Seizure source localization using a hybrid second order blind identification and extended rival penalized competitive learning algorithm

Delaram Jarchi a,*, Reza Boostani a, Mohammad Taheri a, Saeid Sanei b

a Department of Computer Science and Engineering, Shiraz University, Shiraz, Iran
b Centre of DSP, School of Engineering, Cardiff University, Cardiff, UK

1. Introduction

Epilepsy is one of the most common neurological disorders. More than 60 million people worldwide are diagnosed with epilepsy whose hallmark is recurrent seizure [15,16]. EEG, positron emission tomography (PET) and functional magnetic resonance imaging (fMRI) are the most commonly used data acquisition systems for evaluation of brain activity and neurological abnormalities but EEG is the most common modality because of its high temporal resolution and low cost.

The main motivation in seizure localization is reducing the need for invasive intracranial EEGs. Several methods for epileptic localization based on PET [27], single photon emission computed tomography (SPECT) [28], combined magnetoencephalography (MEG) and EEG [26] are applied to find the seizure foci. In this paper we have used scalp EEG signals to localize the seizure foci in the brain based on dipole source localization.

The EEG signals recorded from the scalp are the electrical reflection of spatio-temporal ionic neuron activities. In order to achieve the underlying information of the brain activities, EEG can be considered as a mixture of a limited number of sources within the brain. Therefore, the behavior of source signals can describe the brain activity. Hence, an effective way to obtain the brain source signals is necessary. Several blind source separation (BSS) algorithms such as Infomax [12], SOBI [8], and JADE [13] have been applied to the raw EEG signals for extracting the main brain sources. Most of BSS algorithms assume non-Gaussian distribution of brain source signals. However, the actual statistics of these sources are generally yet unknown [8]. In this study, the well-known SOBI algorithm is used to extract the source signals, mainly because it exploits non-stationarity of the data by performing joint diagonalization of multiple time-lagged covariance matrices of the estimated sources [8].

In these cases where the number of sources is unknown, a common practice in BSS is to assume that the number of brain sources and electrode channels are equal [8]. However, when the number of EEG recording electrodes is reasonably high, it is expected to have a fewer number of sources than electrodes, where some of the estimated sources are likely to be noise and artifacts. Thus, estimation of the number of sources is useful.

In patients with focal seizure, the location of the seizure source remains approximately fixed. Since the row vectors of the unmixing matrix include the information about projection of the source signals into the electrodes’ space, clustering the rows of the resulted unmixing matrices calculated by SOBI algorithm in different windows and then considering each cluster center is helpful to identify the brain sources especially the seizure source. Estimation of the brain sources is performed through multiplication of each cluster center by the electrode signals.
It has been reported that seizure signals exhibit less chaotic (more ordered) behavior during the seizure onset. Therefore, the dynamical changes can be evaluated and used as a measure to detect seizure source. After detecting the estimated seizure source, it can be projected back to the electrode’s space and any dipole localization method such as multiple fixed dipole imaging [19], brain electrical source analysis (BESA) [20] and multiple signal classification (MUSIC) [25] can be used to estimate the seizure dipole location. The main objective of this research is to develop a more accurate estimation of the seizure source. The clustering technique and SOBI algorithm is an effective approach toward obtaining a better estimation of the seizure source. The projected seizure source can be represented by a single dipole within the brain. Therefore, a single dipole source localization method is helpful for localization of the estimated seizure source. In this research, the EEG signals from 3 subjects with focal seizure are recorded. The algorithm for extracting the brain sources includes two stages [17]: first, SOBI algorithm is applied to different time frames of each channel to extract a number of sources in each segment. Then, our extended clustering method, which is the main contribution of this paper, is used to cluster the rows of the unmixing matrices calculated in different frames. By applying Dipole Fitting (DIPFIT) method [19], which is based on a non-linear fitting of a single dipole model to explain the scalp potential distribution, the source location is estimated within a four-shell spherical model of the head [18]. To verify our method, a set of simultaneously recorded intracranial signals was used to test the accuracy of the system.

The remainder of this paper is structured as follows. In Section 2, estimation of the number of independent brain sources is described. In Section 3, the SOBI algorithm is briefly described and in Section 4, an extended clustering method is proposed. Then in Section 5, the stages to project the extracted seizure source onto electrodes’ space is described. In Section 6 the results for both real and simulated data are shown. Finally, Section 7 concludes the paper.

2. Estimation of number of independent sources

Activities of the brain sources are mixed and their spatio-temporal combinations appear over the scalp as the EEG signals. Let \( \mathbf{x}(t) = [x_1(t), \ldots, x_m(t)]^T \) be the EEG recorded from \( m \) electrodes where \( ^T \) represents transpose operation. Thus, each \( x_k(t) \) is a vector of size \( m \times 1 \) for \( 0 \leq t \leq T-1 \) (where \( T \) is the number of time points).

