



*ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN*

Master Ingeniería Biomédica

Trabajo Final de Master:

**ANÁLISIS DE LA FORMA DE ONDA DE
LOS POTENCIALES DE ACCIÓN DE
UNIDAD MOTORA**

Alumno: Miguel Fernández Martínez

Tutor: Armando Malanda Trigueros

Lista de Contenido

INTRODUCCIÓN.....	1
OBJETIVOS.....	11
MATERIAL.....	12
MÉTODOS.....	13
RESULTADOS	17
DISCUSIÓN.....	27
CONCLUSIONES	33
REFERENCIAS.....	34
ANEXOS	36

Lista de Figuras

Figura 1: La fibra muscular	2
Figura 2: El potencial de acción.....	3
Figura 3: Representación esquemática de una fibra inervada por una motoneurona.....	3
Figura 4: Ejemplo de distintas motoneuronas.	4
Figura 5: Representación esquemática de un PAUM.....	5
Figura 6: Reclutamiento de unidades motoras	5
Figura 7: Línea base y duración	6
Figura 8: Amplitud y área.	7
Figura 9: Turns y número de fases.	7
Figura 10: Duración espiga y su área.....	8
Figura 11: Los satélites s.....	8
Figura 12: PAUM Miopático	9
Figura 13: Gold Standard: Registro, sin anotar.	14
Figura 14: Gold Standard: Marcadores temporales.....	15
Figura 15: Gold Standard: Registro con los potenciales seleccionados	15
Figura 16: Gold Standard final, obtenido tras promediar los potenciales seleccionados.....	15
Figura 17: Boxplots de los errores relativos de los registros miopáticos	19
Figura 18: Boxplots de los errores relativos de los registros crónicos.....	20
Figura 19: Boxplots de los errores relativos de los registros subagudos	22
Figura 20: Boxplots de los errores relativos de los registros normales	23
Figura 21: Amplitud del Gold Standard.....	28
Figura 22: Importancia de los marcadores del Gold Standard.....	29
Figura 23: Variación de los errores de amplitud, área y spike_area a lo largo de los métodos y familias	29
Figura 24: Ejemplo Five Closest Averaging.....	30
Figura 25: Ejemplo método EA.....	30
Figura 27: Desalineamiento de los potenciales en los registros.	31
Figura 26: Variación de los errores fases, turns y coeficiente de irregularidad.....	32

Lista de Tablas

Tabla 1: Promedio de los valores de los parámetros para cada tipo de patología.....	17
Tabla 2: Promedio de los parámetros de los registros miopáticos según el método.....	17
Tabla 3: Promedio de los parámetros de los registros crónicos según el método.....	17
Tabla 4: Promedio de los parámetros de los registros subagudos según el método.....	17
Tabla 5: Promedio de los parámetros de los registros normales según el método.....	18
Tabla 6: Coeficientes de correlación para los PAUMs miopáticos.....	18
Tabla 7: Coeficientes de correlación para los PAUMs neurógenos crónicos.....	18
Tabla 8: Coeficientes de correlación para los PAUMs neurógenos subagudos.....	18
Tabla 9: Coeficientes de correlación para los PAUMs normales.....	19
Tabla 10: Valor cuadrático medio de los errores de los registros miopáticos.....	24
Tabla 11: Valor cuadrático medio de los errores de los registros crónicos.....	25
Tabla 12: Valor cuadrático medio de los errores de los registros subagudos.....	25
Tabla 13: Valor cuadrático medio de los errores de los registros normales.....	25
Tabla 14: Resultados finales, Parámetros vs Métodos.....	26
Tabla 15: Resultados finales, Parámetros vs Grupos.....	26
Tabla 16: Resultados finales, Métodos vs Grupos.....	26

INTRODUCCIÓN

La electromiografía (EMG) es una técnica médica que se centra en el estudio de la actividad eléctrica del sistema muscular. Es una herramienta muy útil en el diagnóstico de patologías neuromusculares mediante la contracción muscular por activación eléctrica.

La principal función de los músculos es contraerse y elongarse, para así poder generar fuerza, movimiento y realizar funciones vitales. Se distinguen tres grupos de músculos, según su disposición:

- El músculo cardiaco es un tipo de musculo muy especializado que solamente se encuentra en el corazón, no presenta fatiga, se contrae de forma automática, involuntaria y generalmente rítmica.
- El músculo liso se contrae de forma involuntaria, por orden del sistema nervioso autónomo. Se localiza en zonas como las vísceras, la piel o los vasos sanguíneos.
- El musculo estriado o esquelético, que presenta fatiga, está anclado a los huesos mediante los tendones. Su principal función es generar fuerza o movimiento de forma voluntaria.

El músculo estriado está compuesto por un gran número de fibras o células musculares dispuestas a lo largo y en paralelo a otras fibras. Como se muestra en la Figura 1, cada fibra muscular está compuesta por el sarcolema, que recubre y protege la fibra hasta que termina en tendón. Protegidas por el sarcolema se encuentran las miofibrillas están compuestos por miofilamentos que son un entramado de proteínas llamadas actina y miosina cuya principal propiedad, es la contractilidad, es decir, la capacidad de acortar su propia longitud cuando son sometidas a un estímulo físico, químico, eléctrico o mecánico.

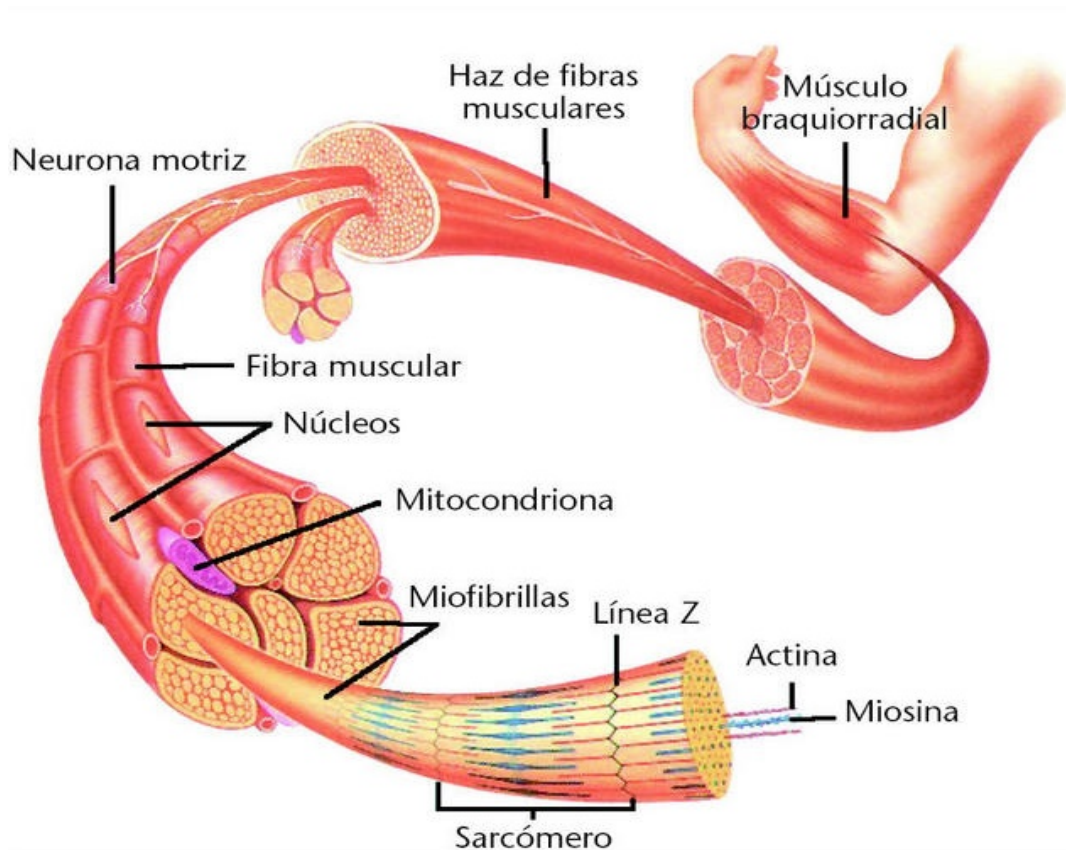


Figura 1: La fibra muscular es el elemento fundamental del músculo que agrupada en haces de fibras lo compone. Cada fibra posee una microestructura formada por filamentos llamados miofibrillas. Estas, a su vez, están provistas de miofilamentos: actina y miosina.

Generalmente, el estímulo que provoca la contracción es un impulso eléctrico enviado desde la motoneurona. Las motoneuronas son neuronas encargadas de la excitación muscular, se encuentran en la espina dorsal y se comunican con los músculos con el fin de activar la contracción de las fibras musculares mediante fibras nerviosas mielinizadas, llamadas axones. La unión entre las dendritas o terminaciones de los nervios y las fibras musculares se denomina unión neuromuscular, y es el responsable de canalizar el impulso eléctrico desde las fibras nerviosas a las fibras musculares mediante la sinapsis. En la sinapsis se liberan neurotransmisores que se unen a receptores especializados, desencadenando una corriente eléctrica en ésta, denominada potencial de acción.

Para poder entender el concepto de sinapsis, hay tener en cuenta que siempre hay una diferencia de potencial entre la parte interna y la parte externa de la membrana celular (aproximadamente -70 mV). En reposo, la carga de la membrana celular se mantiene en valores negativos y varía dentro de unos estrechos márgenes. Cuando el potencial de membrana de una célula excitable se despolariza más allá de un cierto umbral (de -65 mV a -55 mV) la célula dispara un potencial de acción.

Desde un punto de vista interno a la fibra, el potencial se propaga desde el punto de inervación hacia los tendones, en los extremos del músculo. Se distinguen tres casos, reflejados en la Figura 2, según el estado de la membrana. Un estado de reposo, en el que no hay potencial de acción; un estado de despolarización, en el que el potencial de

membrana abandona bruscamente el valor de reposo; y por último, un estado de repolarización en el que lentamente se vuelve a alcanzar el valor de reposo.

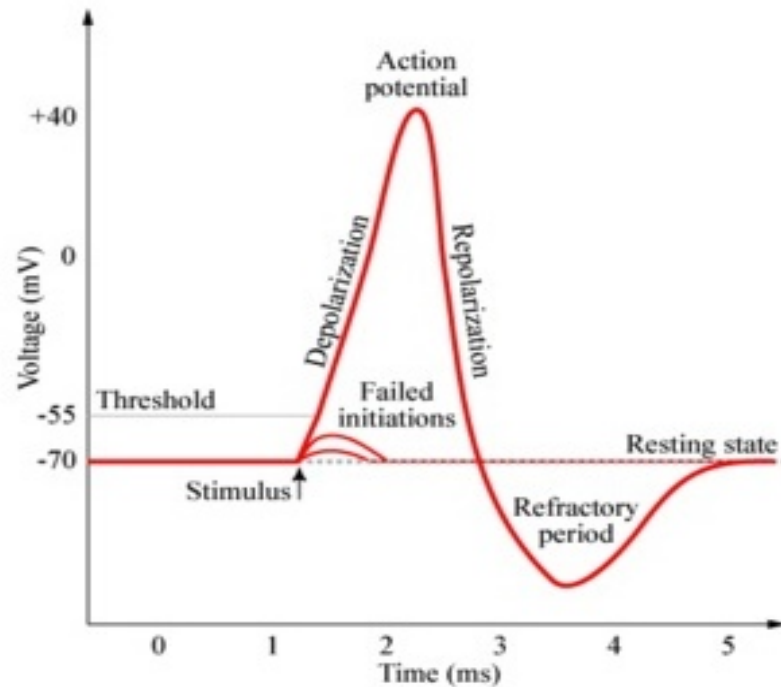


Figura 2: Partiendo de un estado de reposo, el potencial de acción solo se dispara si supera un umbral.

