



Interfaz Gráfica Intuitiva para el Procesamiento de Señales EEG

A. Santillán Guzmán^{1,*}, H. Z. Ramírez Uriarte², J. J. Oliveros Oliveros¹, M. M. Morín Castillo³, H. Ramírez Díaz²

¹ Facultad de Ciencias Físico-Matemáticas, Benemérita Universidad Autónoma de Puebla, Puebla, México,

² Ingeniería en Sistemas, Instituto Tecnológico Superior de Eldorado, Culiacán, Sinaloa, México,

³ Facultad de Ciencias de la Electrónica, Benemérita Universidad Autónoma de Puebla, Puebla, México

*alina_santillan@yahoo.com.mx

Resumen— El presente artículo trata sobre el desarrollo de una nueva interfaz gráfica para el procesamiento de señales electroencefalográficas (EEG) por medio de técnicas simples como filtros digitales (pasa-bajas, pasa-altas, pasa-banda y banda-eliminada), técnicas de descomposición de señales (Análisis de Componentes Independientes, ICA por sus siglas en inglés) y la combinación de ICA con filtros pasa-bajas. Una de las ventajas de esta herramienta es que los comandos, así como las etiquetas y ayuda están en español. Asimismo, cuenta con un ambiente amigable e intuitivo para el usuario y es de fácil uso, permitiendo su manipulación por personas poco familiarizadas con MATLAB, plataforma donde se implementó la interfaz.

Palabras clave— EEG, Filtros, Interfaz Gráfica

I. INTRODUCCIÓN

La actividad cerebral se puede registrar a través de una técnica conocida como Electroencefalografía (EEG). Las señales registradas corresponden a una mezcla de información cerebral, así como artefactos fisiológicos (movimientos oculares, musculares, latido cardiaco, sudor) o técnicos (línea de alimentación, desconexión de electrodos) [1]. Todos estos artefactos necesitan ser suprimidos porque pueden ocultar a la señal fisiológica a analizar. Los filtros digitales son técnicas de fácil uso e implementación, útiles para eliminar distintos tipos de artefactos cuando se conocen tanto las frecuencias a ser analizadas como las que se necesitan eliminar. Por otro lado, cuando los artefactos y las señales fisiológicas a analizar están en el mismo rango de frecuencias, imposibilitando el uso de filtros digitales, se puede recurrir a otras técnicas como el análisis de componentes independientes (ICA, Independent Component Analysis, en inglés) [2, 3]. Esta técnica es muy común en el procesamiento de señales EEG. ICA separa las señales de entrada en una serie de componentes; algunos de ellos contendrán la información fisiológica, y otros, los artefactos.

Actualmente, existen interfaces gráficas de usuario (GUIs, Graphical User Interfaces, en inglés) para poder procesar las señales EEG con alguna de las técnicas mencionadas anteriormente. Una de las GUIs más usadas mundialmente es *EEGLAB* [4]. Con esta interfaz, el usuario puede cargar datos, aplicar filtros de respuesta finita (siempre

estables), aplicar ICA para eliminar artefactos quitando los componentes independientes (ICs) que contengan artefactos, así como observar mapas topográficos cerebrales, entre otras cosas. A pesar de que *EEGLAB* es una herramienta útil debido a todo lo que ofrece, algunos problemas ocurren cuando el formato de los datos de entrada no es compatible con el software o no está contenido dentro del paquete de formatos que ofrece. Además, si el usuario está poco familiarizado con este tipo de interfaces, toma tiempo el empezar a trabajar correctamente con ella (es poco intuitiva).

Otra interfaz gráfica implementada en MATLAB es la llamada *fasticag*, la cual usa *fastICA* (uno de los algoritmos de ICA, que tiene una convergencia rápida) [5]. Con esta herramienta, se estiman y grafican los componentes independientes. Sin embargo, genera problemas al tratar de cargar una nueva serie de datos y se queda anclada, debiéndose cerrar y re-abrir.

