

Single Trial Visual Evoked Potential Extraction by Negentropy Maximisation of Independent Components

RAMASWAMY PALANIAPPAN
Computer Science Department
University of Essex
Wivenhoe Park, CO4 3SQ
UNITED KINGDOM
rpalan@essex.ac.uk

Abstract: - A novel method based on genetic algorithm maximising negentropy function is proposed to perform blind extraction of Visual Evoked Potential (VEP) from background electroencephalogram (EEG) on a single trial basis for use in speller BCI designs. This method is a simpler and rapidly converging alternative to existing independent component analysis (ICA) algorithms that use neural learning algorithms. In the method, binary coded genetic algorithm maximises the negentropy of the extracted signals on a deflationary basis to obtain the inverse of the mixing matrix. To show the validity of the proposed method, the proposed method was applied to both simulated and real EEG (from BCI competition 2003 – data set Iib) signals. The results show significant SNR enhancement of P300 wave in VEP using only as few as three channel (Fz, Pz, Cz) single trial data.

Key-Words: - Electroencephalogram, Genetic Algorithm, Negentropy, Visual Evoked Potential

1 Introduction

In recent years, the trends in biological based computing research have attracted much interest in Brain Computer Interface (BCI) designs using newer type of methodologies based on EEG signals. These BCI designs could be used by individuals to communicate with their external surroundings. In general, BCIs that utilise the brain signals could be divided into different categories: BCI technology using EEG extracted during mental tasks [1], BCI technology using mu and similar rhythms [2] and BCI technology using visual evoked potential (VEP) signals [3].

The common BCI technology for speller BCI designs, i.e. those that translate brain signals into alphanumeric letters is VEP based BCI. Evoked potentials are typically generated by the nervous system in response to motor or sensory stimulation. The stimulus modality can be of visual, auditory, somatosensory, or motor, of which the visual stimulus gives rise to VEP signals.

The technology uses P300 parameters extracted from VEP signals. The P3 (or P300) component is the third positive component within VEP, which normally occurs at between 300 and 600 ms after the stimulus, and reaches its maximum value in the midline parietal area of the brain [4]. This component is evoked when a stimulus is recognised and ‘oddball’ experimental procedures are commonly employed for this purpose.

However, BCI technology using VEP signals suffers from one serious drawback. As it uses P300 parameters, it requires signal averaging from many trials to reduce the random effects of ongoing EEG, which is many times higher in amplitude than the stimulus correlated VEP. This leads to system complexity and higher computational time.

There are many proposed methods for analysing VEP on a single basis like principal component analysis (PCA) [5] and independent component analysis (ICA) [6,7]. Several advantages of these single trial approaches would be the lower complexity and time requirement as only one trial would be needed (the ‘spelled’ meaning of the VEP signal will be found quickly) and these also avoid the problem of inter-trial variation in latency, which might distort the meaningful information in averaged VEP signal from a number of trials.

ICA seems to be the most successful in extracting the P300 wave for use in the VEP based speller BCI [7]. However, ICA uses neural learning methods, which are algorithmically complex and time consuming. Here, a novel independent component extraction algorithm based on genetic algorithm (GA) is proposed to extract P300 wave from ongoing EEG. The method is proposed as a simpler and rapidly converging alternative to neural learning ICA methods. The use of GA with kurtosis maximisation has been proposed in [8] but never applied for any application. Mutual information (MI) minimisation using GA was proposed and

applied to speech. Other functions like Kullback-Leibler (KL) divergence, maximization of maximum likelihood estimation, tensor and non-linear PCA [9-11] have been proposed in ICA. But these methods use neural learning approaches and not GA. The infomax method is used to enhance P300 wave for use in the VEP based speller BCI [7].

