

A differential evolution based energy trajectory planner for artificial limb control using motor imagery EEG signal

Bhattacharyya, S., Konar, A., & Tibarewala, D. N. (2014). A differential evolution based energy trajectory planner for artificial limb control using motor imagery EEG signal. *Biomedical Signal Processing and Control*, *11*, 107-113. https://doi.org/10.1016/j.bspc.2014.03.001

Link to publication record in Ulster University Research Portal

Published in:
Biomedical Signal Processing and Control

Publication Status:
Published (in print/issue): 01/05/2014

DOI:
10.1016/j.bspc.2014.03.001

Document Version
Author Accepted version

General rights
Copyright for the publications made accessible via Ulster University's Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Ulster University's institutional repository that provides access to Ulster's research outputs. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact pure-support@ulster.ac.uk.

Download date: 29/03/2024

A Differential Evolution Based Energy Trajectory Planner for Artificial Limb Control Using Motor Imagery EEG Signal

Saugat Bhattacharyya[1, 2], Amit Konar[1], D.N. Tibarewala[2]

[1]Dept. of Electronics &Telecommunication Engineering, [2]School of Bioscience & Engineering, Jadavpur University, Kolkata, India-700032

E-mail- saugatbhattacharyya@live.com, konaramit@yahoo.co.in, biomed.ju@gmail.com

Corresponding author: Saugat Bhattacharyya

Postal Address: School of Bioscience and Engineering,
Jadavpur University,
Raja S.C. Mullick Road,
Jadavpur, Kolkata
India-700032

Contact No.: +91-9932866197

E-mail: saugatbhattacharyya@live.com

Abstract

Current research on neuro-prosthetics is aimed at designing several computational models and techniques to trigger the neuro-motor rehabilitative aids. Researchers are taking keen interest to accurately classify the stimulated electroencephalography (EEG) signals to interpret motor imagery tasks. In this paper we aim to classify the finger-, elbow- and shoulder-classification along with left- and right-hand classification to move a simulated robot arm in 3D space towards a target of known location. The contribution of the paper lies in the design of an energy optimal trajectory planner, based on differential evolution, which would decide the optimal path for the robot arm to move towards the target based on the classifier output. Each different set of movements consists of a trajectory planner which is activated by the classifier output. The energy distribution of wavelet coefficients of the incoming EEG signals are used as features to be used as inputs in a naïve Bayesian classifier to discriminate among the different mental tasks. The average training classification accuracy obtained is 76.88% and the success rate of the simulated robot arm reaching the target is 85%.

Keywords – Brain-Computer Interfacing, Electroencephalography, Energy Efficient Trajectory Planning, Differential Evolution, Wavelet Transforms, Naïve Bayesian Classifier.

1. INTRODUCTION

Rehabilitative Engineering has travelled a great distance from mechanical prosthetics to neural prosthetics in the last two decades. Such neuroprosthetic aids aim at improving the living standard of people suffering from motor, sensory and cognitive failures by providing rehabilitative solutions for the same [1]-[3]. In this regard, Brain-Computer Interfacing (BCI) [3]-[5] provides the technology to translate the brain signals acquired from the brain to drive various assistive and rehabilitative devices. Among various available modalities to measure brain signals, Electroencephalography (EEG) is widely used among researchers because it is non-invasive, cost effective, portable, and has a good temporal resolution which can be easily used to generate control signals for real world application [5], [6]. Various researchers have employed the EEG based BCI technology to drive a word processor [7]-[9], speech synthesizer [10], [11], keyboard [12] and cursor simulators [13], [14], virtual reality [15], [16] and various robotic devices [17]-[19].

Movement related BCIs have been extensively researched in the past few decades with the aim to provide assistive and rehabilitative solutions to the clinical population suffering from neuro-muscular disorders like Amyotropic Lateral Sclerosis (ALS), Spinal Injury, Paralysis and amputation [20]-[22]. Movement related signals are characterized by specific waveforms at the mu- (8-12 Hz) and central beta- (16-24 Hz) bands, called Event-related Desynchronization and Synchronization (ERD/S), respectively [23], [24]. The recognition of these waveforms leads to the successful classification of various motor imagery signals.

In this study, we have used left-right and finger-elbow-shoulder movement EEG signals to generate the control signals to drive a simulation of an external (prosthetic) device. For this purpose, wavelet transforms [25], [26] have been used to extract the relevant information from the EEG signals and the feature vectors comprises of their energy distribution. The features are used as inputs to the Naïve Bayesian classifier [27], [28] whose outputs are used to move the simulated robot arm. In this study, we have proposed the use of an Energy Efficient Technique [29] based on Differential Evolution [30], [31] to plan the optimal trajectory path of a simulated arm in 3D space. The subject would send control signals from the given list of movements (left finger, left elbow, left shoulder, right finger, right elbow and right shoulder) and the energy efficient technique would plan the optimal trajectory path for

the simulated arm towards the target. While moving towards an object in space, a number of different paths are available to us to reach that object. We have devised a novel approach using both the quantized classifier output and the energy efficient technique to move the robot arm towards the target using an optimum path (the path which requires the minimum energy). This approach finds preference in the sense that in real life implementations an energy optimal path is more relevant to control an artificial limb towards targets present at fixed positions.

The rest of the papers are organized as follows: In Section 2 we have our proposed algorithm developed in this study to control an artificial arm. Section 3 describes the experimental implementation of the developed algorithm followed by a discussion and comparison of the results obtained in Section 4. The concluding remarks and future directions are discussed in Section 5.