The procedure of detecting the number of independent sources [9] of EEG signals is briefly described in the following steps:

Step 1. Calculate the covariance matrix \( \mathbf{C} \) from the measured data matrix \( \mathbf{X} \).

Step 2. Use Singular value decomposition (SVD) to decompose the covariance \( \mathbf{C} \), and calculate all the eigenvalues considering that \( \lambda_1 > \ldots > \lambda_m \).

Step 3. Determine the Information Criterion \( (IC_k) \) value from the eigenvalues of the covariance matrix \( \mathbf{C} \) as below [9]:

\[
IC_k = - \left( T - 1 - k - \frac{2(m-k)}{6(m-k)} + \frac{m-k+2}{6(m-k)} + \sum_{i=1}^{k} \frac{\lambda_i}{\lambda_{m-k}} \right) \cdot \log \left( \frac{\sum_{i=1}^{m-k} \lambda_i}{\sum_{i=k+1}^{m} \lambda_i} \right) - 2d(k,m)C_f(T-1)
\] (1)

where \( d(k,m) = (m-k)(m-k+2)(m-k+3)/2 \), \( k \) is number of assumed dipole sources and \( \lambda_{m-k} \) is the average of \( m-k \) smallest eigenvalues. \( C_f(h|\mathbf{h}=T \text{ or } \mathbf{h}=T-1) \) is a penalty function that can have one of the form:

- \( C_1 = 2 \)
- \( C_2 = 2\log(\log(h)) \)
- \( C_3 = \log(h) \)
- \( C_4 = 2\log(h) \)
- \( C_5 = 3\log(h) \)

For each \( k \) from 1 to \( k_{max} \) (\( k_{max} \) is the maximum number of sources), the related Information Criterion must be calculated \( (IC_k) \).

Step 4. The number of sources that has minimum \( IC_k \) is selected as the estimated number of sources.

3. Second order blind identification

Second order blind identification (SOBI) is able to estimate the mixing matrix as long as the unknown source signals are assumed temporally uncorrelated to each other. But, they have non-zero time-delayed autocorrelations. This is a plausible assumption for the case of EEG and electrocugram (EOG) sources [10].

\[
\mathbf{x}(t) = \mathbf{A} \mathbf{s}(t)
\] (2)

where \( \mathbf{s}(t) \) contains the source signals and \( \mathbf{A} \) is the mixing matrix. The SOBI uses only \( \mathbf{x}(t) \) to produce an unmixing matrix \( \mathbf{W} \) which approximates \( \mathbf{A}^{-1} \), subject to scaling and permutation of the separated sources. Therefore, the estimated sources can be obtained by using

\[
\hat{\mathbf{s}}(t) = \mathbf{W} \mathbf{x}(t)
\] (3)

and \( \mathbf{W} \) is expected to be \( \mathbf{W} = \mathbf{P} \mathbf{D} \mathbf{A}^{-1} \), where \( \mathbf{P} \) and \( \mathbf{D} \) are respectively the permutation and the scaling matrices. The SOBI computes the mixing matrix as the matrix that jointly diagonalizes a set of \( p \) cross correlation matrices \( R(t_i) = \mathbb{E} \{ \mathbf{x}(t) \mathbf{x}(t-t_i)^T \} \), where \( i = 1, \ldots, p \), and \( \mathbb{E} \{ \cdot \} \) is the expectation operator. In this paper we use \( p = [L_0/3] \) where \( L_0 \) is the number of EEG data samples in the analyzed data frame and \( [\cdot] \) denotes the floor operator.

4. Clustering the rows of unmixing matrices

In this section first RPCL [1] clustering method and a cost function-based approach for the RPCL [22] is briefly explained, then, a novel technique is introduced for initialization of the cluster centers. The algorithm is then followed by two conditions proposed for splitting and merging and a restorative technique based on neighborhood information. Finally, the hybrid system will be examined.

4.1. Rival penalized competitive learning (RPCL)

\( k \)-Means [21] is one of the most common optimization methods for clustering algorithm. One approach to \( k \)-means is based on rewarding to winner clusters. In each step, the selected instance rewards to the nearest cluster as the winner. In addition to rewarding the winner, the rival (the 2nd winner) is penalized rewarding to winner clusters. In each step, the selected instance is rewarded based on:

\[
\mathbf{w}_r = \mathbf{w}_d^r + \Delta \mathbf{w}_r^p \quad \rho = 1, 2, \ldots, k
\] (4)

where \( k \) is the number of clusters. \( \mathbf{w}_r \) and \( \mathbf{w}_d \) are new and old centers of the \( \rho \)th cluster respectively. Thus, \( \Delta \mathbf{w}_r^p \) is the location change of \( \rho \)th cluster center.