Este trabajo presenta una nueva interfaz gráfica como herramienta alternativa para el procesamiento de las señales EEG. El nombre de ésta es “Filtrado de señales, Mapeo y Localización de Fuentes” (FIMALOF), puesto que además del filtrado, se pretende también tener un mapeo topográfico cerebral y, mediante modelos y algoritmos matemáticos, hacer una localización de fuentes. La ventaja principal de esta interfaz es el ambiente amigable e intuitivo para el usuario, incluso para aquél que está poco familiarizado en temas de procesamiento digital de señales o que tiene poca experiencia trabajando con MATLAB; además está disponible en español y en inglés. Asimismo, presenta la combinación de ICA-filtro pasa-bajas, con la cual es posible eliminar más eficientemente los artefactos musculares [6].

En las siguientes secciones, la interfaz gráfica propuesta se describe en detalle.

II. METODOLOGÍA

A. Filtros Digitales

Es bien sabido que un filtro digital es un conjunto de operaciones que se aplican a una señal de entrada para que atenúe cierta banda de frecuencias y perturbe lo menos posible la amplitud en el resto de las frecuencias [7]. En la presente interfaz, se han implementado cuatro filtros: 1) Filtro pasa-bajas que dejará pasar todas las frecuencias desde 0 hasta cierta frecuencia de corte (F_c), atenuando las demás frecuencias. 2) Filtros pasa-altas que elimina la amplitud de

las señales desde 0 hasta cierta frecuencia de corte. A partir de ésta existe una atenuación cero, dejando pasar, sin suprimir la amplitud, las señales de alta frecuencia. 3) Filtro pasa-banda, el cual dejará pasar cierta banda de frecuencias, desde una frecuencia de corte inferior, hasta una frecuencia de corte superior, atenuando el resto. 4) Filtros de banda-eliminada, también conocidos como filtros Notch, que eliminarán una frecuencia o un rango de frecuencias a partir de una frecuencia de corte inferior hasta una frecuencia de corte superior. En este caso, al igual que en los filtros pasa-bandas, la diferencia entre ambas frecuencias es el ancho de banda.

Cada uno de los filtros se implementó en MATLAB, usando cuatro tipos de aproximaciones: 1) Butterworth, la cual tiene una respuesta máximamente plana en la banda pasante y una caída monotónica en la banda eliminada. 2) Chebyshev I, el cual presenta rizados en la banda pasante (Ap) y una caída monotónica en la banda eliminada. La amplitud de los rizados está dada en dB y depende del usuario seleccionar su valor. Este tipo de filtro presenta un decaimiento más rápido que el Butterworth. Sin embargo, debido a los rizados existen atenuaciones en la banda pasante. 3) La aproximación Chebyshev II o Chebyshev inverso, presenta una amplitud plana en la banda pasante y rizados en la banda eliminada (As). El valor de As está dado en dB y lo define el usuario. La ventaja de este filtro es que no habrá atenuaciones en la banda pasante, y la desventaja es que habrá menos atenuaciones en la banda eliminada debido a los rizados que presenta. 4) La cuarta aproximación corresponde a la elíptica, la cual presenta rizados tanto en la banda pasante como en la banda eliminada. Usualmente, el valor de Ap es menor al valor de As. Su mayor ventaja es que se necesita un orden menor en comparación con otras aproximaciones. Sin embargo, debido a los rizados que contiene, habrá atenuaciones en la banda pasante y amplificaciones en la banda eliminada.

B. Análisis de Componentes Independientes

La técnica de análisis de componentes independientes, o ICA, hace una estimación de los componentes independientes que conforman a las señales registradas. El modelo para obtener los componentes independientes está dado por [3]:

$$\mathbf{y}(k) = \sum_{i=1}^m c_i s_i(k) = \mathbf{C}\mathbf{s}(k), \quad (1)$$

donde $\mathbf{y}(k) = [y_1(k), \dots, y_n(k)]^T$ corresponde a las señales registradas, k es el instante de tiempo; n es el número de electrodos EEG; $\mathbf{s}(k) = [s_1(k), \dots, s_m(k)]^T$, son los m componentes independientes; y $\mathbf{C} = [c_1, \dots, c_m]$ es una matriz de mezclas de tamaño $n \times m$.

De acuerdo al modelo, se tienen dos incógnitas (\mathbf{C} y \mathbf{s}). Para poder estimar el valor de los componentes independientes, se hacen tres suposiciones principales: *i*) los componentes independientes tienen distribuciones no Gaussianas (máximo un componente puede ser Gaussiano); *ii*) los componentes deben cumplir con la propiedad de

independencia; y *iii*) la matriz de mezclas (\mathbf{C}), debe ser lineal, cuadrada y estacionaria [3].

