Neurocomputing 72 (2009) 1121–1130

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Classification of mental task from EEG data using neural networks based on particle swarm optimization

Cheng-Jian Lin [a,b,*], Ming-Hua Hsieh [b]

[a] Department of Computer Science and Engineering, National Chin-Yi University of Technology, Taichung County, Taiwan 411, ROC
[b] Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung County, Taiwan 413, ROC

ARTICLE INFO

Article history:
Received 5 December 2006
Received in revised form
21 February 2008
Accepted 27 February 2008
Communicated by T. Heskes
Available online 26 March 2008

Keywords:
Brain–computer interface
Neural networks
Particle swarm optimization
Migration
Principle component analysis (PCA)

ABSTRACT

The brain–computer interface (BCI) is a system that transforms the brain activity of different mental tasks into a control signal. The system provides an augmentative communication method for patients with severe motor disabilities. In this paper, a neural classifier based on improved particle swarm optimization (IPSO) is proposed to classify an electroencephalogram (EEG) of mental tasks for left-hand movement imagination, right-hand movement imagination, and word generation. First, the EEG patterns utilize principle component analysis (PCA) in order to reduce the feature dimensions. Then a three-layer neural network trained using particle swarm optimization is used to realize a classifier. The proposed IPSO method consists of the modified evolutionary direction operator (MEDO) and the traditional particle swarm optimization algorithm (PSO). The proposed MEDO combines the evolutionary direction operator (EDO) and the migration. The MEDO can strengthen the searching global solution. The IPSO algorithm can prevent premature convergence and outperform the other existing methods. Experimental results have shown that our method performs well for the classification of mental tasks from EEG data.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

Many people suffer from amyotrophic lateral sclerosis, brainstem stroke, brain or spinal cord injury, cerebral palsy, muscular dystrophies, multiple sclerosis, and numerous other diseases. These diseases impair a person's neural pathways that control muscles or impair the muscles themselves. Moreover, a person suffering from one of these diseases may lose all voluntary muscle control, including eye movements and respiration, and may be completely imprisoned in their bodies, unable to communicate in any way.

Because of new understanding into how the brain functions, low-cost computer equipment, advances in signal process technology, research in brain–computer interface (BCI) has received much interest in the past decade. A BCI is a communication system that does not depend on the brain's normal output pathways of peripheral nerves and muscles [27]. Therefore, BCI system [13] can provide an augmentative communication method for patients with severe motor disabilities. Paralyzed patients can ask for basic necessities like water and food or use a computer by moving the cursor on a monitor screen using a BCI system without any voluntary muscle control.

Since a BCI system is controlled directly by the brain, it needs a method to detect brain activity. The method can be magnetic resonance imaging (MRI), functional magnetic resonance imaging (FMRI), positron emission topography (PET), near-infra-red spectroscopy (NIRS), magnetoencephalography (MEG), or electroencephalogram (EEG). Among these methods, the EEG is relatively inexpensive, has rapid response time, and can function in most environments. At present, the EEG is widely used to monitor brain activity in BCI research [18,1,11]. MRI belongs to the structural imaging. Structural imaging represents a range of measurement techniques, which can display anatomical information. Optical imaging offers some disadvantages over both MEG and FMRI, both of which involve the use of large, heavy, expensive instruments, and a dedicated building to eliminate effects of external magnetic fields. PET is a nuclear medicine imaging modality that provides information about the regional cerebral blood flow and tissue metabolism. NIRS is a non-invasive technique used for monitoring blood oxygenation. PET and NIRS are both unsuited for monitor brain activity.

An electroencephalogram (EEG) is a recording of the very weak (on the order of 5–100 μV) electrical potentials generated by the brain on the scalp. First, it was introduced by Hans Berger in 1929. An EEG is recorded as a potential difference between a signal electrode placed on the scalp and a reference electrode (generally

* Corresponding author at: Department of Computer Science and Information Engineering, Chaoyang University of Technology, Taichung County, Taiwan 413, ROC.
E-mail address: cjlin@nuk.edu.tw (C.-J. Lin).

0925-2312/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2008.02.017

connected to one ear or both ears). The amplitude and frequency of an EEG are related to different brain conditions. We usually consider specific frequency bands thought to be associated with specific brain rhythms: the alpha (8–13 Hz), beta (13–30 Hz), delta (0.5–4 Hz), and theta (4–7 Hz) bands. Alpha rhythm is characteristic of a relaxed state of consciousness. Alpha rhythms are best detected with the eyes closed. Alpha attenuates with drowsiness and open eyes, and is best seen over the occipital (visual) cortex. Beta rhythm is often associated with active, busy or anxious thinking and active concentration. Rhythmic beta with a dominant set of frequencies is associated with various pathologies and drug effects, especially. Delta rhythm is often associated with the very young and underlying lesions. Theta rhythm is associated with drowsiness, childhood, adolescence, and young adulthood.

In Pfurtscheller's study [21], movement or preparation for movement is typically accompanied by a decrease in mu and beta rhythms, particularly contra-lateral to the movement. An alpha-like normal variant called mu is sometimes seen over the motor cortex and attenuates with movement, or rather with the intention to move. This decrease is called "event-related desynchronization" (ERD). A rhythm increase, called "event-related synchronization" (ERS), on the other hand, occurs after movement and with relaxation. Further, the ERD/ERS is not necessary to actual movement. They have been researched occur with motor imagery [20]. Thus, the mu and beta rhythms are commonly used in BCI research.

Many methods [13,18,1,11,24,22,9,16] have been proposed to BCI in the past few years. As a classifier, linear discriminant analysis (LDA) [11,9], support vector machine (SVM), and recently neural network (NN) were used [13,18,1,24,22]. Guger et al. [11] used both recursive least squares and the LDA algorithms to aim at classify left- or right-hand movement. Furthermore, Garrett et al. [9] compared these linear (LDA) and two nonlinear classifiers (NNs and SVMs) applied to the classification of spontaneous EEG during five mental tasks. During mental imagination of specific movements, Dennis et al. [16] used the adaptive multiple regressions where the result EEG recorded from the sensorimotor cortex was classified on-line and used for cursor control. Due to the high-dimensional and artificial noise (e.g., tester blinks his/her eyes) of the EEG signals, the nonlinear classification methods are better than the linear ones [9]. Therefore, we will use the nonlinear NN methods to classify the high-dimensional and artificial noisy EEG signals.

In this paper, we propose a method to classify the EEG of mental tasks for left-hand movement imagination, right-hand movement imagination, and word generation. We expect that the three mental tasks can be classified and that they correspond to three control commands in a BCI system. First, we use principle component analysis (PCA) [6,7] to extract the EEG power spectrum density (PSD) feature. The PCA achieves a linear transformation of a high-dimensional data into a lower-dimensional subspace one whose components are uncorrelated. Next, we classify the PCA feature using artificial NNs. Artificial NNs are deliberately constructed to make use of organizational principles that resemble those of the human brain [12,15,8].

The NN has been trained by back-propagation (BP) algorithm, which uses gradient steepest descent method to train parameters of the network. However, the error curve may converge very slowly or the solution may fall into the local minimum, when it has been trained by BP. In recent years, many researchers studied to avoid falling into local minimum, such as [30] the optimal setting of initial weights optimal learning rates and momentum, finding optimal NN architectures using pruning techniques and construction techniques sophisticated optimization techniques and adaptive activation functions. Therefore, we use an improved particle swarm optimization (IPSO) to search for a better solution of the weights in a NN instead of the BP. The proposed IPSO method consists of the modified evolutionary direction operator (MEDO) and the traditional particle swarm optimization algorithm (PSO). The traditional PSO is an evolutionary algorithm that is based on the flocking of birds or the schooling of fish in nature. Since all the particles have local and global knowledge, PSO can search for and obtain the global solution quickly. The advantages of the proposed IPSO method are as follows: (1) it can prevent premature convergence; (2) it can accelerate the global search capacity using the MEDO; and (3) as demonstrated in Section 4, the IPSO method can find a better solution than the other methods in same iterations.

