldsymbol{x})$ aggregated function. Mean values are displayed with solid lines, whereas the 95% confidence interval of the subjects' repetitions is indicated by the shaded area.

aggregated objective function $F(\boldsymbol{x})$ across the generations. Thus, they estimate the ability of each method to find an optimal solution in the training phase. The detailed convergence curves for each subject can be found in the supplementary material. Concerning the multi-objective meta-heuristics, the evolution of the computed Pareto-fronts over the generations of the algorithms is depicted in the figure 4, also in training phase.

Ranks of selected channels for both single and multi-objective meta-heuristics are displayed in the figure 5, including the common Krusienski's 8-channel set. The rank of a channel is defined as the normalized number of times that the channel was selected in the algorithm repetitions. For multi-objective algorithms, only the ranks of channels that belongs to the repository are included. Scalp distributions of the averaged rank values over the meta-heuristics are depicted for each subject as well.

In order to evaluate the actual performance of the single-objective algorithms

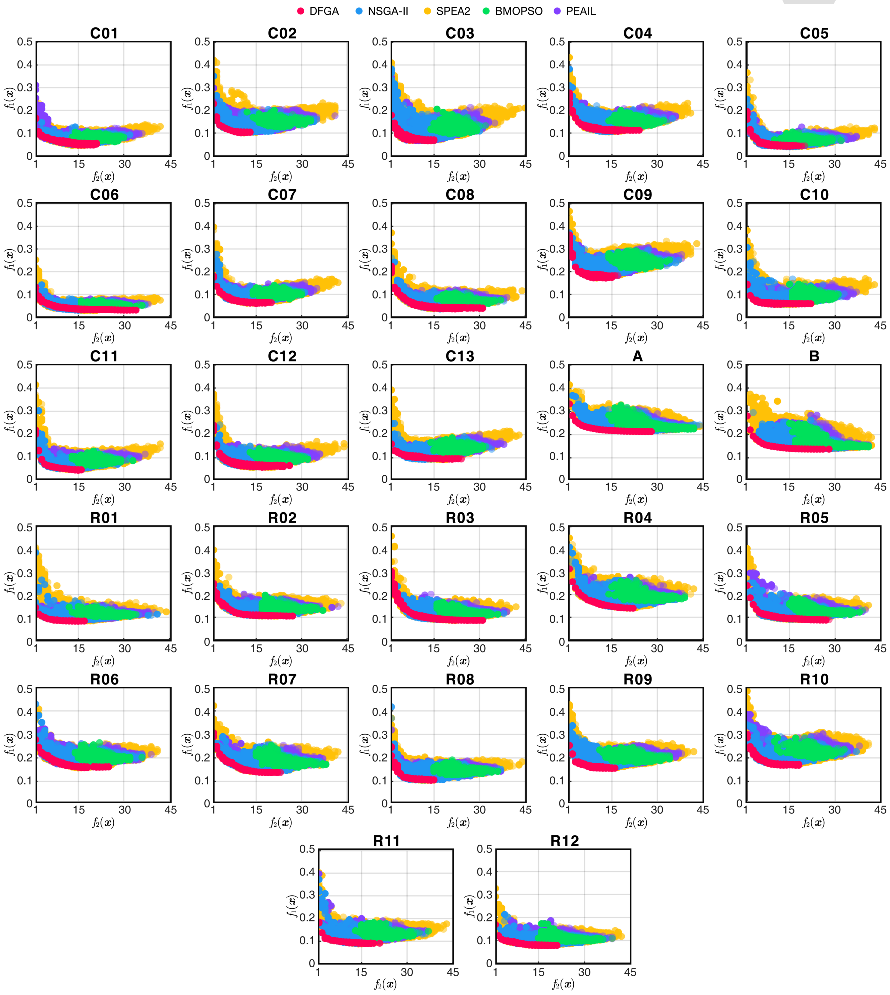


Figure 4: Evolution of Pareto-optimal solutions of the multi-objective meta-heuristics for each subject across all the repetitions: DFGA (red), NSGA-II (blue), SPEA2 (yellow), BMOPSO (green) and PEAIL (purple).

using testing datasets, it is required to select a single solution among the repetitions. Therefore, the solution that reached the minimal $F(\boldsymbol{x})$ value was selected for each single-objective method. Table 3 summarizes the averaged testing accuracies and number of channels of the selected solutions for each subject, in function of the employed method, using the maximum number of sequences available in each database. Regarding the multi-objective algorithms, the final

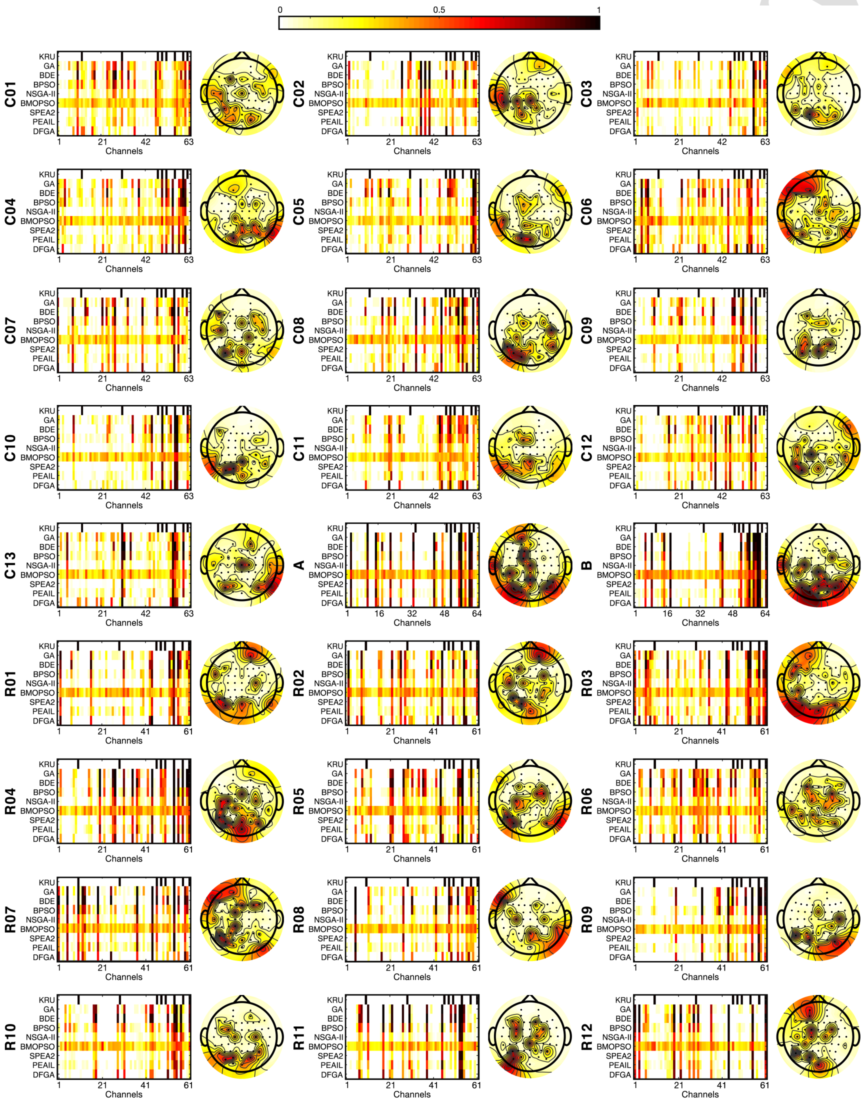


Figure 5: Channel ranks of the selected and the Pareto-optimal solutions for single-objective (GA, BDE, BPSO) and multi-objective (NSGA-II, BMOPSO, SPEA2, PEAIL, DFGA) metaheuristics, respectively. Krusienski's 8-channel set (KRU) is also included. Averaged scalp distributions over the algorithms are depicted as well.