\[
\Delta \mathbf{w}_r^p = \begin{cases} \alpha_r (\mathbf{x}_r^p - \mathbf{w}_d^\rho) & \text{if } \rho = c(n) \\ -\alpha_r (\mathbf{x}_r^p - \mathbf{w}_d^\rho) & \text{if } \rho = r(n) \\ 0 & \text{otherwise} \end{cases}
\] (5)

where \( c(n) \) and \( r(n) \) denote the clusters associated with the winning and rival instance respectively and \( \alpha_r \) is the rewarding parameter.
where $\mathbf{x}^n$ is the instance which can be rewarded and punished, $c(n)$ is the index of the cluster with nearest center to $\mathbf{x}^n$, $r(n)$ is the index of the cluster with next nearest center (the rival) to $\mathbf{x}^n$, and $\alpha_r$ and $\alpha_c$ are the coefficients of reward and penalty terms respectively. The rival penalized mechanism tries to push the rival far away from the cluster towards which the winner is moving.

The cost function-based approach for the RPCL [22] tries to minimize the following functions:

$$E(\mathbf{W}) = E_1(\mathbf{W}) + E_2(\mathbf{W})$$

where

$$E_1(\mathbf{W}) = E_{MSE}(\mathbf{W}) = \frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} f(\mathbf{x}^n, \mathbf{w}_k^{(n)})^2$$

$$E_2(\mathbf{W}) = \frac{2}{p} \sum_{n=1}^{N} \sum_{k=1}^{K} f(\mathbf{x}^n, \mathbf{w}_k^{(n)})^{-p}, \text{ where } k \neq c(n) \in \{1, 2, \ldots, K\}$$

and

$$f(\mathbf{x}^n, \mathbf{w}_m) = (\|\mathbf{x}^n - \mathbf{w}_m\|^2)^{\frac{1}{2}}$$

where $N$ and $K$ are the numbers of instances of the dataset and desired clusters respectively. Parameter $d$ is the number of features (dimension) and $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_K]$ is the set of cluster centers. Moreover, if $\mathbf{x}^n$ is the $n$th instance, $\mathbf{x}_i^n$ is the $i$th feature of the $n$th instance, $\mathbf{w}_m^{(n)}$ represents the nearest cluster center to $\mathbf{x}^n$. Indeed, $c(n)$ is the index of that cluster with nearest center to $\mathbf{x}_i^n$. Finally, $f(a,b)$ is a distance function between $a$ and $b$. Here, Euclidean distance is used as $f$. Using this information, the cluster distances are maximized and within-class distances are minimized. $E(\mathbf{W})$ is therefore, the total error where $E_1(\mathbf{W})$ is the mean-square error (MSE) $E_{MSE}(\mathbf{W})$ introduced in (7) and $E_2(\mathbf{W})$ in [8] approximates the dissimilarities between the clusters. If the similarity between instances in each cluster increases, $E_1(\mathbf{W})$ decreases and when the clusters become more distinguishable $E_2(\mathbf{W})$ decreases. So the clusters become more distinguishable, when we have a large distance between an instance and other cluster centers, therefore $E_2(\mathbf{W})$ decreases. $\alpha_r$ in (5) is related to $\alpha_c$ as

$$\alpha_r = \alpha_c f(\mathbf{x}_i^n, \mathbf{w}_m^{(n)})^{-2}$$

where $P$ is a constant set to 0.2 in this paper. The parameter $\alpha_c$ is set to 0.1 and decreases in each iteration to ensure the convergence. One of the most important features of RPCL is its ability to push the centers of the extra clusters out of the data feature space. Our experiments show that $\alpha_r$ must be much bigger than $\alpha_c$ to avoid throwing exact clusters out of the scope. For this purpose, $f(\mathbf{x}_i^n, \mathbf{w}_m^{(n)})$ must be always bigger than one. Although this condition is not controllable in order to increase its chance in this paper, each data set is scaled such that the minimum distance between the instances is one. In this way, the repeated instances must be removed earlier.

Distance Sensitive RPCL (DSRPCL) is a Distance Sensitive version of the RPCL. In each step of the DSRPCL if all of the rival clusters are penalized, the method is named DSRPCL1 but if only the nearest one (next nearer after winner) is penalized, it is known as DSRPCL2. In the following sections two new techniques for initialization and clustering are introduced, which are the main contributions of this paper.

4.2. Initialization of cluster centers

Unlike in most of clustering algorithms where the initial cluster centers are selected randomly, here, a criterion for initialization is provided. In some places this highly affects the results. For example in RPCL, if the density of cluster centers is selected with a large variance, some of the cluster centers may fall out of the space. Also an unsuitable selection of the initial points may guide the method towards a local (rather than global) solution. Up to now, some methods have been proposed to initialize the cluster centers such as variable division and leader algorithm [2]. Our experiments show that the more distance between the initial centers, the more chance to find the best final clusters. To comply with this requirement, the method proposed by Mico [3] is implemented.