2. THE OPTIMAL TRAJECTORY PLANNER

For our study, we have designed a robot arm with two links and two movable sections each having one degree of freedom to mimic the movement of a 2DOF robot arm in real world. Their movements are described by angles θ and ϕ which are defined with respect to the co-ordinate system as shown in Figure 1.The first section of length *L1* moves in the vertical plane as described by θ measured from the +ve z-axis. The vertical plane in which it moves is displaced from the x-axis by an angle α. The second section of length *L2* moves in the horizontal plane as described by ϕ measured from the +ve x-axis. The arm ends in a gripper whose co-ordinates are given by

$$\begin{aligned} x &= L1\times\sin(\theta)\times\cos(\alpha)+L2\times\cos(\phi) \\ y &= L1\times\sin(\theta)\times\sin(\alpha)+L2\times\cos(\phi) \\ z &= L1\times\cos(\theta) \end{aligned} \tag{1}$$

Here for our analysis we take, without loss of generality, $\alpha = 90°$, so that the upper arm is entirely confined in the Y-Z plane.

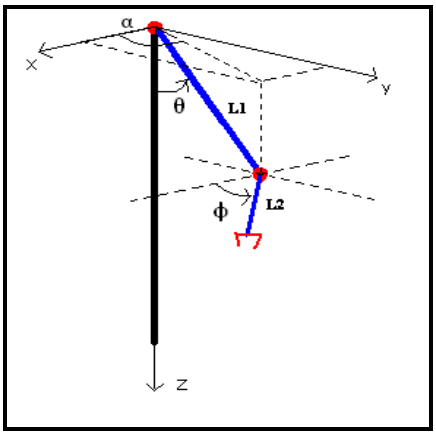


Figure 1. Variation of phi (ϕ) and theta (θ)

2.1. Energy Efficient Trajectory Generation

Here we have developed an energy efficient method of fitting a smooth trajectory to the set of via-points (points between two given location) to move the robot arm towards a target from the initial position. Let us consider that we have n+1via-points (including initial and final points) that are obtained from EEG control. We fit smooth cubic polynomials for θ and φ as functions of time in between each of these points as shown below:

$$\theta = a_0 + a_1 t + a_2 t^2 + a_3 t^3$$
$$\phi = b_0 + b_1 t + b_2 t^2 + b_3 t^3 \quad (2)$$

where the coefficients are determined partly by a set of boundary conditions and partly by energy minimization criterion.

The boundary conditions for the polynomial between the $(i\text{-}1)^{th}$ and i^{th} via-points are as follows:

$$For\ i = 1,2,...,n; \theta = \theta_i, \phi = \phi_i, at\ t = 0$$
$$\overline{\theta_i} = \theta_i - \theta_{i-1}, \overline{\phi_i} = \phi_i - \phi_{i-1},\ \textbf{at}\ t = T \quad (3)$$

where T is the time to move from the $(i\text{-}1)^{th}$ to the i^{th} via-point, and $\overline{\theta}, \overline{\phi_i}$ are the first time derivatives of θ and ϕ, respectively. Condition of continuity of angular displacements is of course necessary, whereas condition of continuity of angular velocity at every via-point avoids jerky movement of the arm. At the initial point angular velocities are zero. Using the boundary conditions, we get the coefficients as:

$$
\begin{aligned}
&For\ i = 1,2,...,n;\ a_0 = \theta_i, b_0 = \phi_i \\
&\qquad a_1 = \overline{\theta_i}, b_1 = \overline{\phi_i} \\
&\qquad a_2 = 1/T^2(\theta_{i+1} - \theta_i - \overline{\theta_i}.T - a_3.T^3) \\
&\qquad b_2 = 1/T^2(\phi_{i+1} - \phi_i - \overline{\phi_i}.T - b_3.T^3)
\end{aligned} \tag{4}
$$

Evidently the above conditions give three equations for four unknown coefficients, and the final equation is provided by the energy term. At any point the mechanical energy of the arm will be given by the summation of kinetic and potential energies of each of the arm sections as shown below:

$$E = \sum_i (K.E._i + P.E._i) \tag{5}$$

where $K.E._i = \frac{1}{2}.m_i.v_i^2 + \frac{1}{2}.I_{CM_i}$

and $P.E._i = m_i.g.h_i$

Here ω_i = angular velocity of i^{th} link, i.e. $\omega_1 = \overline{\theta}$ and $\omega_2 = \overline{\phi}$.

v_i = translational velocity = $\sqrt{\overline{x_i}^2 + \overline{y_i}^2 + \overline{z_i}^2}$, where $\overline{x_i}, \overline{y_i}, \overline{z_i}$ are the first time derivatives of the co-ordinate of the centre of mass (CM) of the i^{th} link.

I_{CM_i} = moment of inertia of the i^{th} link about its CM.

m_i = mass of the i^{th} link.

h_i = height of the i^{th} link from the ground, which is taken as the reference for zero potential energy.

Upon calculations, the total mechanical energy of the system is found to be equal to

$$
\begin{aligned}
E = {} & \frac{1}{2} m_2 L_1^2.\overline{\theta}^2 + \frac{1}{6}\left(m_1 L_1^2.\overline{\theta}^2 + m_2 L_2^2.\overline{\theta}^2\right) \\
& + \frac{1}{2} m_2 L_1 L_2 \cos(\theta)\cos(\phi).\overline{\theta}.\overline{\phi} + gH\left(m_1 + m_2\right) \\
& -gL_1\left(\frac{1}{2} m_1 + m_2\right)\cos(\theta)
\end{aligned} \tag{6}
$$

In this equation we put the expressions for θ, ϕ and $\overline{\theta}, \overline{\phi}$; and replace a_2, b_2 in terms of a_3, b_3 so that E is now a function of time, and a_3, b_3. The energy integrated over a single time interval will give a measure of the total energy consumed and this is a function of a_3 and b_3, using recursive adaptive Simpson quadrature technique. Thus we can find an optimum value of the remaining coefficient by minimizing F_K, where

$$F_K\left(a_3,b_3\right)=\int_0^T E.dt \tag{7}$$

This completes description of the entire trajectory subject to smooth motion and minimization of energy consumed [32], [33].

