


Article

Dimension Reduction Using New Bond Graph Algorithm and Deep Learning Pooling on EEG Signals for BCI

Ahmad Naebi ^{1,*} , Zuren Feng ¹, Farhoud Hosseinpour ²  and Gahder Abdollahi ³

¹ State Key Laboratory for Manufacturing System Engineering, System Engineering Institute, Xi'an Jiaotong University, Xianning West Road No. 28, Xi'an 710049, China; fzf9910@xjtu.edu.cn

² Department of Computing, University of Turku, 20014 Turku, Finland; farhos@utu.fi

³ Payame Noor University of Oskou, Oskou 19395-4697, Iran; kader.abdollahi@yahoo.com

* Correspondence: ahmad.naebi@stu.xjtu.edu.cn; Tel.: +86-15664889104

Abstract: One of the main challenges in studying brain signals is the large size of the data due to the use of many electrodes and the time-consuming sampling. Choosing the right dimensional reduction method can lead to a reduction in the data processing time. Evolutionary algorithms are one of the methods used to reduce the dimensions in the field of EEG brain signals, which have shown better performance than other common methods. In this article, (1) a new Bond Graph algorithm (BGA) is introduced that has demonstrated better performance on eight benchmark functions compared to genetic algorithm and particle swarm optimization. Our algorithm has fast convergence and does not get stuck in local optimums. (2) Reductions of features, electrodes, and the frequency range have been evaluated simultaneously for brain signals (left-handed and right-handed). BGA and other algorithms are used to reduce features. (3) Feature extraction and feature selection (with algorithms) for time domain, frequency domain, wavelet coefficients, and autoregression have been studied as well as electrode reduction and frequency interval reduction. (4) First, the features/properties (algorithms) are reduced, the electrodes are reduced, and the frequency range is reduced, which is followed by the construction of new signals based on the proposed formulas. Then, a Common Spatial Pattern is used to remove noise and feature extraction and is classified by a classifier. (5) A separate study with a deep sampling method has been implemented as feature selection in several layers with functions and different window sizes. This part is also associated with reducing the feature and reducing the frequency range. All items expressed in data set IIa from BCI competition IV (the left hand and right hand) have been evaluated between one and three channels, with better results for similar cases (in close proximity). Our method demonstrated an increased accuracy by 5 to 8% and an increased kappa by 5%.

Keywords: bond graph methodology; algorithms; dimension reduction; feature selection; electrode selection; frequency selection; common spatial pattern (CSP); time domain and frequency domain; frequency bands or filter bank (FB)



Citation: Naebi, A.; Feng, Z.; Hosseinpour, F.; Abdollahi, G. Dimension Reduction Using New Bond Graph Algorithm and Deep Learning Pooling on EEG Signals for BCI. *Appl. Sci.* **2021**, *11*, 8761. <https://doi.org/10.3390/app11188761>

Academic Editor: Sangtae Ahn

Received: 19 August 2021

Accepted: 15 September 2021

Published: 20 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Today, the EEG is used as a non-invasive system to record brain signals from electrodes on the scalp for brain activity. Brain signals aim to control systems for sick and healthy people, playing an essential role in various programs in different fields [1–7]. The primary advantage of processing brain signals is to discover information for prediction and classification. Brain signals have some general information-processing steps: filtering, feature extraction, feature selection, and classification. Most researchers have focused on feature selection for finding new suitable methods and algorithms for improvement [8–14].

Feature selection can appear in three possible methods.

- In the first method, feature selection (one-dimensional and more) after data preprocessing (e.g., filtering) and before feature extraction (if used). If the feature extraction

is not used, the selected features will be sent directly to the classifiers for classification after reducing the dimensions.

- In the second method, the following steps are performed respectively: (1) feature extraction, (2) feature selection, and (3) classification.
- In the third method, after reducing the dimensions of the features (i.e., feature selection), the following steps are performed, respectively: (1) feature extraction, (2) feature selection, and (3) classification.

In some studies, using the first and third methods, a dimensional reduction is used to reduce features, reduce electrodes, or both.

In all of the methods mentioned above, the features from a larger set (including channels, etc.) are converted into better and smaller feature sets by removing the less effective features. In most cases, a certain frequency range (8–30 Hz for the brain signal) is processed for specific purposes (such as left and right-hand imaging). In most studies, all the features along with specific channels or whole channels have been considered [15–20]. In addition, in some studies, one-dimensional or two-dimensional reduction methods are performed for reducing features and the electrodes. However, none of the previous studies examined reducing the frequency range from one to three electrodes.

In some articles, certain frequency ranges have been used. In other words, feature reduction is used by reducing the frequency range with a single electrode. Given that human brain signals are generally a combination of different domains, examining them as a single domain is challenging and has been studied in many previous studies [20–32]. In this study, we propose a new optimization algorithm based on the concept of Bond Graph, which is the modeling and simulation language for multi-domain systems. We examined and analyzed the reduction of frequency intervals of brain signals.

The main contributions of this paper are as follows:

1. Introduction of the Bond Graph algorithm (BGA), which demonstrated a good performance compared to the genetic algorithm (GA) and particle swarm optimization (PSO) in the benchmark functions. The assessment was performed by testing on eight benchmarks and EEG brain signals of right and left-hand perception (two classes). Important features of our proposed algorithm (i.e., BGA) include fast convergence and no trapping in the local optimization.
2. The reduction of features, reduction of electrodes, and frequency range have been evaluated simultaneously.
3. Feature extraction and feature selection for time domain, frequency domain, wavelet coefficients, and autoregression have been investigated as feature reduction, electrode reduction, and frequency range reduction.
4. Initially, the features, electrodes, and frequency are reduced, and then, new signals are made based on them using special formulas for each row. Then, a common spatial pattern (CSP) is performed to remove noise and feature extraction, which is followed by classification.
5. A separate study with a deep learning sampling method has been implemented as feature selection in several layers with different functions and different window sizes. This part also includes feature and frequency reduction.

In cases 1 to 4, the proposed BGA, GA, PSO, and Quantum Genetic Algorithm (QGA) are used for evaluation. In case 5, no algorithm was required, but simple computational functions were used.

The main objective of this study is dimension reduction in single and multi-domain brain signals. For this purpose, five general models (seven model scenarios) have been implemented for testing:

1. One filter bank and single channel from a set of two filter banks (i.e., 2 and 5) with two channels (i.e., right and left hemispheres of the brain) have been investigated by four main algorithms, i.e., BGA, GA, PSO, and QGA. The two channels' features

- are selected from the right and left hemispheres. The reduction of features for two channels is considered separately for each channel.
2. A combination of filter banks with a single channel from a set of four filter banks (i.e., 2, 5, 6, and 9) and two channels (i.e., right and left hemispheres of the brain) have been investigated with the same algorithms. The reduction of features for two channels is considered separately for each channel.
 3. Four general methods, including time domain, frequency domain, wavelet coefficients, and autoregression have been used for feature extraction from a combination of two filter banks (i.e., 2 and 5) as a new signal with a single channel (i.e., right or left hemispheres of the brain), two channels (i.e., right and left hemispheres of the brain), or three channels (i.e., right and left hemispheres and center of the brain). Then, feature selection is done by BGA, GA, PSO, and QGA separately.
 4. Feature selection was done by four main algorithms on a combination of filter banks (from filter banks 2, 5, and 6) and two (i.e., right and left hemispheres of the brain) or three channels (i.e., right and left hemispheres and center of the brain). Then, using special formulas, new signals were formed, which were used as input for CSP. Finally, the ELM classifier classified the extracted features from CSP.
 5. Feature selection was performed by deep learning sampling with five general functions on all filter banks together or each filter bank individually with all channels in three sampling models. In the first sampling model, each filter bank with the same functions was used in all layers. In the second sampling model, the functions were the same selected for all layers and all filter banks. In the third sampling model, the functions were randomly selected for each layer.

The rest of the article is organized as follows: In the Section 2, a summary of previous work on the dimensionality in all items is provided for brain signals. Then, in the Section 3, we review the previous related work and define the Bond Graph and how to implement the proposed algorithm and use it to reduce the dimension in brain signals and reduce the dimension by sampling. The configuration of the experiments and the data set used in this article are described in the Section 4. Finally, in the Section 5, the results of the experiments and beads are analyzed and examined. The last part presents the conclusion.

2. Previous Work

In the article [10], Adham et al. implemented specific individual and specific masks working with the CAR method, standard resource methods, and ELM. The method was a two-dimensional reduction of features and electrodes, which has achieved accuracy in the range of 15 to 32%. In the article [33], Jing Luo et al. performed the feature extraction method by analyzing the wavelet components for two channels (each channel separately) in two modes, first with a dynamic frequency feature selection method and second with a full frequency range (without frequency range selection). They obtained accuracy results of 68% and 67% for the two modes, respectively. So, the feature selection had improved their accuracy. In their work, feature selection was performed after feature extraction, which reduced the channels from 22 to two channels. Liu et al. [34] used the glow algorithm to select the features. First, feature extraction is performed by CSP for all channels and three important and specific channels individually (channels are left hemisphere, right hemisphere, and center of the brain). The GA, PSO, and a Firefly Algorithm obtained 59.85%, 60%, and 70.20% accuracies, respectively. In the article [35], Bashar Awwad used the sixth-order autoregression method with a sample window shift. They reduced the features to 20 features (characteristics) with PCA and then classified them with LDA. They obtained accuracy results between 46.8% and 59% for one channel and between 48% and 62% for two channels. They reported the average of the best accuracies in different non-identical channels as 59.67%. In the article [36], Adham et al. designed an individual and specific work mask that has been shared between individuals, so that part of the data is trained for subjects and tested for the untrained subjects. Features reduction and channel reduction were up to 90%, and the average accuracy was between 73.5% and 74.5%. In

the article [37], Nakisa et al. used frequency-temporal and temporal-frequency feature extraction. They implemented five channels with ACO, DE, GA, SA, and PSO algorithms along with the PNN classifier to select the features. The best result in this study was 65% for four classes. Peterson [38] used the MNE and Infomax preprocessing methods along with feature extraction with spectral power by selecting the feature by GA and SVM classifier. In this paper, for two classes, the number of channels has been reduced from 32 to two. The accuracy range of the subject is between 55% and 67%. By changing the method of the CSP with SVM classifier, Mahnaz Arvandej [39] obtained 70.90% accuracy for three channels, 79.07% for four to 14 channels, and 81.63% accuracy for nine to 19 channels, while the total accuracy for the standard CSP is 79.23%.

Yang et al. [40] used the time domain parameter method to extract the feature and the time-space optimizer to select the optimal channels along with the Fischer analysis classifier. When three channels were selected from 118 channels with BP and TDPS, average results of 71% and 72% were obtained, respectively. When using all the channels, the accuracy was 76%. While in their proposed model, they could increase the accuracy up to 78%. Chen et al. [41] used the method of extracting features from nine filter banks (bands) using the CSP for each filter bank. After feature selection from features collected from all the filter banks, the NBW classifier is used for classification. In their model, the reduction of channels was three or 13 out of 22 channels. The average accuracy was 75% for three specific channels and 87% for 13 channels. In the article [42] by Izabela Rejer, the BSS and PCA methods were used to select the feature. Three channels and 12 filter banks (bands) were used. However, in this study, only one person was examined. For different modes, the accuracy was between 55% and 80% for FSS and between 52% and 87% for PCA. Eslahi et al. [43] have implemented a modified feature extraction method with wavelet subbands and feature selection by GA with four classifiers. A person with three channels was used for the experiment. Results between 68% and 84% were obtained for different classifiers. Wang [44] used the method of the Warp Laser Space Group. The feature extraction is based on the statistical time domain, spectral power, autoregression, and wavelet coefficients in their study. The Warp Laser space group with strategy autoregression model has been used to select channels and features. The total results of the dataset were 83.37%, which reduced the number of channels to 17 and 18 channels, resulting in an accuracy of 84.7%. Kuman [45] used noise elimination cases, cross-correlation normalization by selecting effective channels (from the left and right nipples), and the calculation of data statistics along with ANN and SVM classifiers. Data are collected by a sensory headset device to imagine left and right finger movements. With 14 channels and 420 features, they obtained an average accuracy of 95%. With 10 features and 14 channels, they obtained an average accuracy of 96.69%. With three features and 14 channels, they obtained an average accuracy of 97.34%. In the article by Kasun Amarasinghe [46], where the sensing headset collected the data, the feature selection method was used in addition to the SVM, ANN, and NB classifiers. With 14 channels and 11 features selected for three classes and one person for NB, the accuracy was 82.97% for ANN 83.07% and 83.26% for SVM.

Chin [47] used the channel analysis method to select channels for filtering banks. Three to 14 channels were selected for filtering banks, and feature extraction was done by CSP. Channel reduction was made based on the accuracy of its valid crossover, and the feature reduction was not performed. The total average between 13 and 14 channels is 84.51%, and for three channels, it is 75%. Ang et al. [48] used feature selection from the filter bank common spatial pattern (FBCSP) method with the NBPW classifier. With the original filter bank of CSP (oFBCSP), kappa was 60.7%. With the S filter bank of CSP (sFBCSP), kappa was 61.9%. With the E filter bank of CSP (eFBCSP), kappa was 63.5%.

Wang et al. [49] applied the FDCSP frequency amplitude method. The accuracy results obtained for CSP, FDCSP, SCSP, TRCSP, WTRCSP, and FBCSP were 78%, 82.94%, 81.63%, 78.79%, 78.47%, and 79.30% respectively. Finally, Tang et al. [50] used the neural network model to extract features with five layers of CNN, with different classifiers including Power + SVM, CSP + SVM, and Power + SVM. They achieved accuracy of 81.25%, 82.61%, and

77.17%, respectively, for each classifier, and the average of the proposed method for the two subjects was 86.41%.

We summarize the advantage and disadvantages of the previous methods that we presented above:

Advantages:

- Most of them use heuristic algorithms, which achieved good performance related to their works.
- The heuristic algorithms are suitable for selecting features with or without extracting features using conventional methods.
- In all of them, two general models are used that include (1) feature selection on preprocessed data before the final classification, (2) feature selection after feature extraction on preprocessed data followed by the final classification. It has been demonstrated that the second method is more efficient.
- Most of them used continuous frequency domain for processing.
- All of the previous works only used conventional algorithms for the EEG area.

Disadvantages:

- The heuristic algorithms fail to converge very fast.
- All of them only used a large frequency domain (8 to 30 Hz) that is not effective enough for feature extraction.
- Even though the conventional algorithms are well established, there is still a need for new methods with higher performance and accuracy for processing the EEG data.
- A combination of two general models is not studied.
- Smaller continuous domains reduce the noise and distinctive features between classes and, as a result, reduce the accuracy. Whereas larger continuous domains increase the noise and distinctive features between classes and, as a result, reduce the accuracy. However, a combination of small and large frequency domains to reduce the noise and increase the distinctive features between classes is not studied.

3. Material and Methods

3.1. Bond Graph Introduction

Bond Graph presents a graphical representation of the dynamic behavior of independent domains from physical systems [51–53]. A graphical representation of the Bond Graph is similar to a flow chart diagram with a different meaning for analyzing the systems. The Bond Graph represents the state of space as a dialogue and interaction inside, outside, and between the systems. The Bond Graph uses a graphical model to show and explain details of the system and the relationship of subsystems and elements. This relationship shows the implementation of the calculation of the system for solving problems. Similar to the block diagram, it uses a single-current graph and also represents one-way information. The Bond Graph can integrate multiple domains in the best possible way.

The basis of the Bond Graph relies on bands. These bands connect a single port (port), a double port, and several ports of the elements. A band is a line connection between two and more elements, and the port is the connection point between an element and a band. The band provides energy and power in real time. Power variables consist of pairs of variables distinguished by the band's current (power is calculated based on flow and effort). These variables are flow and effort variables. For example, the variables of flow and effort are electric current and electric voltage in electrical systems, respectively. Meanwhile, the variables of flow and effort are velocity and force in mechanical systems, respectively. Figure 1 shows an example of a Bond Graph architecture for the mechanical field.

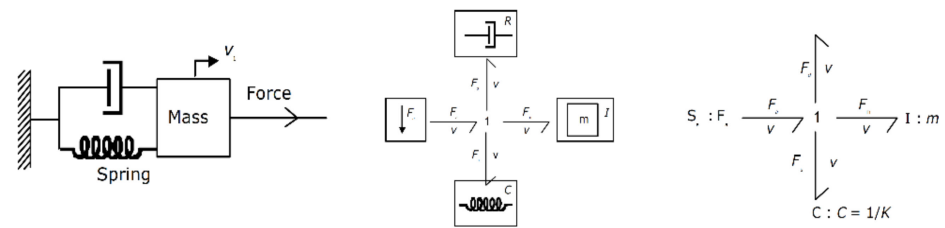


Figure 1. The following figure shows the structure of a mechanical model on the left and the Bond Graph model on the right.

The connection model of the sub-models determines the priority and superiority in the direction of computing the bands. Calculations of the port variables will be determined. The structure of the formula calculation is determined based on the connection model, which contributes to solving the problem. Bond Graphs can be combined with diagram block ports. Bond Graph models can be used as power ports, signal ports, and output signals. In the physical domain, the concept of a band (energy (electric potential) or current (electric current)) can be used to support modeling processes.

For example, in electrical networks, port variables are moved through the Bond Graph elements, the electrical voltage is moved across the element port, and the electric current is moved through the port. A port is an interface from one element to another (connecting points of bands). Power is obtained from the potential multiplication relation of the flow, which is always changing. Power is changed by the system port. A power band connection means that the band notes that the energy changes between the elements. A band is the design of the edges with the elements and notes the direction of the bands in a positive direction from the energy flow. We have a flow source that delivers power to the system, and other elements absorb power.

The Bond Graph consists of two connections.

Connection 1: The flow (current) through all connection bands is the same (the algebraic sum of the currents at the input and output nodes is zero). It means that the algebraic sum of effort (voltage) differences along one closed loop is zero (Kirchhoff’s voltage law).

Connection 0: The input effort (voltage) of the connection of all elements is the same. The effort (voltage) across all the band connections is the same, which means that the algebraic sum of the current is zero (Kirchhoff’s current law).

The variables of flow and effort are electric current and electric voltage in electrical systems, respectively. Figure 1 shows an example of a Bond Graph architecture for a mechanical field.

Table 1 describes several domains for the Bond Graph methodology that show the flow and effort variables in the various domains associated with them. The most basic variables of each domain are introduced as flow and effort variables that can be understood in different domains.

Table 1. Define flow and effect variables in different domains.

Energy Domain		$f(t)$	$e(t)$
Generalized	Name	Generalized flow	Generalized effort
	Symbol	$f(t)$	$e(t)$
Linear mechanical	Name	Velocity	Force
	Symbol	$v(t)$	$F(t)$
Electromagnetic	Name	Current	Voltage
	Symbol	$i(t)$	$V(t)$
Hydraulic pneumatic	Name	Volume flow rate	Pressure
	Symbol	$\varphi(t)$	$P(t)$

We briefly describe the graph structure of the Bond Graph below.

To produce a graph-band model, we start with an ideal physical model. In fact, there is a systematic method that we present here as a method or procedure. This procedure generally consists of identifying the basic domains and elements, generating the connection structure (called the connection structure), placing the elements, and possibly simplifying the diagram. This method is different for mechanical domains compared to other domains. These differences are expressed between the parentheses. This is because elements must be linked to different variables or between global variables. Effort variables in non-mechanical domains and velocities (flow variables) in mechanical domains are the global variables we need.

The Bond Graph includes the following steps. Steps 1 and 2 are about identifying domains and elements. Steps 3 to 5 describe the production of a connection structure, which is called a connection structure.

1. Determine which physical domains exist in the system and identify all the basic elements such as C or C-elements (storage elements such as a capacitor or spring), I or I-elements (storage elements such as an inductor or mass), R or R-elements (dissipate free energy such as resistors), SE (sources effort such as motors), SF (sources flow such as motors), TF (transformer, within the same domain (toothed wheel) or between different domains (electromotor)), and GY (gyrator such as an electromotor, a pump, or a turbine). For this purpose, each element is assigned a unique name to distinguish it from the other.
2. Specify a reference source in the ideal physical model for each domain, given that only resources are directed in the mechanical realm.
3. Identify other variables (mechanical domains: speed) and assign unique names to them.
4. Graphically design these effort variables (mechanical: velocities) rather than the source, zero connection (connection 0), and connection mechanics (connection one).
5. Identify all the effort variables (mechanical: speed difference (=current)) required for the ports of all the elements listed in step 1 of the junction structure.