This paper is organized as follows. Section 2 describes the detailed descriptions of the EEG dataset, which is used in this study. The proposed analysis method, including the extract feature, the classifier, and the learning algorithm, is described in Section 3. The experimental results are shown in Section 4. Finally, conclusion and future work are given in Section 5.

2. EEG dataset

In this study, we use the dataset provided by IDIAP Research Institute (Silvia Chiappa, José del R. Millán) [5] in the Data Competition III.

This dataset contains data from three normal subjects taken during four non-feedback sessions. The subject sat in a normal chair, relaxed arms resting on their legs. All four sessions of a given subject were acquired on the same day, each session lasting 4 min, with 5–10 min breaks in between each session. The subject performed a given task for about 15 s and then switched randomly to another task at the operator's request (see Fig. 1). Three mental tasks were classified in this study:

1. Imagination of the left-hand movements (left).
2. Imagination of the right-hand movements (right).
3. Generation of words beginning with the same random letter (word).

EEG signals were recorded with a Biosemi system using a cap with 32 integrated electrodes located at standard positions of the International 10–20 system. The sampling rate was 512 Hz.

In this dataset, EEG data were not split into trials since the subjects were continuously performing any one of the mental tasks. The data were provided in pre-computed features. The raw EEG potentials were first spatially filtered by means of a surface Laplacian [17]. An EEG electrode actually measures a mixture of signals from several neuronal clusters. Spatial filters, such as a Laplacian filter, are usually applied to concentrate the signals to a single neuronal cluster.

Every 62.5 ms (i.e., 16 times per second) the power spectral density (PSD) in the 8–30 Hz band was estimated over the last second of the EEG data with a frequency resolution of 2 Hz for the eight centro-parietal channels C3, Cz, C4, CP1, CP2, P3, Pz, and P4. The detailed locations of the eight channels are shown in Fig. 2. The PSD was used the Welch periodogram method [19]. Hence, an EEG sample data was 96 dimensions (eight channels in 12

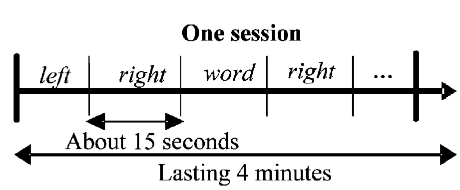

Fig. 1. The procedure of one session of each subject.

frequency components) [5]. The average PSD of all sessions for subject 1 was plotted in Fig. 3. In this figure, the mental tasks of imagination of left-hand movement, imagination of right-hand movement, and word generation are plotted in the left column, middle column, and right column, respectively.

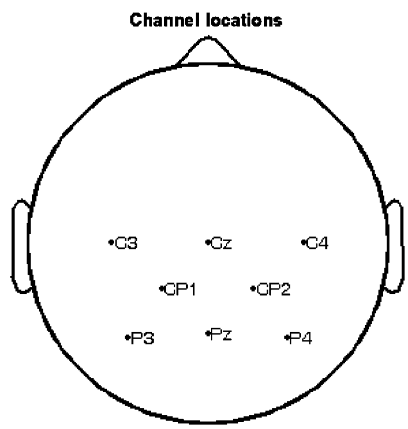

Fig. 2. The placement of the electrodes.

3. Analysis method

In this section, we introduce the proposed analysis method for EEG pattern classification. Fig. 4 shows the flowchart of data-processing procedure. First of all, the EEG data are recoded using downsampling of 2 Hz for later analysis. Second, the efficient PCA can find these features for k dimensions. Third, the mental signal will be classified by using a three-layer NN with IPSO algorithm. After training this network, the classified data can be determined. The detailed introduction will be presented in the following subsections.

3.1. Data preprocessing

Cross-validation is a valid assesses generalization error for stopping the training of a network. During performance analysis of network, cross-validation method can be used for determining large amount of training data. We performed the following cross-validation procedure for training the network as a way to control the over-fitting of training data. NN performance was assessed on both the training and the test validation set. We randomly select 75% of all dataset as training network and 25% of all dataset as test validation for each subject after each training epoch. In order to compare with other methods [28,26,3], the setup of the training

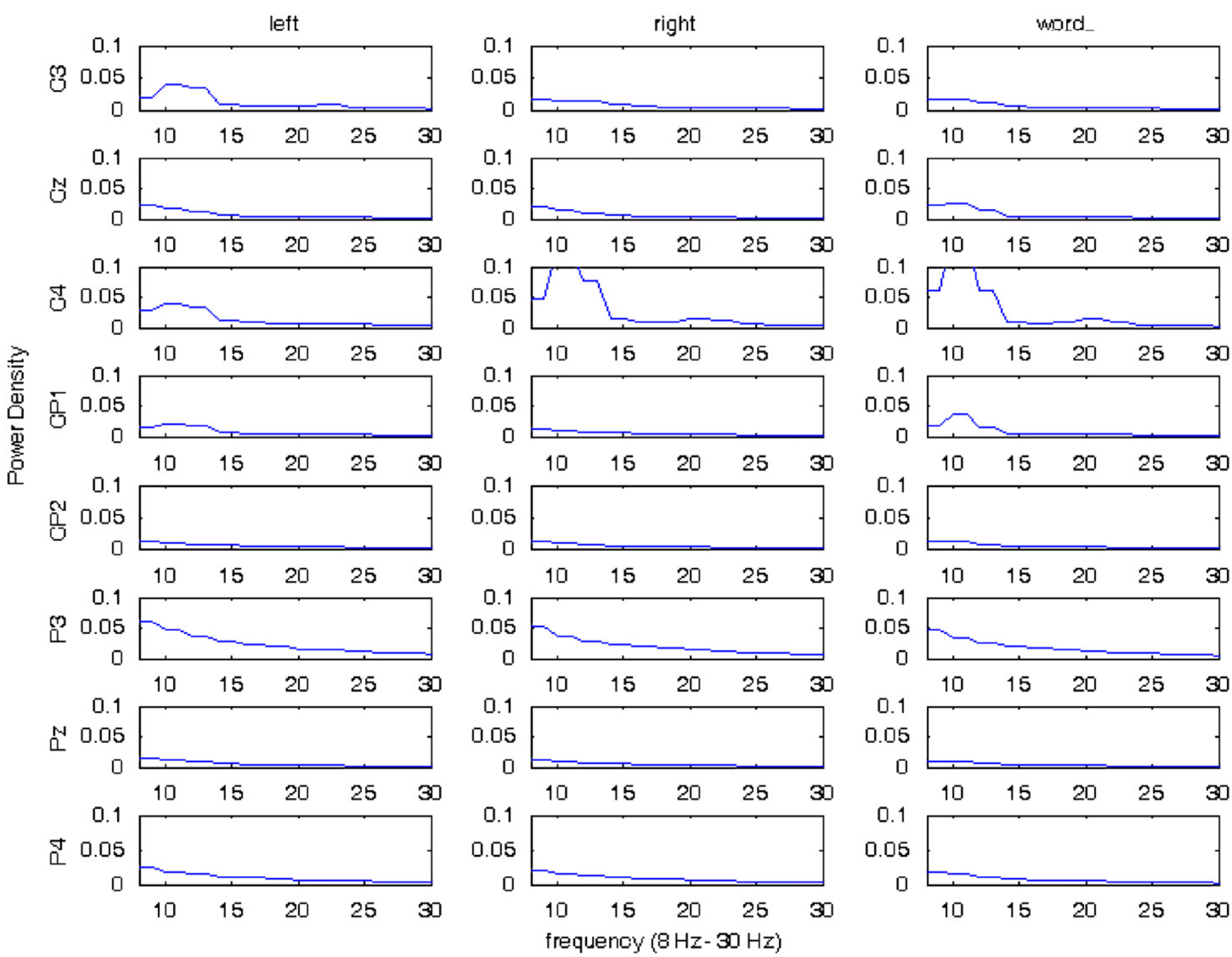

Fig. 3. The PSD value of subject 1.