2.2. The Optimization Technique-Differential Evolution

Like any other evolutionary algorithm, DE starts with a population of *NP D*-dimensional parameter vectors representing the candidate solutions. We shall denote subsequent generations in DE by $G=0,1,...,G_{\max}$ .We represent the i^{th} vector of the population at the current generation as $\overrightarrow{X}_{i,G}=\left[x_{1,i,\mathrm{G}},x_{2,i,\mathrm{G}},...,x_{\mathrm{D},i,\mathrm{G}}\right]$.

The initial population (at *G=0*)should cover the entire search space as much as possible by uniformly randomizing individuals within the search space constrained by the prescribed minimum and maximum bounds:

$$\begin{aligned}&\overrightarrow{X}_{\min}=\left[x_{1,\min},x_{2,\min},...,x_{D,\min}\right]\text{ , and}\\&\overrightarrow{X}_{\max}=\left[x_{1,\max},x_{2,\max},...,x_{D,\max}\right]\end{aligned} \tag{8}$$

Hence we may initialize the j^{th} component of the i^{th} vector as

$$x_{j,i,0}=x_{j,\min}+rand_{i,j}\left(0,1\right).\left(x_{j,\max}-x_{j,\min}\right) \tag{9}$$

where $rand_{i,j}\left(0,1\right)$ is a uniformly distributed random number lying between 0 and 1 and is instantiated independently for each component of the i^{th} vector. The following steps are taken next: mutation, crossover, and selection (in that order), which are explained in the following subsections.

2.2.1. Mutation

After initialization, DE creates a donor vector $\vec{V}_{i,G}$ corresponding to each population member $\overrightarrow{X}_{i,G}$ in the current generation through mutation using arithmetic recombination. It is the method of creation of donor vector that differentiates one DE scheme from another. A popular mutation strategy “DE/rand/1” [31] has been adopted in our paper, which is defined as

$$\vec{V}_{i,G}=\overrightarrow{X}_{r_1^i,G}+F.\left(\overrightarrow{X}_{r_2^i,G}-\overrightarrow{X}_{r_3^i,G}\right) \tag{10}$$

The indices r_1^i, r_2^i and r_3^i are mutually exclusive integers randomly chosen from the range [*1, NP*], and all are different from the base index *i*. The scaling factor *F* is a positive control parameter for scaling the difference vectors. $\vec{X}_{best,G}$ is the vector with the best fitness in the population at generation *G* [30],[31].

2.2.2. Crossover

To increase the potential diversity of the population, a crossover operation comes into play after generating the donor vector through mutation. The donor vector exchanges its components with the target vector $\vec{X}_{i,G}$ under this operation to form the trial vector $\vec{U}_{i,G} = \left[u_{1,i,G}, u_{2,i,G}, ..., u_{D,i,G}\right]$. In our paper we consider binomial crossover where the donor vector exchanges its components with the target vector $\vec{X}_{i,G}$ for each of the *D* variables whenever a randomly picked number between 0 and 1 is less than or equal to the *Cr* value. In this case, the number of parameters inherited from the donor has a (nearly) binomial distribution. The scheme may be outlined as

$$u_{j,i,G} = \begin{cases} v_{j,i,G}, if\ \ rand_{i,j}(0,1) \le Cr \text{ or } j = j_{rand} \\ x_{j,i,G}, \text{ otherwise} \end{cases} \tag{11}$$

where $rand_{i,j}(0,1) \in [0,1]$ is a uniformly distributed random number lying between 0 and 1and is instantiated independently for each j^{th} component of the i^{th} vector. $j_{rand} \in [1, 2, ..., D]$ is a randomly chosen index, which ensures that $\vec{U}_{i,G}$ gets at least one component from $\vec{V}_{i,G}$.

2.2.3. Selection

To keep the population size constant over subsequent generations, the next step of the algorithm calls for selection to determine whether the target or the trial vector survives to the next generation i.e., at *G = G +1*. The selection operation is described as follows:

$$u_{j,i,G} = \begin{cases} \vec{U}_{i,G}, if\ f\left(\vec{U}_{i,G}\right) \le f\left(\vec{X}_{i,G}\right) \\ \vec{X}_{i,G}, if\ f\left(\vec{U}_{i,G}\right) > f\left(\vec{X}_{i,G}\right) \end{cases} \tag{12}$$

where $f\left(\vec{X}\right)$ is the function to be minimized. So if the new trial vector yields an equal or lower value of the objective function, it replaces the corresponding target vector in the next generation; otherwise the target is retained in the population. Hence the population either gets

better (with respect to the minimization of the objective function) or remains the same in fitness status, but never deteriorates.

The above three processes are repeated till maximum number of generations is reached. The parameters selection in the DE algorithm for this study are as follows: scaling factor = 10, population size = 50, crossover ratio = 0.8, maximum number of generations = 10000, stopping criteria = 20 number of same fitness.