El mismo impulso eléctrico que recorre la fibra, se ve de forma distinta desde un punto de vista externo a ella. Como se muestra en la Figura 3, la señal recogida del potencial extracelular de una fibra simple varía principalmente por tres motivos: la separación respecto a la fibra objetivo, si está muy lejos la amplitud será menor; si está muy cerca del punto de inervación o del tendón, varían tanto la forma como la amplitud; de igual modo, la amplitud recogida tiene una fuerte dependencia del grosor de la fibra.

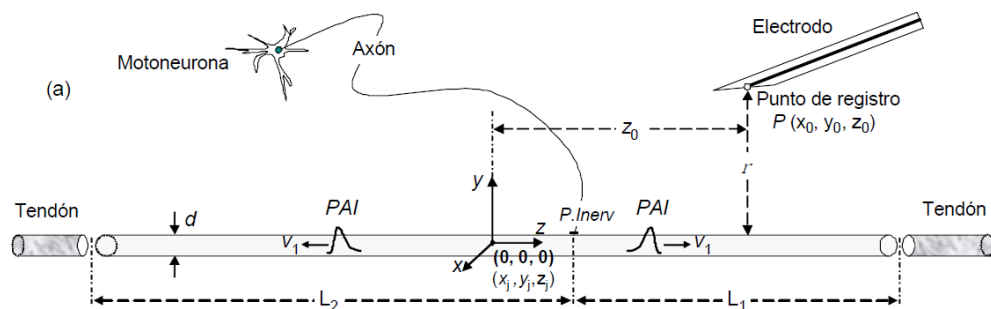
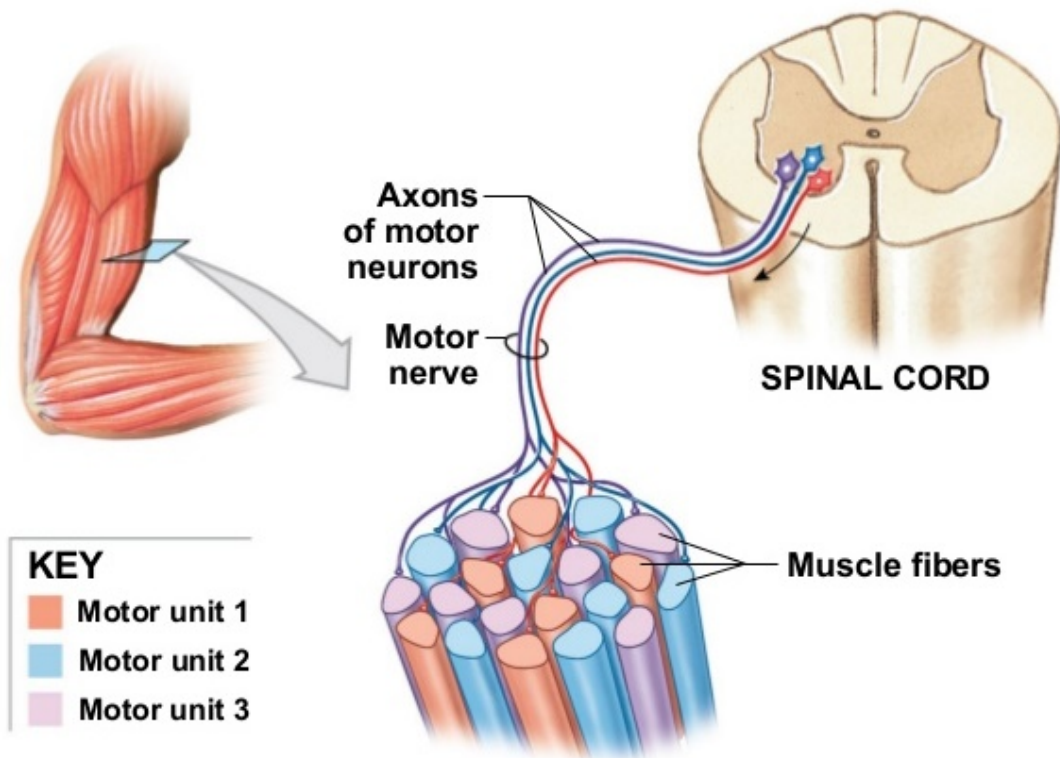


Figura 3: Representación esquemática de una fibra inervada por una motoneurona [1].

Pero en un escenario real, la contracción de un músculo con el nivel de fuerza deseado no es obra únicamente de la contracción repetida de una fibra muscular, sino de las fibras de una, o varias unidades motoras, controladas por su motoneurona alfa. Los músculos que controlan los movimientos finos, como, mover un dedo o cerrar un parpado, utilizan motoneuronas con menor número de fibras que los músculos que requieren más fuerza como los que se activan al correr o saltar.

Una unidad motora (UM), Figura 4, es un sistema formado por una única motoneurona, su axón y las fibras musculares que inerva. Varias unidades motoras de un músculo coexisten en la misma zona sin estar compactadas. Es decir, que las fibras de las unidades motoras se encuentran mezcladas con las fibras de otras unidades motoras.



© 2012 Pearson Education, Inc.

Figura 4: Ejemplo de distintas motoneuronas que nacen en la médula espinal e inervan mediante axones las fibras musculares. Hay que tener en cuenta que una única motoneurona excita varias fibras musculares, y que un músculo está compuesto por fibras inervadas por distintas motoneuronas.

Cuando una unidad motora se activa, el potencial de acción se dirige de la motoneurona, a través de su axón y las terminaciones de éste llegan a las fibras inervadas. Después de transmitirse el potencial de acción, a través de la unión neuromuscular, se dispara un potencial, casi síncrono, en todas las fibras musculares inervadas por la unidad motora particular.

Cada fibra muscular se despolariza, genera un potencial de acción que recorre la fibra y genera un campo eléctrico que puede ser detectado por un electrodo superficial, situado cerca de este campo o por un electrodo de aguja insertado en el músculo.

La suma, en tiempo y espacio, de cada campo eléctrico producido por las fibras que constituyen la unidad motora, se conoce como potencial de acción de la unidad motora (PAUM), donde la aportación de cada fibra dependerá directamente de la posición del electrodo respecto a la fibra, como se explicó anteriormente. En la Figura 5 se muestra un ejemplo.

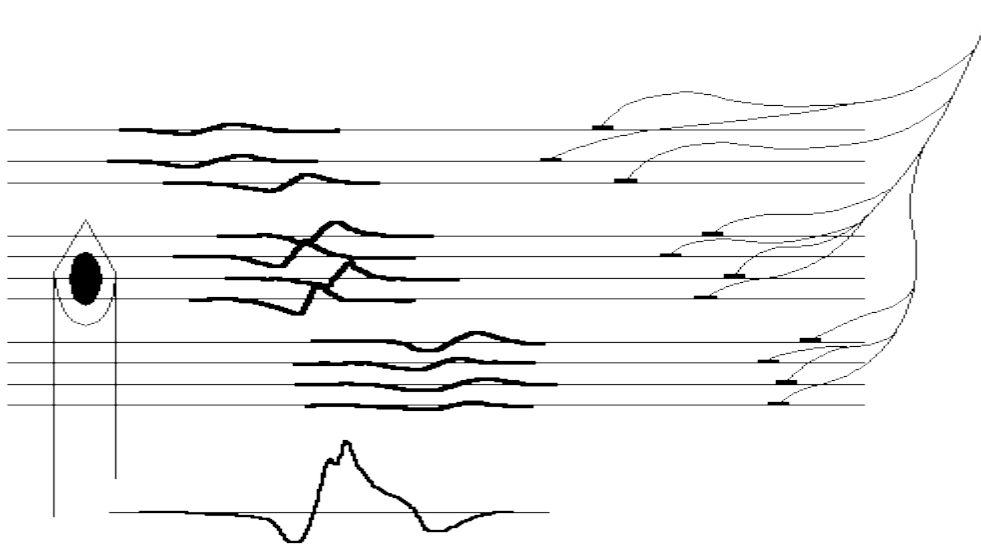


Figura 5: Representación esquemática de un PAUM, donde su morfología proviene de la suma de los potenciales de cada fibra y su aportación tiene una fuerte dependencia de la distancia y punto de inervación. Fuente: <http://www.signal.uu.se/Research/rneurophys.html>

En un escenario de medición real, un músculo mantiene la contracción mediante la activación repetida de las unidades motoras, cada una con una frecuencia de disparo concreta. A medida que el número de unidades motoras reclutadas aumenta, el electrodo registra simultáneamente más trenes de PAUMs y se hace más difícil discernir unas de otras.

El patrón de disparo de cada tren de PAUMs da información de cómo el sistema nervioso central controla la unidad motora. Un patrón normal es regular y rítmico, con poca variabilidad en el disparo.

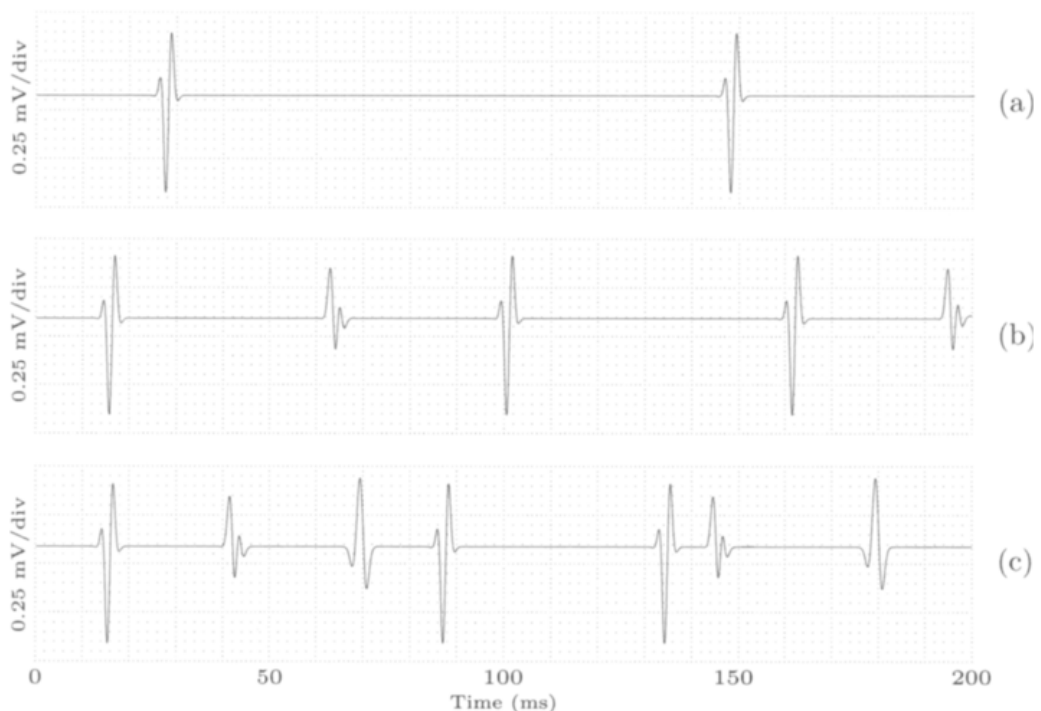


Figura 6: Reclutamiento de unidades motoras. Desde una hasta 3 unidades motoras.