3.2. Bond Graph Optimization and Algorithm

The optimization algorithm proposed in this paper is based on the bond graph methodology. Our proposed algorithm leads to the good convergence and performance of these models. The main structure of this optimizer model and some steps that have been taken to design our algorithm are presented below:

1. Determining the number and type of the physical domains (models): In our model, four specific domains with different formulas for each domain (type of domains) are introduced.
2. Combination of subdomain details: The subdomains are small linear algebraic equations that form bigger algebraic formulas (by addition of them) for calculating the changes in features. Since the small formulas are linear, they lead to the formation of bigger linear formulas. Variables of the small algebraic formulas are effective element attributes.
3. Identifying the basic elements: In our method, the best general element or best local element, the worst general element or any local element, and the average public element or average local element are introduced as basic elements.
4. Identifying all the influencers and naming them: Different combination formulas that are specific to each domain (model) are connected based on ports, and the most influential combinations are determined. Each linear algebraic addition forms a port in our model.
5. Determining the algebraic sum of coefficient of each domain: the algebraic sum of the coefficient of each domain is set to zero or one.
6. Calculation of power based on current or energy: Finally, the power is calculated based on the values obtained multiplied by random numbers. Firstly, values obtained

from formula(s) are multiplied by one random number. Secondly, new element values are calculated by adding the powers of each element and its old element value.

In our algorithm, the Bond Graph concepts are partially used with some modifications. This algorithm can be used similarly to other algorithms to optimize problems in all areas. However, unlike other algorithms, the same or different formula can be used for each dimension. Our algorithm can be customized by modification of the base formulas, domain, and effective elements for other optimization problems.

Our four domains are given below with their formulas and combinations:

All coefficients are calculated as follows.

$$\alpha = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4; \alpha = 0 \text{ or } \alpha = 1 \tag{1}$$

where $\alpha_1, \alpha_2, \alpha_3,$ and α_4 are the coefficients used in the four models.

The values of the coefficients are obtained from the distance of the identified elements or the current element, which is as follows:

$$\alpha_1 = x_{ce} - x_{gb} \tag{2}$$

$$\alpha_2 = x_{ce} - x_{glb} \tag{3}$$

$$\alpha_3 = x_{ce} - x_{nlb} \tag{4}$$

$$\alpha_4 = x_{ce} - x_{gww} \tag{5}$$

$$\alpha_{all} = \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4, \text{ if } \alpha = 1 \tag{6}$$

$$\alpha_1 = \alpha_1 / \alpha_{all}, \alpha_2 = \alpha_2 / \alpha_{all}, \alpha_3 = \alpha_3 / \alpha_{all}, \alpha_4 = \alpha_4 / \alpha_{all} \tag{7}$$

$$\alpha_{all} = \alpha_i + \alpha_j + \alpha_k; i, j, k = 1 \dots 4, i \neq j \neq k, \text{ if } \alpha = 0 \tag{8}$$

$$\alpha_d = -\alpha_{all}, d = 1 \dots 4, d \neq i \neq j \neq k. \tag{9}$$

$x_{ce}, x_{gb}, x_{glb}, x_{gww},$ and x_{nlb} are variables for the current element, global best, global–local best, global worst element, and normal local best. These variables are calculated in a similar way as in the PSO algorithm, but for normal local best (nlb), near to the average of them is selected.

In this section, four domains are considered to introduce one formula or set of formulas for updating each domain. For each domain, the combination of models is considered different. If the calculation formula for a domain is more than one, the calculation for new elements is divided between formulas, respectively. The following ranges and formulas are stated in 4, 5, 6, and 7, which are:

$$\Delta E_{Model_1} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_2 \times (x_{ce} - x_{glb}) + \alpha_3 \times (x_{ce} - x_{nlb}) + \alpha_4 \times (x_{ce} - x_{gww}) \tag{10}$$

$$\Delta E_{Model_2_1} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_2 \times (x_{ce} - x_{glb}) + \alpha_3 \times (x_{ce} - x_{nlb}) \tag{11}$$

$$\Delta E_{Model_2_2} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_2 \times (x_{ce} - x_{glb}) + \alpha_4 \times (x_{ce} - x_{gww}) \tag{12}$$

$$\Delta E_{Model_2_3} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_3 \times (x_{ce} - x_{nlb}) + \alpha_4 \times (x_{ce} - x_{gww}) \tag{13}$$

$$\Delta E_{Model_2_4} = \alpha_2 \times (x_{ce} - x_{glb}) + \alpha_3 \times (x_{ce} - x_{nlb}) + \alpha_4 \times (x_{ce} - x_{gww}) \tag{14}$$

$$\Delta E_{Model_3_1} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_2 \times (x_{ce} - x_{glb}) \tag{15}$$

$$\Delta E_{Model_3_2} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_3 \times (x_{ce} - x_{nlb}) \tag{16}$$

$$\Delta E_{Model_3_3} = \alpha_1 \times (x_{ce} - x_{gb}) + \alpha_4 \times (x_{ce} - x_{gww}) \tag{17}$$

$$\Delta E_{Model_4_1} = \alpha_1 \times (x_{ce} - x_{gb}) \tag{18}$$

$$\Delta E_{Model_4_2} = \alpha_2 \times (x_{ce} - x_{glb}) \tag{19}$$

$$\Delta E_{Model_4_3} = \alpha_3 \times (x_{ce} - x_{nlb}) \tag{20}$$

$$\Delta E_{Model_4_4} = \alpha_4 \times (x_{ce} - x_{gww}) \tag{21}$$

$$\Delta E_{Model_x} = \Delta E_{Model_x} \times rand, x = 1 \dots 4 \tag{22}$$

$$x_{New_Elements} = x_{Old_Elements} + \Delta E_{Model_x} \tag{23}$$

In each domain, the number of formulas depends on your design and the model you implement. In this case, one formula for the first domain, three formulas for the second domain, and four formulas for the third and fourth domains are used to change the values of the elements. When more than one formula is used, the distribution of elements is done randomly. It is possible to distribute all of the elements to one formula in each iteration. However, in fact, the elements distribute between different formulas. Different modes of this model can be used: (1) Using the same formulas (same range) to optimize more dimensions at runtime; (2) Using different formulas (different domains) to optimize more dimensions at runtime; and (3) Using all different formulas (different domains) to optimize more dimensions at runtime. Due to the high complexity of this algorithm, in this article, the first part of the tests is reviewed.

The pseudo-code of the Bond Graph algorithm is defined as follows (Figure 2):

- (1) Define algorithm parameters (algorithm variables, elements, element properties, etc.).
- (2) Initialize the defined parameters (initialize the properties of the elements and some fixed parameters such as selecting the domain as a constant).
- (3) Calculate the cost function based on the proposed functions (evaluating the performance of the algorithm).
- (4) Find the best, worst, and average variables locally and globally.
- (5) For the calculation of changes, the elements are divided between the formulas if there are different formulas. It is up to the user to determine the model of division of the number of elements and the order of division (the order of division is not random but in order). Finally, they are multiplied by a fixed value or random changes to make the changes a little more moderate.
- (6) These changes are applied to the values of the old elements, and new elements are obtained.
- (7) Calculate the cost function based on the proposed functions (evaluate the algorithm's performance).
- (8) Find the variables of best, worst, and average locally and globally.
- (9) Check whether the best element is general and better than the predicted value or not?
- (10) If it is better, go to step 11 and if not, go to step 5.
- (11) Execution is completed, because the best general element is the best answer to solve the problem.

We explain the differences between BGA and PSO in the following:

- Structure of main formula for calculation: In PSO, only the difference between the current particle and the local best particle and global best particle is calculated in each iteration. The coefficients are two fixed random numbers. However, in BGA, for example, in model three, three formulas are used for updating and improving the values of features. The first formula calculates the difference between the current element and the global best and global–local best, and the coefficients α_1 and α_2 are calculated separately for each iteration. The second formula calculates the difference of the current element from the global best and global worst, and the coefficients α_3 and α_4 are calculated separately for each iteration. The third formula calculates the difference between the current element and the normal local best and global worst, and the coefficients α_5 and α_6 are calculated separately for each iteration.
- Comparing of main formula for calculation: In the PSO, only one formula for calculating velocity is used. In BGA, for different elements, different formulas are used to calculate the element attribute values. The formulas are randomly chosen for the calculation of each element.

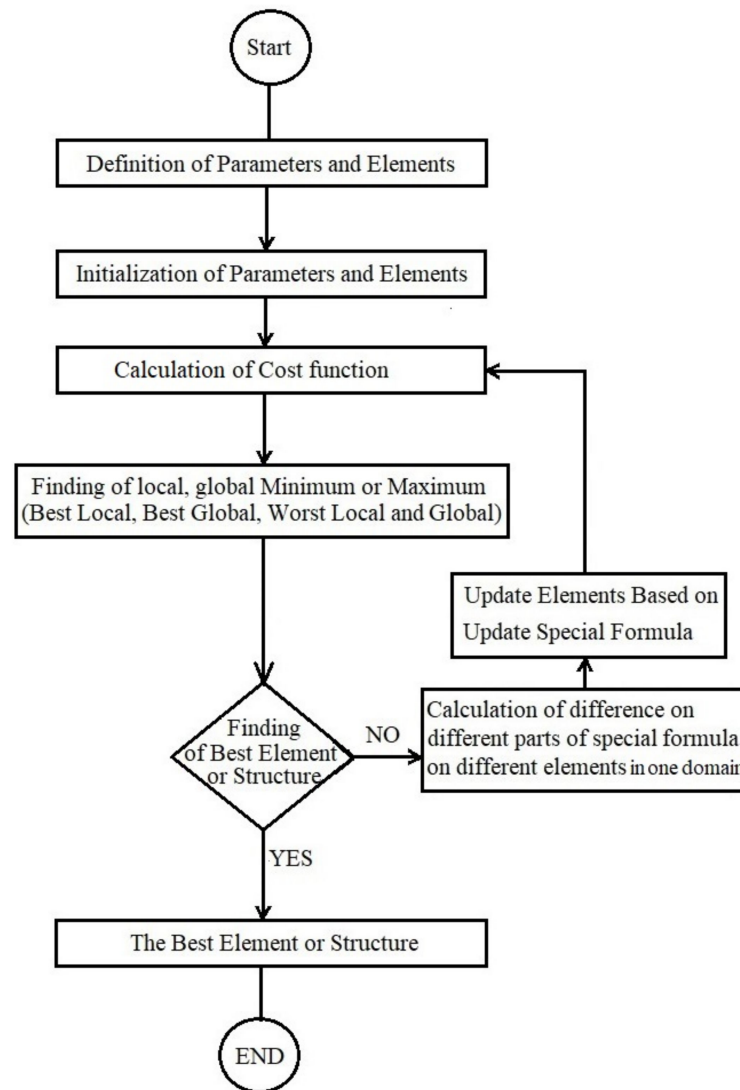


Figure 2. Overview of the Bond Graph algorithm.

3.3. Common Spatial Pattern (CSP)

The common spatial pattern algorithm (CSP) [41,47,48] is known as an efficient and effective EEG signal class analyzer. In other words, it is a feature extraction method that gives signals from several channels below the mapping space, which can maximize the difference between classes and minimize their similarities. This is accomplished by maximizing the variance of one class by minimizing the variance of another class.

The CSP calculation is done as follows

$$C = EE' / trace(EE') \tag{24}$$

where C is the covariance of the normalized space of data input E , which provides raw data from a single imaging period. E is an $N \times T$ matrix. T is the number of electrodes or channels, and N is the number of samples in the channel. Trace (.) is defined to be the sum of the elements on the main diagonal of matrix A . The apostrophe represents the transposition operator. Trace is also a set of diagonal elements of x .

The covariance matrix of both classes C_1 and C_2 is calculated by the average of several imaging periods of the EEG data, and the covariance of the combined space C_c is calculated as follows:

$$C_c = C_1 + C_2 \tag{25}$$

where C_c is real and symmetric and can be defined as follows:

$$C_c = u_c \lambda_c u_c' \tag{26}$$

where u_c is a matrix of special vectors and λ_c is the diameter of the matrix of eigenvalues. p is whitening transformation:

$$p = \sqrt{\lambda^{-1}} u_c'. \tag{27}$$

The variances in space are equalized by u_c' , and all eigenvalues of $p \bar{c}_L p^{prime}$ are equal to 1.

$$S_L = p \bar{c}_L p' \tag{28}$$

$$S_R = p \bar{c}_R p' \tag{29}$$

S_L and S_R are matrix covariance with eigenvector and eigenvalue, provided that if

$$S_L = B \lambda_L B' \tag{30}$$

$$S_R = B \lambda_R B' \tag{31}$$

$$\lambda_L + \lambda_R = I, \tag{32}$$

I is the identity matrix.

Eigenvalues are arranged in descending order, and the projection matrix is defined as W :

$$W = U^T P. \tag{33}$$

The reflection matrix of each training is as follows:

$$Z = W \times i \tag{34}$$

where N rows are selected to represent each period of conception $W_p (p = 1, 2, \dots, N)$ and the covariance P of Z , P components of the feature vectors are calculated for the n th instruction. Normalized variance is used as the algorithm:

$$f_p = \log \left(\text{Var}(Z_p) / \sum \text{Var}(Z_p) \right). \tag{35}$$

3.4. Spatial and Temporal Frequency Domains

This section deals with the formulas of time-domain, frequency-domain, autoregression, and wavelet coefficients because most of the articles explain these methods in detail. In our paper, the formulas are used from papers [28,44]. In the following, we explain the formulas briefly.

These extracting features formulas from 36 to 53 and 54 to 57 belong to the time-domain and frequency-domain, and other formulas that belong to other methods are mentioned above. For wavelet coefficients, the MATLAB function is used. Based on algorithms, features are selecting from different formulas. Using algorithms, the features of some formulas are selected, and the rest are not selected.

$$\text{Absolute values sum} = \sum_{i=1}^N |x_i| \tag{36}$$

$$\text{Mean absolute Value, MAV} = \frac{1}{N} \sum_{i=1}^N |x_i| \tag{37}$$

$$\text{Modified mean absolute value type 1, MAV}_{Type 1} = \frac{1}{N} \sum_{i=1}^N w_i |x_i| \tag{38}$$

$$w_i = \begin{cases} 1, & \text{if } 0.25N \leq i \leq 0.75N \\ 0.5, & \end{cases}$$

$$\text{modified mean absolute value type 2, MAV}_{Type 2} = \frac{1}{N} \sum_{i=1}^N w_i |x_i| \tag{39}$$

$$w_i = \begin{cases} 1, & \text{if } 0.25N \leq i \leq 0.75N \\ \frac{4i}{N}, & \text{slseif } i < 0.25N \\ \frac{4(i-N)}{N} & \end{cases}$$

$$\text{Simple square integral, SSI} = \sum_{i=1}^N x_i^2 \tag{40}$$

$$\text{Variance, } Var = \frac{1}{N-1} \sum_{i=1}^N x_i^2 \tag{41}$$

The absolute value of the 3rd, 4th, and 5th temporal moment

$$TM3 = \left| \frac{1}{N} \sum_{i=1}^N x_i^d \right|, d = 1, 2, 3. \tag{42}$$

$$\text{Root mean square, } RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \tag{43}$$

$$\text{Waveform length, } WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \tag{44}$$

$$\text{Average amplitude change, } AAC = \frac{1}{N} \sum_{i=1}^{N-1} |x_{i+1} - x_i| \tag{45}$$

Difference absolute standard deviation value,

$$DASDV = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N-1} (x_{i+1} - x_i)^2} \tag{46}$$

$$\text{Maximum value, } Max = \max(x_i), i = 1, 2, \dots, N \tag{47}$$

$$\text{Minimum value, } Min = \min(x_i), i = 1, 2, \dots, N \tag{48}$$

$$\text{Standard deviation, } SD = sd(x_i), i = 1, 2, \dots, N \tag{49}$$

$$\text{Mean frequance, } MDF = \frac{1}{2} \sum_{j=1}^M p_j \tag{50}$$

$$\text{Peak frequance, } RKF = \max(P_j) \tag{51}$$

$$\text{Mean power, } MNP = \sum_{j=1}^M \frac{P_j}{M} \tag{52}$$

$$\text{Total power, } TTP = \sum_{j=1}^M p_j \tag{53}$$

$$\text{Autoregressive coefficients, } x_i = \sum_{i=1}^p a_p x_{i-p} + w_i, p \text{ is order of AR} \tag{54}$$

$$\text{Power Spectrum } S_{AA}(f) = \frac{FFT(A) * FFT'(A)}{N} \tag{55}$$

$$ps[m] = \sum_{n=1}^N r_{xx}[n] e^{-j \times 2\pi i m n} N, m = 0, 1, 2, \dots, N \tag{56}$$

$r_{xx}(t)$ and $r_{xx}(n)$ are autocorrelation functions.

$$\text{Coefficient Wavelet } \phi_{j+1,2i+1}(x) = \frac{1}{\sqrt{2N_{j+1}}} \left(\sum_{s=0}^{2N_j-1} \phi_{j,s} \left(\frac{(2l+1)\pi}{2N_j} \right) \phi_{j,s}(x) - \sum_{s=0}^{2N_j-1} \phi_{j,l} \left(\frac{(s)\pi}{N_j} \right) \Psi_{j,s}(x) \right), \tag{57}$$

$$l = 0, 1, \dots, 2N_j-1$$

3.5. Reduce and Extract Features of Brain Signals with Filter Banks and Bands (FBs)

In some studies, feature and electrode reduction methods have been used in combination or separately to process information to reduce data volume. GA and PSO or some other algorithms have been implemented for each person or individual to reduce the dimensions before extracting the features or selecting the features after extracting the features or as the classifiers for classification separately. In this paper, our proposed method is inspired by different methods to reduce the dimensions, while the proposed method is somewhat different from the methods presented in some articles. In a way, it has partial similarities with the items mentioned in the previous. First, in all our models, bandwidth

reduction (filter bank) has been used to extract the feature. Second, in some cases, the reduction of features and electrodes has also been used. Third, selecting the feature is done after extracting the features. For this purpose, we implemented different models, which support all items. These models (five general models) are as follows:

- (1) The selection of features is made through a filter bank with a specific channel. Two filter banks and two channels are used for the purpose (separately for individuals by selecting two filters and two specific channels).
- (2) The selection of features is made through a hybrid filter bank with a specific channel, and a new signal from two filter banks along the channel is used for this purpose. (Separately for a person with a combined signal and two specific channels).
- (3) Reduction of features for two channels with two filter banks (in the range of 8 Hz) is intended for frequency reduction. We have a 63% reduction in the frequency range; i.e., from the whole frequency domain, which is 8 to 30 Hz, we have used a combination of smaller frequency ranges. For example, 8 to 12 Hz and 20 to 24 Hz represent a combination of filter banks 2 and 5. Then, we create a signal matrix with dimensions of 10×100 for each imaging period (approximately 1.33% of the value of imaging period characteristics for a channel) that is used as input to CSP to filter and extract features. This is followed by a classification of features.
- (4) Features reduction for three channels and three bank or band filters (12 Hz interval) has been used to reduce the frequency. From the 22 Hz range (8–30 Hz), the 12 Hz range is selected (45% reduction in the frequency range). Then, a signal matrix of 18×100 is created for each imaging period (generally generated for all imaging periods), which is 2.4% of one channel. The total features are generally slightly more than two channels. Then, the extracted properties are used as two-dimensional matrices as input to the CSP to filter and extract the properties. After extracting the features, classification is done on them.
- (5) Feature extraction from 1 to 3 channels is done separately (for each channel, feature extraction and feature selection are made separately). Time domain, frequency domain, wavelet coefficients, and autoregression coefficients were used to extract the features. From each feature extraction method, a certain amount is considered for feature selection. In other words, four parts of features (features related to each proposed method) are used for each channel separately. Feature extraction and feature selection have been implemented for two channels, 8 parts, and for 3 channels, 12 parts. The amount of feature selection is determined separately from each method and channel, and all features from all channels and methods are used for classification.