3. CONTROL SCHEME AND DATA PROCESSING

3.1. Control Scheme

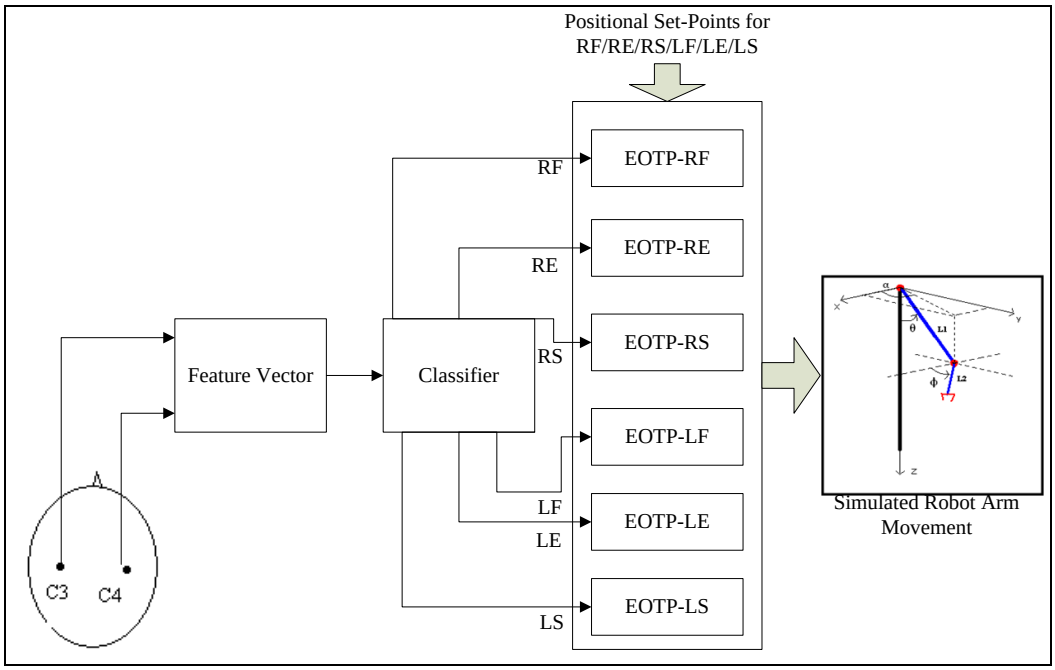


Figure 2. The Proposed Approach to control a robot arm using the Energy Optimized Tranjectory Planner (EOTP).

The complete scheme (Figure 2) is summarized as follows: First, we construct the feature vector from the acquired EEG signals from C3 and C4 electrode location using the energy coefficient of the level 3 and level 4 wavelet detail coefficient. The features are then fed to the classifiers to yield the respective output: Right Finger (RF), Right Elbow (RE), Right Shoulder (RS), Left Finger (LF), Left Elbow (LE), and Left Shoulder (RS). The classifier outputs and their corresponding commands to the simulated robot arm are given in Table 1. The classifiers are trained using the previous EEG signals of the given subject. The classifier output is fed as control signals to the local trajectory planner (i.e., trajectory planners for each

control signal) which simulates the movement of the robot arm based on the proposed energy efficient algorithm discussed in the previous section. The trajectory planner plans the most energy efficient path based on the output of the classifiers and provides a feedback to the subject about the same. The subject would decide the next movement based on the feedback and the process would continue until the goal is reached. Also, the local trajectory planner for each state would interpolate the position of each link of the robot arm (w.r.t. target position) in such a manner so as to minimize the energy consumption.

Table 1. Motor Imagery Outputs and their corresponding commands

Motor Imagery	Encoding	Motion
Right Shoulder	1-100	$\alpha_{sp} = \alpha_{sp} + 10deg$
Right Elbow	1-200	$\theta_{sp} = \theta_{sp} + 10deg$
Right Finger	1-300	Release object
Left Shoulder	2-100	$\alpha_{sp} = \alpha_{sp} - 10deg$
Left Elbow	2-200	$\theta_{sp} = \theta_{sp} - 10deg$
Left Finger	3-300	Grasp object
No motion	1-000/2-000	no motion

3.2. Experimental Procedure

The EEG signal has been recorded using a NeuroWin amplifier comprising of 19 channel (Fp1, Fp2, F8, F4, Fz, F3, F7, T4, C4, Cz, C3, T5, T6, P4, Pz, P3, T7, O2, O1) Ag/AgCl electrodes at a sampling frequency of 250 Hz. The left ear and FPz location is selected as the point for reference and ground electrode. The sensitivity of the amplifier is set to 100 μV and an additional 50 Hz notch filter had been utilized to suppress the line noise.

The experiment is so designed to make the subject move the instructed joint (finger/elbow/shoulder) of any one of the hand (left/right) in a self-paced manner. Six females and seven males (right handed) in the age group of 25±5 years were employed as subjects in our experiment. The subjects were made to fill up a consent form and a simple introduction about the research work and stages of the experiment were given.

The training experiment consists of 3 sessions with 60 trials each conducted on the same day with fifteen minutes break in between. The subjects imagine moving a specific joint

(either Elbow, Finger, Shoulder movement) of either right or left hand based on the instructions given on screen. In each trial, a blank screen was displayed in the first 2 seconds. In the 2nd second a fixation cross '+' was displayed on the screen which indicates the beginning of the trial. From the 3rd second onwards, the instructions are displayed on screen. At that instance, the subject moves their respective joint-limb until the display is blank again. The timing scheme of the visual cue is given below in Figure 3.