Los PAUMs activados de forma voluntaria son el objeto principal de estudio de la electromiografía. Consisten en la suma de distintos potenciales de acción de grupos de fibras musculares que se están contrayendo casi sincrónicamente. Pueden ser monofásicos, bifásicos o trifásicos y, en ocasiones, polifásicos con cinco o más fases. Su duración está comprendida entre 2 ms y 15 ms y su amplitud entre 100 pV y 2 mV, aunque estas magnitudes dependen mucho del tipo de electrodos empleados, del músculo considerado y del número de fibras de la unidad motora [2].

Para analizar un PAUM se puede extraer su forma representativa y cuantificarla en diferentes parámetros como [3]:

- **Baseline o Línea Base:** Al adquirir una señal electromiográfica, otras señales artificiales pueden ser registradas. Estos artefactos pueden generar curvas no deseadas, ruido y fluctuaciones de la línea base. En definitiva, pueden distorsionar la morfología del PAUM. Por eso, la estimación de la línea base es la primera tarea al analizar una señal EGM. El comienzo suele ser abrupto.
- **Duración:** La duración es el intervalo entre el punto inicial y final de los potenciales lentos del PAUM donde se incluye la espiga principal. Se suele ver como el recorrido que hace la traza desde que deja la línea base hasta que vuelve a ella.

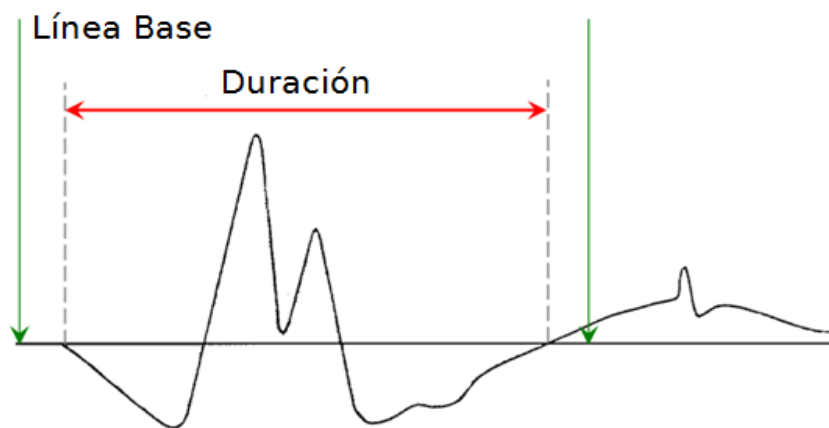


Figura 7: Línea base y duración

- **Área:** El área se relaciona directamente con el diámetro y el número de fibras. Se calcula integrando la señal rectificada y acotada por su duración.
- **Amplitud:** Es la medida máxima del voltaje pico a pico en la señal delimitada por la duración. Como se ha mencionado antes la amplitud ofrece información del número de fibras y su diámetro. Además, presenta una gran variabilidad, en función de la distancia respecto al electrodo.

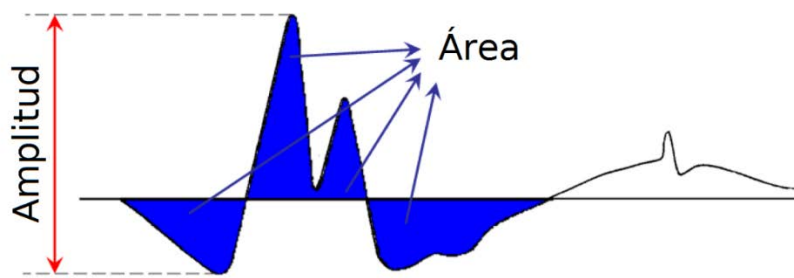


Figura 8: Amplitud y área.

- **Número de fases:** Es el número de veces que el PAUM corta la línea base más uno. Para evitar las fluctuaciones de alta frecuencia se suele establecer un umbral mínimo de amplitud. Un PAUM de un paciente sano no tiene más de tres fases, los que tienen más se llaman PAUM polifásicos.
- **Número de turns:** Los turns son los máximos y mínimos relativos que hay a lo largo de la duración del PAUM. Cada pico tiene que superar un umbral de amplitud, tanto de forma positiva como negativa, respecto al turn anterior. Generalmente, 20 ó 50 μV .

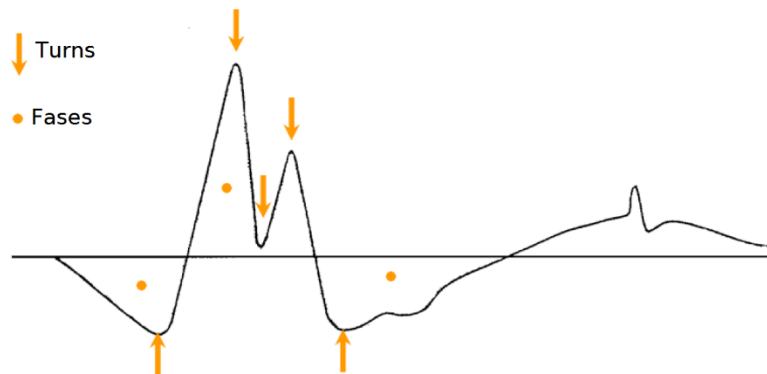


Figura 9: Turns y número de fases.

- **Irregularidad:** Este parámetro cuantifica en un solo coeficiente acumulativo la variabilidad de las pendientes muestra a muestra para ver como varía la amplitud a lo largo del PAUM. Es proporcional al número de fases y de turns, pero ponderados a su amplitud [4].
- **Duración de la espiga:** Se define como el intervalo entre el primer y el último pico positivo del PAUM. Los satélites (Figura 11) se incluyen como parte de la espiga. Este parámetro viene dado por la dispersión temporal del potencial de acción de las fibras más cercanas al electrodo y su forma por el número, grosor y su sincronía.
- **Área de la espiga:** De forma análoga al área, se calcula integrando la señal rectificadas acotada por la duración de la espiga.

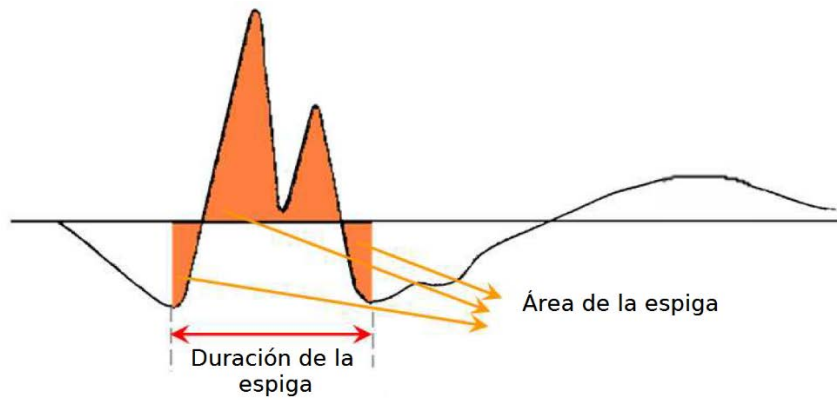


Figura 10: Duración espiga y su área.

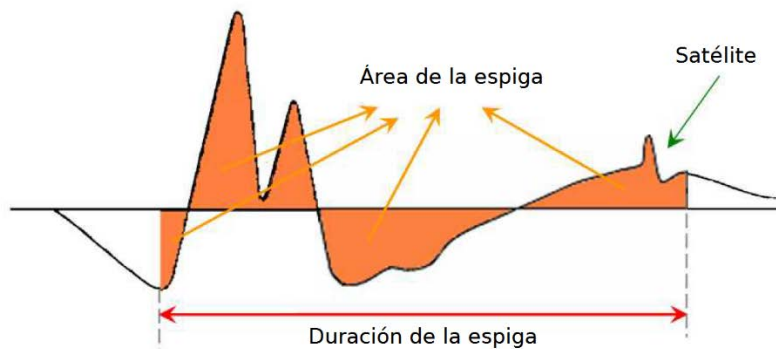


Figura 11: Los satélites son pequeños potenciales, originados por fibras individuales que generan su potencial con retraso y, que aparecen siempre después de la espiga principal.

La fiabilidad del análisis depende en gran medida de la calidad de los métodos de extracción de estos parámetros.

El registro y procesado de señales electromiográficas se ha convertido en una herramienta indispensable en el día a día de los laboratorios de neurofisiología facilitando análisis cuantitativos y facilitando la toma de decisiones en campos como la rehabilitación, medicina deportiva o neurofisiología clínica.

En este trabajo se han tratado patologías neurológicas generales como son las miopatías y neuropatías crónicas y subagudas.

Las enfermedades **miopáticas** se caracterizan por ser un trastorno muscular cuyo principal síntoma es la pérdida de fibras musculares y el aumento en la variabilidad del diámetro de la fibra. La debilidad muscular genera atrofia o hipertrofia de las fibras. Las fibras abandonadas pueden llegar a regenerarse de forma idiopática, en este caso, los axones de las otras unidades motoras pueden competir para reinervarlas.

Se consideran enfermedades sistémicas, ya que también afecta a estructuras como la piel o las articulaciones, aunque el principal órgano afectado es el músculo estriado [5].

Si el electrodo de aguja se sitúa en una zona de pérdida de fibras, el PAUM resultante será de baja amplitud (Fig. 12B). Si está cerca de fibras hipertróficas la amplitud será mayor, pero de mucha menos duración (Fig. 12C). En cambio, si está cerca de una fibra sana, la amplitud será normal pero el PAUM será de corta duración (Fig. 12D). En raras ocasiones, si el electrodo está cerca de un área con regeneración, el PAUM resultante tendrá una forma compleja y larga duración (Fig. 12E).

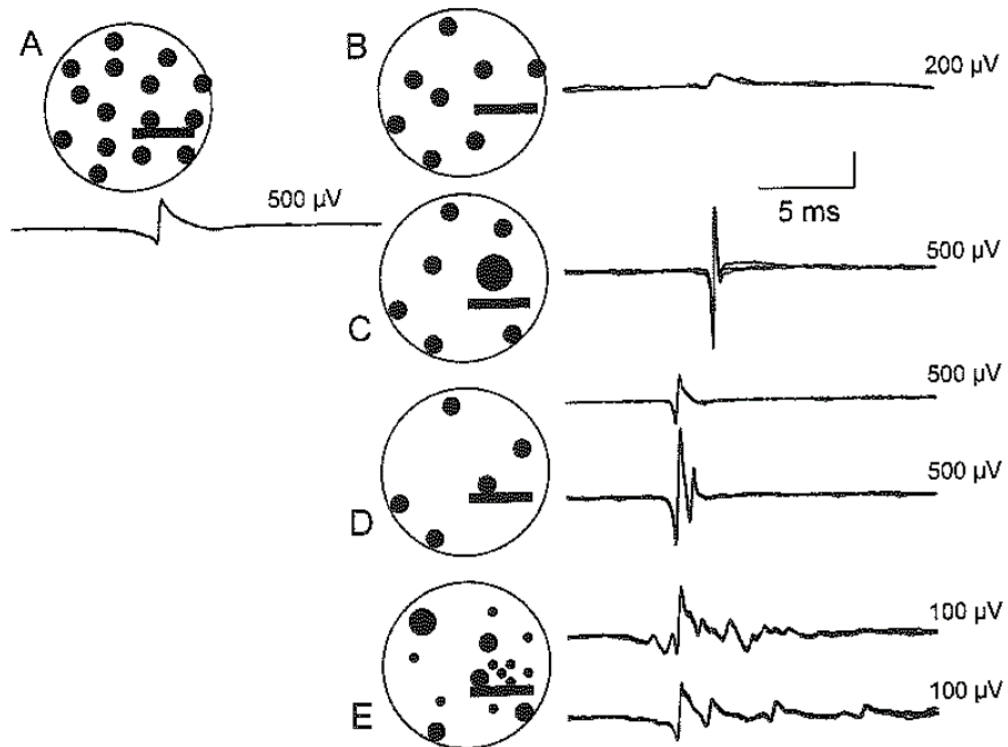


Figura 12: A) Motoneurona normal. B) PAUM con pérdida de FM. C) Registro de PAUM con una fibra hipertrófica. D) Electrodo cerca de fibra muscular sana. E) Registro de zona con regeneración.