Usando el paquete FastICA [5] se hace la estimación de los componentes independientes, tomando en cuenta las suposiciones previamente mencionadas.

Al usar la función *fastica*, del paquete mencionado anteriormente, se pueden especificar ciertos parámetros para realizar la estimación de los componentes independientes. En la interfaz gráfica propuesta, el usuario puede ingresar dichos parámetros o dejar los valores que se proponen por defecto, según lo requiera.

III. RESULTADOS Y DISCUSIÓN

A. Interfaz Principal

La Interfaz Principal contiene las distintas opciones que el usuario puede elegir: Tipo de filtro, ICA, o la combinación de ICA con un filtro pasa bajas (LPF, Low-Pass Filter, en inglés) [6]. Dependiendo de la selección del filtro o técnica que indique el usuario, una nueva interfaz gráfica se abrirá.

Los datos deberán estar en formato *.mat* y con cierta estructura para poder utilizarlos en la interfaz. Es por ello, que previo al procesamiento de los datos, hay una conversión de formato y/o estructura para comenzar a trabajar.

Cabe mencionar que el idioma por defecto es español. Sin embargo, el usuario puede cambiar el idioma al inglés si lo considera necesario.

B. Interfaz de Filtros

Debido a restricciones de espacio, sólo se detallará el funcionamiento del filtro pasa-bajas. Sin embargo, la misma explicación aplica a los demás filtros, teniendo en cuenta que para el filtro pasa-banda y rechaza-banda se necesitan dos frecuencias de corte.

Al elegir la opción de cualquiera de los filtros, se abre una interfaz gráfica similar a la de Fig. 1. Los parámetros que ingresa el usuario son la frecuencia de muestreo (F_s), la frecuencia de corte (F_c), el orden del filtro (Orden), el tamaño pico-a-pico del rizo en la banda pasante (Ap), el tamaño pico-a-pico del rizo en la banda de supresión (As), el punto de inicio de las señales a procesar (Punto de inicio) y el número de segundos a analizar (Segundos). Una vez ingresados estos datos, el usuario deberá seleccionar los canales a los cuales desea aplicar el filtro. Dichos canales pueden formar parte de 4 posibles configuraciones que actualmente se están considerando en la interfaz (ver Fig. 1): EMOTIV Epoc+® (14 canales), sistema 10-20 (19 canales), sistema 10-20 modificado a 32 canales, y sistema a 128 canales, de los cuales el usuario podrá seleccionar el número de canales que requiera. Esta configuración es válida cuando el usuario no tiene conocimiento de la configuración en que las señales EEG están dadas. Según la configuración, se abrirá una interfaz como la que se muestra en Fig. 1, en donde se podrán

seleccionar el(los) canal(es) a filtrar. Una vez seleccionados, el siguiente paso consiste en cargar las señales EEG. Posteriormente, los canales seleccionados serán graficados en la parte superior derecha, y se hará una normalización a tener valor medio de cero y varianza unitaria. A continuación, se selecciona(n) la(s) aproximación(es) de filtro(s) a ser aplicada(s). La respuesta en frecuencia de cada selección se grafica en el color correspondiente a la aproximación, en la parte inferior izquierda. Se aplica(n) los filtro(s) seleccionado(s), y en la parte superior derecha se graficarán las señales filtradas. Además, es posible elegir un canal al cual se le calculará su espectro de potencia, que será graficado en la parte inferior derecha.

A manera de ilustración, se grafican 5 s de una señal EEG de un sujeto sano, correspondiente al canal FC6, cuya configuración seleccionada es la del EMOTIV, tal y como se observa en Fig. 1. Se eligen los filtros Butterworth y Chebyshev I, cuya frecuencia de corte es de 5 Hz, el orden es 4 y la amplitud del rizo en la banda pasante es de 3 dB. Se grafican sus correspondientes señales filtradas, respuesta en magnitud de cada filtro, así como se selecciona un canal para graficar su espectro de potencia. Al aplicar distintos tipos de aproximación, el usuario podrá determinar cuál de ellas otorga mejores resultados de filtrado de las señales EEG, dependiendo qué tipo de señal desee analizar posteriormente.