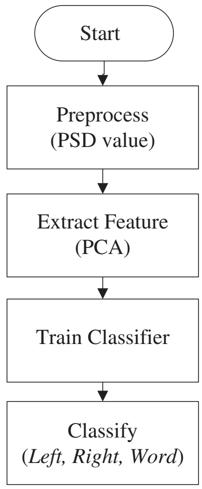

Fig. 4. The block diagram of the analysis procedure.

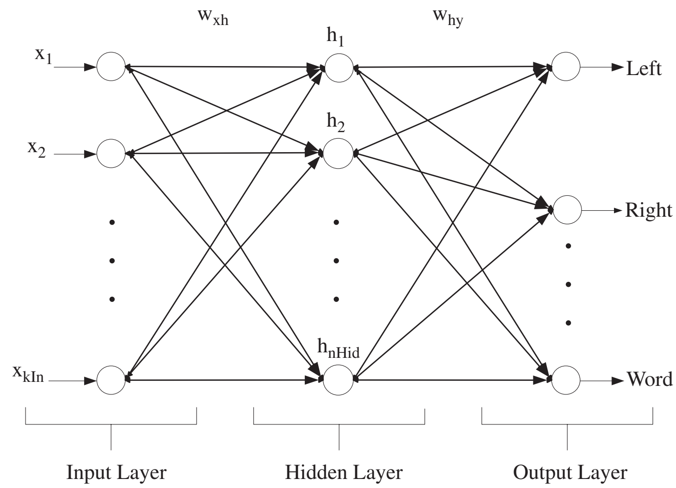

Fig. 5. Three-layer feed-forward neural network.

Table 1
Class distribution of the samples in the training data and testing data

Subject	Class	Number of training data	Number of testing data
1	*L*	366	130
	R	432	128
	W	518	180
2	*L*	370	108
	R	426	144
	W	504	182
3	*L*	426	150
	R	430	146
	W	430	140

set and testing set are recoded down sample to 2 Hz. Let *L* be imagination the left-hand movement, *R* be imagination the right-hand movement, and *W* be generation of words. The class distribution numbers of the training sample and testing sample for three subjects after preprocessing are shown in Table 1.

3.2. Extract feature using PCA

Before classification, we use the PCA to extract the PSD data that will be the input of the classifier. PCA is a statistics method and is widely used in pattern recognition. By using PCA, we can project PSD data from the higher 96-dimension vector to the lower *k*-dimension eigenspace, which is composed of *k* eigenvectors, and retain the feature information. The detailed steps of PCA are as follows:

Step 1: Compute the mean vector

$$\mathbf{m} = \frac{1}{nTr}\sum_{i=1}^{nTr}\mathbf{p}_i, \tag{1}$$

where $\mathbf{p}_i = [p_1 \cdots p_d]^t$ is the *i*th *d*-dimension training sample (i.e., PSD value of all channels) and *nTr* is the total number of the training samples.

Step 2: Compute the covariance matrix

$$\mathbf{\Sigma} = \sum_{i=1}^{nTr}(\mathbf{p}_i - \mathbf{m})(\mathbf{p}_i - \mathbf{m})^t, \tag{2}$$

where $\mathbf{\Sigma}$ is a $d \times d$ matrix.

Step 3: Find the eigenvalue and eigenvector from the covariance matrix by solving

$$\mathbf{\Sigma x} = \lambda \mathbf{x}, \tag{3}$$

which can be rewritten as follows:

$$(\mathbf{\Sigma} - \lambda \mathbf{I})\mathbf{x} = 0, \tag{4}$$

where $\mathbf{I}$ is the identity matrix and 0 is a zero vector. The solution vector $\mathbf{x} = \mathbf{e}_i$ and corresponding scalar $\lambda = \lambda_i$ are the eigenvector and associated eigenvalue, respectively. Because $\mathbf{\Sigma}$ is real and symmetric, there are *d* solution vectors $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_d\}$, each with an associated eigenvalue $\{\lambda_1, \lambda_2, \ldots, \lambda_d\}$, where the eigenvalues are sorted from large to small. The eigenvector with the highest eigenvalue represents the principle component of the dataset. That is, the eigenvectors with the larger eigenvalues represent the most significant relationship between the data dimensions.

Step 4: Generate a $d \times k$ matrix $\mathbf{A}$ whose columns consist of the *k* eigenvectors having the largest eigenvalues:

$$\mathbf{A} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_k]. \tag{5}$$

Step 5: Represent the original data by projecting the data onto the *k*-dimensional subspace according to

$$\mathbf{p}' = \mathbf{A}^t(\mathbf{p} - \mathbf{m}) \tag{6}$$

where $\mathbf{p}' = [\mathbf{p}'_1 \cdots \mathbf{p}'_k]^{\mathrm{T}}$ is the PCA feature vector.

If we need to extract the PCA feature of the testing data, we only do Step 5 again by replacing the training data $\mathbf{p}$ with the testing data.

3.3. NNs

For the classification of three mental tasks, a NN is used as a classifier to distinguish the PCA feature of the PSD value in the EEG. In order to adjust the parameter of the NN efficiently, we used the learning algorithm by IPSO. Before the classification process, we need to normalize the EEG data (i.e., the training data and the testing data) to between 0 and 1.

In this study, we used three-layer feed-forward artificial NN with one hidden layer and one output layer as shown in Fig. 5. Thus, *kIn* and *nHid* represent the number of input nodes and hidden nodes, respectively. An input vector is applied to the input layer, where all of the inputs are distributed to each unit in the hidden layer. All of the units have weight vectors, which are multiplied by these input vectors. Each unit sums these inputs and produces a value that is transformed by nonlinear activation function, for which we used the sigmoid function. The output of the final layer is then computed by multiplying the output vector from the hidden layer by the weights into final layer. More summations and activation of theses units then give the actual output of the network.

In this study, the NN consisted of one input layer, one hidden layer, and one output layer. The number of the input nodes is defined by the size of the PCA feature. Since the EEG is classified into three mental tasks, we set the three output nodes that correspond to the mental tasks *left*, *right*, and *word*, respectively. The three outputs are represented by unit vectors: $left = [1\,0\,0]$, $right = [0\,1\,0]$, and $word = [0\,0\,1]$. The different hidden nodes of the NN are tested and explained in Section 4.

3.4. An IPSO

Particle swarm optimization is an evolutionary computation technique developed by Kennedy and Eberhart [10,14,2]. It is a global search method based on natural systems. PSO originated from studies of the social behavior of animals, such as birds flocking and fish schooling.