During the online experimentation, a target appears randomly on screen and based on its current location, the subject moves the robot arm based on the motor imagery control commands shown in Table 1. Here, on each 500 milliseconds of incoming signal, a feature vector is constructed which is fed to the trained classifier to yield the control commands based on Table 1. Thus, 125 sample-points are accumulated at each update of half a second. These control commands activates the local trajectory planner to move the robot arm towards the target.

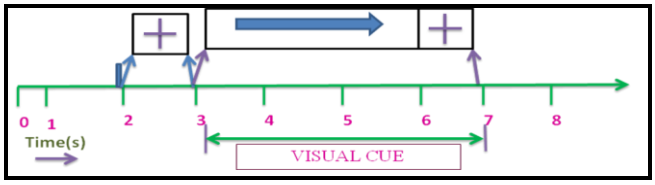


Figure 3. Timing scheme of the experiment

3.3. Data Processing

Researchers have verified that movement related signals (in terms of planning, action or imagination) originates from the sensorimotor area, pre-motor cortex and primary motor cortex and EEG captures the movement related signals from their adjoining locations in the scalp, which are centrally located in the scalp[34]. Signals from electrodes C3 and C4 contains the most information related to left and right hand movement [5] and thus are selected in this study to identify the different movements. Another reason for selecting two electrodes instead of all nineteen of them is to reduce the computational time required by the processor to detect the right output. The signals from the selected electrode are filtered using Common Average Reference [5], followed by band-pass filtering using an elliptical filter (order 14) between 8 and 25 Hz, to reduce the effect of neighboring electrodes and environmental noises, and movement related information are mostly obtained in this bandwidth.

In this study, we have used wavelet transforms for feature extraction the given EEG data. Wavelet Transform (WT) has its obvious advantages over techniques based on time-domain

or frequency-domain as it provides localized frequency related information at a given time. Further details on wavelet transforms are found in [25], [26]. In our study, we have selected the Daubechies mother wavelet of order 4 (db4) to decompose the EEG data. The percentage energy distribution [35] at third (D3) and fourth (D4) level of decomposition are selected to prepare the feature vector because the motor imagery signals are predominant at 8 -12 Hz and 16-24 Hz. The final size of the feature vector during both offline training and online testing is 2 (electrodes) × 2 (energy components from D3 and D4 components).

The feature vectors thus constructed are fed to the classifier to classify among the various movements. Two levels of classifiers, as shown in Figure 4 are used in this study based on a tree formation: first, the left, right hand movements or no movements (Classifier 1) are classified followed by the finger, elbow and shoulder classification (Classifier 2 and 3) for each limb. The classification in both level has been done using the one-versus-rest (OVR) scheme [36], where one movement is classified against rest of the movement. In the offline training phase, Naïve Bayesian (NB), Support Vector Machine (SVM), Radial Basis Function Support Vector Machine (RSVM), Multilayer Perceptron Support Vector Machine (MSVM), k-Nearest Neighbor (kNN), and Linear Discriminant Analysis (LDA) [31] are trained for motor imagery state detection and the best performing classifier among these are used in the online experimentation of moving the robot arm.

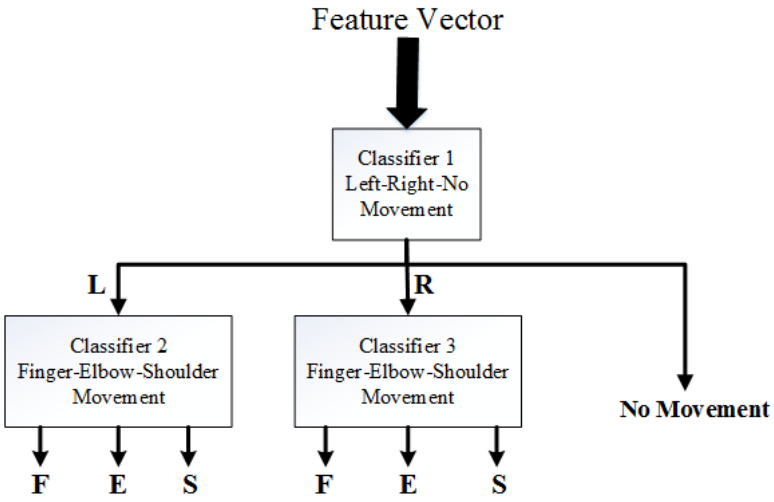


Figure 4. The Classifier scheme (Abbreviations: L- Left movement, R- Right movement, F- Finger movement, E- Elbow movement, S- Shoulder movement)

The performance of the classifier is compared with various other state-of-the-art classifiers in Section 4 and its performance is found to be competitive.

4. RESULTS AND DISCUSSION

The whole experiment is conducted in MATLAB version 7.9 environment. The specification of the system in which the experiment was conducted is as follows: Processor-Intel Core2Duo, 1.19 GHz, 3.2 GB RAM.

4.1. Offline Training

The performance of the trained classifier is measured by the *k*-fold cross-validation technique (Here, *k*=10) [37], using the training data of 3 sessions. The final predicted output is determined by the combination of the outputs from both level of the classification tree. For example, if Classifier 1 produces 'Left' as an output and Classifier 2 produces 'Elbow' as an output for a given instance, then the final output will be 'Left Elbow'. The classification accuracy of the final output is the performance metric used in this study. As observed from Table 2, the classification accuracy of NB as compared with other standard classifiers (as mentioned in Section 3.3) is better. The superiority of NB classifier is further validated statistically by applying Friedman's test [38] on the classifiers with the classification accuracy as the basis of ranking.