C. Interfaz de ICA

En Fig. 2 se muestra la interfaz de ICA, donde el usuario puede ingresar la frecuencia de muestreo, el punto de inicio, la cantidad de segundos, así como otros parámetros y variables a usar.

Dentro del panel “Parámetros Básicos”, una de las variables a utilizar es ‘Aproximación’, la cual puede ser ‘symm’, si se calculan los ICs en paralelo; o ‘defl’, si se calculan uno por uno. Este es el valor por defecto en la interfaz. También se puede determinar el número de ICs a calcular. Por defecto éste corresponde al número de señales EEG de entrada. En el panel de “No linealidades” se puede escoger la función no-lineal para calcular la matriz de mezclas. Dichas funciones son ‘pow3’, ‘tanh’, ‘gauss’, y ‘skew’. Para detalles sobre estas funciones, referirse a [8]. De igual forma, para tener un ‘Ajuste de precisión’ en la estimación de los componentes, se usan las mismas funciones o se hace una deshabilitación si no se necesita hacer un ajuste muy preciso. Como parte de las funciones no-lineales se tienen coeficientes (‘a1’, ‘a2’); y ‘mu’, que es el tamaño de paso para realizar el algoritmo. Otra variable a considerar para poder obtener de manera más efectiva los componentes independientes es ‘Estabilización’, con la cual se controla si el programa usa o no una versión estable. Todos los parámetros mencionados tienen un valor por defecto dentro de la interfaz FIMALOF. Estos valores por defecto son los que en la función *fastica* se proponen. Si el usuario necesita

cambiar los parámetros por defecto, dentro de la interfaz se presentan mensajes de ayuda con los rangos de valores que se pueden ingresar y el tipo de señales para las que sirven las distintas funciones no lineales.

Finalmente, también están los parámetros que controlan la convergencia, tales como: ‘Epsilon’, que es el criterio para parar el algoritmo; ‘Max. Núm. Iteraciones’, que tiene que ver con el número máximo de iteraciones para estimar cada componente independiente; ‘Max. Núm. Precisión’, que corresponde al número máximo de iteraciones para ajustar el algoritmo. Dentro de la misma interfaz, se pueden graficar los ICs, seleccionar aquellos que correspondan a artefactos y reconstruir las señales limpias.

D. Interfaz de ICA-LPF

La interfaz correspondiente a ICA-LPF es similar a la de ICA, detallada anteriormente, a la cual se le ha agregado un botón que permite seleccionar los componentes con artefactos musculares para aplicarles un filtro pasa-bajas. Esta combinación permite la eliminación de artefactos musculares en una forma más eficiente que usando solamente ICA [6].

ICA-LPF permite primero la aplicación de ICA para obtener los componentes de las señales EEG. Posteriormente se seleccionan aquellos componentes que contienen artefactos musculares y se les aplica un filtro pasa-bajas, cuya frecuencia de corte es 30 Hz y un orden de 6, para eliminarlos y evitar que haya mezcla entre señales fisiológicas deseadas y artefactos. De la nueva serie de componentes independientes previamente filtrados, se seleccionan aquellos que aún contengan artefactos oculares y/o musculares. Dichos componentes son eliminados para reconstruir las señales sin artefactos.

IV. CONCLUSION

En el presente trabajo se mostró una interfaz gráfica como herramienta alternativa para el procesamiento de señales EEG. Está implementada en MATLAB y cuenta con un ambiente amigable e intuitivo para el usuario ya que el nombre de los parámetros o variables a introducir indican exactamente lo que el usuario debe ingresar, sin necesidad de saber programar o saber MATLAB para navegar de un módulo a otro. Además, la interfaz se encuentra en idioma español, lo cual representa una ventaja respecto a las interfaces gráficas existentes. Como paso futuro, se pretende lograr que los parámetros correspondientes a ICA sean totalmente intuitivos, integrar la técnica de descomposición empírica de modos (EMD, Empirical Mode Decomposition, en inglés), la implementación del mapeo topográfico y localización y visualización de fuentes. Asimismo, se pretende tener una versión ejecutable para que médicos, clínicos y especialistas en el área biomédica puedan hacer uso de esta herramienta sin necesidad de tener MATLAB.