The system initially has a swarm of random solutions. Each potential solution, called a particle, is given a random initial velocity and is flown through the problem space. At each time step n, the velocity is updated by the following equation:

$$\begin{aligned} v_{i,d}(n+1) = {} & \omega \times v_{i,d}(n) + c_1 \times rand_1 \times (p_{\text{lbest\_}i,d} - x_{i,d}(n)) \\ & + c_2 \times rand_2 \times (p_{\text{gbest},d} - x_{i,d}(n)). \end{aligned} \tag{7}$$

In the updating velocity equation, the new velocity $v_{i,d}$ of particle i in dimension d is computed by summing three components provided by the information from previous searches. The first component is referred to as "inertia" and is the current velocity of the given particle computed from a previous iteration. The second component is referred to as "individual knowledge" and is the best position $p_{\text{lbest\_}i,d}$ found by the particle i; this position is called the local best. The third component is referred to as "group knowledge" and is the best position $p_{\text{gbest},d}$ found by all the particles; this position is called the global best. The basic concept of the PSO technique lies in accelerating each particle towards its local best and the global best locations iteratively. The first, second, and third components are scaled using the constants ω, c_1, and c_2, respectively. Controlling the proportion of the moving velocity between the inertia, individual knowledge and group knowledge will make the solution towards the local or global optimum and influence the convergence speed [23]. In addition, $rand_1$ and $rand_2$ are two uniform random functions and give random search in the problem space.

After the new velocity is obtained, the particle updates the position using the following equation:

$$x_{i,d}(n+1) = x_{i,d}(n) + v_{i,d}(n+1), \tag{8}$$

where $x_{i,d}$ represents a position of the particle.

The traditional evolutionary direction operator (EDO) [29] was used for GA originally. The EDO selects good target points (i.e., particles), and other points will move toward these good target points. Because the GA is a random search method, it cannot obtain optimal solutions efficiently and quickly. The EDO method can enhance GA to search for the optimal solution. The main shortcoming of the EDO is that the new particle created from three arbitrary particles in each generation cannot be certain to have good evolutionary direction. For this reason, the evolutionary direction may not be toward a better direction. Therefore, we propose a MEDO to improve this shortcoming of the traditional EDO. We use the MEDO to enhance the capability of the traditional PSO to find the optimal solution. The new algorithm is called the IPSO. Fig. 6 shows the flowchart of the proposed IPSO learning process. The whole learning process is described step-by-step as follows:

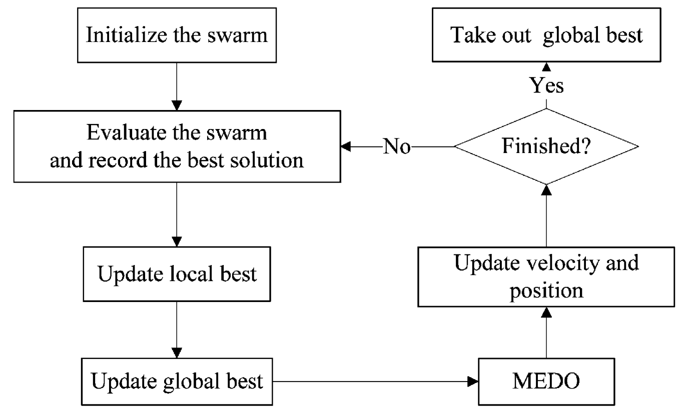

Fig. 6. The procedure of proposed IPSO.

3.4.1. Initialize the swarm

The swarm is initialized using m particles with random positions and velocities of D dimensions in the problem space. In this study, the coordinate positions of a particle in dimension d correspond to each variable (weight and bias) that will be trained in a NN (see Fig. 7).

3.4.2. Evaluate the swarm

For each particle, evaluate its fitness according to the desired optimization. For the evaluation of a NN, the root-mean-square-error ($RMSE$) is used to compute the average output error and is defined as

$$RMSE = \sqrt{\frac{1}{nTr}\sum_{i=1}^{nTr}\sum_{j=1}^{nOut}(t_{i,j} - y_{i,j})^2}, \tag{9}$$

where nTr is the number of training data, $nOut$ the number of network outputs, and $t_{i,j}$ and $y_{i,j}$ the jth target output and real network output of the ith training data, respectively. The fitness function is defined as

$$fitness = \frac{1}{1 + RMSE}. \tag{10}$$

Fitness was used for measuring performance of the NN. After the evaluation of the initial swarm, the local best of each particle and the global best will be recorded. Naturally, in the initial state, the local best of each particle is its initial position. The global best stores the index of the particle, which has the best local best of all the particles.

3.4.3. Update local best and global best

In this step, we update the local best and the global best. If the fitness value of the particle is higher than that of the local best, then the local best will be replaced with the particle; and further, if the local best is better than the current global best, then the global best has to be replaced in the swarm.

3.4.4. MEDO

With the MEDO, we choose the three best solutions in each generation to perform the evolutionary direction operation. The new solution is superior to the original best solution.

After learning, we can obtain the three best particles. These three best particles are ordered according to their fitness and are called the "low," "medium," and "high" particles. Three inputs

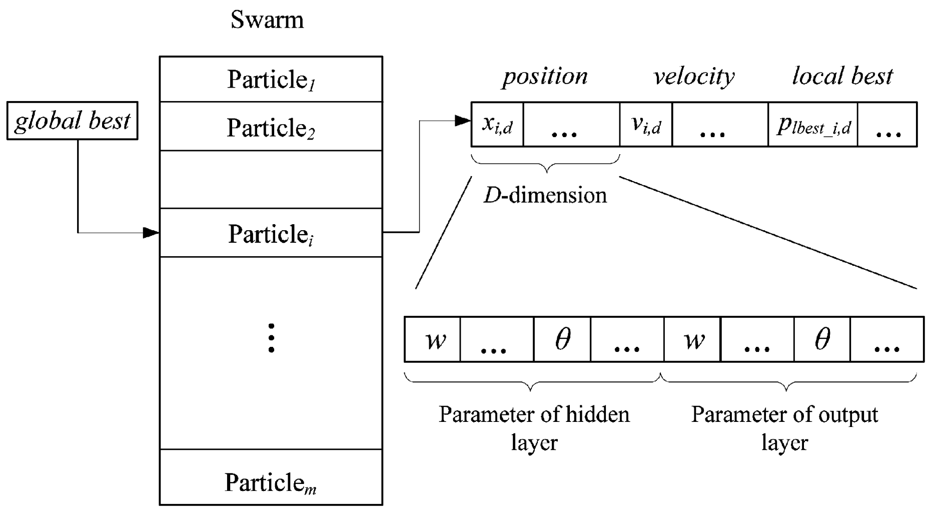

Fig. 7. The structure of the swarm.

(preferred) and the output (created) particles are denoted as follows:

Input particles:

- "low" particle, $C_l = (C_{l1}, C_{l2}, C_{l3}, \ldots, C_{ld})$, with fitness F_l.
- "medium" particle, $C_m = (C_{m1}, C_{m2}, C_{m3}, \ldots, C_{md})$, with fitness F_m.
- "high" particle, $C_h = (C_{h1}, C_{h2}, C_{h3}, \ldots, C_{hd})$, with fitness F_h.

Output particle, $C_o = (C_{o1}, C_{o2}, C_{o3}, \ldots, C_{od})$, with fitness F_o.