Table 3: Averaged testing accuracies and number of channels across users of the selected run for each single-objective method.

Mtd.	Competition		Center		RSVP	
	Acc.	***N***	**Acc.**	***N***	**Acc.**	***N***
GA	92.0%	14.0	97.4%	12.4	84.6%	13.4
BDE	92.0%	14.5	97.9%	12.5	85.5%	13.4
BPSO	92.0%	14.0	96.8%	12.5	85.0%	13.7
ALL	92.0%	64.0	86.5%	63.0	80.3%	61.0
KRU	86.5%	8.0	95.2%	8.0	78.6%	8.0

Mtd.: method, Acc.: accuracy, N: no. of sequences. Results obtained using the maximum number of sequences available for each database (competition: 15, center: 10, RSVP: 10).

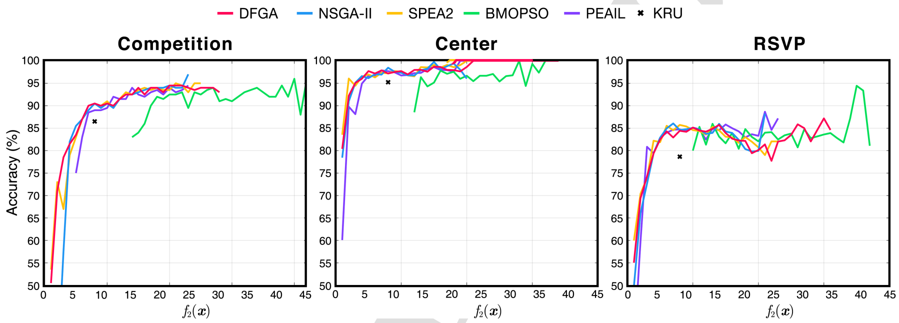


Figure 6: Testing character accuracies of the final Pareto-fronts returned by multi-objective meta-heuristics (DFGA, NSGA-II, SPEA2, BMOPSO, PEAIL) for the averaged subjects of each database. For comparison purposes, Krusienski's set (KRU) is also depicted.

Pareto-front for each subject is composed of the non-dominated solutions of all repetitions. Testing accuracies (i.e., ratio of correctly predicted characters) of the solutions that belongs to the final Pareto-fronts are shown in the figure 6, again using the maximum number of sequences available. Finally, computation costs of all algorithms are detailed in the table 2.

5. Discussion

5.1. Convergence analysis

Regarding the single-objective meta-heuristics, results showed that the inherently discrete algorithms (i.e., GA and BDE) converge to optimal solutions faster than BPSO, and were able to reach the minimal objective value for every single subject. Inherent discrete algorithms are understood as meta-heuristics that employs binary methodologies to improve their solutions (i.e., mutation, crossover). Even though BPSO showed a slower convergence than GA or BDE, the reached $F(\boldsymbol{x})$ values are almost analogous, suggesting that BPSO, GA and BDE would show similar performances in testing phase. It is also noteworthy that, even though the averaged convergence of GA was faster than BDE, the curve reached a standstill over the 100th generation, being overcame by BDE thereafter.

Multi-objective meta-heuristics results showed that DFGA, NSGA-II, SPEA2 and PEAIL algorithms were able to reach similar Pareto-fronts, outperforming BMOPSO. Besides the proper performance of DFGA, NSGA-II, SPEA2 and PEAIL in training phase, they did not converge to their optimal solutions in the same amount of time. DFGA converged faster than the rest, likely due to its deterministic initialization, which allows the algorithm to avoid evaluating solutions that are far from reaching the optimal value. Among the others, NSGA-II converged faster than PEAIL and SPEA2, whose trails spread across higher non-optimal $f_2(\boldsymbol{x})$ values. In contrast to the training performance of these algorithms, BMOPSO did not show a suitable convergence. In fact, their final values were far from matching the Pareto-fronts of DFGA, NSGA-II, SPEA2 and PEAIL. It is noteworthy that BMOPSO fronts keep high $f_2(\boldsymbol{x})$ values, which demonstrates that the algorithm was not able to improve solutions with a few channels. Owing to this behavior, it is not possible to assure that BMOPSO would reach proper performances in testing phase.

Considering these convergence results, it might be argued that meta-heuristics that work with discrete solutions (i.e., GA, BDE, DFGA, NSGA-II, SPEA2, PEAIL) present superior convergence results, which could be somewhat expected due to the nature of the problem. On the one hand, it can be said that local search strategies that rely on mutation, crossover and strength operators favor the convergence in the P300-based BCI channel selection problem [13, 14, 51]. On the other hand, the behavior of BPSO and BMOPSO could imply that the discretization of continuous solutions cannot follow small value changes, hindering the local exploitation of the continuous-based algorithms if their hyperparameters have not been properly fixed.

5.2. Channel distributions

Averaged channel ranks of the figure 5 show that meta-heuristics in general had a slight tendency to mainly select electrodes over the occipital lobe. However, the optimal channel set was clearly different for each subject. This behavior confirms the fact that a customized channel selection procedure prior to the BCI session benefits the subsequent performance. Despite the Krusienski's [8] common 8-channel set is suitable as a general rule of thumb, results did not considered that combination optimal for any subject or database. This fact is reinforced in the testing phase, where both single-objective and multi-objective algorithm solutions outperformed the 8-channel set, as can be noticed in the table 3 and the figure 6. Moreover, the computed Pareto-fronts did not generally kept solutions with more than ~ 20 channels. This fact suggests that a set with few channels is able to reach similar or even better performances than the full set, reducing the dimensionality and the computational cost of the BCI processing framework.

According to the previous analysis, meta-heuristics that converged faster for this optimization problem have also succeeded in finding the most relevant channels for each subject. As can be seen in the figure 5, GA, BDE, BPSO, DFGA, NSGA-II, SPEA2 and PEAIL reiteratively selected a specific combination of channels, which is different for each subject. By contrast, BMOPSO did not show clear differences between channel ranks, which once again indicates a lack of convergence to a global optimum.