Kurtosis computation is the simplest and fastest method for use in ICA especially when using GA. Kurtosis could take positive or negative values for super-Gaussian and sub-Gaussian signal, respectively. Instead of kurtosis, negentropy function is used here. Negentropy is a measure that could be obtained from differential entropy. It is always non-negative and is zero for a Gaussian variable. Though the computation of negentropy is difficult as it involves probability density function, it can be approximated by using cumulants to give the approximation as [10]:

$$\text{Negentropy}(x) \approx \frac{1}{12} E[x^3]^2 + \frac{1}{48} \text{kurt}(x)^2, \quad (1)$$

where x is the zero-mean signal and $\text{kurt}(x)$ is the kurtosis of signal x .

2 Methodology

It is well known in the field of ICA that a mixed signal will have more gaussian behaviour than its independent components [10]. Therefore, by using negentropy as a measure of gaussianity, we could separate the mixed signals into the independent components. The mixing matrix is iteratively improved for source separation using increments in negentropy as a measure of non-gaussian behaviour where negentropy is used as the fitness function to be maximised by the GA. The method will work as long as one of the signals is non-gaussian, i.e. with non-zero negentropy.

The mixing matrix is represented by binary chromosomes converted to real-value that iterates through the GA operators: selection, crossover, mutation and inversion [12,13], to maximise the fitness function given by the negentropy of the independent components. The method relies only on GA, which is simple as compared to existing ICA techniques that use complicated neural learning algorithms.

Consider the VEP corrupted with ongoing EEG signal to be represented in matrix form as

$$[Y]=[A] [X], \quad (2)$$

where $[A]$ is the arbitrary noise mixing matrix, $[X]$ is the matrix containing VEP plus ongoing EEG signals, while $[Y]$ is the matrix of the observed (i.e. recorded) signals.

In ICA methods, the task is to obtain the inverse of matrix $[A]$ to reconstruct the original matrix $[X]$. In this proposed method, deflation approach is used, i.e. instead of the entire matrix $[X]$, signal in matrix $[X]$ will be extracted (reconstructed)¹ one by one as GA iterates to give the signal with the highest negentropy at a time. In other words, GA will iterate in such that the only one output signal will be obtained at a time. Once this signal is extracted, it could be removed and the process repeated to obtain the other independent signals. This is an advantage as compared to using other existing symmetric ICA methods (like using minimisation of MI) as it avoids the extraction of all independent signals, which would be useful if the required signal is extracted earlier.

How this works could be understood with the following example. Assume

$$[Y] = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, [X] = \begin{bmatrix} EEG \\ VEP \end{bmatrix}, \text{inv}[A] = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \quad (3)$$

Then

$$X_{EEG} = a_{21} * y_1 + a_{22} * y_2 + a_{23} * y_3 + a_{24} * y_4. \quad (4)$$

As could be seen from (4), GA will iterate to give only the coefficients a_{21} , a_{22} , a_{23} and a_{24} to reconstruct the signal that maximises negentropy. If this is VEP, then the process is stopped. If not, it is continued with the removal of the extracted signal to obtain the VEP signal.

GA is a computational model inspired by evolution and is based on genetic processes of biological organisms. It is an adaptive method, which may be used to solve search and optimisation problems. Over many generations, natural populations evolve according to the principles of natural selection and "survival of the fittest" [13]. GA requires fitness or objective function, which provide a measure of performance of the population individuals.

The use of GA will now be explained using the example discussed with eqs. (3) and (4). GA operate on the coding of parameters rather than the parameter itself. These parameters or genes, which are known as chromosomes, are a string of values representing potential solutions to a problem. Here,

¹ With different scale factors.

binary coding is used here for convenience but in future studies, it is planned to migrate the method to real-values (continuous) coding.

These certain number of genes (bits) will be used to represent each of the coefficients in $inv [A]$ as in (4). Assuming that 4 signals are observed as in (3) and 6 bits² are used for each coefficient, then each chromosome will have 24 bits. A population will consist of a certain number of these chromosomes, say 20. This is shown in Figure 1.