In all the stated cases, 100 features are considered for the channel using algorithms. In other words, the reduction is done from 875 to 100 features. To evaluate the five models mentioned above, we utilized the proposed BGA, PSO, GA, and QGA with ELM classifier for classification.

If more than one channel is used, the ability to detect 100 features is the same on all channels. The coefficients determine the location of the features. The values of coefficients of the elements in our algorithm, chromosomes of genes in GA and QGA, and attributes of particles in PSO between zero and one change during the execution of the algorithm. However, in order to improve the accuracy in the fifth model, the coefficients related to the time domain and autoregression are set to binary values.

For models one and two, filter banks 2 and 5 and channels c3 and c4 have been used, and their evaluation has been performed on four algorithms, namely the proposed algorithm, PSO, GA, and QGA. However, for the rest of the models, the ELM classification and the proposed algorithm have been evaluated on the filter banks and individuals, etc. Figure 3 shows an overview of the model for better expression and understanding in detail, which is described in more detail in the experimental section. In this figure, BGA is used as the main algorithm. The other three algorithms (GA, PSO, and QGA) are used similarly.

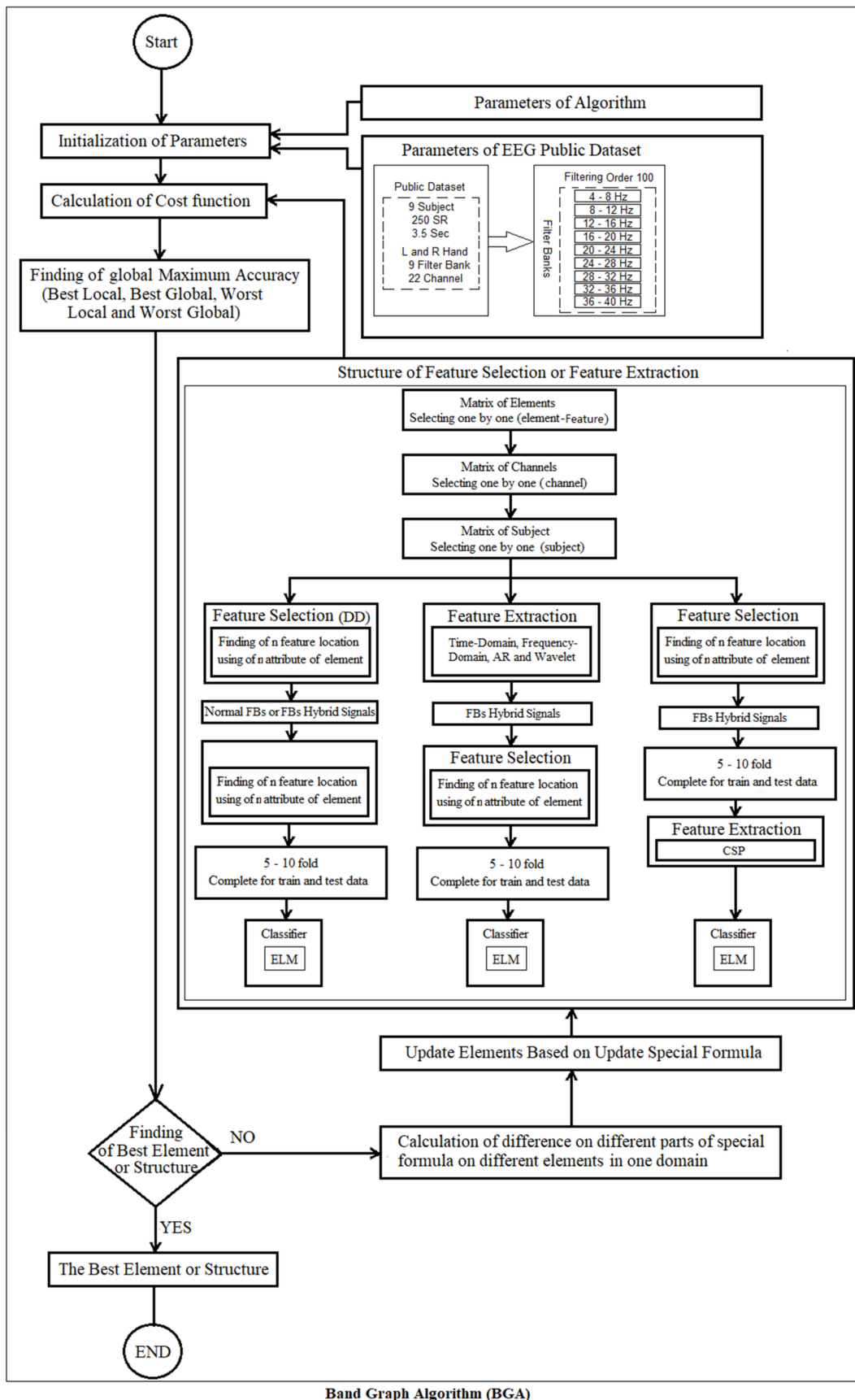


Figure 3. The overview of the dimension reduction using the BGA algorithm and proposed methods for feature selection and feature extraction.

3.6. Reduce Features Using Pooling for Channels

The key idea is to make a sampling that happens randomly at each layer of complexity (deep learning). Common forms of complex (deep learning) sampling are a function of the mean and maximum determinants that select the largest activity in each sampling area. In a random sample, the activity selection consists of a polynomial explanation drawn by the activities in the sampling area. A secondary view of random sampling is similar to standard maximal sampling but with the copying of input images, each of which has short local variations. This is similar to the explicit elastic deformation of input images, which provides excellent performance for some datasets. In addition, the use of random sampling in a multilayer model gives a large number of changes compared to the higher layers. In random sampling, we select the pool mapping response by sampling a distributed polynomial description of activities from each pool area. In maximum sampling, only the strongest activity is taken from the time filters with input from each area, and whether the rest of the activities have any effect is not considered. For this purpose, random sampling is implemented for activities when maximum activities will be useful. You can see an example of this sampling model in Figure 4 [54–56].

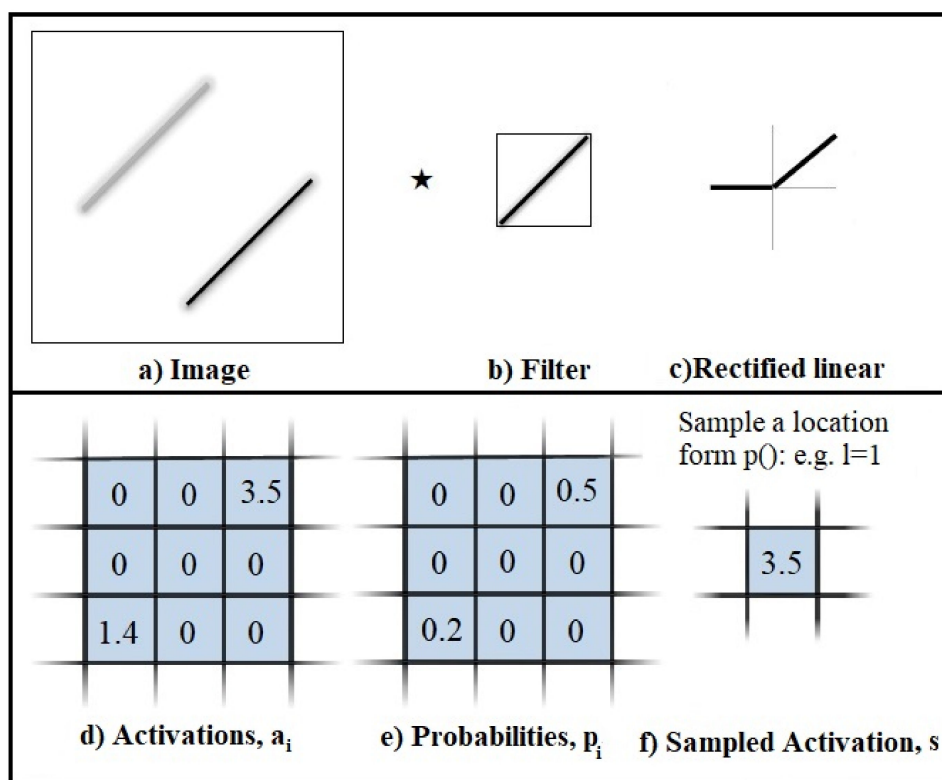


Figure 4. The example of random sampling for deep learning. (a) Image. (b) Filter. (c) Rectified linear. (d) Activation. (e) Probabilities. (f) Sampled Activation.

In some papers, convolutional layers accept arbitrary input sizes. However, they produce variable output sizes. Classifications and artificial neural layers need to be connected to fixed vectors. The spatial pyramid sampling improves the previous method to store space information by sampling in local space depots. Therefore, these depots are proportionate in size. They are relative to the image.

In this paper, we introduce random sampling to select features randomly and statistically. This is because sampling methods are considered in all layers. Each layer reduces the number of features. After the last layer of classification, it begins to gain accurate classification. Most methods use GA and particle cluster optimization to reduce the dimension that the algorithms use. In our study, the feature reduction model is based on deep learning sampling. Deep learning sampling can be used for samples with any channel, i.e., hybrid

channels or individuals. In this article, sampling is done in two ways: (1) for all channels and features for each filter bank individually, and (2) for all channels and features for all filter banks altogether.

In addition to the functions used in older models, i.e., maximum, minimum, and average, two new functions are introduced in this article. The first is the intelligent selection function that selects the maximum or minimum, or average based on a specific formula. In this case, first, we find the average of all the features of the window; then, we calculate the absolute value of the difference between their distance and the average. The maximum value is the selected value. The second function is calculating the average without the participation of the maximum value.

In the following sections, we briefly present a structure of the sampling layer model with classification for our idea:

- (1) In the filter input of banks in general, no preprocessing has been done. After the record of subject, data, or domain, the specified frequencies are filtered.
- (2) Select the size of different molds (from 2 to 6) for each sampling layer (fixed for each layer). However, the active layers are three layers for models with sizes 2 to 4 and five layers for models with sizes 5 to 6.
- (3) Selection of sampling functions for each layer (maximum, minimum, average function, smart maximum or minimum selection, average function without considering the maximum). All sampling models are supported for all layers. Two sampling models for layers are used. First, the layers use only one identical function. Second, the layers use different functions.
- (4) The output of each layer is the input of the next layer: according to the combination of channels and features, the size of the template is a vector, and the output will be one page. Dimensional reduction occurs for features separately.
- (5) Perform steps 2 to 4 for all layers.
- (6) The outputs from the last layers all add up and form a vector (all channels). For classification, if the output is two-dimensional (more than one channel), it is converted to a one-dimensional matrix to send classifiers.
- (7) Selection of training and testing model for classification (10–10 fold method on total data): The whole data set is divided into ten parts. One part is used for testing and the rest are used for training. The whole part of the steps related to testing this model is repeated ten times.
- (8) Classification is done in two classes. The results are different for each person. The results of the first random classification are discussed later in this paper.

An overview of our proposed model for reducing the channel dimension is presented in Figures 5 and 6.

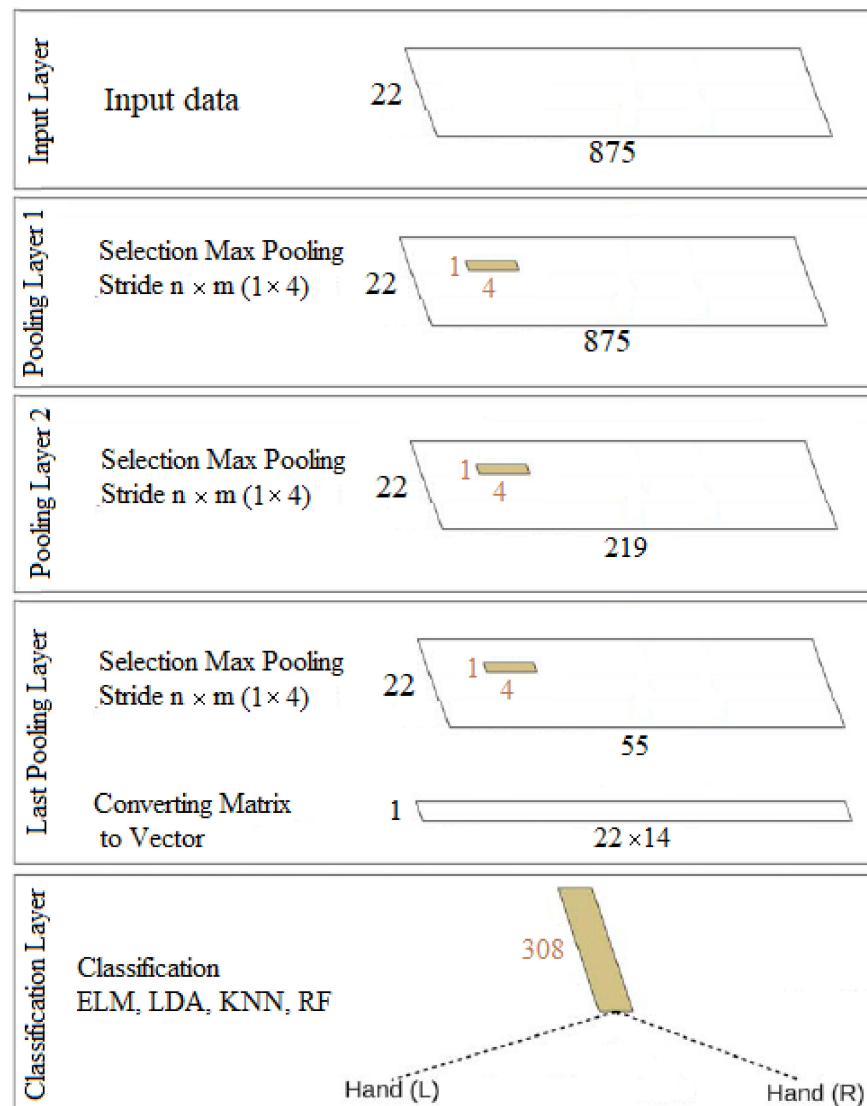


Figure 5. The steps of sampling in different layers using functions.

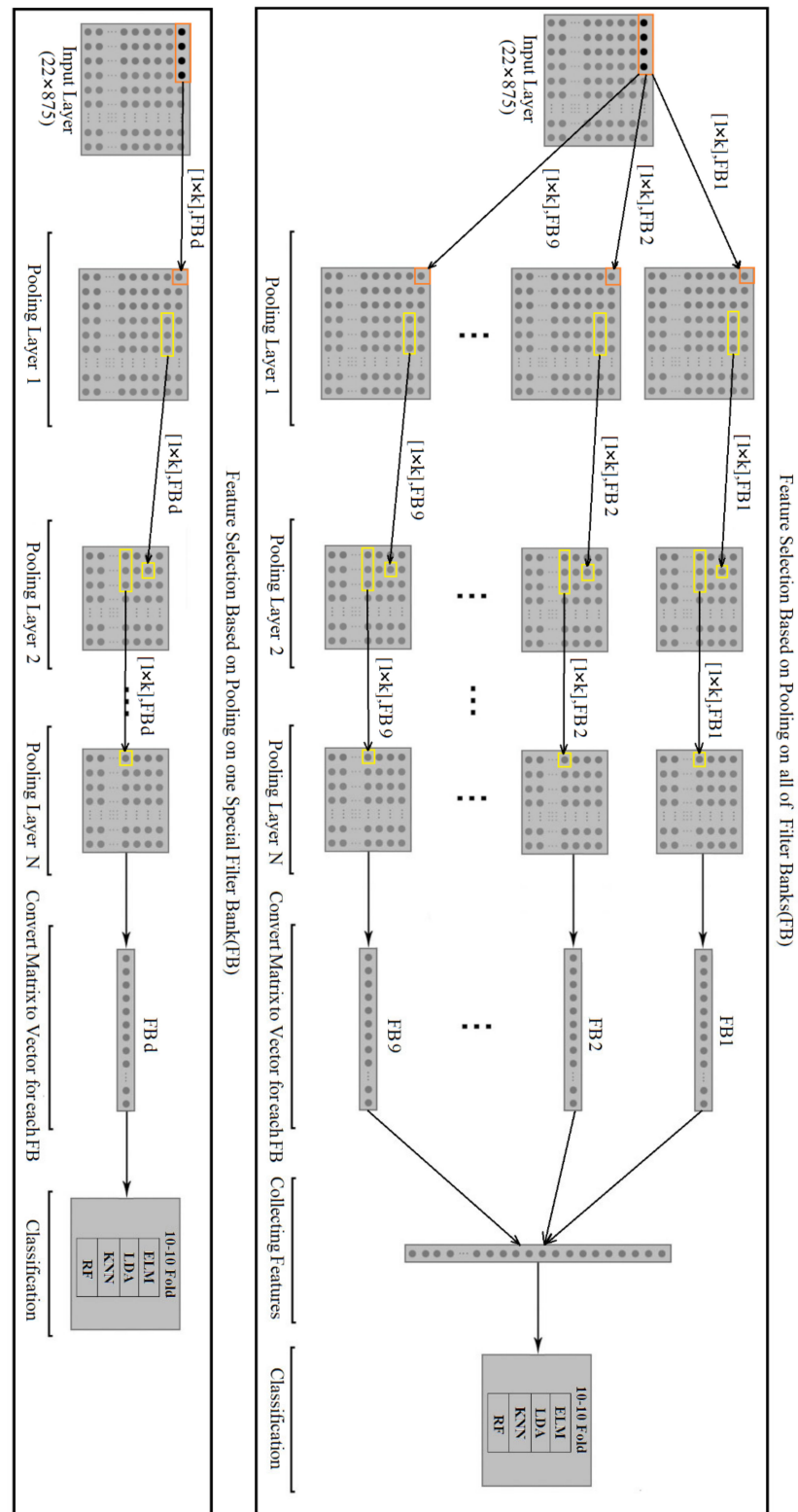


Figure 6. Reduction of channel dimensions based on the deep learning sampling model on a single filter bank or the entire filter banks.

4. Experiment and Results

4.1. Case Study and Numerical Results on Proposed Algorithm

The parameters of the benchmark functions are shown in Table 2. These are 8 general functions for testing optimization regions. Our algorithm with all of the models (domains)

along with GA and PSO [57,58] are run in MATLAB 2016 software on PC, and a comparison is performed. All details of the test scenarios for the algorithms are briefly described in Table 3 [59].

Table 2. Parameters of the benchmark function.

Function	Dimension	Initial Range	Minimum
Sphere	35	$[-100, 100]^D$	0
SumSquares	35	$[-10, 10]^D$	0
Rosenbrock	35	$[-30, 30]^D$	0
Schwefel 2.22	35	$[-10, 10]^D$	0
Rastrigin	35	$[-10, 10]^D$	0
Schwefel	35	$[-500, 500]^D$	-12,569.50
Ackley	35	$[-32.768, 32.768]^D$	0
Griewank	35	$[-600, 600]^D$	0

Table 3. Parameters of the algorithm setting.

Function	Population	Details
GA	40	Single point crossover (0.8), mutation rate (0.01), generation gap (0.9)
PSO	40	Inertia weight (0.6), cognitive and social components (1.8)
BGA	40	Selection model 1 to 4, an average of coefficients (1), initial element (2.5%)

4.2. Bond Graph Algorithm (BGA)

Compared to other algorithms, the BGA for the benchmark functions is rapid convergence and near to the global optimum. In other words, it avoids falling into local traps (the main targets are ranges 3 and 4). It is demonstrated that the performance of this model is better than PSO and GA. Figure 7 shows the convergence diagram and the effect of the BGA compared to the other two algorithms. The convergences of the two benchmark functions over time for better understanding are illustrated in Figure 7.