From the well-defined electrodes that were repeatedly selected for each subject, algorithms showed a special focus on the occipital cortex. From a biological point of view, this tendency is sound. As aforementioned, the $f_1(\boldsymbol{x})$ objective is aimed to maximize the classification performance between target and non-target event-related stimuli, elicited through a visual odd-ball task. The response is therefore modeled as an event-related potential (ERP) composed by several components, such as P1, N1, P2, N2 or P3; which are taken into account when extracting and classifying the features. Among them, P3 (i.e., P300) should be the most prominent one in the RCP [1, 53]. The primary visual cortex, highly specialized in processing information about visual stimuli, static and moving objects; is located at the posterior part of the occipital lobe [54]. Hence, it is expected that occipital electrodes contain relevant discriminative information about target (i.e., ERP is present) and non-target signals (i.e., no ERP should be present) and thus, that they would likely be selected in the channel selection process. Nevertheless, the optimal channel sets are clearly different among subjects, which is relative common in the literature [19, 26, 27, 29]. This fact should not be surprising, since classifiers are frequently optimized for each subject because of the inter-subject and inter-session variability of the EEG signals [38]. Even though the rationale behind the fact that optimal channel sets differ among subjects is not clear, it is believed that EEG is highly sensitive to external factors, such as inter-subject variations in cap positions [55]. In fact, it is common that EEG caps does not correctly fit some users, making some electrodes wobbly and producing noise. Furthermore, it should be taken into account that EEG channels cannot pinpoint neural sources owing to attenuation and volume conduction effects, being limited to a spatial resolution about 5-10 cm and hindering the location of these sources in certain brain areas [1].

In short, we believe that most relevant channels for classification may not necessarily be the same among different users. Like feature selection and classification, results have shown the need to optimize the channel selection stage for each user. Notwithstanding its usefulness as a preliminary approach, the common Kruskienski's set [8], which mainly locates channels over the parietal and occipital cortex, appears to be a suboptimal solution. Note that our study is not intended to propose a general distribution of electrodes for any user, but to emphasize the need to customize the channel set for each subject.

5.3. Testing assessment

Single-objective approaches return a single solution with a certain number of channels, which minimizes the general objective $F(\boldsymbol{x})$. For this reason, averaged testing accuracies of table 3 should be taken into consideration together with the number of channels of each solution. According to the results, even though all the single-objective meta-heuristics reached higher accuracies in comparison to the full set and the Krusienski's 8-channel set, BDE stood out considering the channel-performance trade-off. BDE (competition: 92.0% with 14.5ch, center: 97.9% with 12.5ch, RSVP: 85.5% with 13.4ch) reached the highest average accuracy with a scarce channel set, followed by GA (competition: 92.0% with 14.0ch, center: 97.4% with 12.4ch, RSVP: 84.6% with 13.4ch) and BPSO (competition: 92.0% with 14.0ch, center: 96.8% with 12.5ch, RSVP: 85.0% with 13.7ch). Nevertheless, all methods reached similar or even higher accuracies than the full set of channels, probably due to the drastic increase in dimensionality; outperforming as well the accuracy obtained by the typical 8-channel set. It is worthy to mention, however, that Krusienski's set also used less number of channels. In fact, the increase in testing accuracy of the three methods in comparison with the full set (i.e., ALL) and the Krusienski's set (i.e., KRU) is statistically significant for almost all subjects (i.e., p–value< 0.05, Wilcoxon signed-rank test, false discovery rate corrected by the Benjamini-Hochberg procedure). In particular, the number of subjects (out of 27) that yielded significant differences were: 23 (ALL vs. GA), 27 (ALL vs. BDE), 22 (ALL vs. BPSO), 21 (KRU vs. GA, BDE or BPSO). As expected, the differences among GA, BDE and BPSO results are not significant. A detailed table with the p–values of each subject and comparison is included in the supplementary material. Therefore, it can be assured that GA, BDE and PSO outperformed ALL and KRU; and that their solutions were similar in terms of reached accuracies.

The main advantage of the multi-objective meta-heuristics in comparison with the single-objective ones is that they return a set of optimal solutions for each number of channels, allowing the user to select the most appropriate configuration. In fact, not only these solutions indicate the number of channels that already reaches the maximum performance, but also their scarce solutions overcame the Krusienski's set. According to the figure 6, the typical 8-channel set is outperformed using only 5 channels by DFGA, NSGA-II, SPEA2 and

PEAIL. By contrast, BMOPSO needed 13 channels to outperform it. These results are similar or even better than the individual solutions reached by single-objective algorithms. As can be noticed, DFGA, NSGA-II, SPEA2 and PEAIL reached similar performance results, which improved as the number of channels increased. Those results also outperformed BMOPSO, whose solutions, in spite of using more channels, generally obtained lower accuracies. Results also showed that there is a point for each subject where accuracies come to a standstill. In particular, using more than 15 channels in the competition or RSVP databases could be counter-productive; as well as using more than 20 channels in the center database. This fact reinforces the usefulness of dimensionality reduction techniques, such as channel and feature selection or classifier regularization, to assure a suitable testing performance in BCI systems.

5.4. Related work

Regarding related previous studies, we consider that MI-based [15–23] and auditory potential [28] BCI studies are not comparable in terms of performance, since those control signals are generally less reliable than P300 potentials and thus, obtain significantly lower accuracies. From the P300-based BCI studies, the reached accuracies of our work are similar or even higher than those reported previously, as shown in table 4. The most straightforward comparison comes from the 'III BCI Competition 2005 (dataset II)' (2 subjects), used by [4, 19, 27, 30]. Kee et al. [19] reached an average accuracy of 93.6% with 22.3 channels using GA; and 94.9% with 25.7 channels using NSGA-II. Arican and Polat [30] reached an averaged accuracy of 89.90% with 8 channels using BPSO and a boosted tree classifier. All of them used 15 sequences. A combination of wavelets and BPSO was also used by Perseh and Sharafat [4], obtaining 85% with 31 channels; and Gonzalez et al. [27], 67.5% with 33.5 channels using only 5 sequences. As can be seen, it is hard to compare the accuracies provided each study reported solutions with different number of channels or sequences. In our study, GA, BDE and BPSO yielded an averaged accuracy of 92% with 14 channels. The multi-objective metrics reached 90% of accuracy using 7 (DFGA, SPEA2), 8 (NSGA-II) and 11 (PEAIL) channels, which increased until a maximum of 97% with 23 channels using 15 sequences. There are also studies with custom databases, such as Chaurasiya et al. [29] (9 subjects, 15 sequences), who obtained a mean of 92.8% with 26.1 channels using MOBDE; or Jin et al. [26] (11 subjects, 15 sequences), who tested a RCP-based Chinese speller using PSO and LDA, reaching a mean of 71.09% with 7.63 channels. Besides the III BCI Competition database (2 subjects), our study also comprises the results with two additional databases: Center Speller (13 subjects, 10 sequences), and RSVP Speller (12 subjects, 10 sequences). However, no direct comparison can be made since there are no previous studies that have tested any meta-heuristic for selecting channels with any paradigm apart from RCP. In terms of accuracy, our results for single-objective (center: 97.36% with 12.46 channels, RSVP: 85.03% with 13.5 channels) and averaged multi-objective (center: 97.64% with 8 channels, RSVP: 84.87% with 8 channels) algorithms are similar to the performances reported in the literature [33, 34]. In this context, we would also like to encourage researchers to use our results and these public databases as a benchmark for favoring quantitative comparisons in the BCI channel selection problem.