From now on, $\mathbf{W}^0$ represents the $\rho$th initial point as the cluster center. This method chooses these points from the instances of sample data. At first, $\mathbf{W}^0$ is chosen randomly from the instances and then other points are selected by

$$\mathbf{w}^\rho = \arg \max_{\mathbf{x} \in \mathcal{M}} \left( \sum_{j=1}^{\rho-1} f(\mathbf{x}, \mathbf{w}_j) \right), \quad \rho = 2, 3, \ldots, K$$

where $\mathcal{M}$ is the set of instances of sample data, $\beta_\rho = [\mathbf{w}^1, \mathbf{w}^2, \ldots, \mathbf{w}^{\rho-1}]$ is the set of chosen instances and $K$ is the number of initial clusters. Moreover, in each step, one unselected instance with the largest summation of distances from selected ones is chosen as the next point. But this method seems to be inefficient if $K > 2d$ for a $d$-dimensional feature space. As a simple example, in one-dimensional space depicted in Fig. 1, after selecting A and B as two instances with the largest distance between them, these two points have a similar distance summation. Then, the question is what point is the best.

Therefore, Eq. (10) is changed to:

$$\mathbf{w}^\rho = \arg \max_{\mathbf{x} \in \mathcal{M}} \left( \sum_{j=1}^{\rho-1} \sqrt{f(\mathbf{x}, \mathbf{w}_j)} \right), \quad \rho = 2, 3, \ldots, K$$

With this approach, in Fig. 1, the point with the most similar distance from A and B is chosen as the next point. Also the first point is chosen by the following equations instead of being selected randomly.

$$\mathbf{w}^1 = \arg \max_{\mathbf{x} \in \mathcal{M}} (f(\mathbf{x}, \mathbf{s})) \quad \mathbf{s} = (s_1, s_2, \ldots, s_d)$$

$$s_i = \min_{n=1,2,\ldots,N} (x_i^n) \quad i = 1, 2, \ldots, d$$

At first, a point on the boundary of the dataset or even a corner can be a good candidate to be chosen as $\mathbf{W}^1$. $\mathbf{s}$ is a point at one of the corners of the smallest hyper rectangle which surrounds all of instances of the sample data and $\mathbf{W}^1$ is the farthest instance from $\mathbf{s}$. Also, $s_i$ is the value of $i$th feature of $\mathbf{s}$ and $N$ is the number of feature vectors. The space has $d$ dimensions and $\mathbf{x}_i^n$ identifies the value of $i$th feature of the $n$th feature vector (instance). Eq. (13) identifies a corner which none of its feature values is greater than the same feature of other corners. Then, Eq. (13) can be changed to introduce another corner. To justify this method for selecting $\mathbf{W}^1$, it is considered that $\mathbf{s}$ and $\mathbf{W}^1$ are approximately the heads of one of the corners of the surrounding hyper rectangle. $\mathbf{s}$ is exactly located at the corner but there is often no instance at that point. Unlike $\mathbf{s}$, $\mathbf{W}^1$ is one instance of the sample data, which may not be exactly located at a corner.
4.3. Splitting and merging-based approach

Some clustering methods are not based on merging or splitting but use this technique to improve their performance. The main motivation of iterative self-organizing data analysis (ISODATA) [4] is to improve the k-means method through a split-merge approach. However, ISODATA clustering method uses special conditions for splitting and merging that are neither disjoint nor complement to each other and also they do not consider some properties of the clusters such as their size and distribution. In order to overcome these problems, a method based on merging and splitting that uses new criterion and conditions is proposed here.

4.3.1. Merging condition

Merging is a technique used in many clustering methods to prevent generation of extra clusters. Therefore, many criteria have been developed to check the condition of merging. One of the most widely used one is the distance between the two cluster centers [as used in ISODATA [4]]. In this case, if the cluster centers are nearer than a specified threshold, the main condition for merging is satisfied. This criterion does not consider other properties of the clusters such as their size and distributions.

Here, a novel criterion is introduced which is based on the expected internal distances of a cluster:

\[ D_{\text{MSR}}^A = \text{Mean}(f(x^a, s^n)) \]  
(14)

where \( D_{\text{MSR}}^A \) is the DMSR of cluster A and \( s^n \) is the nearest instance to \( x^a \) such that both of them belong to the same cluster.