Domains 3 and 4 had excellent results near the global optimum. However, the first two domains in our definition had the worst convergence, which needs to be improved (Figure 7 down). Table 4 shows the comparison of 35-dimensional results of GA and PSO and all of the models of the BGA. From this table, we can conclude that the design of the formulas and variables used is very important and influential in convergence.

The BGA optimization algorithm, similar to PSO, and GA is linear with order $O(n)$. Only one loop is used for calculation. BGA is six times more time consuming than PSO and half as time consuming as GA. For example, the average execution time for 400 iterations and 30 executions on Spheres (first bench mark) for GA, BGA, and BGA was 0.83 s, 0.42 s, and PSO 0.07 s, respectively.

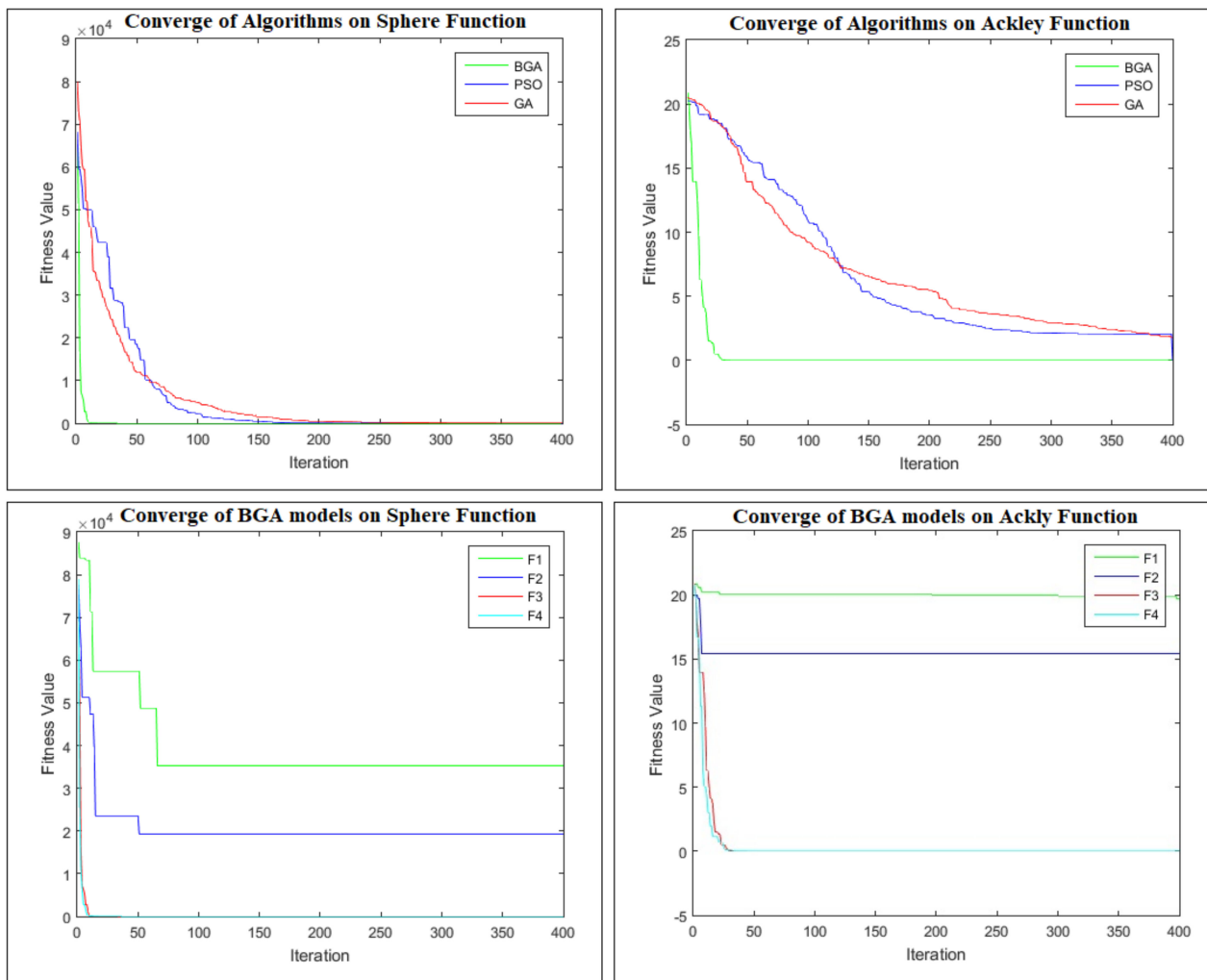


Figure 7. (Top-Left) Convergence of GA, PSO and BGA (third model) on sphere function during 400 steps. (Top-Right) Convergence of GA, PSO, and BGA (third model) on Ackley function during 400 steps. (Down-Left) Convergence of all of BGA models (four models (domains)) on sphere function during 400 steps. (Down-Right) Convergence of all BGA models (four models) on Ackley function during 400 steps.

Table 4. Comparison of 35-dimensional results of GA and PSO and all of the models of the BGA.

Public Function	Parameters	PSO	GA	BGA_Model1	BGA_Model2	BGA_Model3	BGA_Model4
Sphere	Mean	4.72×10^{-2}	43.41	$4.20 \times 10^{+4}$	$1.60 \times 10^{+4}$	5.40×10^{-110}	1.56×10^{-90}
	Std	5.64×10^{-2}	13.74	$5.44 \times 10^{+3}$	$1.11 \times 10^{+4}$	2.00×10^{-109}	7.43×10^{-90}
	Min	2.45×10^{-3}	22.87	$2.43 \times 10^{+4}$	$4.28 \times 10^{+3}$	8.10×10^{-120}	1.40×10^{-102}
	Max	2.24×10^{-1}	76.58	$4.89 \times 10^{+4}$	$4.82 \times 10^{+4}$	1.00×10^{-108}	4.14×10^{-89}
SumSquares	Mean	6.89×10^{-3}	7.15	$8.53 \times 10^{+3}$	$3.04 \times 10^{+3}$	3.80×10^{-111}	3.92×10^{-90}
	Std	9.46×10^{-3}	2.02	$1.40 \times 10^{+3}$	$1.35 \times 10^{+3}$	1.70×10^{-110}	2.17×10^{-89}
	Min	5.11×10^{-4}	3.09	$5.06 \times 10^{+3}$	$8.64 \times 10^{+2}$	1.10×10^{-123}	2.50×10^{-100}
	Max	4.39×10^{-2}	11.20	$1.07 \times 10^{+4}$	$5.86 \times 10^{+3}$	9.30×10^{-110}	1.21×10^{-88}
Rosenbrock	Mean	$2.48 \times 10^{+3}$	$1.78 \times 10^{+3}$	$1.22 \times 10^{+8}$	$2.02 \times 10^{+7}$	33.91	33.93
	Std	$1.07 \times 10^{+4}$	$1.07 \times 10^{+3}$	$3.21 \times 10^{+7}$	$2.17 \times 10^{+7}$	0.05	0.06
	Min	58.93	$8.28 \times 10^{+2}$	$4.82 \times 10^{+7}$	$1.16 \times 10^{+7}$	33.77	33.75
	Max	$5.89 \times 10^{+4}$	$5.40 \times 10^{+3}$	$1.71 \times 10^{+8}$	$1.11 \times 10^{+8}$	33.97	33.98

Table 4. Cont.

Public Function	Parameters	PSO	GA	BGA_Model1	BGA_Model2	BGA_Model3	BGA_Model4
Schwefel 2.22	Mean	0.19	2.04	$3.71 \times 10^{+11}$	$4.21 \times 10^{+3}$	3.80×10^{-56}	7.44×10^{-46}
	Std	0.35	0.28	$1.33 \times 10^{+12}$	$2.29 \times 10^{+4}$	1.40×10^{-55}	1.59×10^{-45}
	Min	0.01	1.58	$1.33 \times 10^{+2}$	34.77	5.31×10^{-61}	9.97×10^{-52}
	Max	1.60	2.59	$5.91 \times 10^{+12}$	$1.27 \times 10^{+5}$	7.65×10^{-55}	5.47×10^{-45}
Rastrigin	Mean	$1.06 \times 10^{+2}$	20.52	$8.36 \times 10^{+2}$	$4.67 \times 10^{+2}$	0.00	0.00
	Std	18.95	2.87	$1.02 \times 10^{+2}$	$1.00 \times 10^{+2}$	0.00	0.00
	Min	59.34	14.04	$6.12 \times 10^{+2}$	$2.90 \times 10^{+2}$	0.00	0.00
	Max	$1.42 \times 10^{+2}$	26.09	$9.83 \times 10^{+2}$	$6.59 \times 10^{+2}$	0.00	0.00
Schwefel	Mean	$-1.00 \times 10^{+50}$	$-2.54 \times 10^{+3}$	$-7.84 \times 10^{+3}$	$-7.99 \times 10^{+3}$	$-8.06 \times 10^{+3}$	$-7.90 \times 10^{+3}$
	Std	$1.95 \times 10^{+50}$	$4.84 \times 10^{+2}$	$2.83 \times 10^{+2}$	$2.31 \times 10^{+2}$	$4.85 \times 10^{+2}$	$5.41 \times 10^{+2}$
	Min	$-7.40 \times 10^{+50}$	$-3.82 \times 10^{+3}$	$-8.65 \times 10^{+3}$	$-8.44 \times 10^{+3}$	$-9.30 \times 10^{+3}$	$-1.01 \times 10^{+4}$
	Max	$-5.80 \times 10^{+50}$	$-1.78 \times 10^{+3}$	$-7.32 \times 10^{+3}$	$-7.62 \times 10^{+3}$	$-7.22 \times 10^{+3}$	$-7.07 \times 10^{+3}$
Ackley	Mean	1.49	2.53	19.36	13.97	-8.90×10^{-16}	-8.90×10^{-16}
	Std	0.75	0.33	0.54	2.81	0.00	0.00
	Min	0.05	1.81	18.25	7.93	-8.90×10^{-16}	-8.90×10^{-16}
	Max	3.71	3.07	20.01	18.15	-8.90×10^{-16}	-8.90×10^{-16}
Griewank	Mean	0.10	1.36	$3.64 \times 10^{+2}$	$1.49 \times 10^{+2}$	0.00	0.00
	Std	0.16	0.09	52.38	70.58	0.00	0.00
	Min	0.01	1.20	$2.91 \times 10^{+2}$	32.94	0.00	0.00
	Max	0.83	1.54	$4.68 \times 10^{+2}$	$2.81 \times 10^{+2}$	0.00	0.00

4.3. Experiments and Scenarios

In this study, data set IIa from BCI competition IV [60] is used for our experiments. This database contains the following details: (1) The number of participants for the test consists of nine subjects. (2) Brain information record electrodes include 22 channels (processing is related to these channels). (3) The training and evaluation data for each participant consist of one session. (4) Each session contains 288 training (including movement imaging activities). (5) The training time frame of each training session is 3 s (in our study, we consider 3.5 s, out of which half a second is considered for the preparation for training). (6) It includes four classes, including the right-hand, left-hand, tongue, and foot. (We have only used two classes of right-hand and left-hand in our experiments, similarly to the studies done in the previous works in the literature.) (7) A Butterworth filter is used with 100th order for nine filter banks, including 4–8 Hz, 8–12 Hz, 12–16 Hz, 16–20 Hz, 20–24 Hz, 24–28 Hz, 28–32 Hz, 32–36 Hz, and 36–40 Hz. The details of the data set are presented in Table 5 [60].

Table 5. The details of data set IIa from BCI competition IV.

Function	Dimension	Initial Range
Volunteers (subjects)	9	9
Electrodes (channels)	22	22
Session–Trials	1–228	1–228
Imagination Time–Sample Rate	4 s–250	After 0.5 s, 3.5 s–250
Each Trial Features	1000	875
Classes	Left–right hand, foot, and tongue	Left–right hand
Filtering	-	9 filter banks
Order of filtering	-	100

In our experiments, details of the new optimizer paradigm are described in Section 3 for the production of models. This is similar to most heuristic-based algorithms for finding

solutions to problems by reducing the dimension for different modes and extracting each case individually. In this approach, a set of features in the elements is used to represent the location of features in the channels.

In the following, we present the details of the models (seven model scenarios) used in our experiments for dimension reduction before the feature extraction (three-dimensional reduction) and the models that are used for dimension reduction after feature extraction:

1. The first model scenario: a filter bank with one channel for each participant is examined by four algorithms with ELM [61,62] classification. Two filter banks, i.e., 2 and 5 with two channels, i.e., 8 and 12 which are representing C3 and C4, respectively, are used.
2. The second model scenario: a new hybrid filter (combination of two, three, or four filter banks) with one channel with four algorithms with ELM classification is examined for each participant. A hybrid filter bank with two channels, i.e., 8 and 12, representing C3 and C4, respectively, is used.
3. The third model scenario: In this model scenario, we first extract the features from channel 8 (C3) or 12 (C4) according to Table 6. Then, from the extracted features, the feature selection is done with graph algorithms, genetics, particle clustering, and quantum genetics (Table 6), and then, the classification is done.
4. The fourth model scenario: In this mode, first, feature extraction from two channels 8 (C3) and 12 (C4) according to Table 6 is performed. Then, from the extracted features, feature selection (i.e., dimension reduction) is done by the specified algorithms. In this approach, the selection of features for each method and each channel is selected separately (Table 6), and their set is sent to the classifier for classification.
5. The fifth model scenario: This model scenario is the same as the previous model scenario except that in this model scenario, we used three channels. First, feature extraction is done from three channels 8 (C3), 10 (CZ), and 12 (C4), according to Table 6. Then, from the extracted features, feature selection (i.e., dimension reduction) is done by the specified algorithms. In this case, we have four methods and three channels forming 12 parts. From each part, the best features are selected for the classification according to Table 6. We used the ELM classifier for classification.
6. The sixth model scenario: In this model scenario, first, the features of channels 8 and 12 are reduced by 100 features (100 features are selected for each channel). These features are selected from the frequency domain of 8–12 Hz, 20–24 Hz, and 24–28 Hz that are within the general range of 8–30 Hz. Subsequently, for both of them, using specific formulas, a matrix of 10×100 is prepared for each period of conception as a part of CSP. The extracted components of CSP are equal to $m = 5$ (10 features). After the feature extraction was done for all imaging periods, the classification operation is performed on it. The formulas of each row in the CSP input matrix are as follows:

$$R_1 = R_{C3 \& FB5} \quad (58)$$

$$R_2 = R_{C3 \& FB6} \quad (59)$$

$$R_3 = R_{C4 \& FB5} \quad (60)$$

$$R_4 = R_{C4 \& FB6} \quad (61)$$

$$R_5 = R_{C3 \& FB5} + R_{C3 \& FB6} \quad (62)$$

$$R_6 = R_{C4 \& FB5} + R_{C4 \& FB6} \quad (63)$$

$$R_7 = R_{C3 \& FB5} + R_{C4 \& FB5} \quad (64)$$

$$R_8 = R_{C3 \& FB6} + R_{C4 \& FB6} \quad (65)$$

$$R_9 = R_{C3 \& FB5} + R_{C4 \& FB6} \quad (66)$$

$$R_{10} = R_{C3 \& FB6} + R_{C4 \& FB5} \quad (67)$$

where R is a row of the CSP input matrix. For R_1 to R_4 , the values are considered from a specific electrode and frequency domain (one filter bank). For the rest, each row is calculated by the addition of two selected electrodes and frequencies (two filter banks).

- Seventh model scenario: This model scenario is similar to model scenario 6, but in this model scenario, first, the features are selected from three channels, i.e., 8, 10, and 12, that are reduced by 100 features. These features are selected from the frequency domain of 8–12 Hz, 20–24 Hz, and 24–28 Hz that are within the general range of 8–30 Hz. After that, for both of them, using specific formulas, a matrix of 18×100 is prepared for each period of conception as a part of CSP. Extracted components of CSP are equal to $m = 5$ (10 features). After the feature extraction was done for all imaging periods, the classification operation is performed on it. These formulas are the same as the previous ten formulas in Part 6, to which eight new formulas have been added. The formulas are as follows:

$$R_{11} = R_{C3 \ \& \ FB2} \tag{68}$$

$$R_{12} = R_{C4 \ \& \ FB2} \tag{69}$$

$$R_{13} = R_{CZ \ \& \ FB2} \tag{70}$$

$$R_{14} = R_{CZ \ \& \ FB5} \tag{71}$$

$$R_{15} = R_{CZ \ \& \ FB6} \tag{72}$$

$$R_{16} = R_{C3 \ \& \ FB6} + R_{C4 \ \& \ FB2} \tag{73}$$

$$R_{17} = R_{C3 \ \& \ FB5} + R_{CZ \ \& \ FB6} \tag{74}$$

$$R_{18} = R_{CZ \ \& \ FB5} + R_{C4 \ \& \ FB6} \tag{75}$$

Table 6. Selection features of all features based on methods and channels (C3 = 8, CZ = 10, and C4 = 12).

Channels/Type Features	Time and Frequency Formulas		Autoregression		Coefficients Wavelet		Fast Fourier Transformation	
	Selected Features	All Features	Selected Features	All Features	Selected Features	All Features	Selected Features	All Features
Channel C3	8/12	20	6	10	20	200	30	200
Channel CZ	10/10	20	6	10	20	200	30	200
Channel C4	12/15	20	6	10	20	200	30	200

The following are the sampling models used to reduce the dimension based on deep learning sampling:

- The first sampling model: Dimensional reduction based on deep learning sampling for each filter bank and participant is done separately with different window sizes and different functions. The same function is used in the layers.
- The second sampling model: Dimensional reduction based on deep learning sampling for all filter banks together and participants separately is done with different window sizes and different functions. The same function is used in the layers.
- Third sampling model: Dimensional reduction based on deep learning sampling for all filter banks together and participants separately is done with different window sizes and different functions. Different functions randomly selected with non-uniform distribution are used in each layer.

Two diagnostic measurements, i.e., accuracy and kappa, are considered for the analysis of each mental task.

4.4. Results on Algorithms

4.4.1. Results of First Model Scenario

Table 7 shows the results of a dimensional reduction by 100 features performed by four algorithms on filter bank number 2 (frequency range was 8 to 12). The average accuracies for channel 8 were calculated as 61.1%, 61.0%, 61.3%, and 60.0%, respectively, and for channel 12, they were calculated as 61.3%, 61.7%, 62.3%, and 61.4%, respectively for four algorithms, i.e., PSO, GA, BGA (with the third model), and QGA [63,64]. The overall best accuracy for the channels 8 and 12 are highlighted for each algorithm. The

average accuracy of our proposed algorithm is better than the others. In channel 8, our algorithm demonstrated a slightly better accuracy compared with PSO and GA (i.e., 0.1%) and outperformed compared with the QGA by about 1%. In channel 12, our algorithm outperformed other algorithms by about 1–2%. However, for some test subjects, the highest accuracy was not obtained by our proposed algorithm. For example, in channel 8, the highest accuracy was obtained by GA for the first, seventh, and ninth subjects, i.e., 62.2%, 61.4%, and 60.8%, respectively. However, in the same channel, our algorithm had better accuracy for the second, third, fifth, and eighth subjects, i.e., 61.7%, 61.9%, 60.8%, and 60.3%, respectively. In channel 12, our proposed algorithm had the highest accuracy for the first, second, third, seventh, eighth, and ninth subjects, i.e., 60.5%, 60.3%, 63.0%, 66.6%, 60.4%, and 62.5%, respectively.

Table 7. Feature selection accuracy of PSO, GA, BGA, and QGA with 100 features, FB2, and ELM for 100 iterations.

