Multi-Objective Particle Swarm Optimization for Channel Selection in Brain-Computer Interfaces

Bashar Awwad Shiekh Hasan and John Q. Gan
School of Computer Science and Electronic Engineering
University of Essex
Wivenhoe Park, Colchester, CO4 3SQ
UK
bawwad@essex.ac.uk

Abstract

This paper presents a novel application of a multi-objective particle swarm optimization (MOPSO) method to solve the problem of effective channel selection for Brain-Computer Interface (BCI) systems. The proposed method is tested on 6 subjects and compared to another search based method, Sequential Floating Forward Search (SFFS). The results demonstrate the effectiveness of MOPSO in selecting a fewer number of channels with insignificant sacrifice in accuracy, which is very important to build robust online BCI systems.

1 Introduction

Non-invasive Brain-Computer Interface (BCI) uses Electroencephalography (EEG) signals to capture the brain signal associated with predefined mental tasks. The number of channels used by an EEG system can vary according to the experiment held and the hardware design. It usually ranges between 16 and 256 channels. According to Event Related Desynchronization/Synchronization (ERD/ERS) research [1], motor imagery experiments can use only the channels at the contralateral hemispheres, which can be as few as 3-5 channels. Using a lot of channels for recording can be useful for medical and diagnostic purposes. For BCI systems and especially when building online systems, the number of channels should be as minimum as possible.

In order to avoid a large number of channels one can choose several electrode positions that are known from neuroscience and psychology studies. Although this approach can be very useful, it ignores the fact that different subjects respond differently and the optimal positioning of the electrodes may vary. The other way around this problem is to use a large number of channels and use some methods to reduce the dimensionality of the input features or to select the best set of channels for each subject.

Common Spatial Patterns (CSP) [2] is a well known spatial filter that is widely used in the BCI community. CSP is useful for channel selection as it can be used to filter out the channels that provide less discriminate data. CSP requires the data from all the channels to be available online before the dimensionality is reduced.

In [3], the author shows that feature selection is advantageous over dimensionality reduction in terms of interpretability. Feature selection (and similarly electrode selection) using several search methods has been used frequently in the literature.

In [4] Digital Particle Swarm Optimization (DPSO) was used, where each particle contained a number of binary variables, (which is equal to the number of channels) and cross validation results were used as the fitness function. In [5] a mixture of CSP and PSO based method was used for channel selection. In [6] SFFS based method was employed for channel selection. SFFS [7] has been well recognized as one of the best feature selection methods [8] [9].

In this paper a directed search method, Multi-Objective Particle Swarm Optimization (MOPSO) is used for channel selection. The competing criteria are cross-validation accuracy and the number of channels selected. The method is compared to SFFS. The results show the usefulness of the multi-objective approach in minimizing the number of channels used with insignificant sacrifice in classification accuracy.

2 Methods

In this section we outline the two search methods used and detail their application on the channel selection problem.

2.1 Multi-Objective Particle Swarm Optimization

The particle swarm optimization (PSO) algorithm is a population-based search algorithm based on the simulation of the social behavior of birds within a flock. In PSO, individuals, referred to as particles are flown through the hyperdimensional search space guided by a leader(s) whose performance affects the speed and direction of the other particles in the population. The position of each particle changes according to its own experience and that of the neighbors. A particle represents a possible solution of the problem, while its position is a multi-dimensional vector where each dimension is a variable in the problem space.

Let $\vec{x_i}(t)$ be the position of particle p_i, at time t. The position of p_i is then updated at time $t+1$ by adding the velocity $\vec{v_i}(t+1)$ to the current position.

$$\vec{x_i}(t+1) = \vec{x_i}(t) + \vec{v_i}(t+1) \tag{1}$$

The velocity vector reflects the socially exchanged information and is defined in the following way [10]:

$$\begin{aligned}\vec{v_i}(t+1) = W\vec{v_i}(t) + C_1 r_1(\vec{x}(pbest_i) - \vec{x_i}(t)) \\ + C_2 r_2(\vec{x}(leader) - \vec{x_i}(t))\end{aligned} \tag{2}$$

where $pbest_i$ is the personal best performance of the particle, *leader* is the particle with the best performance, $r_1, r_2 \in [0,1]$ are random values and W, C_1, C_2 are weights. In order to evaluate particles and hence to order them and select the new leader(s), a fitness function is required so as to evaluate the objective of the search problem (e.g., classification accuracy, etc.).