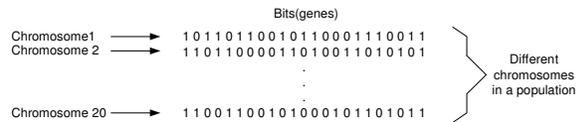


Fig. 1. Genes and chromosomes in a population.

The genes values in the chromosomes of the initial population will be set randomly. These bit valued genes will be converted to real-valued in the range of $[0,1]$ ³. Next, these 4 real-valued gene values are used in (4) to generate 20 independent signals and the kurtosis of each signal is computed. The fitness of each of the 20 chromosomes will be set to these kurtosis values.

Selection (reproduction) operator is performed next based on these fitness values. During this reproductive phase of the GA, chromosomes are selected from the population and recombined, producing offspring chromosomes that will comprise the population for next generation. Selection is applied randomly from the initial population using a scheme that favours the more fit chromosomes (evaluated using the fitness function) to create the intermediate population. Good chromosomes will probably be selected several times in a generation while the poor ones may not be selected at all.

There are a number of ways of performing the parent selection process. The common methods are roulette wheel selection and rank based methods such as tournament selection. Both these selection operators are used here. In tournament selection, certain number of chromosomes are picked randomly (in this case, 5) and the best chromosome (i.e. with the highest fitness) is stored. Since 50% of the new population will be selected here using this

method, this procedure is repeated for 10 times (with the entire population) to obtain 10 chromosomes, where there maybe more than one similar chromosome. The rest half of the new population will be selected using the roulette-wheel method. In this method, the fitness values of each chromosome are cumulatively added into a roulette wheel and when the wheel spins, there are higher chances for the higher fitness chromosomes to be selected. Here, a random number is used to represent the wheel spin and the particular chromosome with the cumulative fitness range denoted by the number will be selected. This is repeated 10 times to add to existing 10 chromosomes from tournament selection.

Crossover process is intended to simulate the exchange of genetic material that occurs during biological reproduction. Here, pairs in the breeding population are mated randomly with a crossover rate. There are a few popular types of crossover techniques like one point, two point and uniform crossover. In this work, uniform crossover is used.

The uniform crossover scheme works as follows. A randomly generated bit string called the crossover mask is used to generalize the process. A bit value of 1 in this bit string indicates that corresponding bits in the parents are to be exchanged while a 0 bit indicates no bit interchange. Here, uniform crossover is applied to two randomly chosen chromosomes to produce two new offspring if a random number generated is less than the crossover probability. This is repeated (with the used parent chromosomes not included) for 10 times. The parent chromosomes that have not been crossed over are kept intact in the population.

Mutation is important to prevent evolutionary dead ends. Most mutations are damaging rather than beneficial; so mutation probability must be. It works by randomly selecting a bit with a certain mutation probability in the string and reversing its value. Mutation is applied to the randomly chosen bit in a randomly chosen chromosome. The mutation is applied if a randomly generated number does not exceed the mutation probability. Inversion works by reversing the bits with a certain inversion rate between two randomly chosen points in a randomly chosen chromosome. It is applied to the bits between two randomly chosen points in a randomly chosen chromosome based on the inversion probability. This probability must be set to lower than mutation probability as it effects are more damaging. Figure 2 shows a block diagram of the GA method.

² Higher number of bits will be more accurate but will increase the computation time.

³ This range is suitable since the extracted signals of all EEG methods are likely to be scaled anyway.

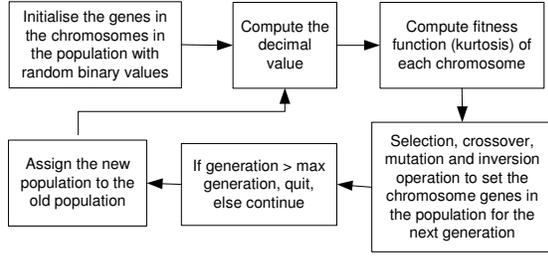


Fig. 2. Block diagram on the use of GA here.