Subject	PSO		GA		BGA		QGA	
	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12
1	60.3	59.1	62.2	59.0	60.0	60.5	59.1	59.8
2	60.9	58.9	60.3	60.3	61.7	60.3	58.8	59.5
3	60.5	61.6	61.3	62.4	61.9	63.0	59.8	61.7
4	65.5	64.5	63.7	64.3	64.2	65.0	63.2	63.8
5	60.1	62.3	58.9	62.8	60.8	62.4	58.3	62.0
6	63.0	60.1	61.7	59.6	61.9	60.1	61.5	60.4
7	61.1	64.4	61.4	65.0	60.8	66.6	60.5	65.0
8	59.1	59.1	58.6	60.1	60.3	60.4	58.8	59.7
9	59.8	61.6	60.8	62.0	60.1	62.5	60.2	61.1
Ave	61.1	61.3	61.0	61.7	61.3	62.3	60.0	61.4

Table 8 shows the results of a dimensional reduction by 100 features performed by four algorithms on filter bank number 5 (frequency range was 20 to 24). The average accuracies for channel 8 were calculated as 60.8%, 61.2%, 61.4%, and 60.1%, respectively, and for channel 12, they were calculated as 61.5%, 61.9%, 63.2%, and 61.1%, respectively for four algorithms, i.e., PSO, GA, BGA (with the third model), and QGA. In general, for each channel and subject, the highest accuracies are about 0.2 to 1% higher than the other algorithms. Our proposed algorithm performed better for most of the cases. However, for some test subjects, the highest accuracy was not obtained by our proposed algorithm. For example, in channel 8, the highest accuracies were obtained by GA for the first, third, and sixth subjects, i.e., 61.6%, 62.0%, and 63.2%, respectively, and for the ninth subject, the highest accuracy, i.e., 61.8%, was obtained by QGA. In channel 12, the highest accuracy, i.e., 65.2%, was obtained for the fourth subject by PSO algorithm. The rest of the highest accuracies was obtained by our proposed algorithm in both channels for different subjects.

Table 8. Feature selection accuracy of PSO, GA, and BGA with 100 features, FB5, and ELM for 100 iterations.

Subject	PSO		GA		BGA		QGA	
	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12
1	60.5	58.7	61.6	59.4	60.6	61.1	60.4	58.1
2	60.7	59.5	59.9	60.0	60.9	60.1	58.3	59.7
3	60.3	62.5	62.0	61.8	60.9	63.3	60.5	61.7
4	65.0	65.2	62.5	64.9	63.8	64.4	60.2	63.3
5	59.6	61.9	59.5	63.5	60.8	63.9	58.9	62.3
6	62.8	61.0	63.2	59.4	62.0	62.8	62.4	60.0
7	60.7	64.6	61.8	66.0	62.2	67.4	60.2	65.4
8	58.5	58.8	59.3	59.5	60.5	61.1	58.4	58.0
9	59.4	61.6	60.7	62.4	61.1	64.6	61.8	61.3
Ave	60.8	61.5	61.2	61.9	61.4	63.2	60.1	61.1

4.4.2. Results of Second Model Scenario

In the following, we will examine the results of the hybrid signals, i.e., a combination of filter banks (a) 2 and 5 presented in Table 9, (b) 2, 5, and 6 presented in Table 10, and (c) 2, 5, 6, and 9 presented in Table 11. Such cases do not exist in nature for recording, and no system can record such models. Hence, we generate them offline. We have highlighted the highest obtained accuracies in these tables, similar to the previous two tables presented above. For these cases, our proposed algorithm mostly had the highest accuracies for channel 12. In addition, our algorithm demonstrated higher average accuracies in all of the cases. In Table 11, similar to Table 9, in channel 12, the majority of the best accuracies was obtained by our algorithm. According to the results presented in Tables 7–11, in general, the algorithms perform better in channel 12 compared to channel 8 (1–3% higher accuracy). Reduction of the dimensions for the brain signals is effective in avoiding the local minimums.

Table 9. Feature selection accuracy of PSO, GA, and BGA with 100 features, FB2 & 5, and ELM for 100 iterations.

Subject	PSO		GA		BGA		QGA	
	FB 2 & 5		FB 2 & 5		FB 2 & 5		FB 2 & 5	
	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12
1	61.2	60.4	62.52	62.09	61.7	60.4	61.9	61.4
2	62.6	59.8	59.53	59.44	61.1	60.2	58.3	58.6
3	60.8	62.4	61.17	64.51	62.0	64.0	61.4	62.8
4	66.5	65.0	62.76	61.45	65.1	65.1	62.2	60.7
5	60.9	62.9	59.77	62.40	62.3	63.2	59.1	61.2
6	61.1	61.3	62.27	60.04	59.9	61.6	61.5	60.1
7	60.2	62.7	60.68	63.84	60.7	66.2	60.9	63.8
8	57.4	58.8	59.4	60.78	60.7	62.4	56.8	58.7
9	59.5	61.8	60.63	61.96	60.8	63.8	61.0	61.5
Ave	61.1	61.7	61.00	61.83	61.6	63.0	60.3	61.0

Table 10. Feature selection accuracy of PSO, GA, and BGA with 100 features, FB2 & 5 & 6, and ELM for 100 iterations.

Subject	PSO		GA		BGA		QGA	
	FB 2 & 5 & 6		FB 2 & 5 & 6		FB 2 & 5 & 6		FB 2 & 5 & 6	
	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12
1	60.7	60.9	62.42	63.32	61.1	60.9	62.2	61.5
2	61.1	58.9	60.31	59.63	61.1	60.5	59.2	60.0
3	60.6	61.7	60.86	62.25	61.5	63.5	59.9	61.4
4	65.5	64.6	60.27	62.48	65.2	64.7	60.0	60.8
5	61.5	62.2	59.52	63.03	61.4	62.5	58.0	61.8
6	62.0	60.6	62.77	59.12	60.8	62.5	61.9	59.9
7	60.6	64.3	61.90	65.66	60.0	66.2	60.0	65.3
8	59.0	60.0	59.56	59.64	61.8	61.2	59.1	58.5
9	59.6	61.1	60.43	62.31	61.1	63.0	60.2	61.3
Ave	61.2	61.6	60.89	61.86	61.6	62.8	60.1	61.2

Table 11. Feature selection accuracy of PSO, GA, and BGA with 100 features, FB2 & 5 & 6 & 9, and ELM for 100 iterations.

Subject	PSO		GA		BGA		QGA	
	FB 2 & 5 & 6 & 9		FB 2 & 5 & 6 & 9		FB 2 & 5 & 6 & 9		FB 2 & 5 & 6 & 9	
	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12
1	60.0	60.6	59.71	62.25	60.6	61.0	58.3	59.9
2	58.7	60.3	59.60	60.75	62.7	60.8	58.6	59.4
3	60.3	61.6	61.18	62.07	61.1	63.0	61.0	61.1
4	61.3	62.3	59.76	61.10	63.9	62.8	58.5	60.2
5	59.9	63.1	61.55	63.32	61.1	64.1	60.1	61.8
6	63.5	60.3	62.34	59.76	62.0	64.7	62.2	59.7
7	61.3	65.8	61.78	65.80	60.3	66.7	60.7	66.2
8	60.7	60.2	60.36	61.99	62.6	61.9	59.5	58.5
9	60.5	61.3	59.56	61.3	61.4	63.2	58.9	61.4
Ave	60.7	61.7	60.65	61.4	61.5	63.1	59.8	60.9

4.4.3. Results of Third Model Scenario

Table 12 presents the results related to the extraction and selection of features, including time domain, frequency domain, autoregression, and wavelet coefficients, on a single channel (C3 = channel 8 or C4 = channel 12) by combining two filter banks, i.e., two and five (making a new signal with eight new frequency ranges). The best average accuracy was obtained by GA on both channels, i.e., 61.90% and 60.97%, respectively. On channel 8, the average accuracies were calculated as 59.84%, 59.78%, and 59.75% for PSO, BGA, and QGA, respectively. On channel 12, the average accuracies were calculated as 61.24%, 61.14%, and 60.89% for PSO, QGA, and BGA, respectively. In channel 12, all other algorithms performed better than our proposed algorithm. In general, in this model, our proposed algorithm did not perform well.

Table 12. The results of algorithms with four methods (TD, FD, wavelet, and autoregression) on a single channel (C3 = 8 or C4 = 12) with FB 2 & 5 (25).

Subject	PSO		GA		BGA		QGA	
	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12	Channel 8	Channel 12
1	62.78	59.16	63.18	59.98	62.00	58.79	62.43	59.13
2	57.24	60.21	56.60	61.04	58.30	59.57	56.38	59.32
3	60.52	63.13	62.21	63.81	61.80	62.10	61.05	64.75
4	59.18	60.00	59.76	60.80	60.00	59.84	59.05	60.03
5	59.59	61.48	60.03	62.10	57.73	60.38	57.77	59.20
6	57.57	57.98	59.75	58.38	56.65	57.89	57.70	57.94
7	56.97	61.46	56.88	62.34	56.94	60.01	56.65	59.68
8	60.71	59.95	61.12	62.09	60.12	59.90	60.29	60.66
9	64.03	67.77	69.17	66.56	64.42	69.51	66.50	69.61
Ave	59.84	61.24	60.97	61.90	59.78	60.89	59.75	61.14

4.4.4. Results of Fourth and Fifth Model Scenarios

Tables 13 and 14 present the results related to the extraction and selection of features, including time domain, frequency domain, autoregression, and wavelet coefficients, on two channels (C3 = 8 and C4 = 12) and three channels (C3 = 8 and CZ = 10 and C4 = 12), respectively. In both cases, two filter banks, i.e., two and five, were combined, making a new signal with eight new frequency ranges.

The GA obtained better average accuracies in both models, i.e., 66.16% and 65.45% for two channels and 66.59% and 66.57% for three channels.

Table 13. The results of algorithms with four methods (TD, FD, wavelet, and autoregression) on two channels (C3 = 8 and C4 = 12) with FB 2 & 5 (25).

Subject	PSO		GA		BGA		QGA	
	M1	M2	M1	M2	M1	M2	M1	M2
1	63.17	63.72	64.62	65.49	62.17	62.88	62.03	62.83
2	60.42	60.67	61.68	61.56	59.20	60.69	59.47	59.73
3	84.64	84.66	85.10	85.10	84.05	83.79	84.61	84.31
4	61.44	60.73	62.40	61.33	60.30	60.23	60.12	60.52
5	61.31	60.18	63.44	59.98	60.61	60.05	60.25	59.79
6	62.49	61.85	62.25	62.34	61.25	61.48	61.12	62.20
7	65.14	59.17	67.48	63.15	61.11	58.41	62.65	59.60
8	61.56	62.99	61.79	62.50	60.47	62.48	62.00	62.22
9	68.94	68.38	66.68	67.60	68.21	66.20	70.82	66.39
Ave	65.46	64.71	66.16	65.45	64.15	64.02	64.78	64.18

Table 14. The results of algorithms with four methods (TD, FD, wavelet, and autoregression) on three channels (C3 = 8 and CZ = 10 and C4 = 12) with FB 2 & 5 (25).

Subject	PSO		GA		BGA		QGA	
	M1	M2	M1	M2	M1	M2	M1	M2
1	63.17	63.36	63.03	63.88	62.19	62.89	62.06	62.50
2	62.30	61.50	61.65	61.15	61.09	59.81	60.08	60.02
3	84.34	85.25	84.88	85.03	82.53	83.11	82.85	83.06
4	63.20	63.05	64.52	63.83	62.09	61.68	62.26	61.66
5	59.31	59.92	59.42	59.85	59.90	58.96	60.56	59.20
6	61.81	62.32	61.72	61.93	61.23	60.91	60.88	60.76
7	61.13	61.40	61.88	62.06	60.75	60.96	61.44	61.62
8	65.41	64.39	65.91	65.34	65.55	63.93	66.06	63.66
9	76.04	76.75	76.31	76.06	74.23	75.22	75.16	75.04
Ave	66.30	66.44	66.59	66.57	65.51	65.27	65.71	65.28

4.4.5. Results of Sixth and Seventh Model Scenarios

Table 15 presents the accuracy of the two models with CSP (sixth and seventh models that we defined earlier). In this case, a feature reduction by 100 features was performed, which was followed by the creation of new signals using the filter banks 2, 5, and 6. As a result, matrices M1 and M2 are created as input to CSP based on two and three primary channels, respectively. Our proposed algorithm shows the best performance, i.e., 73.12% and 79.53% average accuracies for M1 and M2, respectively. Kappa was calculated as 46.24% and 59.05% for M1 and M2, respectively. The average accuracy for M2 was also relatively good for the other three algorithms, i.e., 75.07%, 73.99%, and 71.18% for PSO, GA, and QGA, respectively. While in M1, the accuracy drops by 4 to 5% when using two channels and three filter banks.

Our proposed algorithm's average accuracy improves 6.41% from M1 to M2, and the average Kappa improves 12.82% from M1 to M2. In PSO, this improvement is 5.53% and 11.06% for the average accuracy and kappa, respectively. In GA, this improvement is 4.7% and 9.43% for average accuracy and kappa, respectively. In QGA, this improvement is 5.22% and 10.45% for average accuracy and kappa, respectively. This shows that the addition of the channels and filter banks has a significant impact on improving the accuracy.

Table 15. The accuracy results of algorithms on dimension reduction and features extraction of CSP new model input matrix (using formulas) on two and three channels (C3 = 8 and CZ = 10 and C4 = 12).

Subject	PSO		GA		BGA		QGA	
	M1	M2	M1	M2	M1	M2	M1	M2
1	68.94	75.20	70.88	74.13	70.32	78.16	67.39	71.14
2	67.67	72.03	68.3	72.34	68.50	75.02	64.40	68.12
3	81.66	81.96	71.17	75.37	89.50	90.85	67.32	72.45
4	66.187	73.18	69.73	74.81	68.41	75.21	66.45	70.92
5	65.27	71.12	68.80	70.51	68.20	77.23	65.40	68.64
6	65.36	74.61	67.83	73.58	69.90	77.97	64.52	70.74
7	68.93	72.70	70.72	74.54	70.74	77.09	68.32	72.17
8	64.91	74.59	66.57	73.20	70.32	79.22	62.43	70.90
9	76.98	80.28	69.51	77.47	82.18	85.02	67.38	75.54
Ave	69.54	75.07	69.29	73.99	73.12	79.53	65.96	71.18

4.5. Results on Pooling for Sampling

4.5.1. Results of First Sampling Model

Figures 8 and 9 show the results of selecting four different window sizes for participants, all five functions, and all nine filter banks. The first four functions select one of the features to reduce the dimension, and the fifth function extracts the average of all of the features in each window. Overall, the fifth function, i.e., average function, has the best average accuracy, i.e., 73.17%, compared to the other functions in all filter banks and for all participants. The performance of the rest of the functions, i.e., maximum–minimum automatic function, no-maximum average function, maximum function, and minimum function, are 50.70%, 51.34%, 64.74%, and 63.72%, respectively.

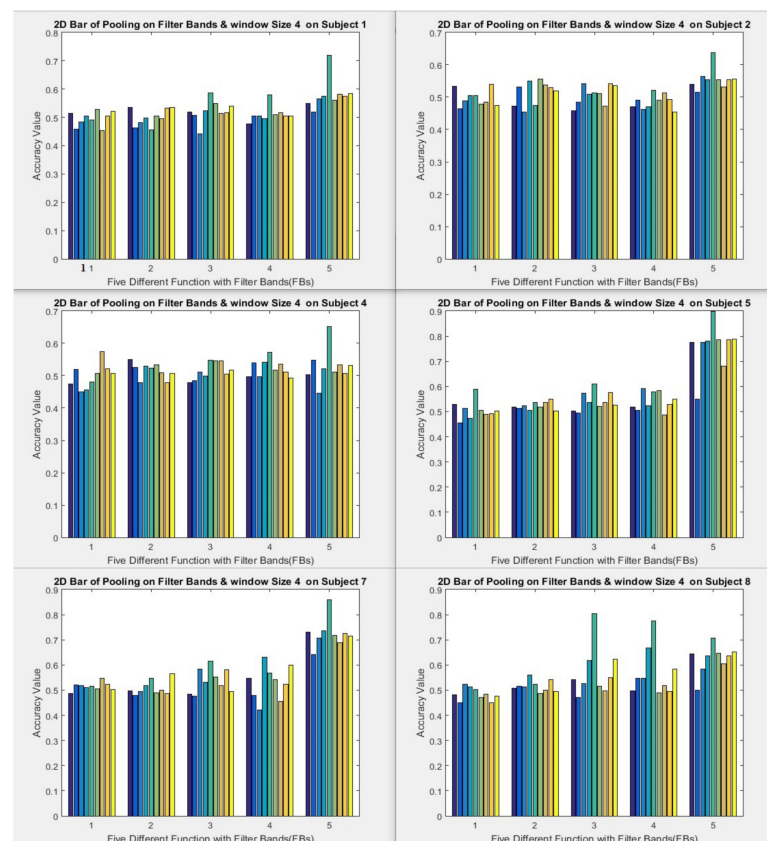


Figure 8. Reduction data based on pooling model on some subjects (1, 2, 4, 5, 7, 8) and filter banks with window size 4 and RF for 10 iterations.

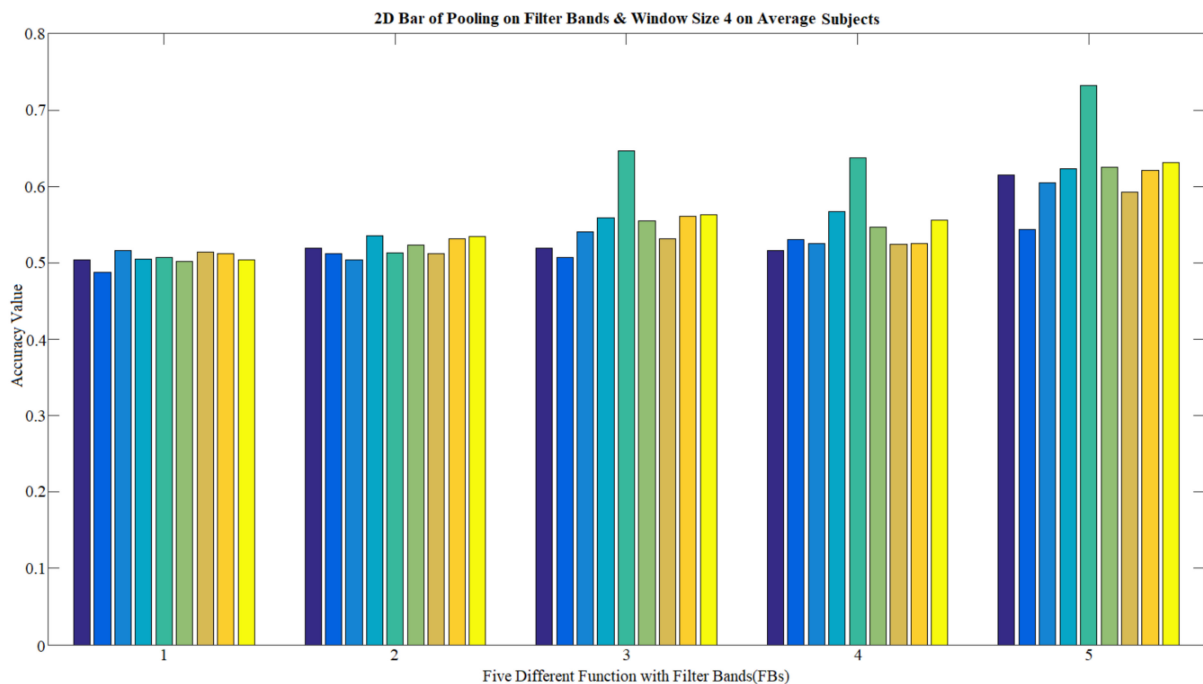


Figure 9. Reduction data based on pooling model on average subjects with filter banks and window size 4 and RF for 10 iterations.

We investigate three well-known functions, i.e., maximum, minimum, and average functions for different participants.

For example, for subject one, the accuracy obtained for maximum, minimum, and average functions were 58.70%, 58.00%, and 71.90%, respectively. While for subject two, these accuracy values were about 10% different.

The accuracy of the first function, i.e., maximum function, is close to 50% in most subjects. The accuracy of the second function, i.e., minimum function, in some cases is 4 or 5% better than the first function. In the second function, the difference in accuracies is 13% and 31% in some subjects. This indicates the effectiveness of the maximum function.

The average accuracy on the average function for all subject on the filter bank 5 is 61.45%, 54.30%, 62.30%, 73.17%, 62.53%, 59.27%, 62.09%, and 63.07%, respectively. So, it can be concluded that filter bank 5 (FB5) has the most information that is related to the perception of the left and right hand. According to our experiment, the frequency range between 8 and 30 Hz contains the most important information that is represented as filter bank 5.