First, the "low" particle is updated using a migration operation to generate a new "low" particle. Next, the "medium" particle and the "high" particle will be set as the moving target direction of the new "low" particle. That is, the new "low" particle will be updated again toward the "medium" and "high" particles. This process improves the global search capacity. Fig. 8 shows the flowchart of the proposed MEDO. The detailed MEDO is described as follows:

Step 1: Set the magnitudes of the two evolutionary directions to 1 (i.e., $D_1 = 1$ and $D_2 = 1$), where D_1 and D_2 are the constant values. Then we also set the initial index of the MEDO to 1 (i.e., $T_s = 1$), the loop number of the MEDO to N_L (termination condition, i.e., $N_L = 10$), and the three particles with the best fitness values from the local best swarm to C_h, C_m, and C_l.

Step 2: The migration operation [4] in the MEDO is used to regenerate a newly diverse population, which prevents individuals from gradual clustering. Thus, the migration operation greatly increases the amount of search space explored for a small swarm. The migrant individuals are generated based on the best individual, $X_i = (x_{i1}, x_{i2}, x_{i3}, \ldots, x_{id})$, by non-uniform random selection. We use Eq. (11) to update the low particle (C_l):

$$x_{id} = \begin{cases} x_{id} + \rho(x_{id}^{L} - x_{id}) & \text{if } r_1 < \dfrac{x_{id} - x_{id}^{L}}{x_{id}^{L} - x_{id}^{U}}, \\ x_{id} + \rho(x_{id}^{U} - x_{id}) & \text{otherwise,} \end{cases} \tag{11}$$

where ρ and r_1 are random numbers in the range of $[0, 1]$. x_{id}^{L} is the lower bound on this particle and x_{id}^{U} is the upper bound on this particle. If the fitness value is not improved and this optimum value is a local optimum, then we use the migration operation to solve this problem. The migration operation is the migrant x_{id} to other points and this idea of migration operation can escape the local extreme value trap.

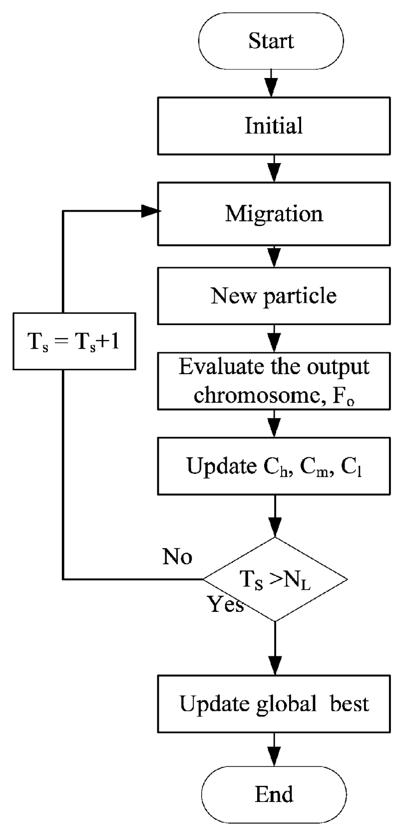

Fig. 8. Flowchart of the proposed MEDO.

Step 3: Compute C_o using

$$C_{oj} = C_{lj} + D_1 \times (C_{lj} - C_{mj}) + D_2 \times (C_{lj} - C_{hj}). \tag{12}$$

Starting from the base point C and with the use of two difference vectors, $D_1 \times (C_{lj} - C_{mj})$ and $D_2 \times (C_{lj} - C_{pj})$, the next evolutionary direction and the next evolutionary step-size can be determined using this parallelogram. Point C_o can then be created along the evolutionary direction with the evolutionary step size.

Step 4: Evaluate the new fitness (F_o) of the newly created output particle (C_o).

Step 5: Update the "low" particle (C_l), "medium" particle (C_m), and "high" particle (C_h). The updating process is as follows:

(1) If $F_o > F_h$, then $C_h = C_o$, $C_m = C_h$, and $C_l = C_m$.
(2) Else if $F_o > F_m$ and $F_o < F_h$, then $C_m = C_o$ and $C_l = C_m$.
(3) Else if $F_o > F_l$ and $F_o < F_m$, then $C_l = C_o$.
(4) Else if $F_o = F_l = F_m = F_h$, then $C_o = C_o + N_r$ ($N_r \in [0,1]$).
(5) Else if $F_o < F_l$, then $D_1 = D_1 \times -0.5$ and $D_2 = D_2 \times -0.5$.

According to statements (1)–(3) in the process above, we update the "low" particle, the "medium" particle, and the "high" particle. In statement (4), when the new particle, the "low" particle, the "medium" particle and the "high" particle all have the same fitness values, the particle will fall into a local optimum. Thus, a random number (N_r) is added to prevent the learning algorithm from falling into a local optimum. If the fitness value of the new particle in statement (5) is not good, we will decrease the moving velocity (i.e., D_1 and D_2) to obtain a good fitness.

Step 6: In this step, we determine whether the MEDO is to be terminated. If the MEDO is terminated, go to step 7. Otherwise, $T_s = T_s + 1$, and go to step 2.

Step 7: In this step, we update the global best. The updating process of the global best is as follows: If the fitness value of the new particle is higher than that of the global best, then the global best will also be replaced with the particle.

Finally, compute the new velocity of the particle according to Eq. (7) and move it to the next position according to Eq. (8).

4. Experimental results

In this section, we will present the classification performance of the proposed analysis method. The selection of the parameters of the IPSO will critically affect the simulation results, and the value will be based on practical experimentation or on trial-and-error tests. In this study, ω is taken to be a constant 0.4 throughout the optimization process. We used $c_1 = 1$ and $c_2 = 2$ in Eq. (7), and these could a further toward best position in the second component of the velocity update equations. The particle numbers could be tuned raise for the study at hand, nevertheless the implementation of the particle swarm optimization is occupied a large memory. Therefore, each run has been conducted with a particle of 50 individuals. Each result is computed by averaging 10 trials. After 1500 iterations, the network state at the iteration for which the validation error is smallest.

Furthermore, we make a preliminary analytical observation of the feature, which was extracted using PCA. In PCA, if the training data has the distinct feature, the first several largest eigenvalue will be clearly larger than others. This indicates that we can project the data into the feature space with lower dimension. The distribution of the 96 eigenvalues that were created by PCA (i.e., 96 dimensions of EEG sample data) is shown in Figs. 9–11 for the data from subject 1, subject 2, and subject 3, respectively. Decoding the EEG signal is not a straightforward task. The signal is very weak and many artifacts can be present just blinking an eye may add noise to the signal. We observe that the data from subject 1 has the largest eigenvalue while the data from subject 3 has the smallest eigenvalue.

Next, we test validation by using different number of hidden nodes and the different dimensions of feature. We defined the classification accuracy as follows:

$$\text{Classification accuracy} = \frac{\text{Number of correct classified mental tasks}}{\text{Number of total mental tasks}}. \tag{13}$$

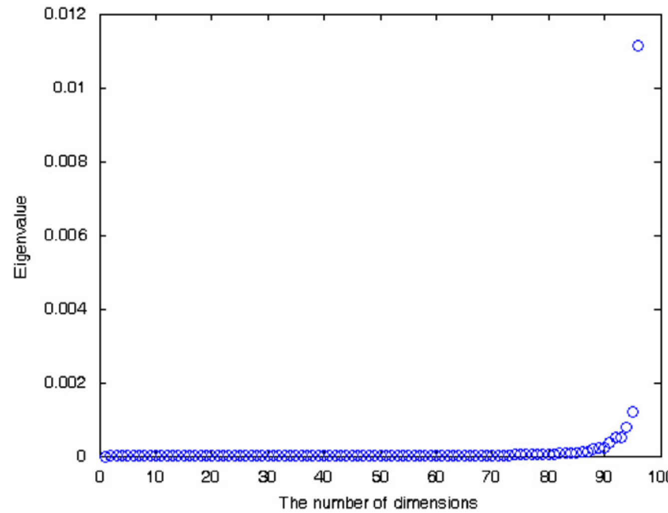

Fig. 9. The distribution of eigenvalues in subject 1.