3 Experiment, Results and Discussion

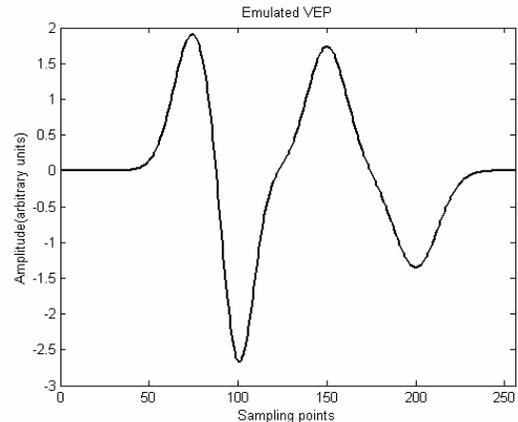
To show the effectiveness of the proposed method, two simulations were conducted. Table 1 summarises the used GA parameters. In the first simulation study, VEP and ongoing EEG were created and added. VEP was created using different gaussian waves, while EEG was created using an 8th order autoregressive model based on studies in [14].

Table 1: Genetic algorithm parameters

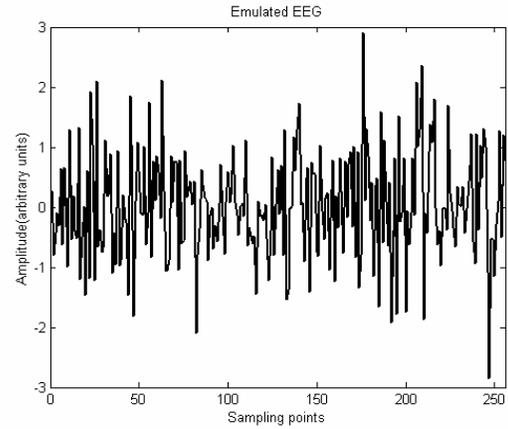
Coding of genes	Binary coding converted to real value [1,0] for fitness computation
Fitness function	Kurtosis (4th order moment of the signal)
Population size	20
No. of genes	6 bits for each gene
Reproduction (selection)	Roulette wheel (50% of population) and tournament selection (50% of population)
Crossover type and rate	Uniform crossover, 0.5
Mutation type and rate	Mutate randomly selected bits, 0.01
Inversion type and rate	Inversion between 2 randomly selected points, 0.01
Convergence	Maximum 50 generations

The values in the mixing matrix were randomly set in the range [0,1]. Figure 3 shows VEP, EEG, VEP+EEG, VEP with the application of PCA and VEP with the application of GA based ICA. It can be seen that PCA improves the SNR to 6.33 dB from -5.95 dB. But GA based ICA improves the SNR further to 25.04 dB. It must be noted that the reconstructed signal will be scaled/inverted (just like other ICA methods). This would be due to the

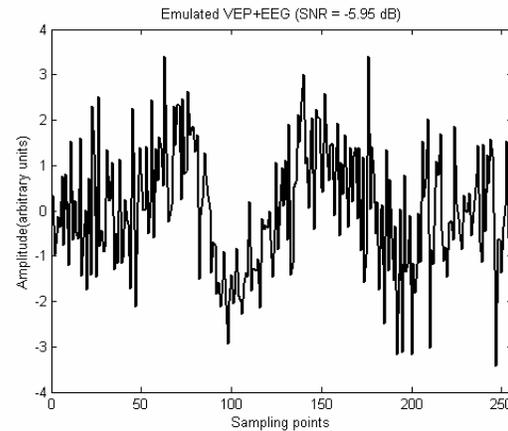
negative/positive coefficients of the inverse of matrix $[A]$ given by GA.