The novel criterion in (14) is the average of Solitude Radiiuses DMSR that is the Mean of Solitude Radiiuses of instances belonged to the specified cluster. Solitude Radius of an instance is its distance from the nearest cluster-mate. Hence, each instance in a cluster is expected to be at DMSR from its nearest cluster-mate. Also, a threshold is needed here; therefore, another criterion is proposed which is based on the external distances of the two clusters. \( D_{\text{min}} \) of two specified clusters A and B is the minimum distance between them, instance to instance, and presented as

\[ D_{\text{min}}(A, B) = \min_{x^a \in A, x^b \in B} (f(x^a, x^b)) \]  
(15)

where

\[ D_{\text{min}}(A, B) \leq \beta \times \max(D_{\text{MSR}}^A, D_{\text{MSR}}^B) \]  
(16)

With these definitions, the merging condition is proposed by (16). Indeed, if two instances from two clusters are closer than a specified threshold, their clusters are suitable to be merged. This threshold is \( \beta \) times greater than the maximum DMSR of these clusters. The parameter \( \beta \) is a constant and considered 1.5 in this paper.

4.3.2. Splitting condition

Splitting is a technique to break a large cluster to a number of smaller ones that in some cases can improve the categorization performance. ISODATA method detects large clusters by checking the variance of their instances. If the variance is more than a specified threshold, that cluster is considered large enough to be divided into two smaller ones. In ISODATA, the conditions for merging and splitting are neither disjoint nor complement, also, they are not even related to each other. The centers of two clusters may be so close to each other that the condition of merging is satisfied but the variance of instances belonged to the produced cluster is greater than the determined threshold such that it can be split. Even two clusters merged previously may be produced again. To ensure that the conditions proposed in this method are complementary, the splitting condition is tried to be inverse of the merging condition. Although the proposed methods here are not completely complementary because of the complexity in the inversion operation, but the experiments show that this approach is very effective to eliminate the steps in merging and splitting and consequently it increases the chance of convergence.

Suppose \( V = \{v_1, v_2, \ldots, v_d\} \) is a set of vectors in d-dimensional feature space such that each \( v_i \) and \( v_j \) are perpendicular where i is not equal to j and size of each vector approximates distribution of sample data in the associated direction. Principle component analysis (PCA) [5] method produces \( V \) by computing the eigenvectors of covariance matrix of instances. To check the splitting condition of a cluster, \( V \) is computed for its corresponding instances. Then, the instances are projected onto each vector separately. After projection, the cluster is considered in d independent directions.

Splitting condition is satisfied for a specified cluster if one of its projections is qualified for splitting. Therefore, the problem is reduced to a one-dimensional problem. In this case, each instance has only one value associated to its unique feature. Then the instances are sorted based on this feature value. Each one-dimensional cluster is qualified to be split if the distance between the two consequent instances is more than a threshold which is \( \beta \) times greater than \( D_{\text{MSR}} \) of the original cluster. \( \beta \) is the constant used in (16). Since the boundary instances are usually far from the cluster centers, only the middle 1/3 of instances are considered for splitting. If a cluster is eligible to be split, its center is removed and the two new centers on both sides of it in the splitting direction are produced. Although this dimension reduction may disturb the inversion process, this approach is very effective in mitigation of the merging-splitting cycles.

In comparison with ISODATA, the proposed criterion and conditions are approximately complementary. In addition, ISODATA parameters are set for all the clusters but our techniques act on each cluster or pair of clusters independently except that \( \beta \) is global. Also the number of parameters which should be set in the proposed technique is considerably less than the number of parameters in ISODATA.

4.4. k-Nearest neighbors restoration

Regardless of which clustering method has been used as the core, according to the complexity of dataset, some methods may result in errors in manipulation of the boundary points. In this paper, a new technique is used to overcome this problem.

A well-known classification method named as k-Nearest Neighbors (KNN) [23] has been extensively used as a powerful non-parametric technique for pattern classification. KNN can classify each unknown instance by investigation in the k nearest instances of the training data. In this method, the goal is to assign a label to a new input data, instance by instance, according to the majority of votes of its K neighbors. This section goes on with a proposed clustering technique inspired by the KNN.

After reaching the stopping condition of the main method, all the instances belonged to a cluster are labeled the same. Finally a post-processing technique based on the KNN method is suggested to avoid producing small extra clusters. This process however, has more benefits as an auxiliary technique in any clustering method. This algorithm is designed sequentially for consecutive instances. In each step, one instance of the sample data is chosen as a new sample in KNN method and other ones assumed to be the training data with their associated labels. Hence, KNN overwrites the label of the selected instance and this process goes on for all the sample data. This post-processing is performed on the instances iteratively until no change in labels is occurred.
This technique is capable of collecting the small clusters surrounding a larger one and even joining them. It can also be useful from other perspectives. For example, in k-means method, each instance belongs to a cluster with the nearest center regardless of their size and distributions. Thus, a small cluster may be known as the container of some boundary instances of a larger cluster since the distance from the centers is considered only. The KNN technique can overcome this shortcoming up to some extent. It also can recognize the clusters with unusual shapes which are not describable by a few parameters in any clustering method. This technique has been proposed previously \([6,7]\) for \(k\) equal to 1. In the main KNN method, all the \(k\) nearest neighbors have the same weights in voting but in this work, these weights are inversely proportional to their distances from the new instance.