Los pacientes con miopatías presentan un número relativamente normal de unidades motoras, por lo tanto, de una se puede obtener un patrón completo de MUAP incluso de un músculo débil.

En cambio, en pacientes con **neuropatías**, el principal síntoma es la pérdida de unidades motoras. Las unidades motoras supervivientes, compiten para reinervar las fibras musculares abandonadas y ganar densidad de fibras en el territorio de la unidad motora. Debido a este incremento de la densidad, la amplitud y duración del PAUM aumentan y en el caso de asincronía entre potenciales de acción, los PAUM serán polifásicos. Además, en los procesos de reinervación, los axones generan nuevas dendritas y puntos de inervación inmaduros pudiendo llegar a errar al transmitir el potencial de acción generando PAUMs inestables.

La mayoría de los pacientes no muestran debilidad muscular incluso tras haber perdido la mitad de las unidades motoras. Estas pueden reinervarse hasta que tengan un tamaño de aproximadamente 20 veces su tamaño normal mostrando un PAUM catalogado como gigante, frecuencia de disparo alta y gran duración y amplitud.

Este tipo de PAUM es más común en pacientes con patologías de lento desarrollo donde las unidades motoras perdidas se compensan con reinervación. En el caso de

que una motoneurona gigante muera, dejará un gran territorio de fibras musculares abandonadas donde difícilmente se llegarán a inervar las fibras del centro del territorio.

Otra posibilidad es que una unidad motora gigante, con el paso de los años, no sea capaz de mantener todo el conjunto de fibras musculares y se pierdan algunas de forma aleatoria. El PAUM será de menor amplitud y duración, pero seguirán siendo mayor que un PAUM de una unidad motora sana.

En una patología con degeneración rápida, como la esclerosis lateral amiotrófica, la reinervación no se realiza al ritmo de la pérdida por lo que los PAUMs reflejaran un ligero aumento en su duración y amplitud, pero habrá una disminución sostenida del número de unidades motoras y por lo tanto del número de trenes.

OBJETIVOS

El objetivo de este trabajo es continuar con el proyecto de investigación de Armando Malanda publicado en 2016, *Sliding window averaging for the extraction of representative waveforms from motor unit action potential trains* [7]. En este artículo, se presentaba un nuevo algoritmo, basado en un enfoque por intervalos, para obtener una forma de onda que represente fielmente el conjunto de potenciales que forman un PAUM.

Para evaluar la eficacia de este nuevo algoritmo se compararon los resultados con los que ofrecían algunos de los métodos más usados en la electromiografía sobre registros de pacientes sanos.

Tras ver el éxito del nuevo método, se presenta este proyecto que pretende continuar evaluando la eficacia del algoritmo con señales EGM de músculos con diferentes patologías.

Para ello se han fijado las siguientes tareas:

Desarrollar una herramienta en Matlab que permita la visualización y la anotación, tanto de simples registros como de bases de datos. Además, pensando en un uso a futuro se han añadido funciones que hacen la herramienta más flexible, como poder cargar datos con otra estructura, cargar desde el área de trabajo o indicar la frecuencia de muestreo de la señal. Anexo a este documento se añaden el manual de usuario y de programador de esta herramienta.

Con la herramienta desarrollada se ha procedido a la fase anotación de los registros para obtener el Gold Standard. En los próximos apartados se profundizan en las características y origen de los registros y en cómo se obtienen el Gold Standard de cada uno.

Después, se ha realizado el procesado que consiste en obtener la forma de onda de cada registro con los distintos métodos. Para cada forma de onda se han calculado algunos parámetros y se han comparado con los resultados de esos mismos parámetros obtenidos del Gold Standard.

Finalmente se ha procedido con el análisis estadístico de los resultados para evaluar si existen diferencias significativas entre los métodos en la extracción de la forma de onda.

MATERIAL

La base de datos utilizada está compuesta de 424 registros, agrupados por patologías: miopáticos (178), neurógenos crónicos (63), neurógenos subagudos (124) y normales (59).

Las señales se obtuvieron de diversos grupos musculares: el músculo tibial anterior, el gastrocnemio medial, el musculo abductor del meñique de la mano, el músculo abductor del dedo gordo, los músculos interóseos dorsales de la mano, los músculos extensores comunes de los dedos, el músculo abductor corto del pulgar y músculo vasto medial.

Dentro de cada grupo, los pacientes estaban diagnosticados con las siguientes patologías:

- Miopáticos: distrofia muscular de cinturas, distrofia muscular escapulooperonea, distrofia miotónica, miopatía mitocondrial y polimitosis.
- Neurógenos crónicos: radiculopatía crónica, atrofia muscular espinal y postpolio.
- Neurógenos subagudos: radiculopatía subaguda, neuralgia amiotrófica y esclerosis lateral amiotrófica con reinervación en curso.

Estas señales se recogieron con el equipo electromiográfico Medelec Synergy de Oxford Instruments Medical, usando electrodos de aguja concéntricos de Medtronic (diámetro 0.46 mm, área de registro 0.07 mm²). Finalmente, las señales se procesaron con un filtro paso banda con su banda de paso desde 3 Hz hasta 10 kHz, con una frecuencia de muestro de 20 kHz y una conversión analógica-digital de 16 bits. Cada registro tiene una duración de 50 ms (1000 muestras) y un número variable de trenes de PAUM.

Finalmente se descartaron 6 registros, 3 crónicos y 3 subagudos, que consideramos como no aptos al tener una apariencia excesivamente ruidosa o formas totalmente diferentes a las esperadas para su grupo.

MÉTODOS

En este proyecto se han usado cuatro métodos distintos para extraer los PAUMs de cada registro a partir de su tren de señales. Para cada método y registro se han obtenido los parámetros anteriormente explicados y se han comparado con un *Gold Standard*. Estos métodos son:

- **Media Aritmética (EA):** Para un número (n) dado de PAUMs, la media aritmética se calcula como la suma de esos PAUMs, dividido entre el número de señales (n).

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i = \frac{x_1 + x_2 + \dots + x_n}{n} \quad (1)$$

- **Mediana (MA):** es una medida de posición central que ordena los PAUMs según su amplitud y en cada muestra elige el valor central. En el caso de que el número de muestras sea par y haya dos valores centrales, se realizará la media de esos valores para obtener un valor único. Las siguientes expresiones recogen ambos casos:

$$M_e = x_{(n+1)/2} \quad (2)$$

$$M_e = (x_{\frac{n}{2}} + x_{\frac{n}{2}+1})/2 \quad (3)$$

- **Promedio de los 5 más próximos (Five Closest Average, FCA):** Este método se basa en el ideado por Nendedkar y Sanders [6] y fundamentalmente realiza un promediado de los 5 potenciales que más se parecen entre sí. Se calcula mediante un proceso iterativo en el que en cada iteración se descarta un potencial. Para determinar los potenciales más parecidos se empieza con todo el tren de PAUMs, se calculan las distancias Euclídeas y se descarta el potencial más distante del resto. Se repite el proceso hasta que quedan los cinco potenciales más parejos. Para maximizar la eficacia, este método solo usa los tramos centrales de los PAUM, concretamente, 2.5 ms antes y 5 ms después del máximo principal.
- **Sliding window (SW):** Método desarrollado por Malanda [7]. Este algoritmo trata de encontrar para cada intervalo el conjunto de muestras idóneas que mejor representan al PAUM. Para ello usa una ventana deslizante de un tamaño prefijado que recorre el PAUM. Para cada ventana se obtiene la mediana con los tramos de dicha ventana y se calculan las distancias entre cada tramo y la mediana obtenida. Posteriormente, se seleccionan aquellos tramos cuya distancia a la mediana sea menor que la desviación estándar del PAUM multiplicada por una constante determinada empíricamente. Con los tramos seleccionados se realiza un cálculo de tendencia central obteniendo así las curvas representativas. Finalmente, se realiza el mismo

proceso desplazando la ventana, se ordenan las curvas y se ensamblan en una única curva representativa. En este trabajo, el tamaño de la ventana es de 100 muestras y se ha realizado dos cálculos de tendencia central, la media (SW1) y la mediana (SW2), para obtener las trazas representativas de cada ventana. Es decir que este método aporta dos curvas representativas para cada registro.

Gold standard

Se definió un Gold Standard con la forma de onda idónea para cada registro, con el fin de poder comparar las características de las formas de onda obtenidas por los diferentes algoritmos. El proceso para determinar cada Gold Standard ha sido el siguiente:

- En cada registro se alinean los potenciales según su pico principal y se añade un offset de forma automática para alinearlos en amplitud, Figura 13.
- Con los trenes de PAUMs alineados y con ayuda del software *PAUM_Analyzer* (ver Anexo), se identifican manualmente los instantes de inicio y fin del PAUM, Figura 14.
- Se selecciona el conjunto de potenciales que más se asemejen entre ellos o bien con formas similares, Figura 15.
- Finalmente, se toma como Gold Standard el promediado de este conjunto acotado por los instantes de inicio y fin seleccionados, Figura 16.

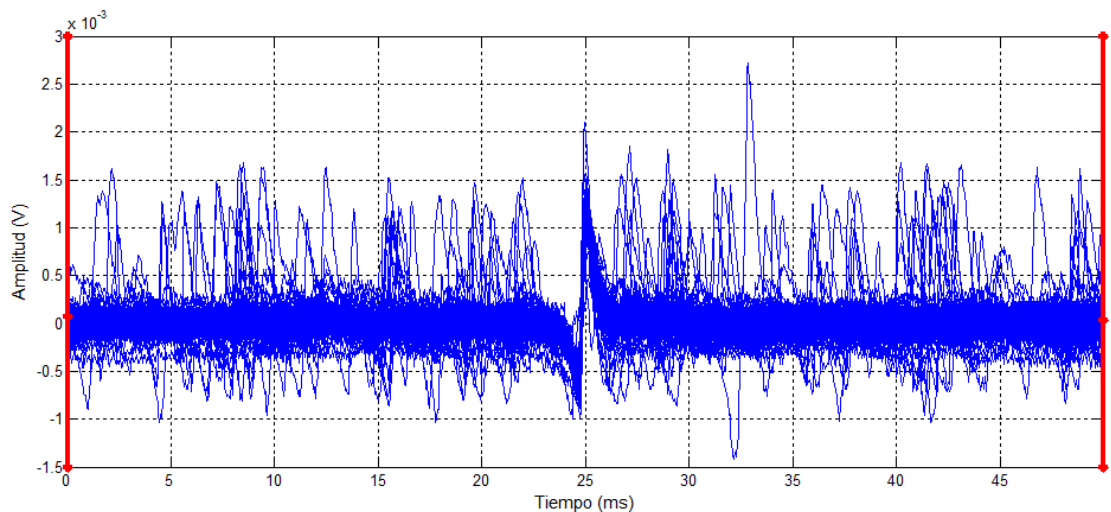


Figura 13: Registro, sin anotar, con los trenes de PAUMs alineados.