Recently, deep learning has started to make a breakthrough in the P300-based BCI field due to its ability to achieve superior performances, especially in the decoding stage [12, 56–61]. In the era of deep learning, one may wonder whether a channel selection based on classical machine learning methodologies is still relevant. In that respect, many of the deep learning approaches for P300-based BCIs do not get rid of pre-processing and channel selection stages, but are applied as a direct substitute of the feature extraction and classification stages. However, there is a possibility in which the interpretation of the deep neural network (DNN) may lead to channel and feature selection alternatives.

For instance, Cecotti and Gräser [12] addressed the channel selection optimization by rejecting the smallest weights from the first hidden layer. However, a direct analysis of weights has disadvantages that should not be overlooked, as it is completely dependent on the DNN architecture used to classify P300 potentials. Today, it is generally accepted by the academia that bringing understand-

Table 4: Comparison between channel selection meta-heuristics applied in P300-based BCIs.

Study	Database	N_s	Method	Accuracy	N_c
Kee et al. [19]	Comp.	15	GA	93.60%	22.3
			NSGA-II	94.90%	25.7
Arican and Polat [30]	Comp.	15	BPSO	89.90%	8.0
Perseh and Sharafat [4]	Comp.	15	BPSO	85.00%	31.0
Gonzalez et al. [27]	Comp.	5	BPSO	67.50%	33.5
Chaurasiya et al. [29]	Custom, 9HS	15	MOBDE	92.80%	26.1
Jin et al. [26]	Custom, 11HS	15	BPSO	71.09%	7.6
Our study	Comp.	15	GA	92.00%	14.0
			BDE	92.00%	14.5
			BPSO	92.00%	14.0
			DFGA*	94.50%	20.0
			NSGA-II*	94.50%	20.0
			SPEA2*	94.00%	16.0
			BMOPSO*	92.50%	20.0
			PEAIL*	94.00%	14.0
	Center	10	GA	97.40%	12.4
			BDE	97.90%	12.5
			BPSO	96.80%	12.5
			DFGA*	97.88%	7.0
			NSGA-II*	98.46%	8.0
			SPEA2*	97.72%	9.0
			BMOPSO*	97.82%	16.0
			PEAIL*	100.0%	19.0
	RSVP	10	GA	84.60%	13.4
			BDE	85.50%	13.4
			BPSO	85.00%	13.7
			DFGA*	85.73%	14.0
			NSGA-II*	86.05%	7.0
			SPEA2*	85.73%	8.0
			BMOPSO*	84.80%	18.0
			PEAIL*	85.82%	15.0

N_s: number of sequences, N_c: averaged number of channels, Comp.: III BCI Competition 2005 (dataset II), HS: healthy subjects, GA: genetic algorithm, NSGA-II: non-sorting genetic algorithm 2, BPSO: binary particle swarm optimization, BA: bees algorithm, ABC: artificial bee colony, BAS: binary ant system, FA: firefly algorithm, MOBDE: multi-objective binary differential evolution, BDE: binary differential evolution, DFGA: dual-front sorting algorithm, SPEA2: strength pareto evolutionary algorithm 2, BMOPSO: binary multi-objective particle swarm optimization, PEAIL: Pareto Evolutionary Algorithm based on Incremental Learning. * The selected solution for the multi-objective approaches was the one that maximized the accuracy in the range $N_c \in [5, 20]$. The rest of the accuracies can be checked in Figure 6.

ing to DNN models is still a very challenging issue. While single-layer linear transformations can be easily interpreted by looking at the learned weights, multiple layers with non-linear interactions on each layer involve disentangling a complicated nested structure [62]. In general, the more interpretable the model, the simpler and less accurate [62, 63]. Although this approach would be feasible in simple convolutional neural networks (CNN) where spatial information is processed in a single layer [12], it would be a serious challenge in more complex and recent architectures (e.g., CNN-BLSTM, DeepConvNet, EEGNet, EEG-Inception [56–61]). Since the complexity of DNN architectures is increasing rapidly, it is expected that the spatial information from the EEG is likely to be processed multiple times in different layers [61]. Thus, the difficulty of evaluating the importance of each channel and its contribution to the final result could increase exponentially.

Moreover, the efficacy of this approach when using simple DNN architectures compared to the proposed meta-heuristics is not clear, as they reported an average accuracy of 87.0% (8 ch.) in the 'III BCI Competition 2005 (dataset II)'. As shown in the figure 6, the Krusienski's set already achieved a mean accuracy of 86.5% (8 ch.), i.e., without performing channel selection. Similarly, the solutions achieved by most of the multi-objective meta-heuristics using only 8 channels clearly outperformed the accuracy of Cecotti and Gräser [12]: 90.5% (DFGA, NSGA-II, SPEA2) and 89.0% (PEAIL); even being obtained by an LDA, a machine learning method much simpler than a CNN.

To the best of our knowledge, none of the DNN-based approaches applied in P300-based BCIs performed a different channel selection approach [56–61], instead they used the full set of channels available. By contrast, Zhang et al. [64] recently published a channel selection method based on adding sparse regularization to squeeze-and-excitation blocks in a CNN and applied it to MI decoding. However, whether this method is feasible for P300-based BCIs is an open question yet.

Noteworthy, a DNN-based channel selection procedure would presumably require using the initial number of electrodes (e.g., 64) whenever the model is tested with new observations. This problem could be solved by retraining the DNN with the "relevant" channels. However, if this taxonomy is done using explainable DNNs without a wrapper test of the solution, it cannot be claimed that the performance would be maximized. In this case, we believe deep learning does not provide enough incentives to be a substitute for the proposed meta-heuristics. In this context, the proposed meta-heuristics can be applied in an initial channel selection stage for P300-based BCIs, as they are totally independent of the DNN architecture. It should be also noted that there are still many BCI applications that use classical machine learning methods (e.g., LDA, SVM), either due to limited computational power, few training examples or even preference, which would also clearly benefit from our proposal.