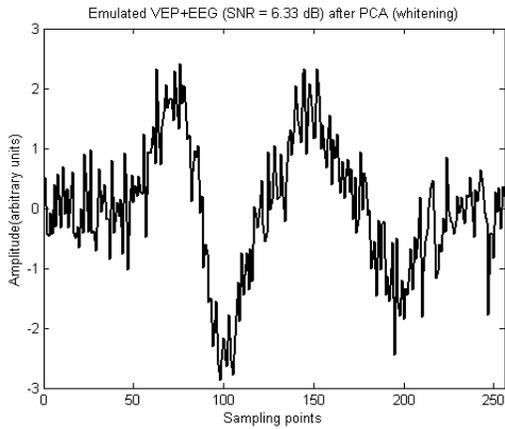
(a)



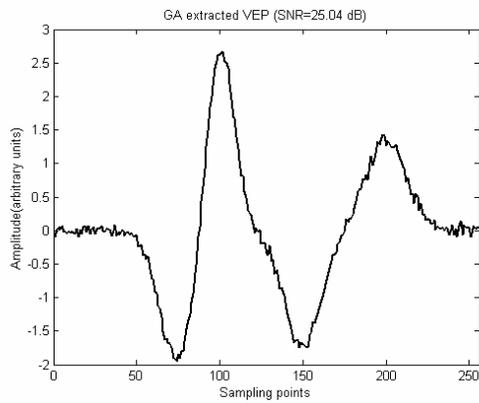
(b)



(c)



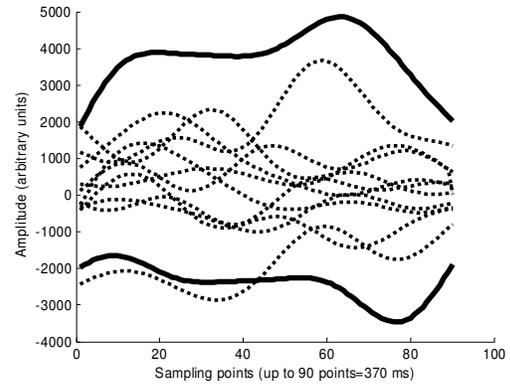
(d)



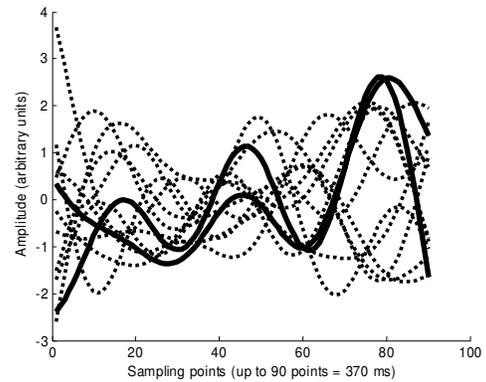
(e)

Fig. 3 (a) VEP (b) EEG (c) VEP+EEG (d) VEP after PCA (e) VEP after using GA based ICA (inverted).

In the next simulation study, the proposed method was applied to extract VEP from ongoing EEG using the BCI competition 2003–data set IIB [3], which is available online at http://ida.first.fraunhofer.de/projects/bci/competition_n_iii. Sixty-four channel signals from a single trial are chosen from this dataset; these represent the recordings when the subject focuses on a character in 6 rows and 6 columns. The data are band-pass filtered (2–8 Hz) and 275–370 ms window after stimulus onset is used; these are based on studies in [7]. Fig. 4 shows the results with averaging and with GA based ICA. It could be seen that the P300 amplitudes for target row and column are higher than non-targets but this is not the case for averaging where only the target column is higher.



(a)



(b)

Fig. 4.(a) With averaging from 15 trials (b) with GA based ICA with single trial. Note in (b), P300 responses for both the target column and row are higher than all non-targets. This is not the case in (a) where only the P300 amplitude for target column is higher. Target row and column are in bold.

PCA is used to reduce the dimension from 64 to 22 based on studies in [7]. Next, GA based ICA method gives 22 independent signals. These are averaged and P300 peak is located in the window 275–370 ms. Since the independent signals could be inverted during ICA, the independent signal is inverted before averaging if the P300 peak is a downward curve for any of the independent signal. As a comparison, averaged data from 64 channels from 15 trials are shown in Figure 4 (a). As could be seen from Figures 4 (a) and (b), the P300 component is higher for the target row and column using GA based ICA method but this is not the case using averaging method.