The null hypothesis here, states that all the features are equivalent. The Friedman statistic distributed according χ_F^2 to with k-1 degrees of freedom, is given by

$$\chi_F^2 = \frac{12N}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right] \tag{14}$$

where, *k* is the number of classifiers and *n* is the number of datasets. From Table 2, it is noted that k=6 and N=13 and based on (14), χ_F^2 =46.598, which is > $\chi_{5,0.05}^2$ = 11.07. So, the null hypothesis claimed is rejected to a level of 5% confidence level and the classifiers are not equivalent. It is evident from Table 2 that NB is ranked one and outperforms the other classifiers. Thus, NB classifier is selected for online experimentation of this study.

4.2. Trajectory Planning

The subject sees the target on the screen and imagines one step of movement towards the goal based on the current position of the arm. The classification continues to take place until the arm reaches its target or the algorithm reaches its stopping criteria, thus, failing to reach the target. The subjects perform the online experiment in 20 different runs and the target positions are selected randomly from the following (θ, ϕ) values: (30, 30), (30, 60), (60, 60), (60, 90) and (90, 90). The online experimentation is analyzed using the average of the following metrics: % of successful hits, % deviation from goal on unsuccessful hit and

computational time required to reach the five different target. Table 3 gives the average of metrics for 13 subjects. An example of the robotic arm moving towards the target is demonstrated in Figure 5. The green dot in the illustrations is the target to be reached.

Table 2. Classification Accuracies of the classifiers along with their ranking (given in brackets)

Subject ID	**NB**	**SVM**	**RSVM**	**MSVM**	**kNN**	**LDA**
1	80.00 (1)	72.50 (2)	70.00 (3)	65.71 (6)	66.67 (5)	67.50 (4)
2	75.00 (1)	70.00 (3)	65.00 (4.5)	55.00 (6)	73.33 (2)	65.00 (4.5)
3	67.50 (1)	57.50 (5)	62.50 (3)	65.00 (2)	50.00 (6)	60.00 (4)
4	76.47 (3)	78.38 (2)	80.00 (1)	60.00 (5)	56.67 (6)	69.70 (4)
5	75.00 (2)	77.50 (1)	70.00 (3)	57.50 (6)	60.00 (5)	62.50 (4)
6	80.00 (1)	75.00 (3)	77.50 (2)	67.50 (4)	66.67 (5)	60.00 (6)
7	83.75 (1)	77.00 (2)	75.00 (3)	62.50 (5)	60.00 (6)	71.25 (4)
8	72.25 (1)	69.00 (3)	70.00 (2)	64.75 (5)	63.33 (6)	66.75 (4)
9	69.00 (2)	73.50 (1)	67.50 (3)	57.00 (6)	60.50 (5)	63.50 (4)
10	76.75 (1)	74.50 (3)	75.00 (2)	59.25 (5)	57.75 (6)	64.00 (4)
11	81.25 (1)	79.25 (2)	77.50 (3)	62.50 (4)	58.00 (6)	60.00 (5)
12	82.00 (1)	77.50 (2.5)	77.50 (2.5)	66.67 (5)	62.00 (6)	66.75 (4)
13	80.50 (2)	81.50 (1)	74.75 (3)	72.00 (5)	70.00 (6)	73.50 (4)
Mean	**76.88**	74.08	72.48	62.72	61.92	65.41
Average Rank (R_j)	1.38	2.35	2.69	4.92	5.38	4.27

Table 3. Average of performance metrics for online trajectory planning for 13 subjects

% of successful hits	% deviation from goal for unsuccessful hits	Computational time (in seconds) to reach the goal when target is at (θ, φ) position				
		(30, 30)	(30, 60)	(60, 60)	(60, 90)	(90, 90)
85	4.75	73	81	87	89	96