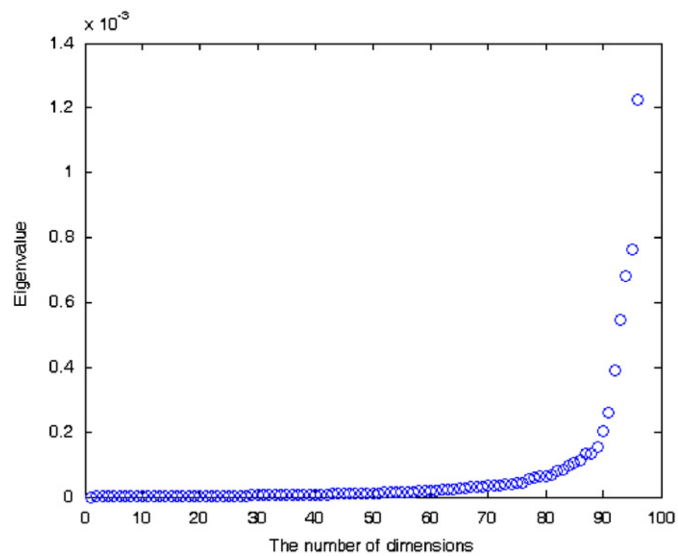

Fig. 10. The distribution of eigenvalues in subject 2.

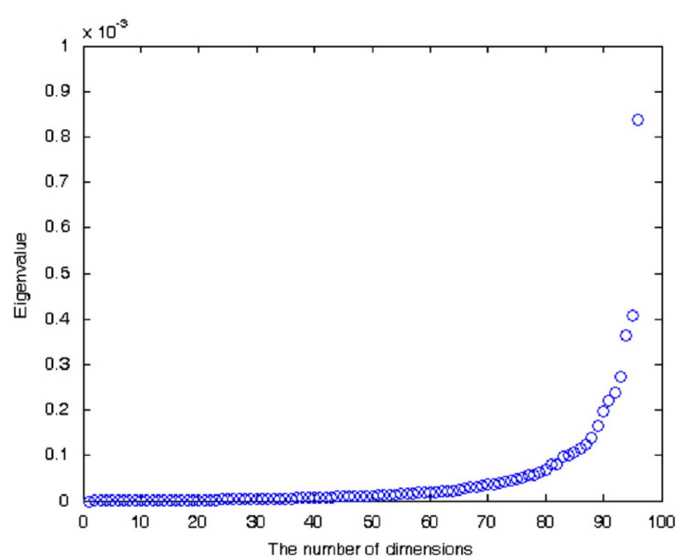

Fig. 11. The distribution of eigenvalues in subject 3.

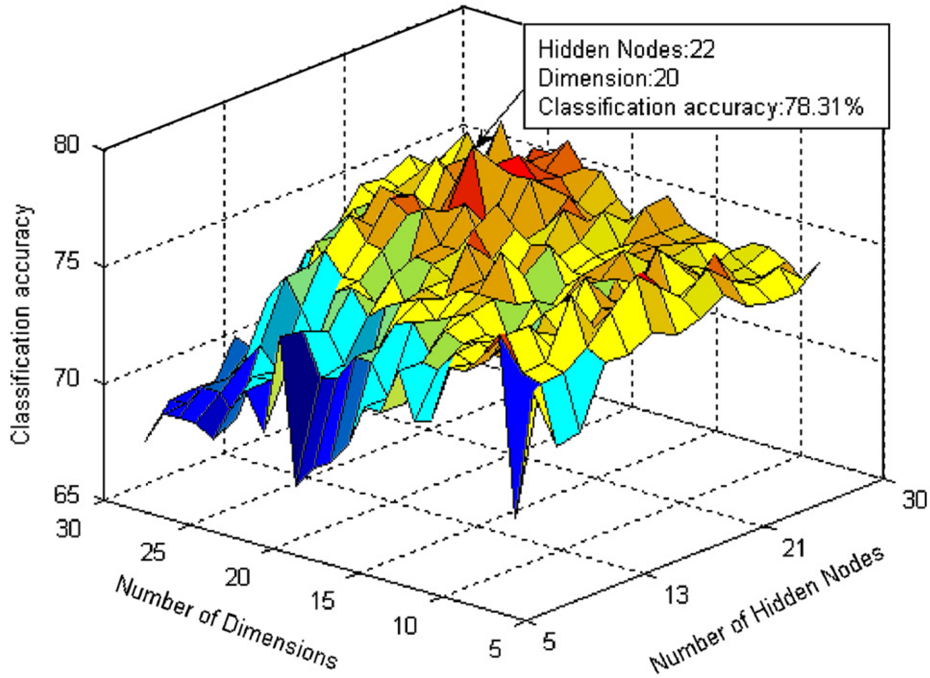

Fig. 12. The classification accuracy for different feature dimensions and hidden nodes for the subject 1.

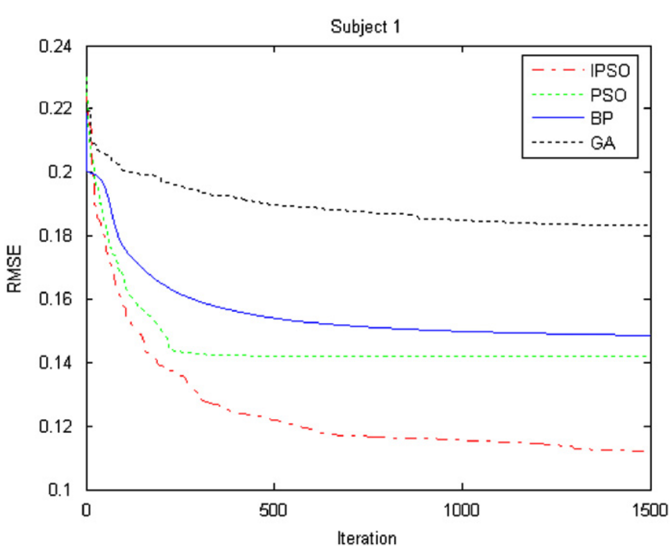

Fig. 13. The learning curve of subject 1.

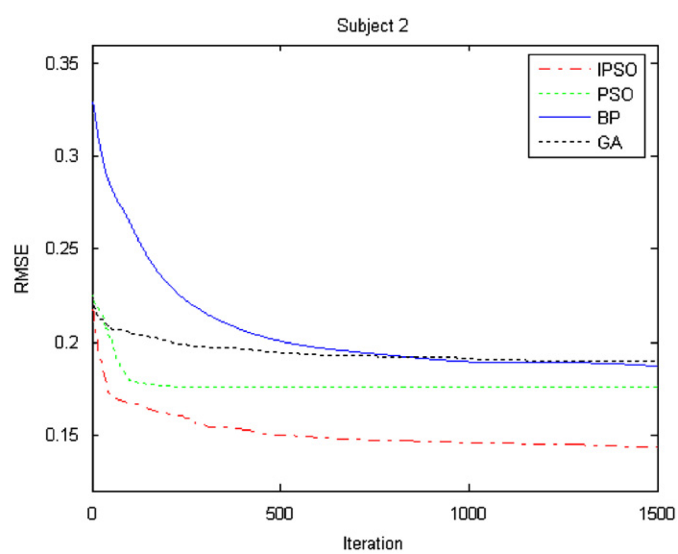

Fig. 14. The learning curve of subject 2.