4 Conclusion

In the simulation study, the proposed method successfully separated a mixed signal consisting of artificially generated VEP and EEG. To validate the method further, an experiment was conducted with EEG signals from competition 2003 – data set IIB, which also gave good results in extracting P300 component. Since the method does not assume any property (i.e. completely blind), it could be applied to extract/separate any type of linearly mixed signal. For future work, chromosomes with continuous values will be explored instead of binary chromosome converted to real-values as done here.

5 Acknowledgement

The author is thankful to researchers at Wadsworth Center and the organizers of BCI competition 2003 for providing the data on the internet.

References:

- [1] R.Palaniappan, R.Paramesran, S.Nishida, and N.Saiwaki, A New Brain-Computer Interface Design Using Fuzzy ARTMAP, *IEEE Transactions on Rehabilitation Engineering*, vol. 10, no. 3, 2002, pp.140-148.
- [2] G.Pfurtscheller, C.Neuper, C.Guger, W.Harkam, H. Ramoser, A.Schlogl, B.Obermaier, and M.Pregenzer, Current trends in Graz brain-computer interface (BCI) research, *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, 2000, pp. 216-219.
- [3] E.Donchin, K.M.Spencer and R.Wijesinghe, The Mental Prosthesis: Assessing the Speed of a P300-Based Brain-Computer Interface, *IEEE Transactions on Rehabilitation Engineering*, vol. 8, no. 2, 2000, pp.174-179.
- [4] J.Polich, P300 in Clinical Applications: Meaning, Method and Measurement, *American Journal of EEG Technology*, vol. 31, 1991, pp. 201-231.
- [5] D.H.Lange and G.F.Inbar, Variable single-trial evoked potential estimation via principal component identification, *Proceedings of 18th Annual IEEE EMBS International Conference*, 1996, pp.954-955.
- [6] T-P. Jung, S.Makeig, M. Westerfield, J. Townsend, E. Courchesne and T.J. Sejnowski, Analysis and visualization of single-trial event-related potentials, *Human Brain Mapping*, vol. 14, 2001, pp.166-185.
- [7] N.Xu, X.Gao, B.Hong, X.Miao, S.Gao and F.Yang, BCI competition 2003 – data set IIB: enhancing P300 wave detection using ICA-based subspace projections for BCI applications, *IEEE Transactions on Biomedical Engineering*, vol. 51, no.6, 2004, pp.1067-1072.
- [8] X-Y. Zeng, Y-W. Chen, Z. Nakao, and K. Yamashita, Signal separation by independent component analysis based on a genetic algorithm, *Proceedings of 5th International Conference on Signal Processing*, vol. 3, 2000, pp.1688-1694.
- [9] F.Rojas, C.G.Puntonet, M.Rodriguez-Alvarez, I.Rojas, and R. Martin-Clemente, Blind source separation in post-nonlinear mixtures using competitive learning, simulated annealing and a genetic algorithm, *IEEE Transactions on SMC –Part C: Application and Review*, vol. 34, no. 4, 2004, pp.407-416.
- [10] A.Hyvarinen, J.Karhunen, and E.Oja, *Independent Component Analysis*, John Wiley & Sons, 2001.
- [11] A.Cichocki and S.Amari, *Adaptive Blind Signal and Image Processing*, John Wiley & Sons, 2001.
- [12] R.L.Haught and S.E.Haupt, *Practical Genetic Algorithm*, John Wiley & Sons, 1998.
- [13] D.E.Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.
- [14] P.A.Karjalainen, J.P.Kaipio, A.S.Koistinen, and M.Vauhkonen, Subspace regularization method for the single-trial estimation of evoked potentials, *IEEE Transactions on Biomedical Engineering*, vol. 46, no.7, 1999, pp.849-860.