Figures 10–13 show the results of two filter banks, i.e., 5 and 6, with all the functions and different types of sampling window sizes for participants. In practice, we have five different types of sampling window sizes in the range of 3–7. We present the results of the average accuracy of all subjects with varying window sizes for all functions and the filter bank 5 in Figure 10.

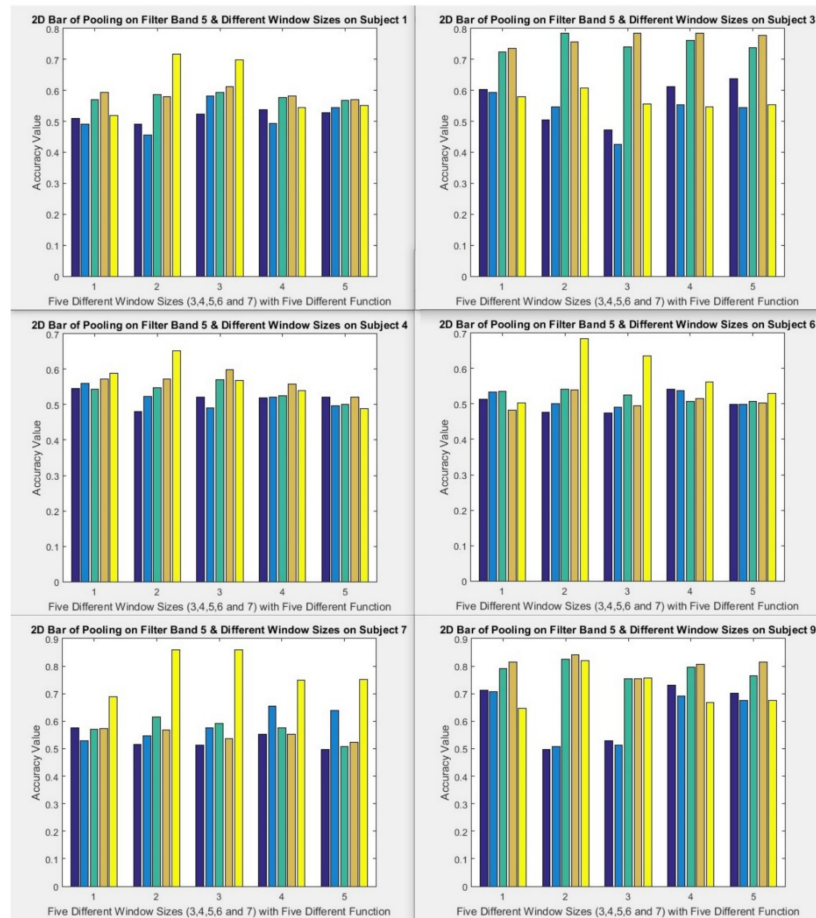


Figure 10. Reduction data based on pooling model on some subjects (1, 3, 4, 6, 7, 9) and filter banks with different window sizes for FB 5 and RF for 10 iterations.

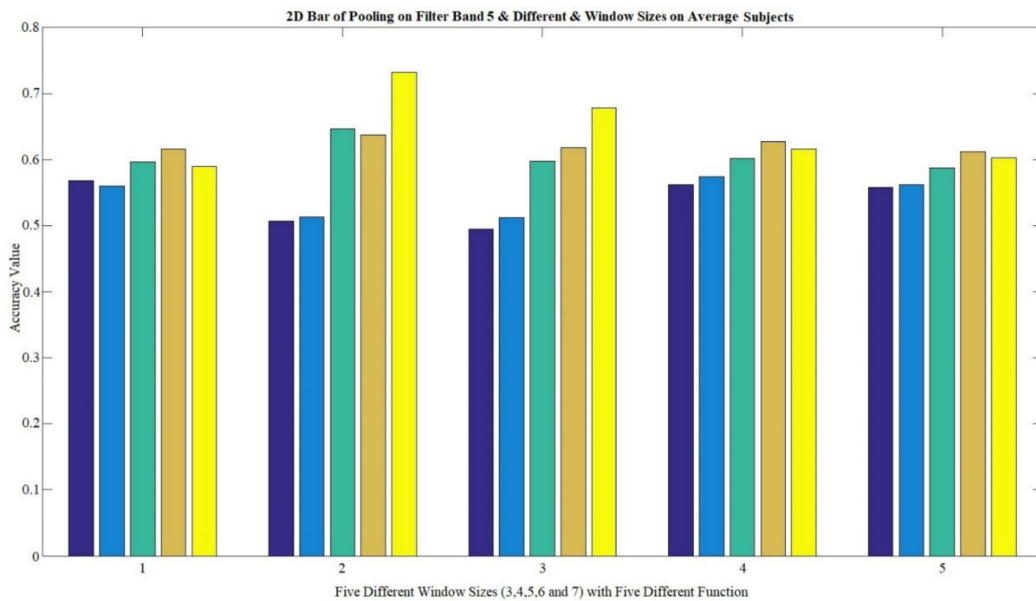


Figure 11. Reduction data based on pooling model on average subjects with filter banks with different window sizes for FB 5 and RF for 10 iterations.

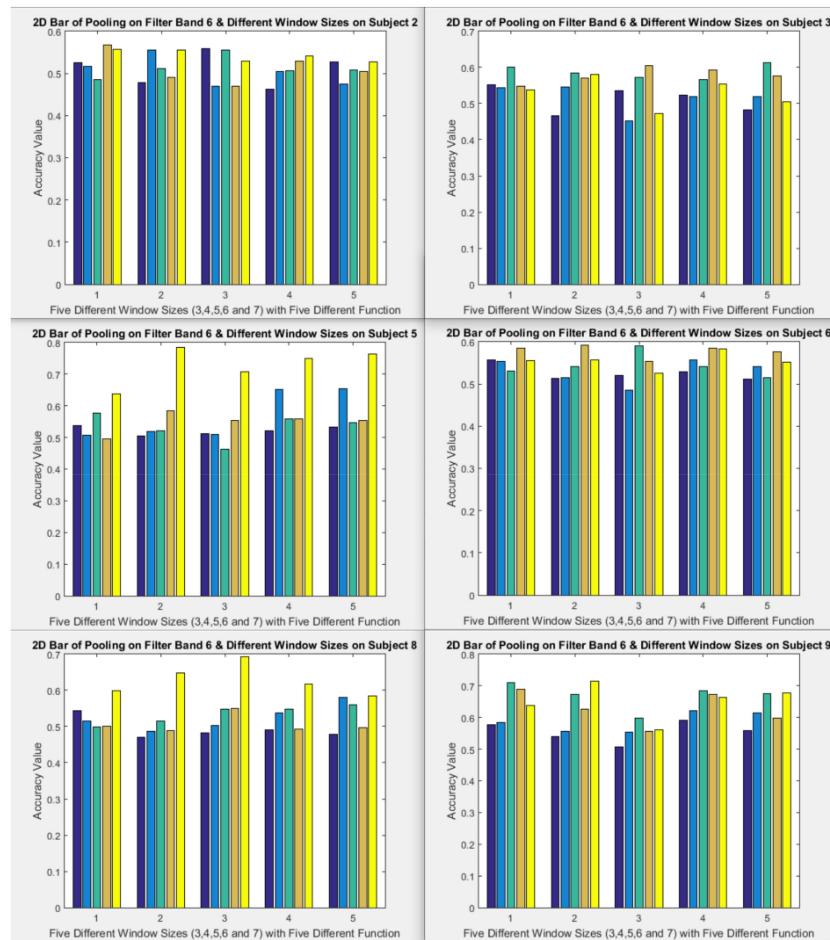


Figure 12. Reduction data based on pooling model on some subjects (2, 3, 5, 6, 8, 9) and filter banks with different window sizes for FB 6 and RF for 10 iterations.

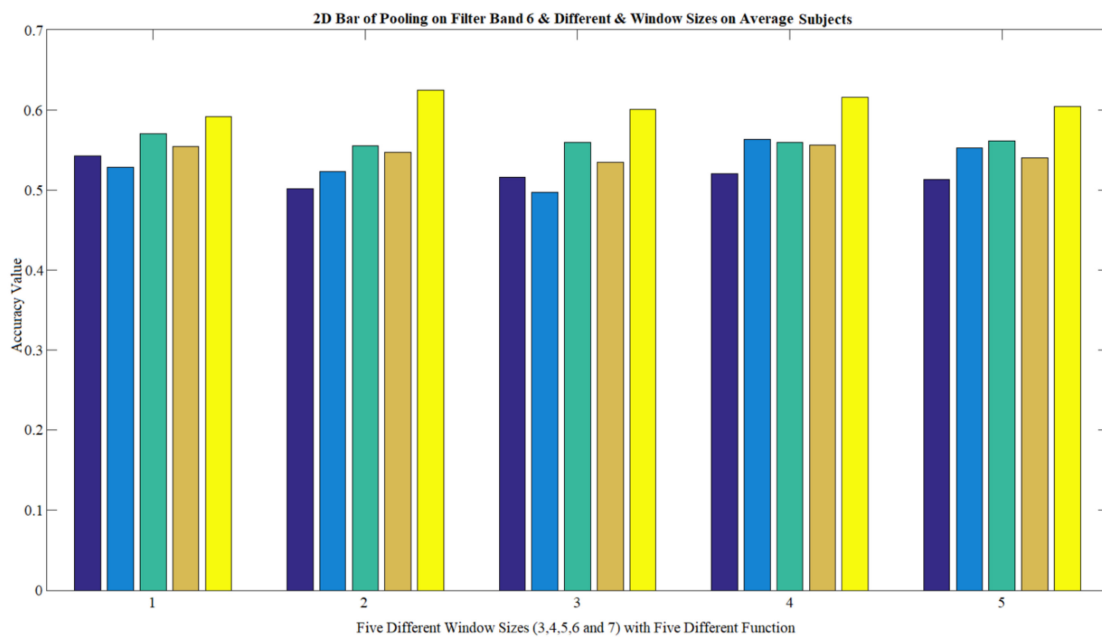


Figure 13. Reduction data based on pooling model on average subjects with filter banks with different window sizes for FB 6 and RF for 10 iterations.

The highest accuracy obtained for the first, second, third, fourth, and fifth functions and filter bank 5 were related to window sizes 3, 6, 4, 4, and 4, respectively, i.e., 56.79%, 57.43%, 64.74%, 63.72%, and 73.17%, respectively (Figure 12). Similarly, the results for filter bank 6 is presented in Figures 12 and 13.

4.5.2. Results of Second and Third Sampling Models

Table 16 presents the result of two different approaches: first, all filter banks with the same function used in all layers, and second, all filter banks with randomly selected functions for each layer. Both cases are applied for different types of sampling window sizes for participants. The best-obtained accuracies are highlighted in the table. According to this experiment, we conclude that the window size and the selected functions significantly impact the accuracy. The best accuracies in this scenario, i.e., using all filter banks, are in the range of 55–62%, while this range is between 60 and 73% when using only one filter bank, i.e., filter bank 5 (Figure 11). This demonstrates that effective features are selected in filter bank 5, contributing to better accuracies.

Table 16. Reduction data based on pooling model on filter banks with different window sizes for all filter banks with the same function in layers or randomly FB 5 and all of FBs and RF for 10 iterations.

Subject		FB-Same					FB-Hybrid				
	Size/Function	F1	F2	F3	F4	F5	R1_HF	R2_HF	R3_HF	R4_HF	R5_HF
Avg	Size 3	0.545	0.541	0.59	0.595	0.59	0.557	0.545	0.554	0.559	0.554
	Size 4	0.523	0.56	0.592	0.589	0.623	0.535	0.516	0.522	0.524	0.534
	Size 5	0.506	0.523	0.569	0.577	0.59	0.526	0.518	0.511	0.512	0.511
	Size 6	0.537	0.587	0.588	0.604	0.617	0.571	0.555	0.55	0.542	0.578
	Size 7	0.529	0.567	0.587	0.581	0.604	0.575	0.548	0.547	0.545	0.571

5. Results and Discussion

5.1. Discussion of Algorithm on Brain Signals

Table 17 shows the best results for four different algorithms for different settings, i.e., different combinations of filter banks and channels. Our proposed algorithm (BGA) on filter bank 5 and channel 12 has obtained the best accuracies in most subjects. The average best accuracy, in this case, is 63.2%. The second-best result is related to GA with filter bank 5 and channel 12 (i.e., 61.9%). Subsequently, BGA and PSO have achieved the best results of 61.6% and 61.7%, respectively, when the combination of filter banks 2 and 5 are used. This indicates that the combination of filter banks 2 and 5 provides valuable information, but the results are still 1.5% less than the case using only filter bank 5.

In Table 18 and Figure 14, the results of accuracy for GA, BGA, and PSO with two channels and two proposed models (i.e., FM and CSP) are compared with the methods of some articles [33,65]. BGA with CSP obtained the best accuracies in most subjects with the best average accuracy, i.e., 73.12%, which is about 5–6% more than the other three algorithms, while the results related to GA and PSO are within the same range as the other three algorithms.

Table 17. The best results of different methods for feature selection (with feature extraction, feature reduction, and channels, etc.) on a single channel 8 or 12 by combining different select bands.

Subject	PSO FB2 & 5 & 6 Ch8	PSO FB 2 & 5 Ch12	GA FB 5 Ch8	GA FB 5 Ch12	BGA FB 2 & 5 Ch8	BGA FB 5 Ch12	QA FB 2 & 5 Ch8	QA FB 2 Ch12
1	60.7	60.4	61.6	59.4	61.7	61.1	61.9	59.8
2	61.1	59.8	59.9	60.0	61.1	60.1	58.3	59.5
3	60.6	62.4	62.0	61.8	62.0	63.3	61.4	61.7
4	65.5	65.0	62.5	64.9	65.1	64.4	62.2	63.8
5	61.5	62.9	59.5	63.5	62.3	63.9	59.1	62.0
6	62.0	61.3	63.2	59.4	59.9	62.8	61.5	60.4
7	60.6	62.7	61.8	66.0	60.7	67.4	60.9	65.0
8	59.0	58.8	59.3	59.5	60.7	61.1	56.8	59.7
9	59.6	61.8	60.7	62.4	60.8	64.6	61.0	61.1
Ave	61.2	61.7	61.2	61.9	61.6	63.2	60.3	61.4
Std	1.75	1.75	1.3	2.37	1.43	2.11	1.74	1.8

The table is comparing first and second model scenarios using algorithms. FB I & J means that the new signal is the combination of filter bank I with J. Ch 12 and Ch 8 mean Channel 12 and Channel 8, respectively.

Table 18. The best results of different methods for feature selection (along with feature extraction, feature reduction, channels, etc.) on two channels 8 and 12 by combining different select bands.

Subject	PSO FM2	GA FM2	BGA FM2	PSO CSP1	GA CSP1	BGA CSP1	RF & DFFS	RF, 8–30 Hz	RQNN
1	63.17	64.62	62.17	68.94	70.88	70.32	63.69	66.52	61.11
2	60.42	61.68	59.20	67.67	68.30	68.50	61.97	56.62	61.11
3	84.64	85.10	84.05	81.66	71.17	89.50	91.09	90.36	79.17
4	61.44	62.40	60.30	66.18	69.73	68.41	61.72	59.83	60.42
5	61.31	63.44	60.61	65.27	68.80	68.20	63.41	57.63	71.53
6	62.49	62.25	61.25	65.36	67.83	69.90	66.11	65.19	61.11
7	65.14	67.48	61.11	68.93	70.72	70.74	59.57	64.86	58.33
8	61.56	61.79	60.47	64.91	66.57	70.32	62.84	66.57	67.36
9	68.94	66.68	68.21	76.98	69.51	82.18	84.46	77.69	79.36
Avg	65.46	66.16	64.15	69.54	69.29	73.12	68.32	67.25	66.59
Std	7.21	6.98	7.45	5.53	1.46	7.07	10.65	10.08	7.77
p-value	0.086	0.191	0.020	0.285	0.398	0.005	-	0.242	0.199

p-value, the paired t-test between results of RF & DFFS [33], with fourth (four methods with two channels (FM2)) and sixth (CSP with two channels (CSP1)) model scenarios and some previous methods.

In Table 18, the statistical significance of performance between RF and DDFS with two other methods [33] with the fourth model scenario (four feature extractions, i.e., time domain, frequency domain, autoregression, and wavelet coefficients with two channels), and the sixth model scenario (CSP with two channels) have been calculated. A paired t-test is calculated for comparing p-values. The methods are considered of statistical significance when the p-value is less than 0.05. BGA with CSP with the fourth and sixth model scenarios could achieve p-values 0.020 and 0.005, respectively.

In Table 19 and Figure 15, the results of kappa for GA, BGA, and PSO with three channels and two proposed models (i.e., FM and CSP) are compared with the methods of some articles that use different channels [39]. BGA with CSP obtained the best kappa in most subjects with the average kappa, i.e., 59.06%, which is only 0.84% less than standard kappa (i.e., 60%). This result is significantly higher (about 18%) than the article [39] with three channels, while this difference is only about 0.5–1% compared to the results of the article [39] that uses all or 8.55 channels.

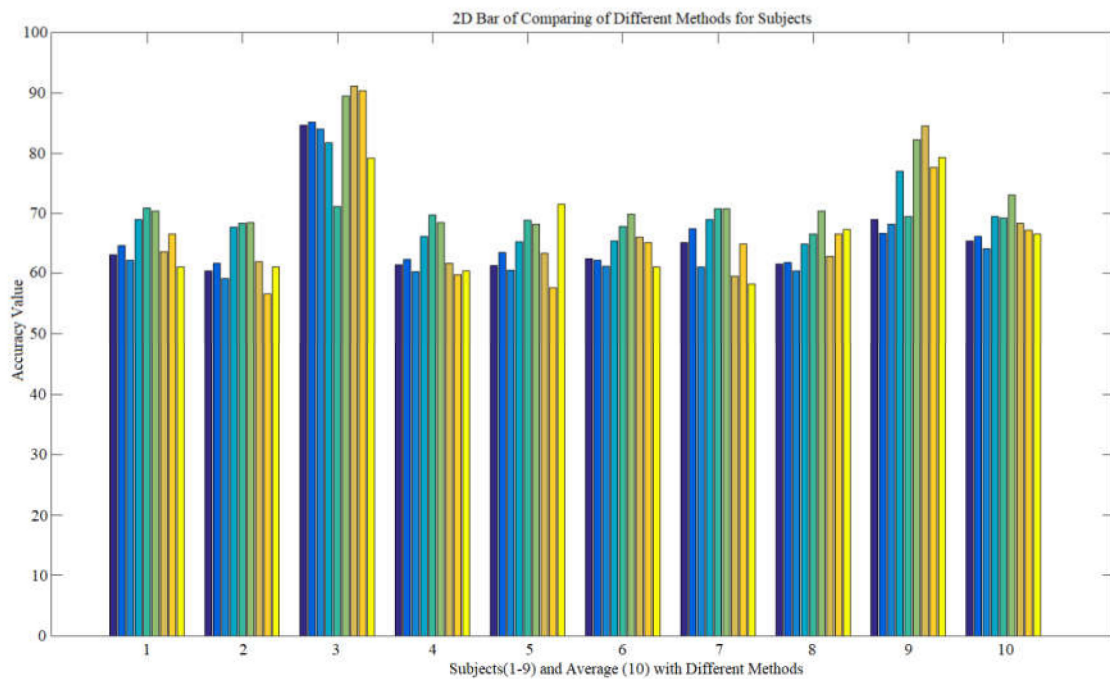


Figure 14. The best results of different methods for feature selection (along with feature extraction, feature reduction, channels, etc.) on two channels, 8 and 12, by combining different select bands.

Table 19. Examining the best kappa different methods for selecting features (along with feature extraction, reducing features and channels, etc.) on three channels, 8, 10, and 12, by combining different selected bands.