The classification accuracy for different feature dimensions and hidden nodes is shown in Fig. 12. In this figure, we find that 20 feature dimensions and 22 hidden nodes used can obtain the best performance for subject 1.

In order to show the effectiveness and efficiency of the proposed IPSO-NN, we applied the BP algorithm, the genetic algorithm (GA) and traditional PSO to the same problem. In the BP, the learning rate η was set to 0.1. The one point crossover rate and mutation rate of the GA were set to 0.5 and 0.1. In the PSO, the parameters of IPSO and PSO were set to the same. In the BP, GA, PSO, and IPSO, the evolutionary learning processed for 1500 generations and was repeated 10 times. For the evaluation of the classifier during the optimization, a 10-fold 10-cross-validation was computed. The comparison results are shown in Figs. 13–15; the data used in the figures were taken from the datasets of subject 1, subject 2, and subject 3, respectively. In these figures, we find that the proposed method converges quickly and obtains a lower *RMSE* than other methods.

Recently, several researchers [28,26,3] have proposed many classifiers to improve the classification accuracy of the EEG of two mental tasks of left-hand movement imagination and right-hand movement imagination. In [28], Xiaomei et al. used the Fisher discriminant analysis (FDA) as the base learner to distinguish the mental tasks. Firstly, the eight consecutive PSD samples in each frame of 62.5 ms are averaged t in every 0.5 s. Secondly, according to the Fisher ratio of the averaged PSD within the three classes, the most reactive PSD features are selected to form the new feature vectors for further classification. Finally, by the

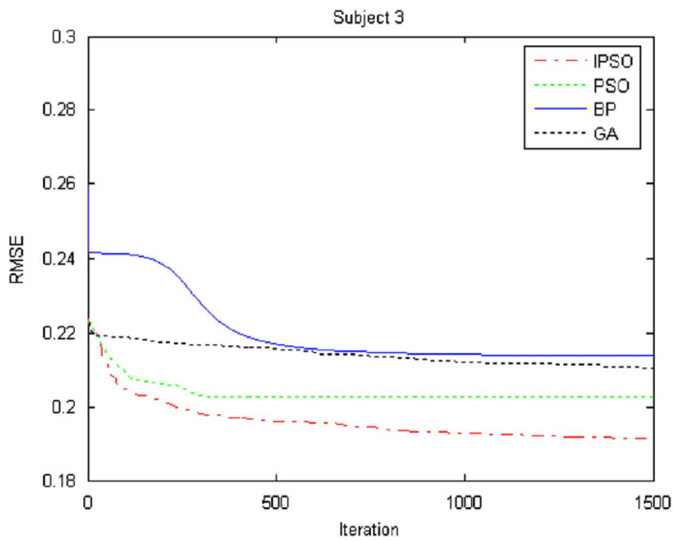

Fig. 15. The learning curve of subject 3.

Table 2
The classification accuracy using various methods

Subject methods	Accuracy of subject 1 (%)	Accuracy of subject 2 (%)	Accuracy of subject 3 (%)	Average (%)
IPSONN	78.31	70.27	56.46	68.35
PSONN	75.98	69.78	53.83	66.33
BPNN	76.02	65.89	51.14	64.35
GANN	69.315	60.32	44.40	58.01
FDA	76.03	69.36	51.61	65.67
RDA	78.08	63.83	52.72	64.91
SVM	77.85	66.36	53.44	65.90

kernel-based FDA, the two mental tasks could be discriminated. The kernel parameters are selected by cross-validation to minimize the classification error. The regularized discriminant analysis (RDA) [26], applied to motor imagery data of the subjects. The classifier can separate better these two-class motor imagery data. Their task was to imagine left- or right-hand movements depending on the cue. This suggests that the effort needed to find regularizing parameters for RDA. The SVMs [3] has also been used to classify the electroencephalogram of mental tasks. The complexity of the SVM is characterized by the number of a subset of training data rather than the dimensionality of the transformed feature space.

In this paper, a neural classifier based on IPSO is proposed to classify an electroencephalogram of three mental tasks for left-hand movement imagination, right-hand movement imagination, and word generation. We compare our method with RDA, FDA and SVM. Table 2 shows a performance comparison of various existing models. The regularization and the degree parameters were both set to 0.01 in RDA. In SVM, the margin constraint was set to 200. As shown in Table 2, the accuracy rate of the proposed method is better than those of the methods given in RDA, FDA, and SVM. The result indicates that the nonlinear classification methods (IPSO and SVM) are better than the linear ones (FDA and RDA). But SVM cannot be analyzed under electroencephalogram of three mental tasks. In other algorithms, our approach searches the ability to solve solution space more than other methods. GA needs more learning iterations consuming than IPSO. The classification accuracy of GA is lower than other evolutionary algorithms in 1500 iterations. Our method outperforms other methods in performance.

5. Conclusion and future work

This paper has proposed the IPSO to train NNs for EEG mental task classification. We use PCA to extract the feature of the PSD value of an EEG. Using PCA, we can reduce the data from 96 dimensions to 20 dimensions. Then the PCA feature is classified using a three-layer feed-forward NN. The parameters of the NN are trained using the IPSO algorithm. In the experimental result, the average classification accuracy reached 78.31% for subject 1, 70.27% for subject 2, and 56.46% for subject 3. The results show that IPSO has faster convergence speed than BP and can find a better solution in same iterations. However, the proposed IPSO method is used on off-line work presently. In this study, we attempt to emphasize the methodology of the proposed IPSO learning method. Therefore, we will extend the proposed method to work on-line in the future work. These experiments will be also performed on multiple subjects and multiple sessions.

In order to obtain good simulation results, the search parameters are an important concern. In this study, the selection of the parameter of the IPSO will critically affect the simulation results. Although we cannot guarantee the minimum performance in all cases, the adjustable parameters will be based on practical experimentation or on trial-and-error tests. However, many methods can be envisioned, especially in connection with the kind of the search parameter used. For example, Taguchi method [25] could improve the quality of a product, processes, and equipment. The fundamental principle of Taguchi method is to improve the quality of a product by minimizing the effect of the causes of variation without eliminating the causes. In the future work, we will adopt Taguchi method to solve the search parameter problem.