PSO, as has been introduced so far is a single objective optimization method. In order to extend it to multi-objective problems it should be modified. In [10] a survey of the MOPSO methods is presented. In this paper we adopt the approach in [11], where a Pareto dominance and crowding factor approaches are used for the selection of leaders, a mutation operator, and $\epsilon-$dominance concept are incorporated in the optimization method.

In a multi-objective configuration the output of the fitness function should include a separate evaluation of each objective. The solutions might also be constrained as well depending on the problem in hand.

2.2 Sequential Floating Forward Search

SFFS is a well recognized search method for feature selection. It has been widely applied for feature selection in BCI [6] [12]. SFFS has two phases. The first phase is growing phase where at each step a channel is added to the previously selected best set of channels and the cross-validation test is done. The result of this phase is a new set of channels that has the best accuracy. The second phase is the pruning phase, where for each step a channel is removed from the already best selected channels, and the evaluation criteria is tested. The pruning continues while removing channels is enhancing the criteria. Both phases are repeated until a maximum number of channels is selected or a pre-defined number of cycles was reached.

3 Experiments

3.1 Dataset

In this study the dataset1 of BCI Competition IV was used. The challenge is the classification of continuous EEG without trial structure. The dataset is divided into training data and testing data. The calibration data are synchronous trials for 7 subjects (3 of the datasets are synthesized data). The evaluation data are soft-cued trials. For each subject 3 motor imagery tasks (right hand, left hand and foot, where the foot side was chosen by the subject) were recorded but only the most separable 2 were provided. 59 channels were used to record EEG data. As the purpose here is to test the channel selection method, the evaluation data were not used and rather the calibration data were used. More technical details and information about data acquisition and recording method can be found in [13].

3.2 Feature Extraction and Classification

The original dataset was sampled at $1000Hz$. Here we used another downsampled version (at $100Hz$) provided by the authors. Autoregressive features of order 6 were extracted, with a 4 samples shift window, resulting in 25 samples per second.

Linear Discriminant Analysis (LDA) was used to classify the extracted features. When the number of features is more than 20, Principle Component Analysis (PCA) was used to reduce the dimensionality of the input.

3.3 MOPSO and SFFS for channel selection

The application of SFFS is straight forward. 4-fold cross validation test was used as the evaluation criterion and the maximum number of channels accepted is 20.

For MOPSO, the number of particles was set to 59 (which was chosen arbitrarily), the initial number of archive was set equal to the number of particles, the maximum number of iterations was set to 200, and the perturbation index (which is a parameter for the mutation operator used in MOPSO, this technique is borrowed from genetic algorithms and is used to control the generation of the offspring) is set to 0.5. For this method the weight W is set to a random value in the range $[0.1, 0.5]$, C_1 and C_2 are set to random values in the range $[1.5, 2.0]$. These parameter values were set as recommended in [11].

Each particle contains 59 binary variables with each representing a channel, whose value can be 1 (the channel is selected) or 0 (the channel is not selected). The first objective is set to $1-$ cross validation result so that the method will minimize both objectives. The second objective is the number of selected channels. The number of channels selected by each particle is constrained to be in the range $[1, 30]$. The goal is to find a set of solutions that minimize these two objectives.

4 Results

The 7 subjects were named as "a", "b", "c", "d", "e", "f", "g". Subject "e" was not included in the study as the size of data is different from the other datasets. Table 1 shows the different solutions when using MOPSO, th higlighted rows are the ones considered the "best" solution in terms of accuracy and channel number. Figure 1 shows the solutions provided by MOPSO for the 6 subjects, these solutions are the resulted Pareto front for each subject. Table 2 presents the results when using SFFS.

Due to space limit we can not include figures from all the subjects. Therefore, without losing of generality, figures were chosen arbitrarily.