Subject	PSO FM3	GA FM3	BGA FM3	PSO CSP2	GA CSP2	BGA CSP2	CSP ALL	CSP 3.0 Channels	CSP 8.55 Channels
1	26.72	26.06	24.38	50.40	48.26	56.32	81.94	51.38	83.32
2	23	23.3	22.18	44.06	44.68	50.04	12.5	6.94	20.82
3	70.5	69.76	65.06	63.92	50.74	81.7	93.04	86.1	94.28
4	26.1	29.04	24.18	46.36	49.62	50.42	45.82	36.1	41.66
5	19.84	18.84	19.8	42.24	41.02	54.46	27.76	6.94	26.38
6	24.64	23.44	22.46	49.22	47.16	55.94	27.76	22.22	22.22
7	22.8	23.76	21.5	45.4	49.08	54.18	59.72	15.26	56.94
8	28.78	31.82	31.1	49.18	46.4	58.44	94.44	73.6	90.26
9	53.5	52.62	48.46	60.56	54.94	70.04	83.32	77.76	87.50
Avg	32.88	33.18	31.01	50.14	47.98	59.06	58.47	41.81	58.15
Std	16.26	15.89	14.61	6.96	3.69	9.73	29.47	29.64	29.44
<i>p</i> -value	0.112	0.114	0.076	0.176	0.270	0.032	0.003	-	0.003

p-value, the paired *t*-test between results of CSP (C3, C4, CZ (all 3)) [39], with the fifth (four methods with three channels (FM3)) and seventh (CSP with three channels (CSP2)) model scenarios and some previous methods.

In Table 19, the statistical significance of performance between CSP with three channels (C3, C4, and CZ) with two other methods [39], the fifth model scenario (four feature extractions, i.e., time domain, frequency domain, autoregression, and wavelet coefficients with three channels), and the seventh model scenario (CSP with three channels) have been calculated. The methods are considered statistically significant when the *p*-value is less than 0.05. BGA with CSP in the seventh model scenario and two models from other papers could achieve *p*-values of 0.032, 0.003, and 0.003, respectively.

Table 20 and Figure 16 show the results of kappa for GA, BGA, and PSO with three channels, and two proposed models (i.e., FM and CSP) are compared with the methods of some articles that use different channels and different filter banks [39,48]. The best average kappa in this experiment is related to FBCSP in [48], which was 4.41% higher than

BGA with CSP. In this scenario, the other methods, in general, had 1.5–3% better kappa compared to BGA with CSP. This is because in BGA, we use only three channels with 60% of the frequency range, while in other studies, 22 channels with the entire frequency range have been used. Considering the significant difference in the channels and frequency range, the BGA had only 1.5% less kappa compared to the smallest kappa in the other works.

Table 21 and Figure 17 show the results of kappa for various implementation methods with CSP [66]. The kappa values are in the range of 50% to 63%. BGA with CSP, three channels, and 60% of the frequency range obtained the second-best average kappa, i.e., 59%. While in all the other methods, all channels with all frequency ranges have been used.

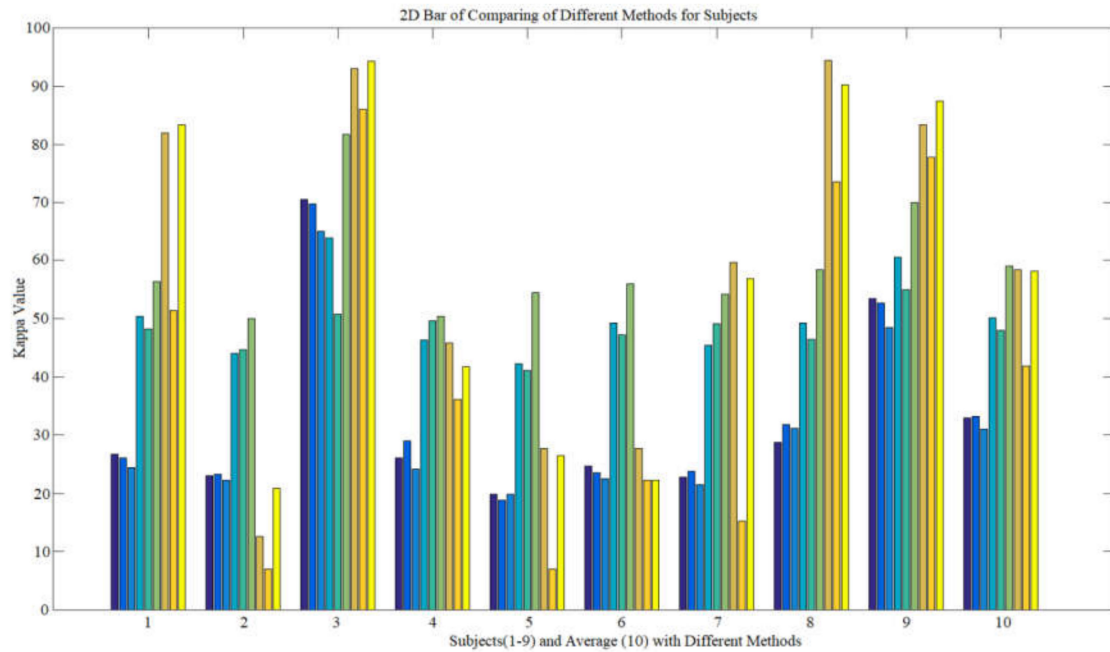


Figure 15. Examining the best kappa different methods for selecting features (along with feature extraction, reducing features and channels, etc.) on three channels, 8, 10, and 12, by combining different selected bands.

Table 20. Examining the best kappa different methods for selecting features (along with feature extraction, reducing features and channels, etc.) on three channels with more than three channels by combining different selected bands and so on.

Subject	GA FM2	BGA FM2	PSO CSP2	BGA CSP2	CSP ALL	CSP 8.55 Channels	CSP 13.22 Channels	oFBCSP	sFBCSP	eFBCSP
1	26.06	24.38	50.4	56.32	81.94	83.32	83.32	72.00	72.10	74.70
2	23.3	22.18	44.06	50.04	12.5	20.82	34.72	38.90	39.50	41.60
3	69.76	65.06	63.92	81.7	93.04	94.28	95.82	82.20	81.60	82.40
4	29.04	24.18	46.36	50.42	45.82	41.66	44.44	38.10	38.40	40.00
5	18.84	19.8	42.24	54.46	27.76	26.38	30.54	56.30	59.20	60.80
6	23.44	22.46	49.22	55.94	27.76	22.22	33.34	25.50	28.70	30.90
7	23.76	21.5	45.4	54.18	59.72	56.94	69.44	80.00	83.00	84.90
8	31.82	31.1	49.18	58.44	94.44	90.26	94.44	78.50	78.60	78.70
9	52.62	48.46	60.56	70.04	83.32	87.50	83.32	74.70	76.00	77.20
Avg	33.18	31.01	50.15	59.06	58.48	58.15	63.26	60.70	61.90	63.50
Std	15.89	14.61	6.96	9.73	29.47	29.44	25.84	20	20	20

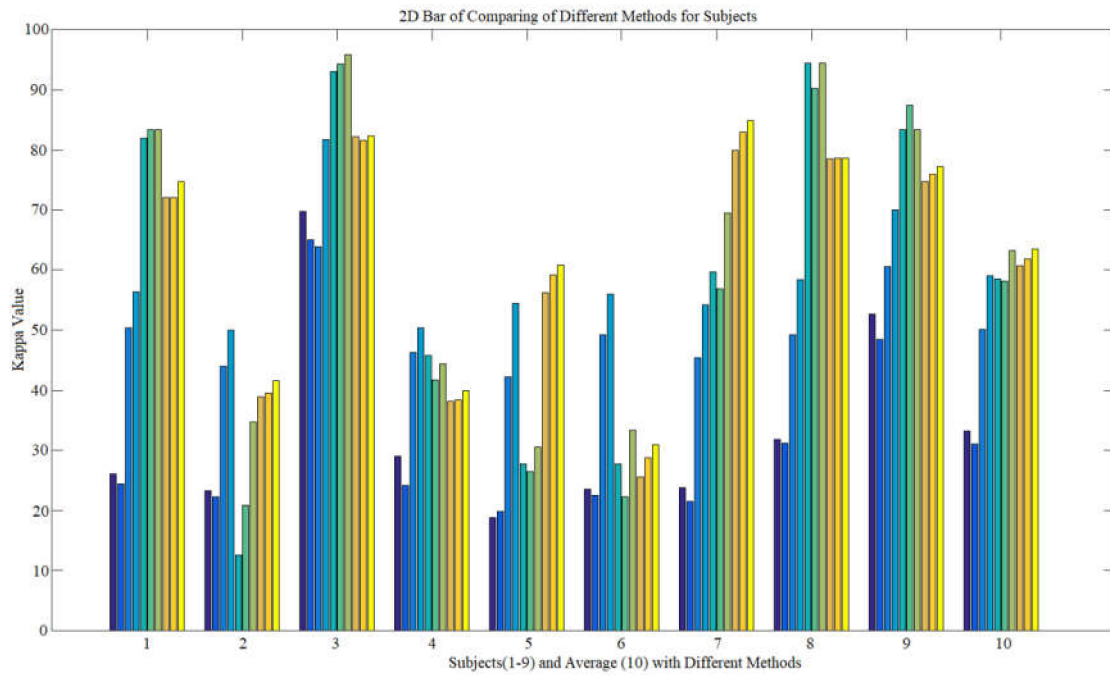


Figure 16. Examining the best kappa different methods for selecting features (along with feature extraction, reducing features and channels, etc.) on three channels with more than three channels by combining different selected bands and so on.

Table 21. Checking the best kappa results of different methods compared with the proposed method [66].

Methods	S1	S2	S3	S4	S5	S6	S7	S8	S9	Avg	Std	p-Value
CSP	77.8	2.8	93.1	40.3	9.7	43.1	62.5	87.5	87.5	56	32.09	0.013
GLRCSP	72.2	16.7	87.5	34.7	11.1	30.6	62.5	87.5	76.4	53.2	28.47	0.028
CCSP1	72.2	20.8	87.5	13.9	1.4	30.6	62.5	87.5	76.4	50	31.71	0.112
CCSP2	77.8	6.9	94.4	40.3	8.3	36.1	58.3	90.3	80.6	54.8	31.71	0.013
DLCSPauto	77.8	2.8	93.1	40.3	13.9	43.1	63.9	87.5	87.5	56.6	31.46	0.011
DLCSPcv	77.8	1.4	93.1	40.3	11.1	25	62.5	87.5	73.6	52.5	32.13	0.046
DLCSPcvdiff	77.8	1.4	93.1	40.3	11.1	25	62.5	87.5	73.6	52.5	32.13	0.046
SSRCSP	77.8	6.9	94.4	40.3	12.5	37.5	58.3	94.4	80.6	55.9	31.49	0.008
TRCSP	77.8	8.3	93.1	41.7	25	34.7	62.5	91.7	83.3	57.6	29.41	0.005
WTRCSP	77.8	9.7	93.1	40.3	31.9	23.6	62.5	91.7	81.9	56.9	29.54	0.009
SRCSP	77.8	26.4	93.1	33.3	26.4	27.8	56.9	91.7	84.7	57.6	27.87	0.004
SCSP	81.9	12.5	93	45.8	27.8	27.8	59.7	94.4	83.3	58.5	29.45	0.003
SCSP1	83.3	34.7	95.8	44.4	30.5	33.3	69.4	94.4	83.3	63.3	25.84	0.002
SCSP2	83.3	20.8	94.3	41.7	26.4	22.2	56.9	90.3	87.5	58.2	29.44	0.003
BGA-CSP2	56.3	50	81.7	50.4	54.5	55.9	54.2	58.4	70	59.1	9.73	0.032
CSP (C3,C4, CZ)	51.38	6.94	86.10	36.1	6.94	22.22	15.26	73.60	77.76	41.81	29.64	-

p-value, the paired t-test between results of CSP (C3, C4, CZ (all 3)) [39], with the seventh (CSP with three channels (CSP2)) model scenario and some previous methods.

In Table 21, the statistical significance of performance between CSP (C3, C4, CZ) [39] with two of our model scenarios and other methods is evaluated. We used a paired t-test for the comping of p-value. All of the methods have a p-value of less than 0.05 (p-value < 0.05).

Figure 18 shows one training epoch of the increasing accuracy of subjects 3 and 8 with algorithms using the seventh model scenario (CSP with three channels (CSP2)). BGA has the best converging to reach optimal results.

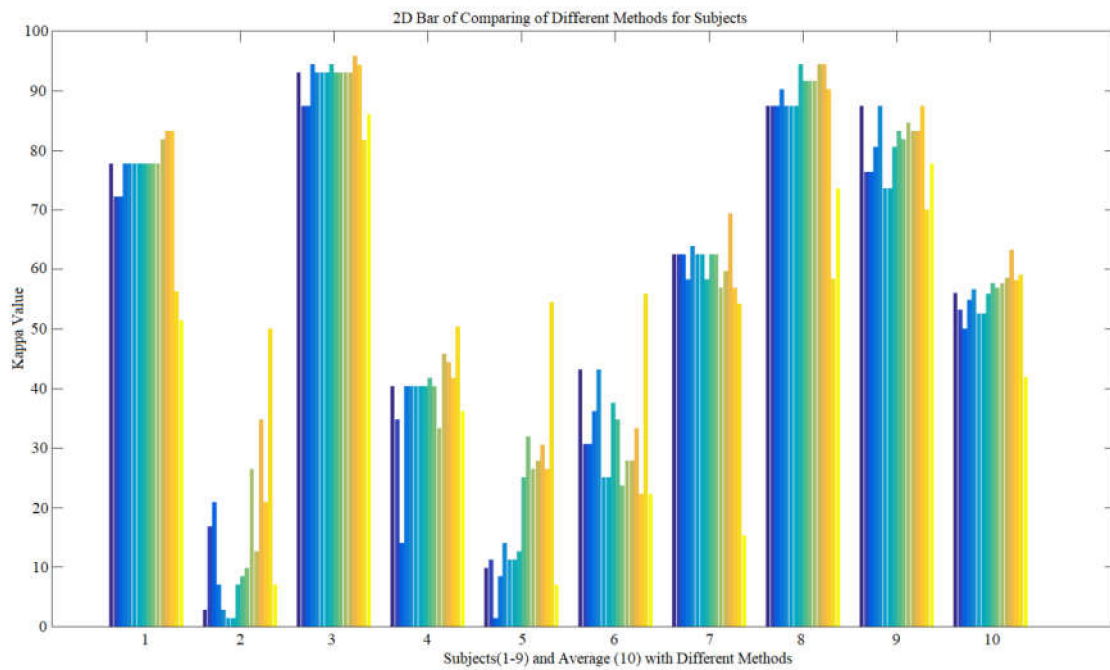


Figure 17. Checking the best kappa results of different methods compared with the proposed method.

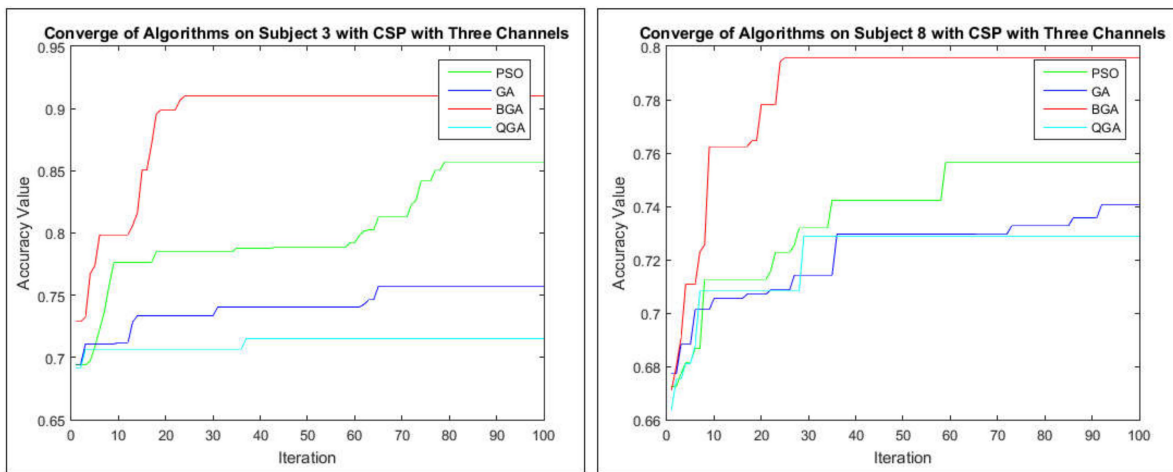


Figure 18. Converging of subjects 3 and 8 with algorithms on the seventh model scenario (CSP with three channels (PSO-CSP2, GA-CSP2, BGA-CSP2, and QGA-CSP2)).

In most studies involving filter banks, first, all filter banks are applied to noise space filters to reduce noise, and then, features are extracted from them. Next, all the features of all filter banks are added together, and we select a set of features from them. In addition, in reducing the features for channels or selecting a channel, an interval of eight to 30 frequencies is considered. In this study, two or three filter banks with two important channels are applied to reduce the noise and extract more important information. We examined various algorithms tested on brain signals. We also proposed a new algorithm based on the Bond Graph method, i.e., BGA. Our design goal for the new algorithm is to effectively reduce the dimension of features and the effect of specified filter banks and channels.

Most of these algorithms show close convergence with each other. Although the accuracy values for some subjects were low in some cases, it was still far enough from the local minimum trap. In most cases, BGA showed the best performance compared to the other algorithms.

5.2. Discussion Sampling (Pooling) on Brain Signals

In all our study cases, sampling is part of deep learning to reduce the dimension in the layers. It is used during the deep learning phase. We use samplings based on four main functions, i.e., maximum, minimum, maximum–minimum automatic function, and no-maximum average function, on the data along with filter banks to reduce the dimension of the channels. The number of channels is fixed, and the dimensional reduction of each channel depends on the window size. The output window sizes for each channel are as follows: window size 3 with four features, window size 4 with one feature, window size 5 with seven features, window size 6 with five features, and window size 7 with three features.

Window size 4 holds the lowest number of features (i.e., one feature) with 99.88% dimension reduction. So, in our experiment, a total of 22 features out of a total of 19,250 features have been selected for classification for window size 1. Window size 4 holds the highest number of features (i.e., seven features), with 99.2% dimension reduction. So, in our experiment, a total of 154 features out of a total of 19,250 features have been selected for classification for window size 4.

In addition to the earlier functions, we also used an average function for the sampling in each channel. In the average function, only the average value of the features within each window size is selected. So, the number of output samples for the average function is 22 features.

The average accuracy of the functions mentioned above is in the range of 55% to 73%. This can be due to the excessive reduction of features in each channel. Therefore, in future works, more features must be considered to examine the main performance. However, despite the very small selection of features, when using a single filter bank, the accuracy of 61.45% to 64.74% is obtained, which is very good considering the very small selected features.

Compared to the articles on the basic quantum neural network method [65], where the average accuracy of all individuals based on feature extraction is 66.59%, all channels and total data have been involved in the extraction, which is only 2.2% better than our sampling methods with a single filter bank and significant dimension reduction.

Compared to Jing Luo's [33] article, feature selection was selected after feature extraction from two channels, i.e., 8 and 12, using a wavelet package. The average accuracy in the range of 67–68% was obtained using a random forest classifier in the frequency range of 8–30, which is only 3–4% better than our sampling methods with a single filter bank and significant dimension reduction.

In Adham Atyabi's article [36], a Mask model has been used on 118 channels to reduce the electrodes and features. As a result, the feature reduction rate was 99%, which is near to our work (i.e., 99.88%). They obtained accuracy in the range of 63–86% for five subjects. The average accuracy was 74% in their work, which is only 1% more than our best average accuracy. However, the dimensional reduction in our work was more than their work.

5.3. Strength and Weakness of the Bond Graph Algorithm

Based on our experiments for the new BGA, BGA could achieve a very fast convergence compared to GA and PSO on eight benchmarks with models 3 and 4. It means that the formulation of two domains is suitable for converging, whereas the convergence of models 1 and 2 was worse than the others. Based on our experience, introducing a formula structure is very effective for improving the convergence.

It is possible to change parameters to find out the most suitable parameters for convergence. However, it may not have a significant effect on their performance, because their structure remains unchanged.