References

[1] E. Abbas, G. Mahdi, EEG signals can be used to detect the voluntary hand movements by using an enhanced resource-allocating neural network engineering in medicine and biology society, Proc. IEEE Int. Conf. Ann. 1 (2001) 721–724.
[2] D.W. Boeringer, D.H. Werner, Particle swarm optimization versus genetic algorithms for phased array synthesis, IEEE Trans. Anten. P. 52 (2004) 771–779.
[3] R. Boostani, B. Graimann, M.H. Moradi, G. Pfurtscheller, A comparison approach toward finding the best feature and classifier in cue-based BCI, Med. Biol. Eng. Comput. 45 (4) (2007) 403–412.
[4] C.L. Chiang, Improved genetic algorithm for power economic dispatch of units with vale-point effects and multiple fuels, IEEE Trans. Power Syst. 20 (4) (November 2005) 1690–1699.
[5] J. del R. Millán, On the need for on-line learning in brain–computer interfaces, in: Proceedings of the International Joint Conference on Neural Networks, 2004.
[6] K.I. Diamantaras, S.Y. Kung, Principal Component Neural Networks: Theory and Applications, Wiely, Inc., New York, 1996.
[7] R.O. Duda, P.E. Hart, D.G. Stork, Pattern Classification, Wiely, Inc., New York, 2001.
[8] T. Fukuda, T. Shibata, Theory and applications of neural networks for industrial control systems, IEEE Trans. Ind. Electron. 39 (1992) 472–489.
[9] D. Garrett, A.P. David, C.W. Anderson, M.H. Thaut, Comparison of linear, nonlinear, and feature selection methods for EEG signal classification, IEEE Trans. Rehab. Eng. 11 (2) (2003) 141–144.
[10] V.G. Gudise, G.K. Venayagamoorthy, Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks, in: Proceedings of the 2003 IEEE on Swarm Intelligence Symposium, 2003.
[11] C. Guger, A. Schlögl, C. Neuper, D. Walterspacher, T. Strein, G. Pfurtscheller, Rapid prototyping of an EEG-based brain–computer interface (BCI), IEEE Trans. Rehab. Eng. 9 (2001) 49–58.
[12] M.T. Hagan, H.B. Demuth, M. Beale, Neural Network Design, International Thomson Publishing Inc., Boston, Mass., USA, 1996.
[13] J. Kalcher, D. Flotzinger, S. Gölly, C. Neuper, G. Pfurtscheller, Graz brain–computer interface (BCI) II, in: W.L. Zagler, G. Busby, R.R. Wagner (Eds.),

Computers for Handicapped Persons, Springer, Berlin, Germany, 1994, pp. 170–176 ICCHP'94.
[14] J. Kennedy, R. Eberhan, Particle swarm optimization, in: IEEE International Conference on Neural Networks, 1995, pp. 1942–1948.
[15] C.T. Lin, C.S.G. Lee, Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent System, Prentice-Hall, Englewood Cliffs, NJ, 1996.
[16] D.J. McFarland, J.R. Wolpaw, Sensorimotor rhythm-based brain–computer interface (BCI): feature selection by regression improves performance, IEEE Trans. Rehab. Eng. 13 (3) (2005) 372–379.
[17] D.J. McFarland, L.M. McCane, S.V. David, J.R. Wolpaw, Spatial filter selection for EEG-based communication, Electroen. Clin. Neurophysiol. 103 (1997) 386–394.
[18] J.R. Millan, J. Mourino, M. Franze, F. Cincotti, M. Varsta, J. Heikkonen, F. Babiloni, A local neural classifier for the recognition of EEG patterns associated to mental tasks, IEEE Trans. Neural Networks 13 (3) (2002).
[19] S.K. Mitra, Digital Signal Processing: A Computer-Based Approach, second ed, McGraw-Hill, Inc., New York, 2001.
[20] G. Pfurtscheller, C. Neuper, Motor imagery activates primary sensorimotor area in humans, Neurosci. Lett. 239 (1997) 65–68.
[21] G. Pfurtscheller, F.H. Lopes da Silva, Event-related EEG/MEG synchronization and desynchronization: basic principles, Clin. Neurophysiol. 110 (1999) 1842–1857.
[22] G. Pfurtscheller, J. Kalcher, C. Neuper, D. Flotzinger, M. Pregenzer, On-line EEG classification during externally-placed hand movements using a neural network-based classifier, Electroen. Clin. Neurophysiol. 99 (1996) 416–425.
[23] A. Ratnaweera, S.K. Halgamuge, H.C. Watson, Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients, IEEE Trans. Evol. Comput. 8 (2004) 240–255.
[24] P. Shenoy, R.P.N. Rao, Dynamic Bayesian Networks for Brain–Computer Interfaces. NIPS 2004 (Advances in NIPS 17, pp. 1265–1272, 2005).
[25] J.T. Tsai, T.K. Liu, J.H. Chou, Hybrid Taguchi-genetic algorithm for global numerical optimization, IEEE Trans. Evol. Comput. 8 (4) (August 2004) 365–377.
[26] C. Vidaurre, R. Scherer, R. Cabezal, A. Schlögl, G. Pfurtscheller, Study of discriminant analysis applied to motor imagery bipolar data, Med. Biol. Eng. Comput. 45 (1) (2007) 61–68.
[27] J.R. Wolpaw, N. Birbaumer, W.J. Heetderks, D.J. McFarland, P.H. Peckham, G. Schalk, E. Donchin, L.A. Quatrano, C.J. Robinson, T.M. Vaughan, Brain–computer interface technology: a review of the first international meeting, IEEE Trans. Rehab. Eng. 8 (2000) 164–173.
[28] P. Xiaomei, Z. Chongxun, X. Jin, B. Guangyu, Adaboost for improving classification of left and right hand motor imagery tasks 2005, in: First International Conference on Neural Interface and Control, Wuhan, China, May 2005, pp. 26–28.
[29] K. Yamamoto, O. Inoue, New evolutionary direction operator for genetic algorithms, AIAA J. Tech. Notes 33 (10) (1995) 1990–1993.
[30] X.-H. Yu, G.-A. Chen, Efficient backpropagation learning using optimal learning rate and momentum, Neural Networks 10 (3) (April 1997) 517–527.

Cheng-Jian Lin received the B.S. degree in electrical engineering from Ta-Tung University, Taiwan, ROC, in 1986 and the M.S. and Ph.D. degrees in electrical and control engineering from the National Chiao-Tung University, Taiwan, ROC, in 1991 and 1996. From April 1996 to July 1999, he was an Associate Professor in the Department of Electronic Engineering, Nan-Kai College, Nantou, Taiwan, ROC. From August 1999 to January 2005, he was an Associate Professor in the Department of Computer Science and Information Engineering, Chaoyang University of Technology. From February 2005 to July 2007, he was a full Professor in the Department of Computer Science and Information Engineering, Chaoyang University of Technology. Currently, he is a full Professor of Electrical Engineering Department, National University of Kaohsiung, Kaohsiung, Taiwan, ROC. He served as the chairman of Computer Science and Information Engineering Department, Chaoyang University of Technology from 2001 to 2005. He served as the library director of Poding Memorial Library, Chaoyang University of Technology from 2005 to 2007. Dr. Lin served as the Associate Editor of International Journal of Applied Science and Engineering from 2002 to 2005. His current research interests are soft computing, pattern recognition, intelligent control, image processing, bioinformatics, and FPGA design. He has published more than 150 papers in the referred journals and conference proceedings. Dr. Lin is a member of the Phi Tau Phi. He is also a member of the Chinese Fuzzy Systems Association (CFSA), the Chinese Automation Association, the Taiwanese Association for Artificial Intelligence (TAAI), the IEEE Systems, Man, and Cybernetics Society, and the IEEE Computational Intelligence Society. He is an executive committee member of the Taiwanese Association for Artificial Intelligence (TAAI) from 2003 to 2008. He is an executive committee member of the Chinese Fuzzy Systems Association (CFSA) from 2007 to 2008. Dr. Lin has received several honors and awards, including the 2006 Outstanding Paper Award of the 11th Conference on Artificial Intelligence and Applications, the 2007 Outstanding Paper Award of the 12th Conference on Artificial Intelligence and Applications, and the 2006 Best Paper Award of International Trans. on Computer Science and Engineering (vol. 32(1)).

Ming-Hua Hsieh was born in Taipei, Taiwan, ROC in 1984. He received the B.S. degree in Biomedical Engineering from the Yuanpei University of Technology, Taiwan, ROC, in 2006. He received the M.S. degree in Computer Science and Information Engineering from Chaoyang University of Technology, Taiwan, ROC, in 2008. His research interests are neural networks, fuzzy systems, pattern recognition, and bioinformatics.