5.5. Hyperparameters

The main drawback of most meta-heuristics is the need to fix hyperparameters, which usually depend on the context of the problem. Poorly chosen values may cause convergence issues, limiting the performance of the algorithm [13]. Since all algorithms were able to converge before reaching their own limit of generations, the number of individuals and generations were appropriate. In this context, the quality of a meta-heuristic must not be only assessed according to the performance results, but also taking into account the number of required hyperparameters. The less hyperparameters, the more likely it is to assure reliable and generalizable results. GA, DFGA, NSGA-II and SPEA2 only require mutation and crossover rates to be fixed. Fortunately, these parameters are widely studied in the literature, where often are $1/N$ for the mutation rate, and 0.90–0.95 for the crossover rate [13, 48, 49]. A similar approach is followed in BDE, whose extra parameters are intended to perform a mutation procedure [42]. PEAIL also requires an additional parameter: the learning rate, which controls the confidence on the best individual [52]. BPSO and BMOPSO add three hyperparameters more (i.e., personal and global confidences and maximum velocity). Although there are several studies that tried to find global relations among their values, further endeavors should be made in order to make PSO algorithms problem-independent [13, 44]. Since these hyperparameters directly weigh the velocity of the particles, which is then used as an input of a transfer function, care must be taken in order to limit their values in the range $[0, 1]$.

Otherwise, particles will not tend to improve their solutions, restricting global and local exploitation. In this study, we fixed the hyperparameters according to the recommendations of the literature, as indicated in the table 1. These values yielded suitable performances, but could have been improved by means of a hyperparameter optimization or following an adaptive approach. It is also worthy to mention that the weights ω_1 and ω_2 of $F(\boldsymbol{x})$ were heuristically set to 0.7 and 0.3, respectively, in view of preliminary results [7, 31, 35]. Note that the value difference between ω_1 and ω_2 would cause a strengthen of solutions in a certain $f_2(\boldsymbol{x})$ range, while avoiding the search in other spaces. The supervisor could vary the ω_1/ω_2 ratio to obtain different optimal solutions, simulating the search over the $f_2(\boldsymbol{x})$ spectrum as multi-objective approaches do.

5.6. Computational cost

A comparison between computational costs of different meta-heuristics is tricky, forcing to consider several aspects at the same time. On the one hand, the table 2 details the approximate duration of a generation and the number of evaluations that comprised a generation. Note that the number of evaluations per generation differs depending on the meta-heuristic strategy and thus, algorithms can only be compared in terms of the duration of a single evaluation. For this reason, the number of generations of each algorithm has been adapted in order to assure a fair comparison among them, so that every single method performs a total of 4000 evaluations. On the other hand, it is also essential to consider additional aspects, such as the convergence of the algorithms, their search depth and the programming approach. When abstracting a meta-heuristic as a black box, the total time of the execution varies according to the required number of generations to reach a suitable convergence. These differences usually affect the computational cost in a higher extent than the individual duration of a evaluation, making it essential to be taken into consideration. Moreover, a correct implementation of these meta-heuristics should employ a hash table to match the previously computed solutions with their fitnesses. Note that an intense search depth will inevitable generate repeated solutions. The hash map acts as a remainder and allows avoiding unnecessary evaluations. Noteworthy, the computation time of a generation should decline exponentially when the algorithm goes on. Note that table 2 measurements were made without considering any hash table (i.e., an initial generation).

The overall complexity of DFGA in asymptotic notation behaves as $O(N_oN^2)$, where N_o is the number of objectives. An analysis of the complexity of each DFGA step is detailed in the supplementary material, which demonstrates that the exponential increase in operations is mainly due to the dual-front sorting procedure. However, this trend is similar to other recent multi-objective algorithms, such as NSGA-II, SPEA2, PEAIL and BMOPSO, whose complexity behaves as $O(N_om^2)$ [65, 66]. Note that m indicates the population size which, in case of DFGA, equals to N (the no. channels) due to the deterministic initialization. This fact partly explains why DFGA performs a higher number of evaluations in a single generation. It is also noteworthy that the presented asymptotic complexity analysis involves only the advancing of a single generation, allowing the comparison among different algorithms, since their convergences are not deterministic [66]. Moreover, note that these complexities indicate the worst cases, which usually decrease as generations increase due to the hash table implementation.

According to the table 2, NSGA-II, PEAIL and DFGA were the least time-consuming, spending less than 600 ms per evaluation with the selected hyperparameters. In addition, they demonstrated excellent convergence abilities, making them excellent multi-objective approaches to address this problem. By contrast, BPSO and BMOPSO not only were the most time-consuming algorithms, but also their performances were inferior. For the single-objective approaches, it is worthwhile to use GA or BDE, whose convergence abilities balanced out their evaluation costs. In any case, the overall duration of these algorithms restrict their application to the calibration session, where the weights of the classifier are optimized for each subject. Then, the selected channels should be further applied in the testing sessions.

5.7. Guidelines

A series of guidelines or practical recommendations for the application of meta-heuristics to BCI systems are derived from the discussed results:

1. Multi-objective algorithms should be used instead of single-objective meta-heuristics if computation time is not an issue. Otherwise, it is preferable even using deterministic algorithms, such as BE, to provide sub-optimal but acceptable solutions.

2. Discrete algorithms that use mutation, crossover or strength operators should be preferred (e.g., single-objective: GA, BDE; multi-objective: DFGA, NSGA-II, SPEA2, PEAIL).

3. Whether discretization is required to adapt a continuous-based meta-heuristic to the BCI framework, avoid using transfer functions and attempt to redefine the equations (section 3.5). Still, if conversion via transfer functions is used, care should be taken with the hyperparameters values. Assure that the input of the function always lies within the range $[0, 1]$. Otherwise, the probability of change of the solutions would increase drastically, hindering the convergence of the algorithm. Furthermore, distance metrics should not be employed after applying the discrete transformation unless it is a Hamming distance.

4. For single-objective algorithms, use an aggregation approach to minimize two objectives at one: number of channels and performance error (section 3.2). AUC-based modeling of the performance should be preferred instead of accuracies in order to increment the resolution of the objective values.

5. Multi-objective repository limitation strategies, such as crowding distances or distance sought, are not necessary in the BCI channel selection problem and should be avoided to prevent worthless computational costs (section 3.5).

6. A hybrid meta-heuristic that also employs deterministic methods, such as DFGA, should be preferred. DFGA reached similar accuracies than NSGA-II, SPEA2, or PEAIL, but converged faster (section 4).

7. A convergence detection method to stop the iterations is recommended rather than using a maximum generation limit for practical purposes (e.g., none or petty changes along the n last generations).

8. Repeated solutions across generations are unavoidable. It is required to implement a hash map (e.g., dictionary) for matching previously computed solutions with their fitnesses, in order to avoid unnecessary evaluations.