We obtained the best results for feature selection in one or two channels with model 3 when the selected features are sent directly to the ELM classifier. In addition, we obtained the best results for feature selection in two or three channels with model 3 when the selected features are sent directly to the CSP feature extraction. Then, the extracted features are sent

to the ELM classifier. In both cases, the BGA converged significantly faster to the global optimum compared to the other algorithms.

However, the BGA converges slower and may fall in local optimum when the feature selection is done based on four different parts, i.e., time domain, frequency domain, autoregression, and wavelet coefficients. In this case, if some parts are selected based on the binary model and some other parts are selected based on the random mode, the BGA performs poorly compared to the other algorithms. However, if all of the parts are selected based on either binary or random models, the results are better than the other algorithms. So, to solve this problem, two different models for feature selection should not be used simultaneously.

6. Conclusions

In this article, a new algorithm called BGA is introduced and implemented to reduce the feature dimensions in brain signals. This algorithm shows a better performance than GA and PSO on general functions. Our algorithm performs better compared to other algorithms when reducing the dimension to 100 features on the specific filter banks. Some scenarios for testing this algorithm were implemented, including the following. (1) Reductions of features, electrodes, and frequency range have been evaluated simultaneously for brain signals. (2) Feature selection (with algorithms) and feature extraction by time domain, frequency domain, wavelet coefficients, and autoregression have been studied on some electrodes and filter banks. (3) Feature, electrodes, and the frequency range are reduced, which is followed by the construction of new signals based on the proposed formulas. Then, the CSP is used for the extraction of features. (4) Finally, a separate experiment with the deep learning sampling method was implemented as feature selection in several layers. The dimensional reduction was performed by sampling using three general functions and two new functions. All scenarios expressed in the left hand and right hand have been evaluated between one and three channels. Our algorithm outperformed by up to 5 to 8% accuracy and had 5% better kappa compared to the other studies with the same or similar settings.

For future works, first, we will investigate even smaller sizes and different combinations of filter banks to optimize the noise reduction and increase the distinctive patterns. Second, we aim to examine new functions of deep learning sampling models with the aim of increasing accuracy and performance.

Author Contributions: A.N. conceived the ideas. Z.F., G.A., and F.H. helped to complete ideas. A.N. performed the numerical simulations. Z.F. validated the results. G.A. and A.N. wrote the original draft. F.H. and A.N. edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Natural Science Foundation of China, grant number 61876147.

Institutional Review Board Statement: No applicable.

Informed Consent Statement: No applicable.

Data Availability Statement: The data are publicly available in the link [Doi]. For analyzing and implementation, it needs to request from the corresponding author for explaining. Due to the bond graph methodology and different models, It is necessary to explain more details for implementation.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krauledet, M.; Schroder, M.; Blankert, B.; Muller, K.R. *Reducing Calibration Time for Brain-Computer Interface: A Clustering Approach*, *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2007; Volume 19, pp. 753–760.
2. Krauledet, M.; Tangermann, M.; Blankert, B.; Muller, K.R. Towards zero training for brain-computer interfacing. *PLoS ONE* **2008**, *3*, e2967.

3. Birbaumer, N.; Cohen, L.G. Brain-Computer interfaces: Communication and restoration of movement in paralysis. *J. Physiol.* **2007**, *579*, 621–636. [[CrossRef](#)] [[PubMed](#)]
4. Varkuti, B.; Guan, C.; Pan, Y.; Phua, K.S.; Ang, K.K.; Kuah, C.W.K.; Chua, K.; Ang, B.T.; Birbaumer, N.; Sitaram, R. Resting state changes in functional connectivity correlate with movement recovery for BCI and robot-assisted upper-extremity training after stroke. *Neurorehabil. Neural Repair* **2013**, *27*, 53–62. [[CrossRef](#)]
5. McCane, L.M.; Heckman, S.M.; McFarland, D.J.; Townsend, G.; Mak, J.N.; Sellers, E.W.; Zeitlin, D.; Tenteromano, L.M.; Wolpaw, J.R.; Vaughan, T.M. P300-Based brain-computer interface (BCI) event-related potentials (ERPs): People with amyotrophic lateral sclerosis (ALS) vs. age-matched controls. *Clin. Neurophysiol.* **2015**, *126*, 2124–2131. [[CrossRef](#)] [[PubMed](#)]
6. Milovanovic, I.; Robinson, R.; Fetz, E.E.; Moritz, C.T. Simultaneous and independent control of a brain-computer interface and contralateral limb movement. *Brain Comput. Interfaces* **2015**, *2*, 174–185. [[CrossRef](#)]
7. Yin, E.; Zhou, Z.; Jiang, J.; Chen, F.; Liu, Y.; Hu, D. A novel hybrid BCI speller based on the incorporation of SSVEP into the P300 paradigm. *J. Neural Eng.* **2013**, *10*, 026012. [[CrossRef](#)] [[PubMed](#)]
8. Wang, M.; Daly, I.; Allison, B.Z.; Jin, J.; Zhang, Y.; Chen, L.; Wang, X. A new hybrid BCI paradigm based on P300 and SSVEP. *J. Neurosci. Methods* **2015**, *244*, 16–25. [[CrossRef](#)] [[PubMed](#)]
9. Brumberg, J.S.; Burnison, J.D.; Pitt, K. Using Motor Imagery to Control Brain-Computer Interfaces for Communication. In *International Conference on Augmented Cognition*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 14–25.
10. Atyabi, A.; Luerssen, M.H.; FitzGibbon, S.P.; Powers, D.M.W. The Impact of PSO Based Dimension Reduction on EEG Classification. In *International Conference on Brain Informatics*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 220–231. [[CrossRef](#)]
11. Jin, J.; Wang, X.; Zhang, J. Optimal Selection of EEG Electrodes via DPSO Algorithm. In *Proceedings of the 7th World Congress on Intelligent Control and Automation, Chongqing, China, 25–27 June 2008*; pp. 5095–5099.
12. Mistry, K.; Zhang, L.; Neoh, S.C.; Lim, C.P.; Fielding, B. A Micro-GA Embedded PSO Feature Selection Approach to Intelligent Facial Emotion Recognition. *IEEE Trans. Cybern.* **2016**, *47*, 1496–1509. [[CrossRef](#)]
13. Al-Ani, A. Feature Subset Selection Using Ant Colony Optimization. 2005. Available online: <https://opus.lib.uts.edu.au/handle/10453/6181> (accessed on 10 January 2020).
14. Sabeti, M.; Boostani, R.; Katebi, S.D. A New Approach to Classify the Schizophrenic and Normal Subjects by Finding the Best Channels and Frequency Bands. In *Proceedings of the 15th International Conference on Digital Signal Processing, Wales, UK, 1–4 July 2007*; pp. 123–126. [[CrossRef](#)]
15. Yom-Tov, E.; Inbar, G. Feature selection for the classification of movements from single movement-related potentials. *IEEE Trans. Neural Syst. Rehabilitation Eng.* **2002**, *10*, 170–177. [[CrossRef](#)] [[PubMed](#)]
16. Dias, N.S.; Jacinto, L.R.; Mendes, P.M.; Correia, J.H. Feature Down Selection in Brain Computer Interface. In *Proceedings of the 4th International IEEE EMBS Conference on Neural Engineering, Antalya, Turkey, 29 April–2 May 2009*; pp. 323–326.
17. Largo, R.; Munteanu, C.; Rosa, A. CAP event detection by wavelets and GA tuning. *IEEE Int. Workshop Intell. Signal Process.* **2005**, *2005*, 44–48.
18. Hasan, B.A.S.; Gan, J.Q.; Zhang, Q. Multi-Objective Evolutionary Methods for channel selection in brain Computer interface: Some Preliminary Experimental Results. In *Proceedings of the 2010 IEEE Congress on Evolutionary Computation (CEC), Barcelona, Spain, 18–23 July 2010*; pp. 1–6.
19. Al Moubayed, N.; Hasan, B.A.S.; Gan, J.Q.; Petrovski, A.; McCall, J. Binary-SDMOPSO and its application in channel selection for Brain-Computer Interfaces. In *Proceedings of the UK Workshop on Computational Intelligence (UKCI), Colchester, UK, 8–10 September 2010*. [[CrossRef](#)]
20. Atyabi, A.; Luerssen, M.; Fitzgibbon, S.; Powers, D.M.W. Evolutionary feature selection and electrode reduction for EEG classification. In *Proceedings of the IEEE Congress on Computational Intelligence, Washington, DC, USA, 17–18 November 2012*. [[CrossRef](#)]
21. Atyabi, A.; Luerssen, M.; Fitzgibbon, S.; Powers, D.M.W. Adapting subject-independent task-specific EEG feature masks using PSO. In *Proceedings of the IEEE Congress on Computational Intelligence, Washington, DC, USA, 17–18 November 2012*. [[CrossRef](#)]
22. Atyabi, A.; Luerssen, M.; Fitzgibbon, S.P.; Powers, D.M.W. Dimension reduction in EEG data using Particle Swarm Optimization. In *Proceedings of the IEEE World Congress on Computational Intelligence (WCCI), Brisbane, Australia, 10–15 June 2012*. [[CrossRef](#)]
23. Qi, Y.; Ding, F.; Xu, F.; Yang, J. Channel and Feature Selection for a Motor Imagery-Based BCI System Using Multilevel Particle Swarm Optimization. *Comput. Intell. Neurosci.* **2020**, *2020*, 1–11. [[CrossRef](#)]
24. Yin, Z.; Liu, L.; Chen, J.; Zhao, B.; Wang, Y. Locally robust EEG feature selection for individual-independent emotion recognition. *Expert Syst. Appl.* **2020**, *162*, 113768. [[CrossRef](#)]
25. Islam, R.; Tanaka, T.; Molla, K.I. Multiband tangent space mapping and feature selection for classification of EEG during motor imagery. *J. Neural Eng.* **2018**, *15*, 046021. [[CrossRef](#)] [[PubMed](#)]
26. Baig, M.Z.; Javed, E.; Ayaz, Y.; Afzal, W.; Gillani, S.O.; Naveed, M.; Jamil, M. Classification of left/right hand movement from EEG signal by intelligent algorithms. In *Proceedings of the 2014 IEEE Symposium on Computer Applications & Industrial Electronics (ISCAIE), Penang, Malaysia, 7–8 April 2014*. [[CrossRef](#)]

27. Dupres, A.; Cabestaing, F.; Rouillard, J. Supervision of time-frequency features selection in EEG signals by a human expert for brain-computer interfacing based on motor imagery. In Proceedings of the 2016 IEEE Systems, Man and Cybernetics (SMC), Budapest, Hungary, 9–12 October 2016. [\[CrossRef\]](#)
28. Phinyomark, A.; Phukpattaranont, P.; Limsakul, C. Feature reduction and selection for EMG signal classification. *Expert Syst. Appl.* **2012**, *39*, 7420–7431. [\[CrossRef\]](#)
29. Gan, J.Q. Feature dimensionality reduction by manifold in brain-computer interface design. In Proceedings of the 3rd International Workshop on Brain-Computer Interface, Colchester, UK, 21–24 September 2006; pp. 28–29.
30. Jun, L.; Meichun, L. Common spatial pattern and particle swarm optimization for channel selection in BCI. In Proceedings of the 3rd International Conference on Innovative Computing Information and Control, Dalian, China, 18–20 June 2008; p. 457.
31. Jain, A.; Zongker, D. Feature selection: Evaluation, application, and small sample performance. *IEEE Trans. Pattern Anal. Mach. Intell.* **1997**, *19*, 153–158. [\[CrossRef\]](#)
32. Liu, H.; Yu, L. Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 491–502. [\[CrossRef\]](#)
33. Luo, J.; Feng, Z.R.; Zhang, J.; Lu, N. Dynamic frequency feature selection based approach for classification of motor im-ageries. *Comput. Biol. Med.* **2016**, *75*, 45–53. [\[CrossRef\]](#)
34. Liu, A.; Chen, K.; Liu, Q.; Ai, Q.; Xie, Y.; Chen, A. Feature Selection for Motor Imagery EEG Classification Based on Firefly Algorithm and Learning Automata. *Sensors* **2017**, *17*, 2576. [\[CrossRef\]](#) [\[PubMed\]](#)
35. Hassan, B.A.S.; Gan, J.Q. Multi-Objective particle Swarm Optimization for channel selection in Brain-Computer Interface. In Proceedings of the UK Workshop on Computational Intelligence (UKCI 2009), Nottingham, UK, 7–9 September 2009.
36. Atyabi, A.; Luerssen, M.H.; Powers, D.M. PSO-based dimension reduction of EEG recordings: Implications for subject transfer in BCI. *Neurocomputing* **2013**, *119*, 319–331. [\[CrossRef\]](#)
37. Nakisa, B.; Rastgoo, M.N.; Tjondronegoro, D.; Chandran, V. Evolutionary computation algorithms for feature selection of EEG-based emotion recognition using mobile sensors. *Expert Syst. Appl.* **2018**, *93*, 143–155. [\[CrossRef\]](#)
38. Peterson, D.A.; Knight, J.N.; Kirby, M.J.; Anderson, C.W.; Thaut, M.H. Feature Selection and Blind Source Separation in an EEG-Based Brain-Computer Interface. *EURASIP J. Adv. Signal Process.* **2005**, *2005*, 218613. [\[CrossRef\]](#)
39. Arvande, M.; Guan, C.; Ang, K.; Quek, C. Optimizing the Channel Selection and Classification Accuracy in EEG-Based BCI. *IEEE Trans. Biomed. Eng.* **2011**, *58*, 1865–1873. [\[CrossRef\]](#) [\[PubMed\]](#)
40. Yang, Y.; Kyrgyzov, O.; Wiart, J.; Bloch, I. Subject-Specific Channel Selection for Classification of Motor Imagery Electroencephalographic Data. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2013), Vancouver, BC, Canada, 26–31 May 2013; pp. 1277–1280.
41. Chin, Z.Y.; Ang, K.K.; Wang, C.; Guan, C.; Zhang, H. Multi-Class filter bank common spatial pattern for four-class motor imagery BCI. In Proceedings of the 31st Annual International Conference of IEEE EMBS, Minneapolis, MN, USA, 2–6 September 2009; pp. 571–574. [\[CrossRef\]](#)
42. Rejer, I. EEG Feature Selection for BCI Based on Motor Imaginary Task. *Found. Comput. Decis. Sci.* **2012**, *37*, 283–292. [\[CrossRef\]](#)
43. Eslahia, S.V.; Dabanloob, N.J.; Maghooli, K. A GA-based feature selection of the EEG signals by classification evaluation, Application in BCI system. *arXiv* **2019**, arXiv:1903.02081v1.
44. Wang, J.; Xue, F.; Li, H. Simultaneous Channel and Feature Selection of Fused EEG Features Based on Sparse Group Lasso. *BioMed Res. Int.* **2015**, *2015*, 1–13. [\[CrossRef\]](#)
45. Lokman, M.; Dabag, A.; Ozkurt, N.; Najeeb, S.M.M. Feature Selection and Classification of EEG Finger Movement Based on Genetic Algorithm. In Proceedings of the 2018 Innovations in Intelligent Systems and Applications Conference (ASYU), Adana, Turkey, 4–6 October 2018.
46. Amarasinghe, K.; Sivils, P.; Manic, M. EEG Feature Selection for Thought Driven Robots using Evolutionary Algorithms. In Proceedings of the IEEE 9th International Conference on Human System Interactions (HSI), Portsmouth, UK, 7 July 2016.
47. Chin, Z.Y.; Ang, K.K.; Wang, C.; Guan, C. Discriminative Channel Addition and Filter Bank Common Spatial Pattern in Motor imagery BCI. In Proceedings of the 36th Annual IEEE International Conference of Engineering in Medicine and Biology Society (EMBC), Chicago, IL, USA, 26–30 August 2014; pp. 1310–1313.
48. Ang, K.K.; Chin, Z.Y.; Zhang, H.; Guan, C. Filter Bank Common Spatial Pattern (FBCSP) algorithm using online adaptive and semi-supervised learning. In Proceedings of the International Joint Conference on Neural Networks 2011, San Jose, CA, USA, 31 July–5 August 2011; pp. 392–396. [\[CrossRef\]](#)
49. Wang, J.; Feng, Z.R.; Lu, N. Feature extraction by Common Spatial Pattern in Frequency Domain for Motor Imagery Tasks Classification. In Proceedings of the 29th Chinese Control And Decision Conference (CCDC), Chongqing, China, 28–30 May 2017. Electronic ISSN: 1948-9447.
50. Tang, Z.; Li, C.; Sun, S. Single-Trial EEG classification of motor imagery using deep convolutional neural networks. *Optik* **2017**, *130*, 11–18. [\[CrossRef\]](#)
51. Broenink, J.F. Bond Graph modelling in Modelica. In Proceedings of the 9th European Simulation Symposium, Passau, Germany, 19–22 October 1997; pp. 137–141.
52. Broenink, J.F. Object-Oriented modelling with bond graph and modelica. In Proceedings of the International Conference on Bond Graph Modelling ICBGM'99, San Francisco, CA, USA, 1 January 1999; Volume 31, pp. 163–168.
53. Broenink, J.F. Introduction to Physical Systems Modelling with Bond Graphs. *SiE Whitebook Simul. Methodol.* **1999**, *31*, 2.

54. Zeiler, M.D.; Fergus, R. Stochastic Pooling for Regularization of Deep Convolutional Neural Network. *arXiv* **2013**, arXiv:1301.3557v1.
55. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [[CrossRef](#)]
56. Garrett, D.; Peterson, D.A.; Anderson, C.W.; Thaut, M.H. Comparison of Linear, Nonlinear, and feature Selection Methods for EEG Signal Classification. *IEEE Trans. Neural Syst. Rehabil. Eng.* **2003**, *11*, 2. [[CrossRef](#)] [[PubMed](#)]
57. Eberhart, R.; Kennedy, J. A New Optimizer Using Particle Swarm Theory. In Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS'95, Nagoya, Japan, 4–6 October 1995.
58. Kennedy, J.; Eberhart, R. A Discrete Binary Version of The Particle Swarm Algorithm. In Proceedings of the 1997 IEEE International Conference on Systems, Man and Cybernetics, Orlando, FL, USA, 12–15 October 1997.
59. Zhang, H.; Zhu, Y.; Chen, H. Root growth model: A novel approach to numerical function optimization and simulation of plant root system. *Soft Comput.* **2013**, *18*, 521–537. [[CrossRef](#)]
60. Naeem, M.; Brunner, C.; Leeb, R.; Graimann, B.; Pfurtscheller, G. Seperability of four-class motor imagery data using independent components analysis. *J. Neural Eng.* **2006**, *3*, 208–216. [[CrossRef](#)]
61. Huang, G.-B.; Zhou, H.; Ding, X.; Zhang, R. Extreme Learning Machine for Regression and Multiclass Classification. *IEEE Trans. Syst. Man Cybern. Part B Cybern.* **2011**, *42*, 513–529. [[CrossRef](#)] [[PubMed](#)]
62. Huang, G.-B.; Zhu, Q.-Y.; Siew, C.-K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [[CrossRef](#)]
63. Narayanan, A.; Moore, M. Quantum-Inspired genetic algorithms. In Proceedings of the IEEE International Conference on Evolutionary Computation, Nagoya, Japan, 20–22 May 1996. [[CrossRef](#)]
64. Han, K.-H.; Kim, J.-H. Genetic quantum algorithm and its application to combinatorial optimization problem. In Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512), La Jolla, CA, USA, 16–19 July 2000.
65. Gandhi, V.; Prasad, G.; Coyle, D.; Behera, L.; McGinnity, T.M. Quantum Neural Network-Based EEG Filtering for a Brain–Computer Interface. *IEEE Trans. Neural Networks Learn. Syst.* **2013**, *25*, 278–288. [[CrossRef](#)] [[PubMed](#)]
66. Lotte, F.; Guan, C. Regularizing Common Spatial Patterns to Improve BCI Designs: Unified Theory and New Algorithms. *IEEE Trans. Biomed. Eng.* **2010**, *58*, 355–362. [[CrossRef](#)] [[PubMed](#)]