This technique runs iteratively until no change takes place any longer. Sometimes the termination condition never meets, therefore, the maximum number of iterations should be limited to stop the algorithm.

### 4.5. A combinatorial clustering system

Here, a clustering system that uses techniques described in this paper has been explained. This system uses DSRPCL2 as the main method although a simpler method such as k-means can also be used. Since DSRPCL2 is used, sample data is scaled such that the minimum distance of instances is 1. Then, some points from the instances are chosen as the initial cluster centers by the technique proposed in Section 3. Afterwards, iteratively, DSRPCL2, KNN restoration and merging are applied to all samples. If no merging is occurred, the splitting condition is checked. In the final iteration, no merging or splitting condition is satisfied and consequently, the system result converges unless the number of iterations exceeds a specified limit. The flow chart of the proposed algorithm is depicted in Fig. 2.

### 5. Seizure localization

In this section the SOBI algorithm \([10]\) is combined with the proposed clustering technique in order to separate the EEG signals and select the desired source (i.e. focal epileptic seizure source). The extended clustering algorithm is used to cluster the rows of the unmixing matrices calculated by the SOBI algorithm. On the other hand, the seizure source is identified in places where the STLmax has its minimum value often after a down trend in the measurement of Lyaponov exponents.

Therefore, in order to determine the dynamical changes in the brain source signals, STLmax value is employed to quantify the nonlinear dynamic characteristics of the source signals \([11]\). Moreover, the number of independent brain sources is estimated by the method described in Section 2 before applying the SOBI algorithm. The main seizure source signal has the lowest STLmax compared to others source signals. During onset of seizure (the ictal interval), EEG signals behave like a burst signals, which is more rhythmic (in time domain) than during the pre and post-ictal intervals and also include the regular harmonics in the frequency domain. When the regularity of a signal is increased, its complexity is decreased. Therefore, the signal with the lowest complexity can be considered the main seizure source signal. In the next stage, only the main seizure source signal is projected back to the electrode positions. Since the other source signals do not have a low value of STLmax and just contain the background EEG information or noise and artifact, therefore, they are removed. The projected electrode signals are used to localize the place of the seizure source in the brain. Projecting any desired estimated source onto the electrodes’ space can be performed by the following steps:

#### Step (1).
Keep the desired source and set the other sources to zero and form a new matrix named \(Y\).

#### Step (2).
Multiply inverse of the unmixing matrix by \(Y\) \(\left( Z = W^{-1} \times Y \right)\). \(Z\) is the projection of the desired source onto the space of electrodes.

Here, after clustering the rows of the unmixing matrices in several time frames, we can form a new unmixing matrix named \(U\) with each row as a cluster center. This new unmixing matrix is an \(n \times m\) matrix \(n\) is the number of final estimated sources and \(m\) is the number of sensors) estimated for all segments of the recorded data.

Multiplying \(U\) by the electrode signals leads to estimation of the estimated sources. Therefore, each row of \(U\) is used in order to construct one estimated source. Obviously, \(U\) is not square (as far as the number of EEG recording electrodes is reasonably high it is expected to have a fewer number of sources than sensors), hence, we cannot multiply its inverse by \(Y\) according to Step 2 above. To solve this problem instead of the steps above, the following stages are performed in order to project back the seizure source signal to the electrodes:

Stage (1) Compute the unmixing matrix called \(Q\) over the whole specified number of time frames where the number of sources is set equal to the number of electrodes. \(Q\) is an \(m \times m\) matrix.

Stage (2) For the seizure source, we compare its corresponding row in matrix \(U\) to the rows of the estimated unmixing matrix from Stage 1. This can be carried out by computing the

---

**Fig. 2.** Block diagram of the combinatorial clustering system used in this paper.
correlation coefficient between the equivalent row of the seizure source in matrix \( \mathbf{U} \) and each rows of \( \mathbf{Q} \) according to the following equation:

\[
\rho = \frac{\sum_m (a_m - \bar{a})(b_m - \bar{b})}{\sqrt{\left(\sum_m (a_m - \bar{a})^2\right)\left(\sum_m (b_m - \bar{b})^2\right)}}
\]

where \( a_m \) is the \( m \)th entry of the vector \( \mathbf{a} \) and \( \mathbf{b} \) is a row vector of \( \mathbf{U} \). After multiplying \( \mathbf{U} \) by the electrode signals all of the sources including the seizure source is generated. \( \bar{a} \) is the mean of \( \mathbf{a} \) entries. Similarly, \( \mathbf{b} \) is a row vector of \( \mathbf{Q} \) and \( \bar{b} \) is the mean of \( \mathbf{b} \) entries.