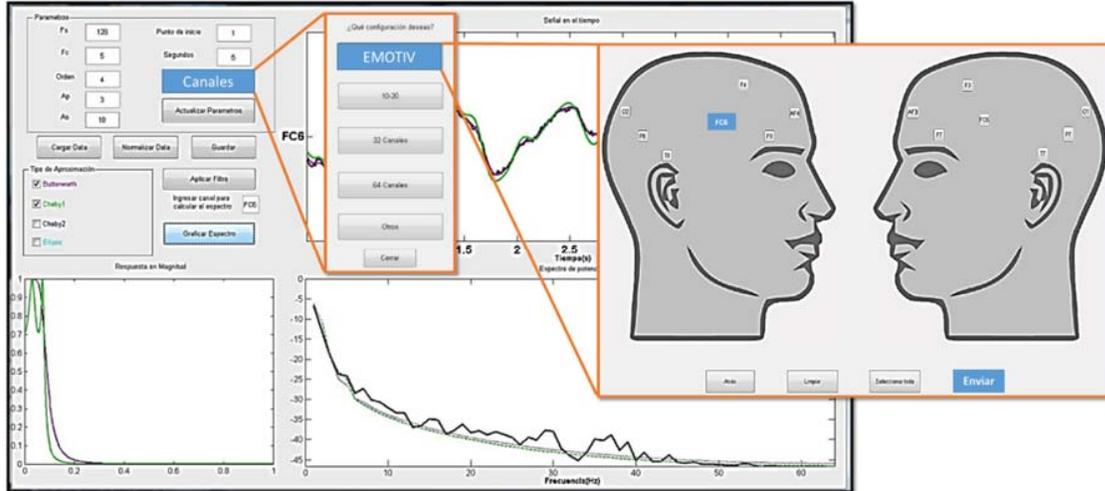


Fig. 1. Interfaz correspondiente al filtro pasa-bajas, mostrando la sub-interfaz con las configuraciones de los canales, así como la configuración del EMOTIV.



Fig. 2. Interfaz correspondiente a ICA.

RECONOCIMIENTOS

A. Santillán Guzmán agradece a PRODEP por el apoyo brindado a través del convenio 511-6/17-14391, complemento del documento DSA/103.5/16/14599 y al Proyecto VIEP 100408844VIEPBUP. H. Ramírez Díaz agradece por el apoyo brindado a través del Proyecto Número 293384 para el Laboratorio SEDEAM, registrado en el Directorio del Programa de Laboratorios Nacionales de CONACYT 2018.

BIBLIOGRAFÍA

- [1] S. Sanei and J.A. Chambers, "EEG signal processing", John Wiley & Sons: England, 2007.
- [2] Hyvärinen A, Oja E. Independent component analysis: Algorithms and applications. Neural Networks. 13:411-430, 2000. DOI: [https://doi.org/10.1016/S0893-6080\(00\)00026-5](https://doi.org/10.1016/S0893-6080(00)00026-5).
- [3] Vigário R, Särelä J, Jousmäki V, Hämäläinen M, Oja E. Independent component approach to the analysis of EEG and MEG recordings. IEEE Transactions on Biomedical Engineering. 47(5):589-593, May 2000. DOI: 10.1109/10.841330.
- [4] Delorme A, Makeig S. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. J Neurosci Methods. 134(1):9-21, Mar 2004. DOI: 10.1016/j.jneumeth.2003.10.009.
- [5] FastICA. FastICA MATLAB package, Disponible en: <https://research.ics.aalto.fi/ica/fastica/code/dlcode.shtml>
- [6] Santillán Guzmán A, Heute U, Stephani U, Galka A. Comparison of different methods to suppress muscle artifacts in EEG signals. Signal, Image, and Video Proc. Journal. 11:761-768, 2017. DOI: 10.1007/s11760-016-1020-4.
- [7] A. P. Malvino Principios de Electrónica, Mc Graw Hill, España, 2000.
- [8] Hyvärinen A. Fast and Robust Fixed-Point Algorithms for Independent Component Analysis, IEEE Trans. on Neural Networks. 10(3):626-634, 1999. DOI: 10.1109/72.761722.