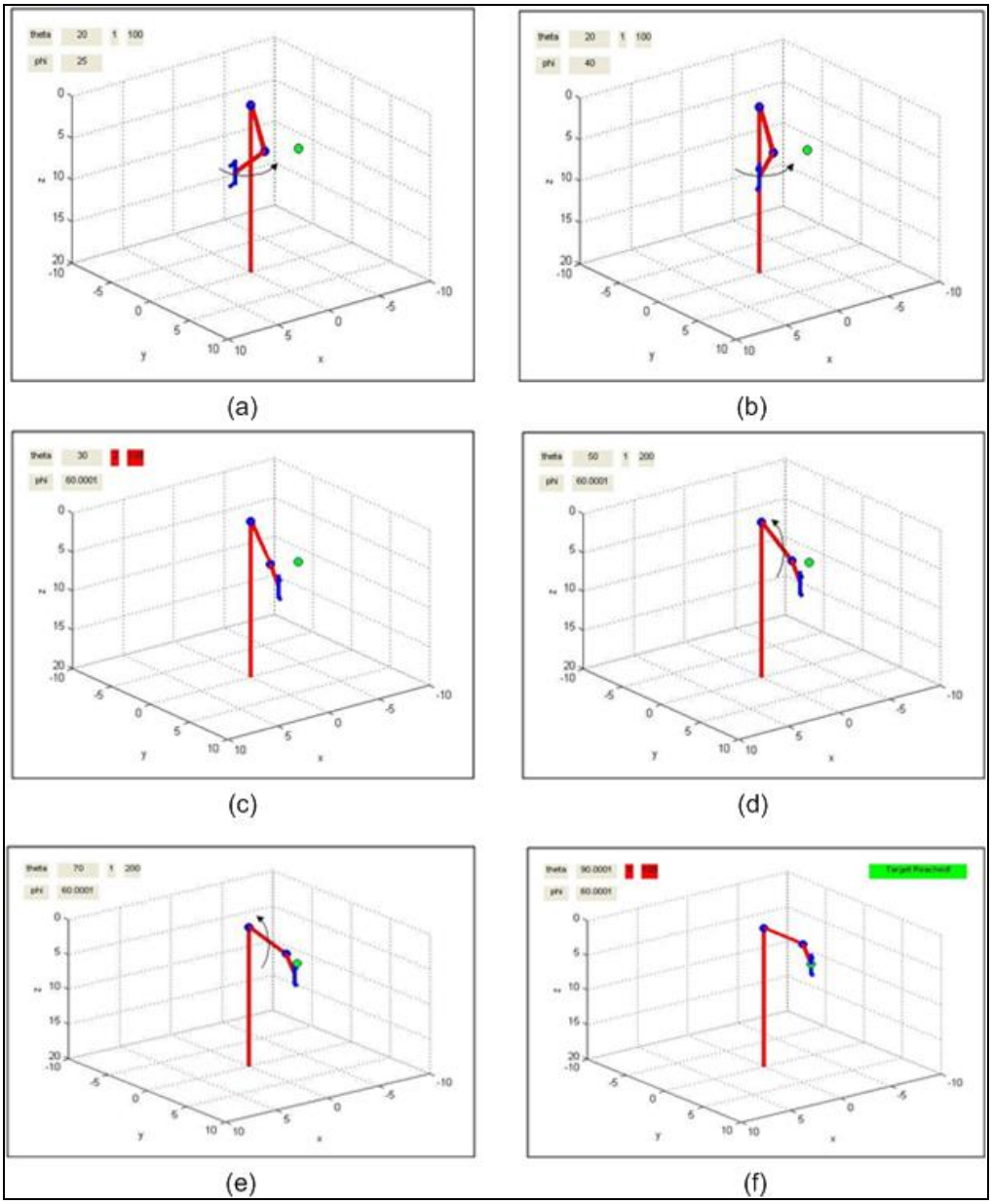


Figure 5.Illustrations of the trajectory simulation of the robot arm showing six instances during execution of movement towards the target (green dot). Initial and final positions are θ, ϕ =20°,25° and θ, ϕ = 90°, 60° (a) Initial Position (θ=20°, ϕ =25°), (b) Rotation in ϕ (θ =20°; ϕ = 40°), (c)Final ϕ attained, further motion in ϕ restricted (θ =30°, ϕ = 60°), (d) Rotation in θ (θ =60°, ϕ =60°), (e) Rotation in θ (θ =70°, Phi=60°) (f) Goal reached (green dot), further rotation stopped (θ =90°, ϕ =60°)

Table 4 shows the performance of the optimizer relative to two other popular algorithms Particle Swarm Optimization (PSO) and Invasive Weed Optimization (IWO) [30] in terms of the total number of energy consumed in the trajectory as determined by the algorithms and the average number of function evaluations required in doing so. It is seen that DE gives superior solution and hence our choice is justified.

Table 4. Efficiency of Optimizer

	DE	PSO	IWO
Units of energy consumed	1.039e+04	1.353e+04	1.228e+04
Average Number of Function Evaluations	260	390	310

5. CONCLUSION

In this paper, we have successfully classified three different joint movements (viz. finger, elbow and shoulder) along with the left/right upper limbs with an average accuracy of more than 70% and simulated the "thought-control" of a robot arm towards a known target subject to conditions of minimum cost in energy with a success hit rate of 85%. In our future work, we will be concentrating on targets whose locations are not known or when the target position is dynamic. Further the energy optimal path planning as devised by our team finds particular relevance in practical implementations and armed with the results and performance analysis of classifiers and trajectory planners obtained here we aim to ultimately design and develop a complete real time BCI system.

Acknowledgements

This study has been supported by University Grants Commission, India, University of Potential Excellence Programme (Phase II) in Cognitive Science, Jadavpur University and Council of Scientific and Industrial Research, India.

References

1. M. Tavella, R. Leeb, R. Rupp, J.R. Millan, Towards natural non-invasive hand neuroprostheses for daily living, in: 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2010, pp. 126-129.
2. G. R. Muller-Putz, R. Scherer, G. Pfurtscheller, C. Neuper, Temporal coding of brain patterns for direct limb control in humans, F. Neurosci. 4 (2010) 1-11.
3. A. Nijholt, D. Tan, Brain-computer interfacing for intelligent systems, IEEE Int. Sys. 23(3) (2008) 72-79.
4. G. Pfurtscheller , C. Neuper, N. Birbaumer, Human brain-computer interface, in: A. Riehle, E. Vaadia (Eds.), Motor Cortex in Voluntary Movements: A Distributed System for Distributed Functions, CRC Press, Florida, 2005, pp. 367-401.
5. G. Dornhege, J.R. Millan, T. Hinterberger, D.J. McFarland, K.R. Muller, Toward Brain-Computer Interfacing, first ed., MIT Press, Massachusetts, 2007.
6. S. Sanei, J.A. Chambers, EEG Signal Processing, first ed., John Wiley & Sons, West Sussex, 2007.