Figures 2 and 3 show the selected channels using the two referred methods for subjects "a" and "c". Circled channels are the non-selected channels. The left-headed arrows are the channels selected by MOPSO (one of the solutions). The right-headed arrows channels are the ones selected by SFFS. The up arrows are the channels selected by both methods.

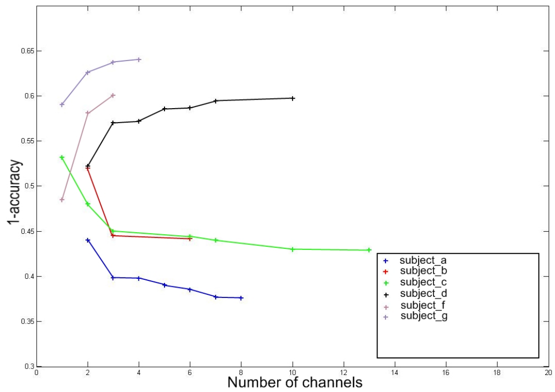

Figure 1: MOPSO solutions

Figures 4 and 5 show the frequency of the selected channels using MOPSO (all the solutions) and SFFS for subjects "a" and "b". The blue points are the frequency of MOPSO selected channels over all the solutions. The red points are the channels selected via SFFS. As SFFS results in only one solution the frequency is set to 5 (for presentation purposes only). The histograms show that some of the frequently selected channels by MOPSO are also selected by SFFS.

Figures 6 and 7 demonstrate the change of the objective/fitness function values over time for subject "a" and "f". It should be noted that only the particle with the best fitness at each step was drawn. The graphs show the effectiveness of the optimization method in minimizing both objectives at the same time. Figure 6 show how the perturbation index helps avoiding local minima in objective2.

Looking closely at the results, the assumptions of Wilcoxon rank sum statistical test hold (the two sets has the same median). Wilcoxon rank sum test was applied on the resulted accuracies (the best selected by MOPSO and SFFS), the test showed insignificant difference ($p=0.3939$). On the other hand there is a clear difference between the average number of channels selected using MOPSO (5.5 ∓ 2.9496) and SFFS (10 ∓ 5.4772). This supports the claim that MOPSO can achieve similar classification results with smaller number of channels.

5 Discussion and Conclusion

The results show that MOPSO can in general select a much smaller number of channels with insignificant sacrifice in classification accuracy. This is due to the nature of MOPSO as a multi-

Subject	No. of Selected Channels	Accuracies
a	2	0.56
	3	0.602
	4	0.6024
	5	0.6097
	6	0.615
	7	**0.6231**
	8	0.6239
b	2	0.48
	3	**0.555**
	6	0.558
c	1	0.468
	2	0.52
	3	0.55
	6	0.556
	7	0.56
	10	**0.57**
	13	0.571
d	2	0.522
	3	0.57
	4	0.572
	5	0.5859
	6	0.5869
	7	**0.594**
	10	0.5969
f	1	0.4848
	2	0.581
	3	**0.6007**
g	1	0.59
	3	**0.6371**
	2	0.626
	4	0.64
average best	**5.5**	**0.5967**
std. best	**2.9496**	**0.0310**

Table 1: Results using MOPSO

Subject	No. of Selected Channels	Accuracy
a	12	0.635
b	16	0.5812
c	15	0.5767
d	10	0.6014
f	3	0.6298
g	4	0.6493
avg.	**10**	**0.6122**
std.	**5.4772**	**0.0301**

Table 2: Results using SFFS

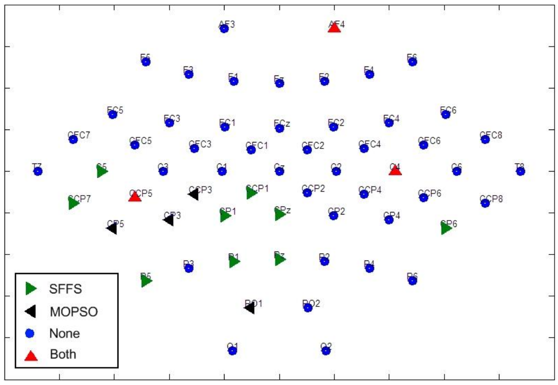

Figure 2: selected channels using both methods for subject "a"