5.8. Contributions, limitations and future work

According to the experimental outcomes, it has been demonstrated the utility of meta-heuristics to find an optimal combination of channels in P300-based BCI systems. The importance of selecting an optimal channel set for each user has been highlighted as well. Moreover, to the best of our knowledge, this is the first study that provides a comprehensive comparison of different meta-heuristics that can be applied to the BCI channel selection problem. Previous studies have isolated the application of BPSO, BDE, GA and NSGA-II, but no comparison has been performed; whereas this manuscript has included a total of 3 single-objective (i.e., GA, BDE, BPSO) and 5 multi-objective (i.e., DFGA, NSGA-II, SPEA2, PEAIL, BMOPSO) algorithms. As a result, GA, BDE, BPSO, DFGA, NSGA-II, SPEA2 and PEAIL have reached high performances in testing phase, outperforming the full set and the common Krusienski's 8-channel set in three databases with different paradigms. Due to the characteristics of the BCI framework, none of the well-known methods can be applied in a productive way without a proper modification. For that reason, DFGA, a new multi-objective meta-heuristic, has been especially designed to optimize channel or feature sets in BCI systems. Moreover, results have shown that the meta-heuristics that exhibited better convergences repeatedly selected the same distribution of channels, which clearly depended on the subject. It is thus suggested that Krusienski's set (or any set that covers the occipital lobe) is a general rule of thumb solution that could led to acceptable performances, but definitely suboptimal. This fact reinforces the importance of performing a channel optimization for each user to maximize the performance of the system. In fact, due to the high inter-session and inter-subject variability of the EEG, the optimization of signal processing stages such as feature selection and classification for each user are a common practice. Hence, we would want to encourage researchers to integrate channel selection procedures to those optimization pipelines. In this context, the supervisor could apply DFGA in the first session, select an appropriate channel set and avoid placing extra electrodes for the next BCI sessions. Noteworthy, in order to ease the application of meta-heuristics in these systems, a series of guidelines have been detailed.

Despite the aforementioned strengths, several limitations can be pointed out. Firstly, since the purpose of the manuscript was focused on the channel selection procedure, only basic feature extraction (i.e., down-sampling) and classification (i.e., LDA) methods have been applied. Testing accuracies, especially those than belong to crowded channel sets, could have been improved by using regularization techniques [4–6] or deep learning approaches [58, 67]. Moreover, the algorithms entail high computational costs. Further endeavors should be aimed at assigning stopping criteria that could avoid the computation of worthless generations, allowing a better estimation of the total duration for each model. The computational cost is mainly caused by the wrapper nature of the algorithms, which evaluate the quality of a solution by training and testing different LDA models [68]. Embedded techniques (e.g., heuristic search methods), which look for optimal sets inside the classifier constructor, are less intensive than wrappers [68, 69]. A future endeavor could be aimed at developing new embedded techniques that could reduce the computational cost by modifying the training procedures of certain classifiers. It is worth mentioning that Deb and Jain [70] proposed an extension of NSGA-II, called NSGA-III, to handle many-objective (i.e., four or more objectives) optimization problems. Although its application to this problem could be also interesting as a further research line, we applied

NSGA-II as it is oriented to solve multi-objective optimization problems. Care should be also taken when using transfer functions, such as those used in BPSO or BMOPSO, since they could be fruitless to the proper exploitation of the discrete space. Another research line could be focused on providing a comprehensive comparison of alternative genetic operators and strategies to maximize the performance of DFGA. Hyperparameters were fixed according to the recommendations of the literature. However, an optimization of these values would be beneficial to the final performance of the algorithms. Adaptive approaches that vary the hyperparameters in function of the generation could also enhance the results. It should be also mentioned that the competition database contains more training trials than those that are commonly in practice. It would be also interesting to explore the usefulness of interpretable deep learning approaches to infer the significance of the selected channels in the classification stage [63]. Another future research line could be focused on assessing the performance of these methods with less training trials. Finally, although it has not been explored in study, results suggest that the proposed meta-heuristics could be also applied to feature selection problems.

6. Conclusions

A comprehensive comparison among 8 different meta-heuristics applied to the P300-based BCI channel selection problem has been performed in this study. In particular, 3 single-objective and 5 multi-objective algorithms have been included. Due to the discrete characteristics of the BCI framework, the majority of them have been modified in different ways in order to adapt them to the aforementioned problem. For this reason, a series of guidelines or practical recommendations have been detailed as an aid for further adaptations. A novel multi-objective algorithm, DFGA, has been especially developed for BCI systems. Methods have been tested with three public databases that used different stimulation paradigms: competition (2 users with 64ch., RCP), center speller (13 users with 63ch., CS paradigm) and RSVP speller (12 users with 61ch., RSVP). Results showed that meta-heuristics are able to provide solutions that simultaneously use few number of channels and reach high accuracies. In fact, the full set of channels and the common Krusienki's 8-channel set have been outperformed by all methods, demonstrating their usefulness to provide an optimized channel set for each user.

The main findings of the study can be summarized as follows:

1. Optimal channel sets show a high inter-subject variability, which makes essential the optimization for each individual, instead of using a common set for all of them.

2. Inherently discrete algorithms (i.e., GA, BDE, DFGA, NSGA-II, SPEA2, PEAIL) usually reach higher performances due to the dichotomous nature of the problem.

3. Among single-objective meta-heuristics, GA, and BDE provide suitable convergences and high accuracies. Regarding multi-objective algorithms, DFGA, NSGA-II, SPEA2 and PEAIL provided competitive results.

4. A balanced combination of deterministic and stochastic techniques is beneficial. DFGA reaches an excellent performance, as well as NSGA-II, SPEA2 and PEAIL, but converges considerably faster to their optimal solutions.

5. Hyperparameter tuning is crucial. BMOPSO could not converge to an optimal solution, whereas it is possible to guarantee the convergence of the rest in a single run.

Acknowledgments

This study was partially funded by projects RTC2019-007350-1 of the 'Ministerio de Ciencia, Innovación y Universidades' and 0702_MIGRAINEE_2_E of the European Commission, as well as by CIBER-BBN through 'Instituto de Salud Carlos III' co-funded with FEDER funds. Eduardo Santamaría-Vázquez was in receipt of a PIF grant by the 'Consejería de Educación de la Junta de Castilla y León'.

References

[1] J. Wolpaw, E. W. Wolpaw, Brain-computer interfaces: principles and practice, OUP USA, 2012.

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

[3] H. Cecotti, B. Rivet, M. Congedo, C. Jutten, O. Bertrand, E. Maby, J. Mattout, A robust sensor-selection method for P300 brain-computer interfaces, Journal of Neural Engineering 8 (2011) 16001.

[4] B. Perseh, A. R. Sharafat, An Efficient P300-based BCI Using Wavelet Features and IBPSO-based Channel Selection, Journal of Medical Signals and Sensors 2 (2012) 128–143.

[5] F. Tahernezhad-Javazm, V. Azimirad, M. Shoaran, A review and experimental study on application of classifiers and evolutionary algorithms in EEG based brain-machine interface systems, Journal of Neural Engineering (2017).

[6] B. Blankertz, S. Lemm, M. Treder, S. Haufe, K. R. Müller, Single-trial analysis and classification of ERP components - A tutorial, NeuroImage 56 (2011) 814–825.