7. C.J. James, S. Wang, Performance analysis of a P300 BCI speller through single channel ICA, in: IET International Conference on Advances in Medical, Signal and Information Processing, 2008, 1-4.
8. C.C. Postelnicu, D. Talaba, P300 based brain-neuronal computer interaction for spelling applications, IEEE Trans. Biomed. Engg., 60(2) (2013) 534-543.
9. A. Lenhardt, M. Kaper, H.J. Ritter, An adaptive P300 based online brain-computer interface, IEEE Trans. Neural Systems and Rehab. Engg., 16(2) (2008) 121-130.
10. J. Brumberg, A. Nieto-Castanon, P.R. Kennedy, F.H. Guenther, Brain-computer interfaces for speech communication, J. Speech Comm. 52(4) (2010) 367-379.
12. A. Porbadnigk, M. Wester, J.P. Callies, T. Schultz, EEG-based speech recognition-impact of temporal effects, in: BIOSIGNALS, INSTICC press, 2009, pp. 376-381.
13. B. Chambayil, R. Singla, R. Jha, Virtual keyboard BCI using eye blinks in EEG, In: 2010 IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communication (2010) 466-470.
14. J. Long, Y.Li, T.Yu, Z.Gu, Target selection with hybrid feature for BCI-based 2-D cursor control, IEEE Trans. Biomed. Engg., 59(1) (2012) 132-140.
15. Y. Li, J. Long, T. Yu, Z. Yu, C. Wang, H. Zhang, C. Guan, An EEG-based BCI system for 2-D cursor control by combining mu/beta rhythm and P300 potential, IEEE Trans. Biomed Engg. 57(10) (2010) 2495-2505.
16. D. Friedman, R.Leeb, G. Pfurtscheller, M. Slater. Human-Computer Interface issues in controlling virtual reality with brain-computer interface, Human-Computer Interaction 25(1) (2010) 67-94.
17. D.P. Bos et al, Looking around with your brain in a virtual world, in: 2011 IEEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind & Brain, 2011, pp. 1-8.
18. F. Galan, M. Nuttin, E. Lew, P.W. Ferrez, G. Vanacker, J. Phillips, J del R. Millan, A brain-actuated wheelchair: Asynchronous and non-invasive brain-computer interfaces for continuous control of robots, J. Clin. NeuroPhy. 119 (2008) 2159-2169.
19. Y. Chae, J. Jeong, S. Jo, Toward brain-actuated humanoid robots: Asynchronous direct control using an EEG-Based BCI, IEEE Trans. Robotics 28(5) (2012) 1131-1144.
20. A.Finke, A.Knoblauch, H. Koesling, H. Ritter, A hybrid brain interface for a humanoid robot assistant, in: 2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2011, pp. 7421-7424.
21. J. Conradi, B. Blankertz, M. Tangermann, V. Kunzmann, G. Curio, Brain-computer interfacing in tetraplegic patients with high spinal cord injury, Int J Bioelectromagnetism 11(2) (2009) 65-68.
22. G. Prasad, P. Herman, D. Coyle, S. McDonough, and J. Crosbie, Applying a brain-computer interface to support motor imagery practice in people with stroke for upper limb recovery: a feasibility study, J. Neuroeng. And Rehab. 7 (1) (2010) 60-76.
23. N. Birbaumer, A.R. Murguialday, L. Cohen, Brain-computer interface in paralysis, J. Cur. Op. Neurology: 21 (2008) 634-638.
24. G. Pfurtscheller, F.H. Lopes da Silva, Event-related EEG/MEG synchronization and desynchronization: basic principles, J. Clin. Neurophysiology 110 (1999) 1842-1857.

25. C. Qiang, P.Hu, F. Huanqing, Experiment study of the relation between motion complexity and event related desynchronization/synchronization, in: First International Conference on Neural interfaces and Control Proceedings (2005).
26. S. Darvishi, A. Al-Ani, Brain-computer interface analysis using continuous wavelet transform and adaptive neuro-fuzzy classifier, in: 29th Int. Annual International Conference of the IEEE Engineering in Medicine and Biology Society (2007) 3220-3223.
27. W. Sweldons, The lifting scheme: a custom-design construction of biorthogonal wavelets, J. App. Comp. Harmonic Analysis 3(2) (1996) 186-200.
28. N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifier, J. Machine Learning 29 (1997) 131-161.
29. R.O. Duda, P.E. Hart, D.G. Stork, Pattern Recognition, second ed.,Wiley-Interscience, New York, 2000.
30. A. Sengupta, T. Chakraborti, A. Konar, A.K. Nagar, Energy efficient trajectory planning by a robot arm using invasive weed optimization technique, in: IEEE Third World Congress on Nature and Biologically Inspired Computing, 2011, pp. 311-316.
31. R. Storn, K.V. Price, Differential Evolution–a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optimization 11(4) (1997) 341–359.
32. R. Storn., K.V. Price, J. Lampinen, Differential evolution–A practical approach to global optimization, Springer-Verlag, 2005.
33. Y. Saab, M. VanPutte. Shortest path planning on topographical maps, IEEE Trans. Systems, Man & Cybernetics– A 29(1) (1999) 139-150.
34. Z. Mao, T.C. Hsia, Obstacle avoidance inverse kinematics solution of redundant robots by neural networks, J. Robotica, 15 (1997) 3-10.
35. J.C. Tamraz, Y.G. Comair, Atlas of regional anatomy of the brain using MRI with functional correlations, Springer, 2006.
36. C. Kocaman, M. Ozdemir, Comparison of statistical methods and wavelet energy coefficients for determining two common PQ disturbances: Sag and Swell, in: 2009 International Conference on Electrical & Electronics Engineering (2009) I-80-I-84.
37. R. Rifkin, A. Klautau, In defense of one-vs-all classification, J. Machine Learning Research 5 (2004) 101-141.
38. E. Alpaydin, Introduction to Machine Learning, second ed., MIT Press, Massachusetts, 2009.
39. M. Hollander, D.A. Wolfe, Nonparametric Statistical Methods, second ed., Wiley-Interscience, 1999.