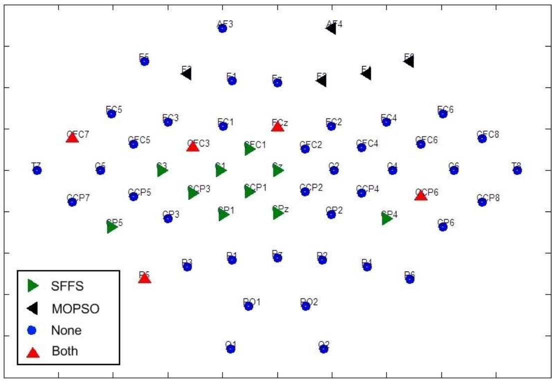

Figure 3: selected channels using both methods for subject "c"

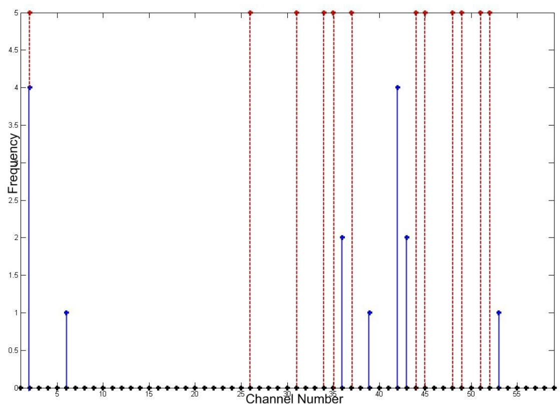

Figure 4: histogram of channels selected using both methods for subject "a"

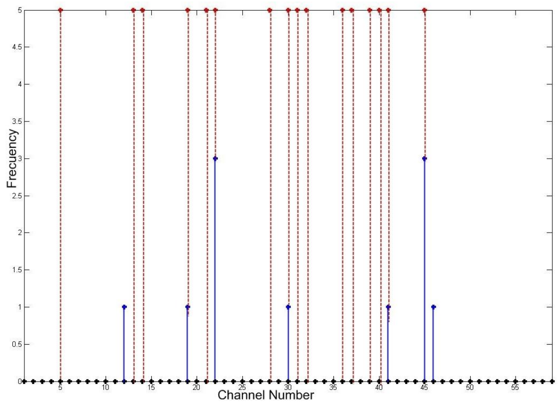

Figure 5: histogram of channels selected using both methods for subject “b”

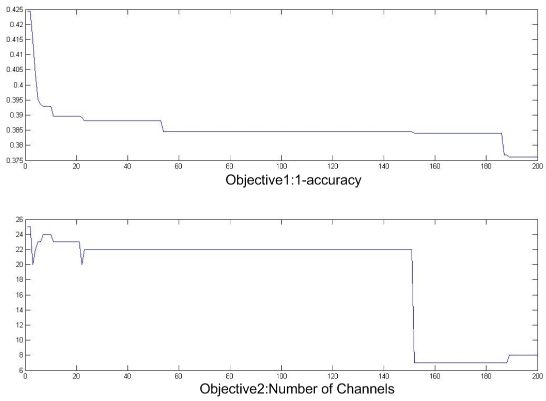

Figure 6: change of fitness function values over time for subject “a”

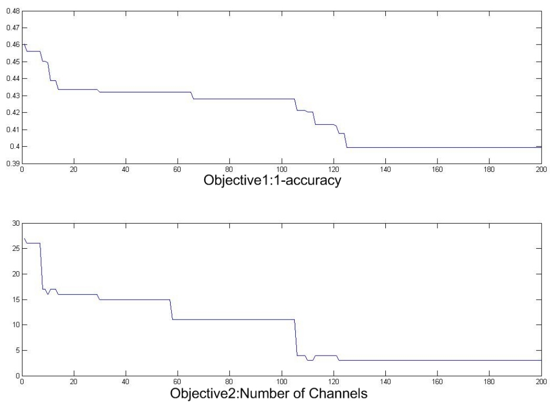

Figure 7: change of fitness function values over time for subject “f”

objective optimization method, where it can simultaneously work on minimizing the two objectives. Figures 1, 6 and 7 show the effectiveness of the optimization method in minimizing the objective functions.