Stage (3) Find the row index of \( \mathbf{Q} \) which generates the maximum \( \rho \). Keep the seizure source and set the rest of the outputs to zero.

Stage (4) Compute \( \mathbf{Z} = \mathbf{Q}^{-1} \mathbf{Y} \), where \( \mathbf{Z} \) is the projection of seizure source onto the electrodes’ space.

The projected seizure source, \( \mathbf{Z} \), can be represented by a single dipole localization to find its location. In the dipole fitting method, the EEG source is modelled as the current dipole, and then, a least squares (LS) fit to the data is performed. Here we have used the DIPFIT method [19], and the source location is estimated within a four-shell spherical model of the head [18].

6. Simulation and experimental results

To show the effectiveness of our proposed method, this approach is applied to the real and simulated (synthetic) data. The real data contains one subject with focal seizure. In simulating the seizure signals, some regular harmonics are considered as the main source and some electrode signals are produced according to their distances from the source location. What follows, is the results of application of our approach to both data sets.

6.1. Simulated data

In this section our novel approach is applied to two synthetic data sets in order to calculate the mean square error (MSE) between the localized and real source positions. The names and the positions of the electrodes used in the real data are shown in Fig. 6. For the simulated data, four electrodes (FP1, FP2, F7, F8) which are located on frontal lobe of the head are selected. Approximate positions of the selected electrodes are provided in the Table 1. Each electrode’s position is denoted by 3 coordinates values \([x \ y \ z]\). These electrodes record mixture of the sources inside the head. To simulate the EEG signals for different electrodes for a patient with focal seizure, the main assumption is that the electrode signals are generated from two main sources. One source is noise while the other source is a sinusoid with three different harmonics. The sinusoid is generated via the following equation:

\[
s(t) = \sin \left(\frac{\pi t}{2}\right) + \sin \left(\frac{\pi t}{4}\right) + \sin \left(\frac{3\pi t}{2}\right)
\]

The two sources are placed in different positions in the brain. The electrode signals are generated considering the simulated source signals and their distances from the electrodes. The mixing matrix is inversely proportional to (square of) the distances from each source to all the electrodes [24]. After constructing the mixing matrix, it is possible to generate the simulated EEG signals.

Our initial data set consisted of 40 simulated EEG signals. For 20 of them, the sinusoidal source was placed close to FP1 and F7 electrodes and for the other 20 signals, the sinusoidal source was placed close to FP2 and F8 electrodes. After source extraction using our first hybrid method (hybrid of the extended clustering algorithm and SOBI), the method of dipole localization is performed in order to find the location of the extracted source. For comparison, the hybrid method of source selection is also performed using a second hybrid method (hybrid of \( k \)-means clustering algorithm and SOBI). However in some cases the MSE difference for the above two kinds of hybrid methods was not significant and in average, our proposed hybrid method significantly outperforms the second hybrid method.

Because of uncertainty in the noise level in some cases the results of localization of the sources and equivalent MSE are shown in Figs. 4 and 5. The results of localization using both hybrid systems are illustrated in Table 4.

Table 1

<table>
<thead>
<tr>
<th>Electrode name</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP1</td>
<td>[-24.41 92.26 11.31]</td>
</tr>
<tr>
<td>FP2</td>
<td>[29.49 89.46 12.37]</td>
</tr>
<tr>
<td>F7</td>
<td>[-66.74 52.49 3.348]</td>
</tr>
<tr>
<td>F8</td>
<td>[66.61 49.57 0.6124]</td>
</tr>
</tbody>
</table>

Table 2

Sources and their corresponding positions inside the head (First simulated data),

<table>
<thead>
<tr>
<th>Source</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sinusoidal</td>
<td>[-40 70 14]</td>
</tr>
<tr>
<td>Noise</td>
<td>[10 40 15]</td>
</tr>
</tbody>
</table>

Table 3

Sources and their corresponding positions inside the head (Second simulated data),

<table>
<thead>
<tr>
<th>Source</th>
<th>Position</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sinusoidal</td>
<td>[70 30 5]</td>
</tr>
<tr>
<td>Noise</td>
<td>[-10 40 15]</td>
</tr>
</tbody>
</table>
Fig. 4. Localized source and its MSE for simulated data 1 using the first hybrid method.