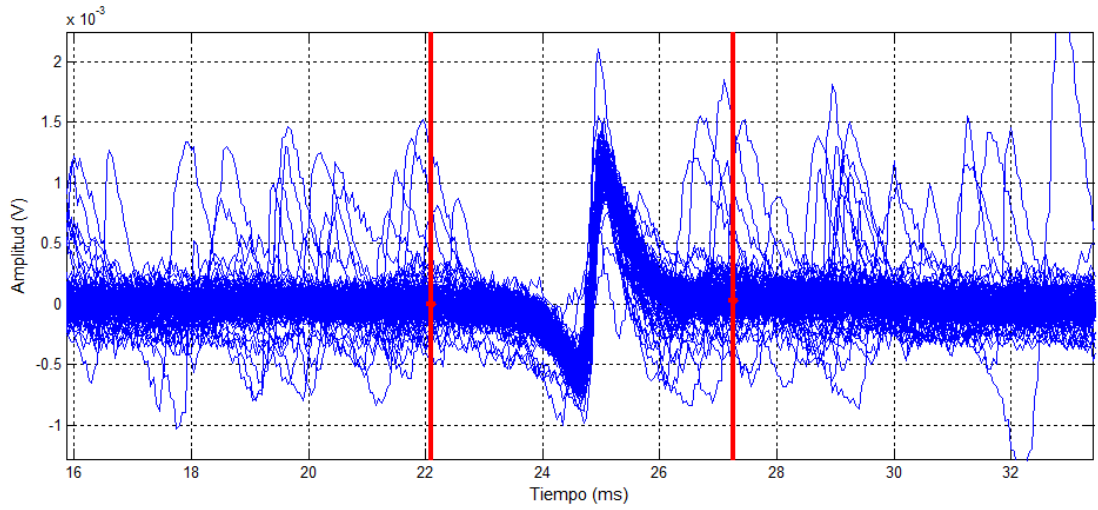


Figura 14: Barras verticales de marcadores de inicio (M1) y fin (M2). En este ejemplo, M1: 22.1 ms y M2: 27.2 ms.

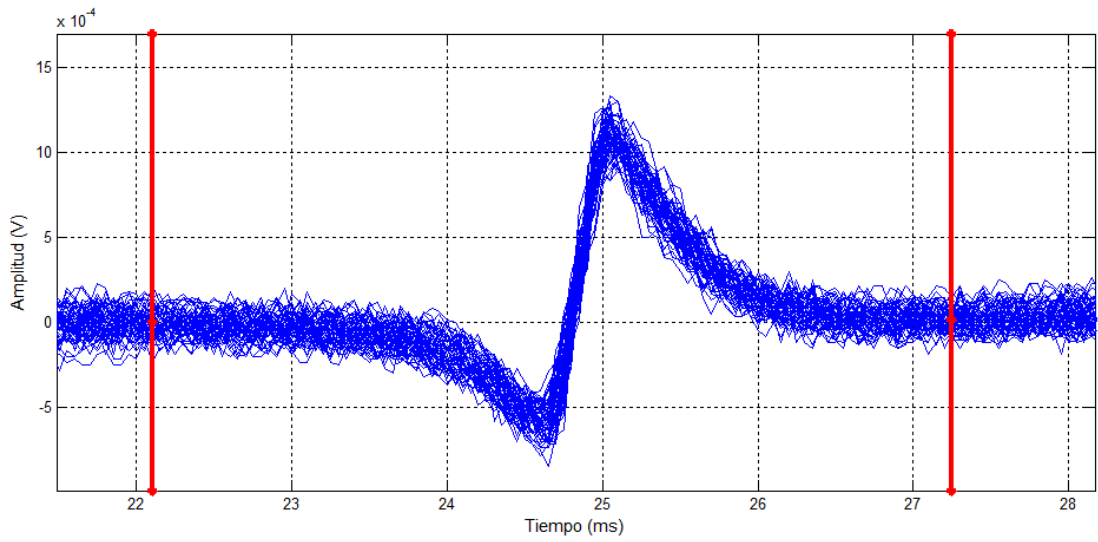


Figura 15: Registro con los potenciales seleccionados. En este ejemplo 55 de 114.

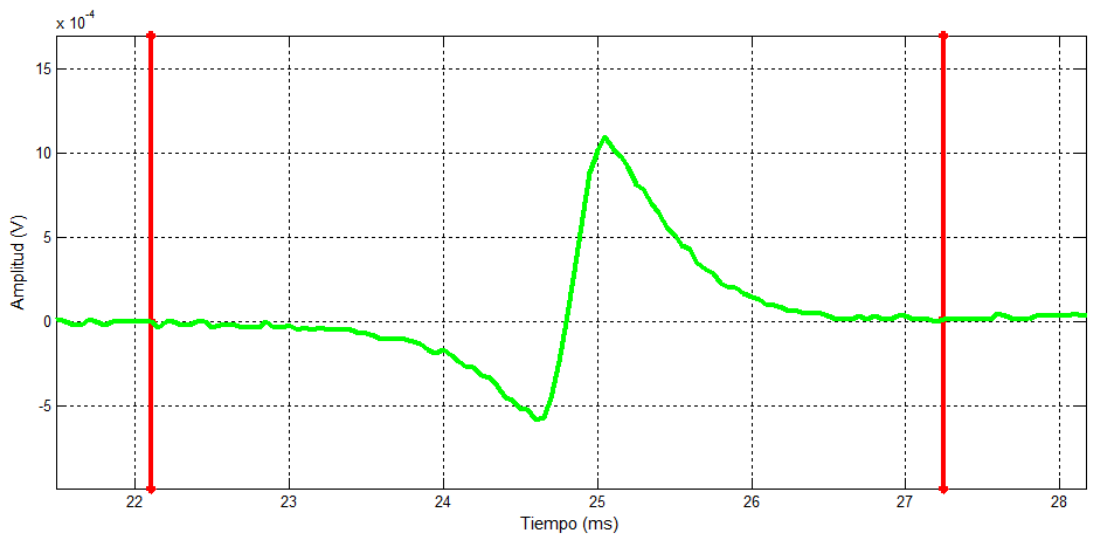


Figura 16: Gold Standard final, obtenido tras promediar los potenciales seleccionados.

Figuras de mérito

Para poder comparar las formas de onda de los PAUMs generados por los distintos métodos con la forma de onda del Gold Standard, se ha decidido utilizar como figuras de mérito varios de los parámetros descriptivos que más se utilizan en la electromiografía cuantitativa [8, 9, 10]. Este grupo se compone de 7 parámetros:

- Amplitud pico a pico (mV): diferencia entre el valor máximo y mínimo.
- Área (mV · ms): se calcula integrando el PAUM rectificado a lo largo de su duración. La integración se realiza sumando las muestras de amplitud del PAUM y multiplicando la suma por el periodo de muerto de la señal.
- Número de fases: definidas como los periodos comprendidos entre dos cruces por la línea de base superando un umbral para evitar el ruido o pequeñas fluctuaciones de la traza. En este trabajo se ha usado un umbral de 20 μ V [7].
- Número de turns: Un turn se calcula buscando un máximo o mínimo cuyo pico adyacente sea de polaridad opuesta y mantenga una diferencia en amplitud mínima, en este estudio de 50 μ V. Para minimizar la detección de artefactos se ha añadido como condición que los turns consecutivos disten como mínimo una distancia de 0.25 ms.
- Coeficiente de irregularidad: este parámetro mide la longitud de la curva del PAUM normalizada por su máximo y se calcula como la suma acumulada de las distancias entre las muestras consecutivas y normalizadas.

En este estudio, este parámetro no se ha calculado muestra a muestra, sino cada 3 muestras, lo que facilita la eliminación del ruido [11]. Para cada intervalo de muestras se calcula el valor medio que lo representa. Con el vector de medias de los intervalos se calcula la pendiente y se va acumulando en el coeficiente de irregularidad.

- Duración de la espiga: Se calcula como el intervalo entre el primer pico, ya sea positivo o negativo, y el último pico negativo [12]. Si el PAUM tiene menos de 2 picos, se considera que la espiga abarca toda la señal.
- Área de la espiga (mV · ms): se calcula integrando la forma de onda rectificada a lo largo de la duración de la espiga.

Los resultados de los parámetros de cada registro se compararon con los resultados obtenidos con la forma de onda del Gold Standard para ese mismo registro, con el fin de cuantificar los errores cometidos por cada método.

Para el número de turns se han contemplado los dos escenarios posibles: los turns por exceso, aquellos turns obtenidos pero que no existen en el Gold Standard; los turns por defecto o turns perdidos, que son los que se han encontrado en el Gold Standard, pero no en la señal construida con el método bajo análisis.

RESULTADOS

En la Tabla 1, se muestra el promedio de los valores obtenidos para los distintos parámetros según su patología y calculados a partir de los PAUMs del Gold Standard.

	Amplitud (mV)	Área (mV·ms)	Fases	Turns	Irregularidad	Spk_Dur (ms)	Spk_Area (mV·ms)
Miopáticos	0,410	0,430	3,320	2,983	2,957	7,981	0,279
Crónicos	4,323	6,513	4,833	6,383	2,969	6,334	3,859
Subagudos	0,852	1,245	4,694	5,207	3,514	6,593	0,706
Normales	0,516	0,715	3,153	2,746	2,707	6,865	0,411

Tabla 1: Promedio de los valores de los parámetros del PAUM obtenidos con mediciones manuales para cada tipo de patología.

Se observa una gran diferencia en los parámetros de la familia de neurógenos crónicos, concretamente en la *amplitud*, el *área* y el *área de la espiga*. Además, hay una diferencia considerable entre *el número de turns* en los PAUMs crónicos y subagudos, por una parte, y los normales y miopáticos, por otra.

En las Tablas 2, 3, 4 y 5, se muestran los promedios de los parámetros de los PAUMs obtenidos con los distintos métodos y con el Gold Standard y clasificados según su patología.

MIOPÁTICOS							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
GOLD	0,410	0,430	3,320	2,983	2,957	7,981	0,279
EA	0,403	0,434	3,326	2,871	2,935	8,122	0,282
MA	0,408	0,433	3,303	2,933	2,922	7,905	0,279
FCA	0,410	0,444	3,416	3,011	3,011	8,169	0,293
SW1	0,407	0,431	3,275	2,927	2,876	8,126	0,281
SW2	0,409	0,432	3,298	2,933	2,915	8,030	0,281

Tabla 2: Promedio de los parámetros de los registros miopáticos según el método.

CRÓNICOS							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
GOLD	4,323	6,513	4,833	6,383	2,969	6,334	3,859
EA	4,290	6,484	4,950	6,167	2,932	6,791	3,901
MA	4,296	6,479	4,683	5,883	2,920	6,253	3,822
FCA	4,282	6,531	5,283	7,067	3,031	7,418	4,059
SW1	4,304	6,483	4,733	6,183	2,950	6,268	3,841
SW2	4,312	6,484	4,733	6,250	2,959	6,365	3,855

Tabla 3: Promedio de los parámetros de los registros neurógenos crónicos según el método.

SUBAGUDOS							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
GOLD	0,852	1,245	4,694	5,207	3,514	6,593	0,706
EA	0,828	1,247	4,405	4,983	3,462	6,599	0,702
MA	0,836	1,238	4,364	5,198	3,464	6,487	0,697
FCA	0,856	1,232	4,876	5,455	3,537	7,225	0,731
SW1	0,839	1,237	4,355	5,190	3,411	6,611	0,701
SW2	0,844	1,237	4,438	5,281	3,462	6,563	0,701

Tabla 4: Promedio de los parámetros de los registros neurógenos subagudos según el método.