In spite of the pruning phase of SFFS, it appears to be better to use multi-objective optimization when the number of channels required for online BCI systems should be minimal.

There are some common channels selected using SFFS and MOPSO. These channels can though be considered the most useful for classification. This also shows that both methods can reach these important channels with their different approaches. It must be noted though that SFFS has a sequential feature meaning that the currently selected channel can affect the selection of the next channel(s), MOPSO on the other hand has a perturbation index which plays the role of mutation rate in evolutionary methods in adding some randomness to the optimization method which might result in better scanning of the search space.

The results strongly show that the selected channels varies a lot among subjects. This actually supports the claim that different subjects have different optimal positioning of electrodes.

Acknowledgements

This work is part of the project “Adaptive Asynchronous Brain Actuated Control” funded by UK EPSRC. First author’s study is funded by Aga Khan Foundation.

To implement the methods jMetal (http://jmetal.sourceforge.net/) was used, which contains state-of-the-art implementation of multi-objective optimization methods, along with our own Matlab code for pre/post processing and classification.

References

[1] C. Neuper and G. Pfurtsheller. Motor imagery and ERD. *Handbook of Electroencephalography and Clinical Neurophysiology- Event-related Desynchronization*, pages 303–325, 1999.

[2] G. Dornhege, B. Blankertz, M. Krauledat, and et al. Combined optimization of spatial and temporal filters for improving brain-computer interfacing. *IEEE Trans. Biomed. Eng.*, pages 2274–2281, 2006.

[3] J.Q. Gan. Feature dimensionality reduction

by manifold learning in brain-computer interface design. In *Proc. of the 3rd International Workshop on Brain-Computer Interfaces*, pages 28–29, 2006.

[4] J. Jin, X. Wang, and J. Zhang. Optimal selection of eeg electrodes via DPSO algorithm. In *Proceedings of the 7th World Congress on Intelligent Control and Automation*, 2008.

[5] L. Jun and L. Meichun. Common spatial pattern and particle swarm optimization for channel selection in BCI. In *Innovative Computing Information and Control, 2008. ICICIC '08. 3rd International Conference on*, pages 457–457, 2008.

[6] M. Dyson, F. Sepulveda, J.Q. Gan, and S.J. Roberts. Sequential classification of mental tasks vs. idle. In *International IEEE EMBS Conf. on Neural Engineering*, pages 351–354, 2009.

[7] P. Pudil, J. Novovicova, and J. Kittler. Floating search methods in feature selection. *Pattern Recognition Letters*, pages 1119–1125, 1994.

[8] A. Jain and D. Zongker. Feature selection: evaluation, application, and small sample performance. *IEEE Trans. on PARMI*, (2):153–158, 1997.

[9] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. on Knowledge and Data Engineering*, (4):491–502, 2005.

[10] M. Reyes and C. A. Coello Coello. Multi-objective particle swarm optimizers: A survey of the state-of-the-art. *International Journal of Computational Intelligence Research*, 2:287–308, 2006.

[11] M. Reyes and C.A. Coello Coello. Improving PSO-based multi-objective optimization using crowding, mutation and ϵ-dominance. In C.A. Coello, A. Hernández, and E. Zitler, editors, *Third International Conference on Evolutionary MultiCriterion Optimization, EMO 2005*, volume 3410 of *LNCS*, pages 509–519. Springer, 2005.

[12] B. Awwad Shiekh Hasan, M. Dyson, T. Balli, and J.Q. Gan. A study via feature selection on the separability of approximate entropy for brain-computer interfaces. In *The UK Workshop on Computational Intelligence (UKCI2008)*, pages 189–194, 2008.

[13] B. Blankertz, G. Dornhege, M. Krauledat, K. Mller, and G. Curio. The non-invasive Berlin brain-computer interface: fast acquisition of effective performance in untrained subjects. *NeuroImage*, 37:539–550, 2007.