<table>
<thead>
<tr>
<th>Simulated source 1</th>
<th>Simulated source 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated location</td>
<td>MSE</td>
</tr>
<tr>
<td></td>
<td>-36.99  71.09  9.498</td>
</tr>
</tbody>
</table>

Fig. 5. Localized source and its MSE for simulated data 2 using the first hybrid method.

<table>
<thead>
<tr>
<th>Simulated source 1</th>
<th>Simulated source 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimated location</td>
<td>MSE</td>
</tr>
<tr>
<td></td>
<td>62.48  43.23  -0.2723</td>
</tr>
</tbody>
</table>

Table 4
The Localization error of the sinusoidal sources when applying the two hybrid methods.

<table>
<thead>
<tr>
<th>Method</th>
<th>Simulated source 1</th>
<th>Simulated source 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOBI + the proposed clustering method</td>
<td>Estimated location</td>
<td>MSE</td>
</tr>
<tr>
<td>SOBI + k-means</td>
<td>Estimated location</td>
<td>MSE</td>
</tr>
</tbody>
</table>

Fig. 6. Channel locations by number (left) and by name (right).
6.2. Real data

In this part scalp EEG signals of three patients with focal seizure are examined. The data is recorded from the Department of Neurophysiology, King’s College Hospital, London, UK. Each EEG signal recordings has 2 min long (with sample frequency of 200 Hz) from 16 channels which are sorted in the following order. FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6. Channel locations by number and name are shown in Fig. 6.

The STLmax, as a widely used measure of chaos for time series [11], is employed here to estimate the seizure onset from the signals. The duration of the data segment to estimate the STLmax values was 10 s. Thus, STLmax measurement has been performed for the different channels of scalp signals. The other parameters here, have the same values used for estimating the STLmax in [11].

The signal lengths are limited, therefore, to have a better estimation of STLmax value, the signals are segmented into the segments of 12 s duration with 80% overlap.

The estimated STLmax values of channel 2 is nearly the same as of channel 1 except that the minimum value of STLmax is in slightly before 60 s. Therefore, the start time of seizure onset is considered approximately at 60th second [17]. The scalp signals from 40 s to 80 s are divided into 8 segments with 10% overlap to solve the permutation problem of BSS.

The number of sources estimated by the method described in Section 2 for eight successive time frames of the EEG signals are 8, 8, 8, 10, 10, 12, 13, 13.

As expected, the number of sources varies from one frame to another. The SOBI with the given number of sources in each time frame is applied to the eight time frames. Then, the new auxiliary clustering method is applied to the rows of the unmixing matrices of all the time frames. After clustering, each cluster center is used to extract a source signal by multiplying that by the EEG signals to find the corresponding estimated source. Our main interest is to find the location of the seizure source, and it is clear that for a focal seizure, the projection of seizure position on the scalp remains in the same place and cluster center of seizure source can give us the main seizure source signal. The resulted sources are shown in Fig. 7. STLmax values have been measured from the second signal of Fig. 7 (from top). The STLmax values show that a drop is appeared for the second signal around 14th second. Therefore, we can consider this signal as the main seizure source.

The results of applying the four stages above on seizure source are shown in Fig. 8. From this figure it can be seen that seizure source is propagated to FP1, FP2, F3, F4, F7, F8 electrodes more than the other electrodes. The spherical 4 shell head model [18] is used to locate the dipole of the projected seizure source. For the projected seizure source, one dipole is found as the main source.
and it is what we expect from a focal seizure. Dipole plot of seizure source is shown in Fig. 9.

7. Discussion and conclusion

Seizure source location plays an important role in surgically removing the epileptic zone. In this paper, a new two-stage method for localizing the seizure signal sources from the scalp EEGs is proposed. The main goal of this paper is localization of the seizure source. The main contribution here however is developing a reliable clustering algorithm which is essential for better estimation of the seizure source. In this application, an accurate source localization directly depends on the accuracy of the seizure source estimation. Therefore, the extended clustering method used in this paper is effective in order to find the most relevant brain sources especially the seizure source non-invasively. The extended clustering algorithm was validated based on standard data sets such as Iris, wine and glass and in some cases the performance of the proposed extended clustering algorithm was better than the DSRPCL algorithm. The results of localization on the simulated EEG signals using two hybrid methods suggest that our extended clustering algorithm is effective. However the results of the
localization for the selected simulated data does not significantly differ. The main objective here is to have a reliable and effective clustering algorithm, which automatically determines the best number of clusters through merging and splitting approach. This is very important for a complex system. Therefore, the hybrid method of seizure source estimation along with a dipole localization method to find the seizure foci seems to be very effective. However it is needed to apply the method of localization to a larger number of scalp signals and evaluate its simultaneously recorded intracranial signals in order to completely validate the proposed method. This is achievable for the cases where both the intracranial and scalp EEGs are recorded simultaneously.

References