NORMALES							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
GOLD	0,516	0,715	3,153	2,746	2,707	6,865	0,411
EA	0,498	0,723	3,169	2,763	2,679	6,643	0,409
MA	0,508	0,718	3,068	2,763	2,689	6,664	0,411
FCA	0,513	0,736	3,254	2,847	2,723	6,747	0,414
SW1	0,508	0,714	3,068	2,746	2,637	6,869	0,413
SW2	0,512	0,715	3,085	2,831	2,675	6,856	0,414

Tabla 5: Promedio de los parámetros de los registros normales según el método.

Se observa como los métodos que más se alejan del valor marcado por el Gold Standard son el Five Closest Average (FCA) y la Media Aritmética (EA) para todas las familias patológicas.

Los coeficientes de correlación de los parámetros obtenidos con la colección de trenes PAUM del Gold Standard y agrupados por patología se recogen en las Tablas 6, 7, 8 y 9.

MIOPATICOS							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
Amplitud	1,000	0,596	0,321	0,341	-0,196	-0,359	0,512
Área		1,000	0,371	0,569	0,105	-0,151	0,863
Fases			1,000	0,643	0,592	-0,253	0,243
Turns				1,000	0,545	-0,400	0,364
Irregularidad					1,000	0,001	-0,025
Spk_Dur						1,000	0,234
Spk_Area							1,000

Tabla 6: Coeficientes de correlación para los PAUMs miopáticos

CRÓNICOS							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
Amplitud	1,000	0,808	-0,097	-0,098	-0,277	-0,230	0,871
Área		1,000	-0,066	-0,045	-0,158	-0,093	0,927
Fases			1,000	0,758	0,809	0,583	-0,102
Turns				1,000	0,698	0,681	-0,005
Irregularidad					1,000	0,622	-0,192
Spk_Dur						1,000	-0,017
Spk_Area							1,000

Tabla 7: Coeficientes de correlación para los PAUMs neurógenos crónicos

SUBAGUDOS							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
Amplitud	1,000	0,932	0,039	0,129	-0,210	-0,151	0,923
Área		1,000	0,033	0,170	-0,121	-0,069	0,982
Fases			1,000	0,652	0,631	-0,139	0,026
Turns				1,000	0,516	-0,065	0,220
Irregularidad					1,000	-0,040	-0,140
Spk_Dur						1,000	0,016
Spk_area							1,000

Tabla 8: Coeficientes de correlación para los PAUMs neurógenos subagudos

NORMALES							
	Amplitud	Área	Fases	Turns	Irreg	Spk_Dur	Spk_Area
Amplitud	1,000	0,535	0,115	0,123	-0,219	-0,171	0,696
Área		1,000	0,125	0,212	0,092	-0,390	0,713
Fases			1,000	0,548	0,534	0,005	0,187
Turns				1,000	0,652	-0,412	0,072
Irregularidad					1,000	-0,226	-0,185
Spk_Dur						1,000	0,115
Spk_Area							1,000

Tabla 9: Coeficientes de correlación para los PAUMs normales

Como tónica general entre los cuatro grupos, existe una fuerte correlación entre la *Amplitud*, *Área* y *Spike_Area*. Aunque, en el caso de los PAUMs miopáticos, el grado de correlación entre *amplitud* y *área* es menor. Además, los coeficientes de correlación indican una dependencia considerable entre los *turns*, las *fases* y la *irregularidad* y en el caso particular de los crónicos existe cierta relación entre los *turns*, la *irregularidad* y la *duración de la espiga*.

Para cuantificar la eficacia de cada método generando la forma de onda representativa de cada registro, se han calculado los errores relativos de cada parámetro respecto a los obtenidos con el Gold Estándar para ese mismo registro. En las Figuras 17, 18, 19 y 20 se resumen y comparan las distribuciones de los errores relativos para cada parámetro.

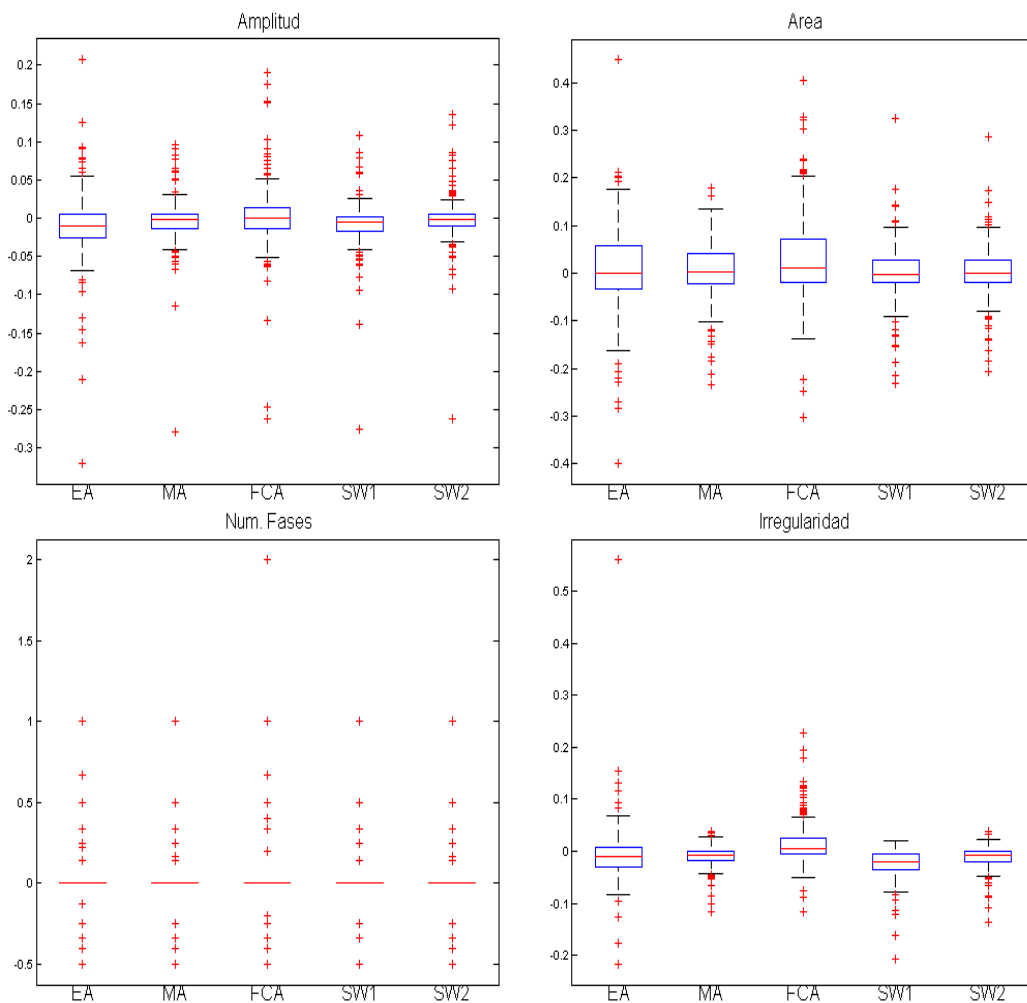


Figura 17: Boxplots de los errores relativos de los registros miopáticos (Continua)

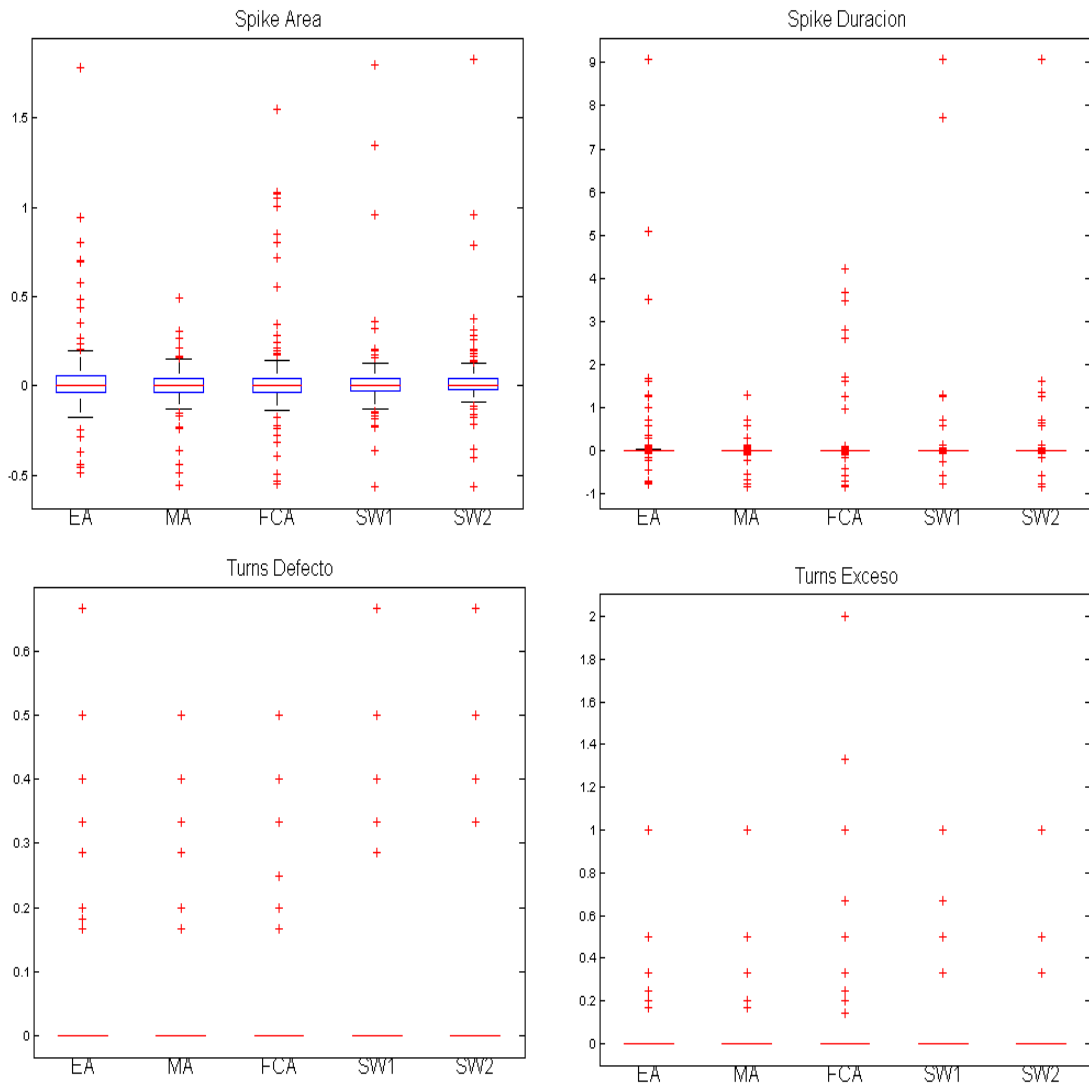


Figura 17: Continuación

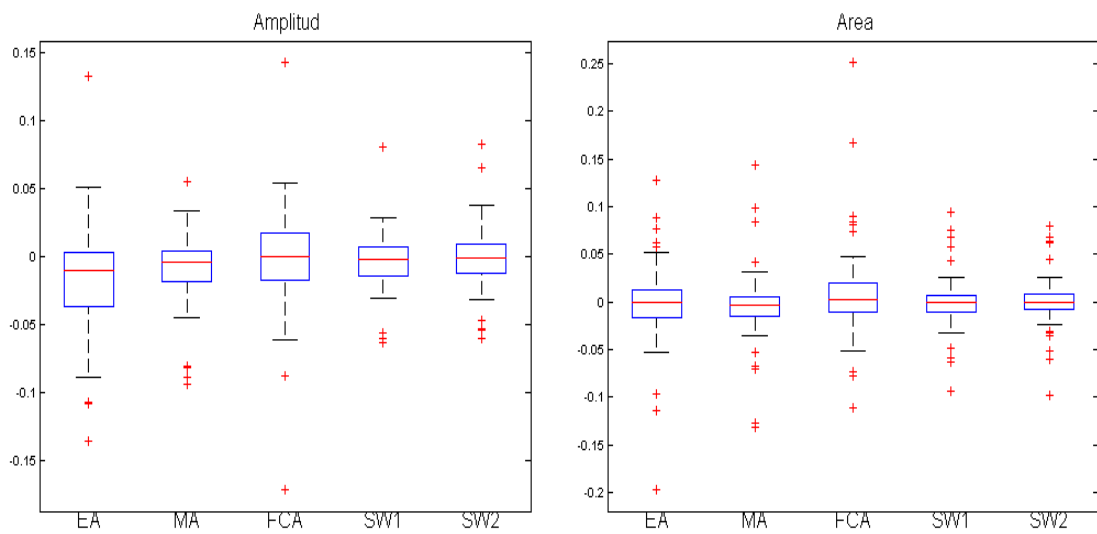


Figura 18: Boxplots de los errores relativos de los registros crónicos (Continúa)