[7] V. Martínez-Cagigal, R. Hornero, P300-based Brain-Computer Interface channel selection using swarm intelligence, Revista Iberoamericana de Automática e Informática Industrial 14 (2017) 372–383.

[8] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, J. R. Wolpaw, Toward enhanced P300 speller performance, Journal of Neuroscience Methods 167 (2008) 15–21.

[9] J. D. Jobson, Applied multivariate data analysis: volume I: Regression and Experimental Design, Springer, 1991.

[10] L. Yu, H. Liu, Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution, Proceedings, Twentieth International Conference on Machine Learning 2 (2003) 856–863.

[11] H. Zou, T. Hastie, Erratum: Regularization and variable selection via the elastic net (Journal of the Royal Statistical Society. Series B: Statistical Methodology (2005) 67 (301-320)), Journal of the Royal Statistical Society. Series B: Statistical Methodology 67 (2005) 768.

[12] H. Cecotti, A. Gräser, Convolutional neural networks for P300 detection with application to brain-computer interfaces, IEEE Transactions on Pattern Analysis and Machine Intelligence 33 (2011) 433–445.

[13] X.-S. Yang, Nature-Inspired Optimization Algorithms, 1 ed., Elsevier Inc., 2014.

[14] O. Bozorg-Haddad, M. Solgi, H. A. Loaiciga, Meta-Heuristic and Evolutionary Algorithms for Engineering Optimization, 2017. URL: www.wiley.com.

[15] J. Lv, M. Liu, Common Spatial Pattern and Particle Swarm Optimization for Channel Selection in BCI, 2008 3rd International Conference on Innovative Computing Information and Control (2008) 457.

[16] B. A. S. Hasan, J. Q. Gan, Multi-Objective Particle Swarm Optimization for Channel Selection in Brain-Computer Interfaces, The UK Workshop on Computational Intelligence (2009) 2–7.

[17] B. A. S. Hasan, J. Q. Gan, Q. Zhang, Multi-objective evolutionary methods for channel selection in brain-computer interfaces: some preliminary experimental results, IEEE Congress on Evolutionary Computation (CEC-2010) (2010) 1–6.

[18] Q. Wei, Y. Wang, Binary Multi-Objective Particle Swarm Optimization for Channel Selection in Motor Imagery Based Brain-Computer Interfaces, 2011 4th International Conference on Biomedical Engineering and Informatics (BME I) (2011) 667–670.

[19] C.-Y. Kee, S. G. Ponnambalam, C.-K. Loo, Multi-objective genetic algorithm as channel selection method for P300 and motor imagery data set, Neurocomputing 161 (2015) 120–131.

[20] R. Aler, I. M. Galván, Optimizing the number of electrodes and spatial filters for Brain-Computer Interfaces by means of an evolutionary multi-objective approach, Expert Systems with Applications 42 (2015) 6215–6223.

[21] A. Franklin Alex Joseph, C. Govindaraju, Channel selection using glow swarm optimization and its application in line of sight secure communication, Cluster Computing 22 (2019) 10801–10808.

[22] L. Zhang, Q. Wei, Channel selection in motor imaginary-based brain-computer interfaces: a particle swarm optimization algorithm, Journal of integrative neuroscience 18 (2019) 141–152.

[23] J. González, J. Ortega, M. Damas, P. Martín-Smith, J. Q. Gan, A new multi-objective wrapper method for feature selection – Accuracy and stability analysis for BCI, Neurocomputing 333 (2019) 407–418.

[24] A. Albasri, F. Abdali-Mohammadi, A. Fathi, EEG electrode selection for person identification thru a genetic-algorithm method, Journal of Medical Systems 43 (2019).

[25] Z. A. A. Alyasseri, A. T. Khader, M. A. Al-Betar, O. A. Alomari, Person identification using EEG channel selection with hybrid flower pollination algorithm, Pattern Recognition 105 (2020) 107393.

[26] J. Jin, B. Z. Allison, C. Brunner, B. Wang, X. Wang, J. Zhang, C. Neuper, G. Pfurtscheller, P300 Chinese input system based on Bayesian LDA, Biomedizinische Technik 55 (2010) 5–18.

[27] A. Gonzalez, I. Nambu, H. Hokari, M. Iwahashi, Y. Wada, Towards the classification of single-trial event-related potentials using adapted wavelets and particle swarm optimization, Proceedings - 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC 2013 (2013) 3089–3094.

[28] A. Gonzalez, I. Nambu, H. Hokari, Y. Wada, EEG Channel Selection Using Particle Swarm Optimization for the Classification of Auditory Event-Related Potentials, Scientific World Journal 2014 (2014) 350270.

[29] R. K. Chaurasiya, N. D. Londhe, S. Ghosh, Multi-objective binary DE algorithm for optimizing the performance of Devanagari script-based P300 speller, Biocybernetics and Biomedical Engineering 37 (2017) 422–431.

[30] M. Arican, K. Polat, Binary particle swarm optimization (BPSO) based channel selection in the EEG signals and its application to speller systems, Journal of Artificial Intelligence and Systems 2 (2020) 27–37.

[31] V. Martínez-Cagigal, E. Santamaría-Vázquez, R. Hornero, A novel hybrid swarm algorithm for P300-based BCI channel selection, in: IFMBE Proceedings, volume 68, 2018. doi:10.1007/978-981-10-9023-3_8.

[32] B. Blankertz, K.-R. Müller, D. J. Krusienski, G. Schalk, J. R. Wolpaw, A. Schlögl, G. Pfurtscheller, J. D. R. Millán, M. Schröder, N. Birbaumer, The BCI competition III: Validating alternative approaches to actual BCI problems, IEEE Transactions on Neural Systems and Rehabilitation Engineering 14 (2006) 153–159.

[33] M. S. Treder, N. M. Schmidt, B. Blankertz, Gaze-independent brain-computer interfaces based on covert attention and feature attention., J Neural Eng 8 (2011) 66003.

[34] L. Acqualagna, B. Blankertz, Gaze-independent BCI-spelling using rapid serial visual presentation (RSVP), Clinical Neurophysiology 124 (2013) 901–908.

[35] V. Martínez-Cagigal, R. Hornero, A Binary Bees Algorithm for P300-Based Brain-Computer Interfaces Channel Selection, in: Advances in Computational Intelligence. IWANN 2017. Lecture Notes in Computer Science, 1 ed., Springer International Publishing AG, Cádiz, Spain, 2017, pp. 453–463. URL: https://link.springer.com/chapter/10.1007/978-3-319-59147-6{_}39. doi:10.1007/978-3-319-59147-6_39.

[36] K. A. Colwell, D. B. Ryan, C. S. Throckmorton, E. W. Sellers, L. M. Collins, Channel selection methods for the P300 Speller, Journal of Neuroscience Methods 232 (2014) 6–15.

[37] B. G. Marcot, A. M. Hanea, What is an optimal value of k in k-fold cross-validation in discrete Bayesian network analysis?, Computational Statistics 36 (2021) 2009–2031.