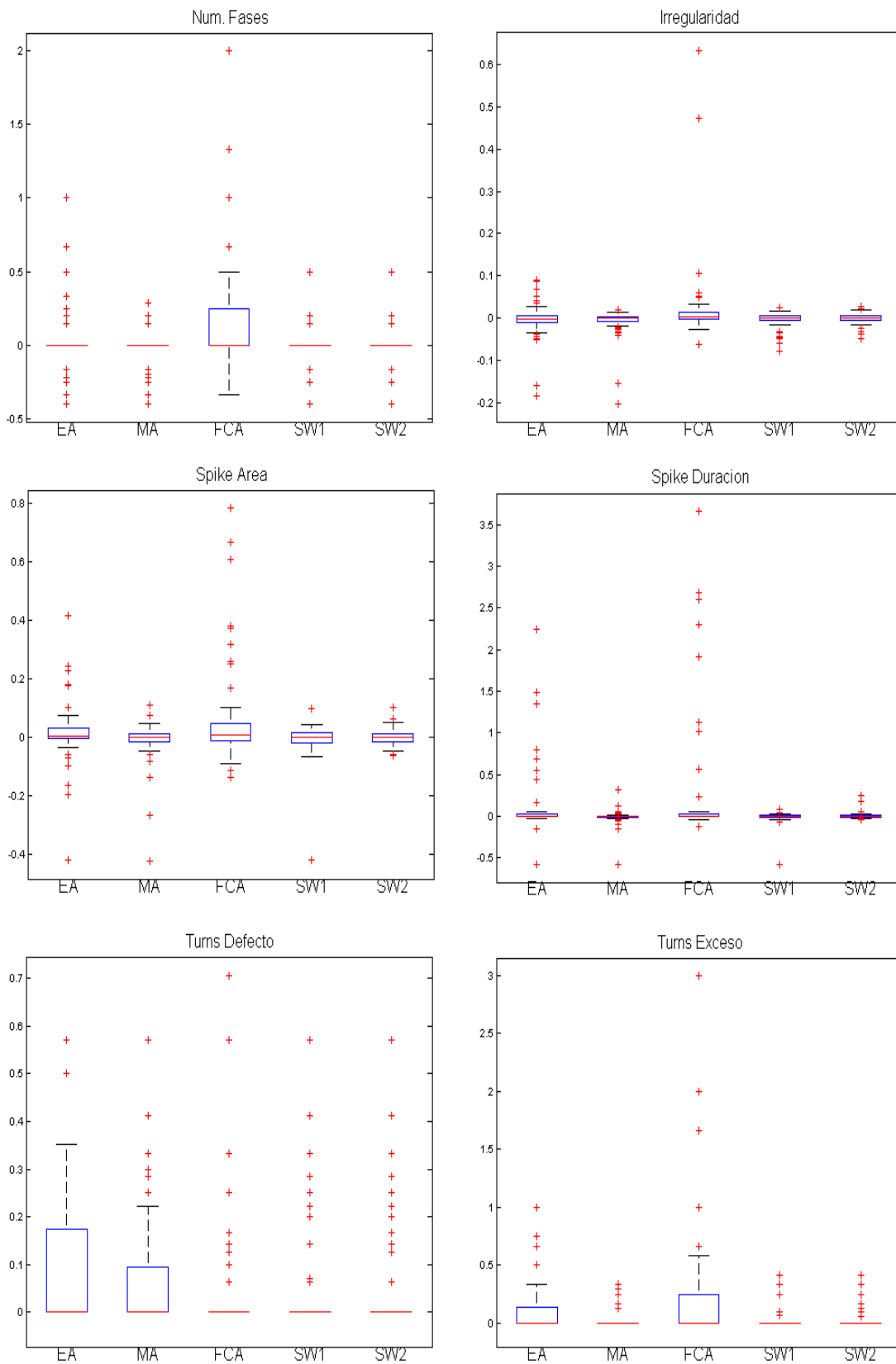


Figura 18: Continuación

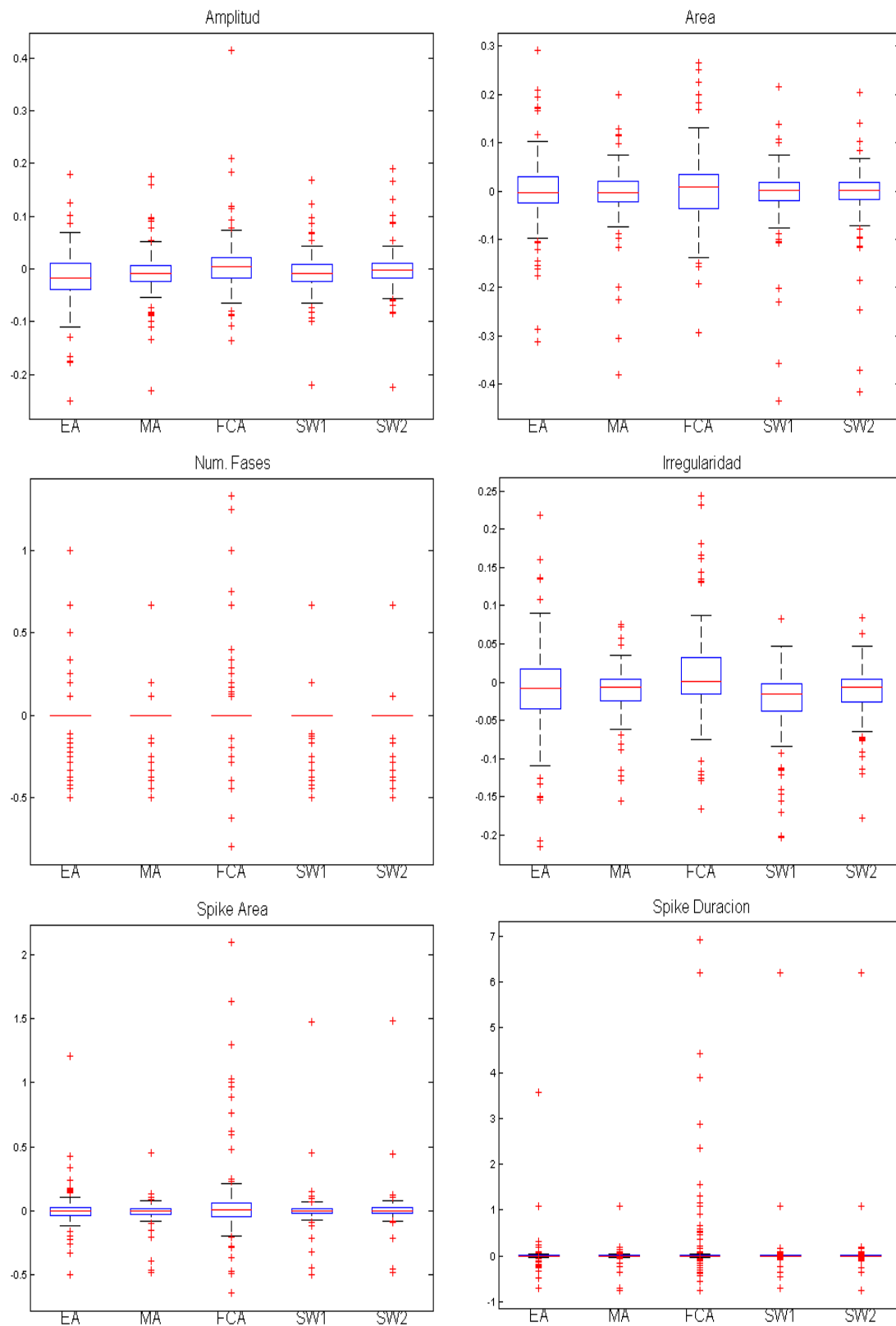


Figura 19: Boxplots de los errores relativos de los registros subagudos (Continua)