[38] V. Martínez-Cagigal, J. Gomez-Pilar, D. Álvarez, R. Hornero, An asynchronous P300-based Brain–Computer Interface web browser for severely disabled people, IEEE Transactions on Neural Systems and Rehabilitation Engineering 25 (2017) 1332–1342.

[39] C. a. Coello, M. Reyes-Sierra, Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art, International Journal of Computational Intelligence Research 2 (2006) 287–308.

[40] J. H. Holland, Genetic algorithms, Scientific american 267 (1992) 66–73.

[41] R. Storn, K. Price, Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization 11 (1997) 341–359.

[42] L. Wang, X. Fu, Y. Mao, M. Ilyas Menhas, M. Fei, A novel modified binary differential evolution algorithm and its applications, Neurocomputing 98 (2012) 55–75.

[43] J. Kennedy, R. Eberhart, Particle swarm optimization, Neural Networks, 1995. Proceedings., IEEE International Conference on 4 (1995) 1942—-1948 vol.4.

[44] M. Clerc, J. Kennedy, The Particle Swarm–Explosion, Stability, and Convergence in a Multidimensional Complex Space, IEEE Transactions on Evolutionary Computation 6 (2002) 58–73.

[45] R. Poli, J. Kennedy, T. Blackwell, Particle swarm optimization, Swarm Intelligence 1 (2007) 33–57.

[46] J. Kennedy, R. C. Eberhart, A Discrete Binary Version of the Particle Swarm Algorithm, 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation 5 (1997) 4–8.

[47] S. Mirjalili, A. Lewis, S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization, Swarm and Evolutionary Computation 9 (2013) 1–14.

[48] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2002) 182–197.

[49] K. Deb, Multi-Objective Optimization, 2005. URL: http://link.springer.com/10.1007/0-387-28356-0{_}10. doi:10.1007/0-387-28356-0_10. arXiv:arXiv:1011.1669v3.

[50] M. Reyes-Sierra, C. A. Coello, Improving PSO-Based Multi-objective Optimization Using Crowding, Mutation and E-Dominance, Lecture Notes in Computer Science 3410 (2005) 505–519.

[51] E. Zitzler, M. Laumanns, L. Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm, Technical Report, ETH Zurich, Zurich, 2001. doi:10.3929/ETHZ-B-000225616.

[52] L. Rong-Juan, J. Shou-Feng, Z. Bao-Lin, A Pareto evolutionary algorithm based on incremental learning for a kind of multi-objective multidimensional knapsack problem, Computers and Industrial Engineering 135 (2019) 537–559.

[53] J. Polich, Updating P300: An Integrative Theory of P3a and P3b, Clin. Neurophysiol. 118 (2007) 2128–2148.

[54] S. Standring, Gray's anatomy: the anatomical basis of clinical practice, 41 ed., Elsevier Ltd, London, 2016.

[55] T. W. Picton, The P300 wave of the human event-related potential, Journal of Clinical Neurophysiology 9 (1992) 456–479.

[56] R. Manor, A. B. Geva, Convolutional neural network for multi-category rapid serial visual presentation BCI, Frontiers in Computational Neuroscience 9 (2015) 1–12.

[57] M. Liu, W. Wu, Z. Gu, Z. Yu, F. F. Qi, Y. Li, Deep learning based on Batch Normalization for P300 signal detection, Neurocomputing 275 (2018) 288–297.

[58] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, B. J. Lance, EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces, Journal of Neural Engineering 15 (2018).

[59] E. Santamaría-Vázquez, V. Martínez-Cagigal, J. Gomez-Pilar, R. Hornero, Deep Learning Architecture Based on the Combination of Convolutional and Recurrent Layers for ERP-Based Brain-Computer Interfaces, in: J. Henriques, N. Neves, P. de Carvalho (Eds.), XV Mediterranean Conference on Medical and Biological Engineering and Computing – MEDICON 2019, Springer International Publishing, Cham, 2020, pp. 1844–1852.

[60] D. Borra, S. Fantozzi, E. Magosso, Convolutional Neural Network for a P300 Brain-Computer Interface to Improve Social Attention in Autistic Spectrum Disorder, in: J. Henriques, N. Neves, P. de Carvalho (Eds.), XV Mediterranean Conference on Medical and Biological Engineering and Computing – MEDICON 2019, Springer International Publishing, Cham, 2020, pp. 1837–1843.

[61] E. Santamaría-Vázquez, V. Martínez-Cagigal, F. Vaquerizo-Villar, R. Hornero, EEG-Inception: A Novel Deep Convolutional Neural Network for Assistive ERP-Based Brain-Computer Interfaces, IEEE Transactions on Neural Systems and Rehabilitation Engineering 28 (2020) 2773–2782.

[62] A. Adadi, M. Berrada, Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI), IEEE Access 6 (2018) 52138–52160.

[63] I. Sturm, S. Lapuschkin, W. Samek, K. R. Müller, Interpretable deep neural networks for single-trial EEG classification, Journal of Neuroscience Methods 274 (2016) 141–145.

[64] H. Zhang, X. Zhao, Z. Wu, B. Sun, T. Li, Motor imagery recognition with automatic EEG channel selection and deep learning, Journal of Neural Engineering 18 (2021).

[65] P. K. Tripathi, S. Bandyopadhyay, S. K. Pal, Multi-Objective Particle Swarm Optimization with time variant inertia and acceleration coefficients, Information Sciences 177 (2007) 5033–5049.

[66] D. M. Curry, C. H. Dagli, Computational complexity measures for many-objective optimization problems, Procedia Computer Science 36 (2014) 185–191.

[67] E. Santamaría-Vázquez, V. Martínez-Cagigal, J. Gomez-Pilar, R. Hornero, Deep Learning Architecture Based on the Combination of Convolutional and Recurrent Layers for ERP-Based Brain-Computer Interfaces, in:

Mediterranean Conference on Medical and Biological Engineering and Computing, Springer, 2019, pp. 1844–1852.

[68] Y. Saeys, I. Inza, P. Larrañaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (2007) 2507–2517.

[69] I. Guyon, A. Elisseeff, An Introduction to Variable and Feature Selection, Journal of Machine Learning Research (JMLR) 3 (2003) 1157–1182.

[70] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints, IEEE Transactions on Evolutionary Computation 18 (2014) 577–601.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

HIGHLIGHTS

- Meta-heuristics can find an optimal channel set in P300-based BCI systems.
- Six single- and four multi-objective algorithms are compared for the first time.
- Dual-Front Sorting Algorithm (DFGA) demonstrated an excellent performance.
- Discrete approaches performed better due to the dichotomous nature of the problem.
- Meta-heuristics cannot be used in their common form and should be carefully adapted.

CRediT author statement

Víctor Martínez-Cagigal: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization; **Eduardo Santamaría-Vázquez:** Investigation, Writing – review & editing; **Roberto Hornero:** Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.