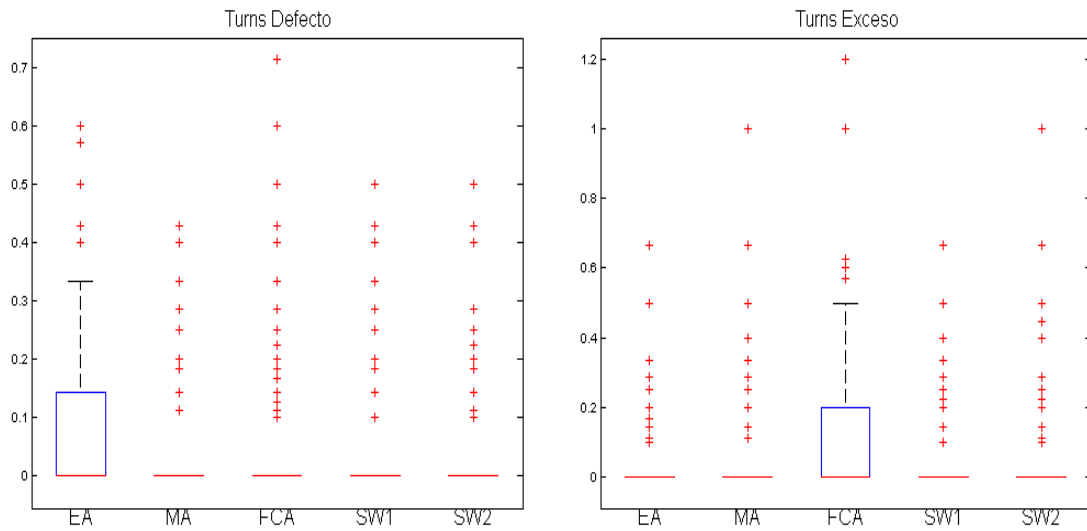


Figura 19: Continuación

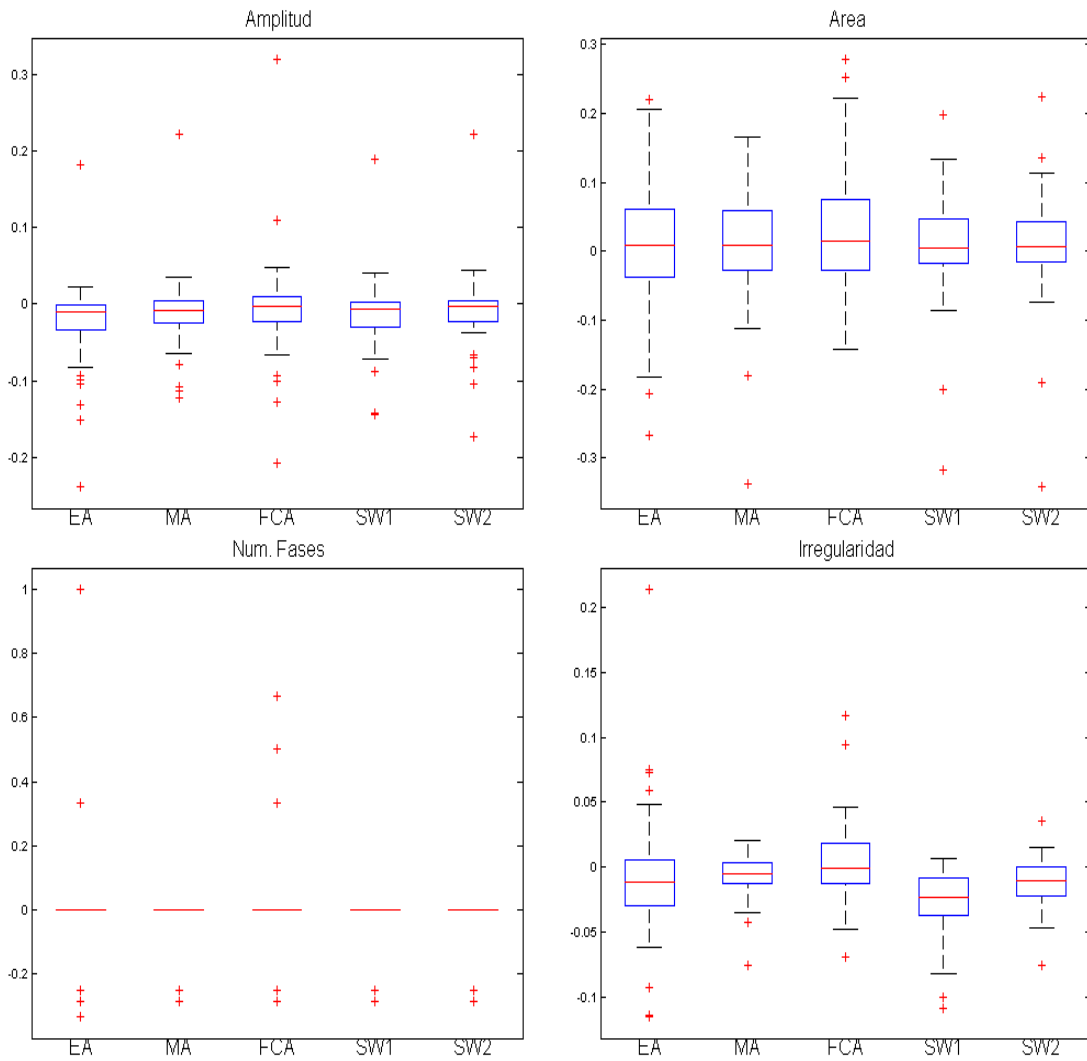


Figura 20: Boxplots de los errores relativos de los registros normales (Continúa)

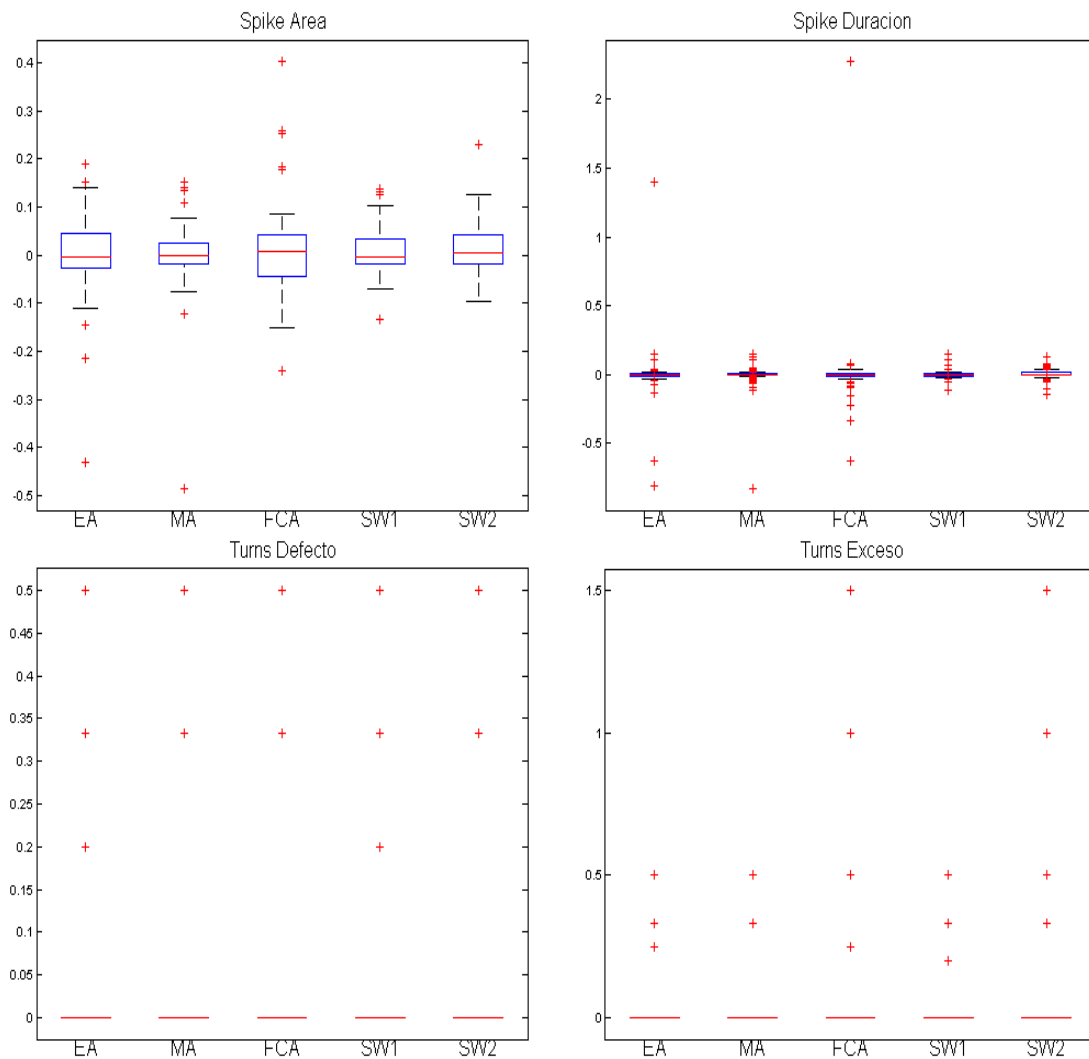


Figura 20: Continuación

Para determinar si alguno de los métodos se aproxima más al Gold Estándar, es decir, que tiene un error relativo significativamente menor, se ha optado por utilizar una Prueba T para cada parámetro y patología comparando cada método con el mejor (menor error relativo). Para evitar el efecto del signo y seleccionar el mejor método se ha calculado el valor cuadrático medio de los errores de cada método y posteriormente seleccionado el método de menor valor. En las Tablas 10, 11, 12 y 13 se presentan los valores de la media cuadrática de los errores.

MIOPATICOS								
	Amplitud	Área	Fases	Irreg	Spk_Area	Spk_Dur	Turns_Def	Turns_Exc
EA	0,05038	0,09533	0,19021	0,06173	0,21847	0,87305	0,15923	0,17023
MA	0,03445	0,06296	0,12208	0,02439	0,10571	0,16463	0,11043	0,14158
FCA	0,04832	0,10059	0,24176	0,04858	0,25298	0,63159	0,09804	0,24096
SW1	0,03455	0,06311	0,12223	0,04010	0,20388	0,91013	0,11011	0,10072
SW2	0,03415	0,05802	0,13483	0,02611	0,19115	0,71640	0,11636	0,09757

Tabla 10: Valor cuadrático medio de los errores de los registros miopáticos

CRÓNICOS								
	Amplitud	Área	Fases	Irreg	Spk_Area	Spk_Dur	Turns_Def	Turns_Exc
EA	0,04418	0,04481	0,24798	0,04291	0,10434	0,42885	0,16052	0,24551
MA	0,02931	0,04006	0,11791	0,03523	0,07361	0,09126	0,14059	0,07112
FCA	0,03875	0,05246	0,41538	0,10524	0,18538	0,80590	0,13519	0,57459
SW1	0,02423	0,02925	0,11823	0,01899	0,06072	0,07787	0,12467	0,08845
SW2	0,02494	0,02777	0,11823	0,01434	0,02849	0,04237	0,12491	0,09481

Tabla 11: Valor cuadrático medio de los errores de los registros con neuropatías crónicas.

SUBAGUDOS								
	Amplitud	Área	Fases	Irreg	Spk_Area	Spk_Dur	Turns_Def	Turns_Exc
EA	0,05727	0,08043	0,19871	0,06225	0,15266	0,35609	0,18240	0,14135
MA	0,04699	0,06948	0,15560	0,03620	0,09527	0,14664	0,10774	0,15895
FCA	0,05915	0,08171	0,27790	0,06343	0,36436	1,10329	0,16278	0,30555
SW1	0,04187	0,07314	0,15626	0,05237	0,16079	0,58037	0,12202	0,14367
SW2	0,04310	0,07138	0,13896	0,03605	0,15888	0,57975	0,10639	0,16939

Tabla 12: Valor cuadrático medio de los errores de los registros con neuropatías subagudas.

NORMALES								
	Amplitud	Área	Fases	Irreg	Spk_Area	Spk_Dur	Turns_Def	Turns_Exc
EA	0,05837	0,09361	0,21446	0,04637	0,08929	0,22768	0,10505	0,12512
MA	0,04554	0,07959	0,06754	0,01738	0,08111	0,11462	0,11064	0,12837
FCA	0,06185	0,09994	0,17950	0,03013	0,10012	0,31382	0,10177	0,28560
SW1	0,04414	0,07310	0,06754	0,03520	0,04953	0,03494	0,08245	0,08245
SW2	0,04530	0,07558	0,05918	0,02103	0,05333	0,04099	0,11064	0,28291

Tabla 13: Valor cuadrático medio de los errores de los registros normales.

La Prueba T pareada se ha realizado con un p-valor ≤ 0.05 , suponiendo que los datos siguen distribuciones normales, simétricas y con varianzas distintas. Se ha tomado como hipótesis nula la igualdad de medias en los errores cometidos por los métodos.

Para evitar un doble efecto en los errores en los turns, las variables *Turns_Def* y *Turns_Exc* se han agrupado en una sola, *Turns_accu*.

Este estudio cuenta con tres dimensiones: *parámetros*, *métodos* y *grupos de patología*. Por eso se ha optado por presentar proyecciones a pares. En las Tablas 14, 15 y 16 se recogen los porcentajes de los casos en los que hay evidencias de diferencias de medias. La última fila y columna muestran los promedios finales de los parámetros, los métodos o los grupos.

PARÁMETROS vs MÉTODOS						
	EA	MA	FCA	SW1	SW2	Total general
Amplitud	100,00%	100,00%	75,00%	100,00%	25,00%	80,00%
Área	50,00%	75,00%	50,00%	100,00%	100,00%	75,00%
Fases	50,00%	100,00%	0,00%	100,00%	100,00%	70,00%
Turns_Accu	25,00%	75,00%	37,50%	87,50%	75,00%	60,00%
Irregularidad	100,00%	100,00%	0,00%	100,00%	100,00%	80,00%
Spk_Dur	50,00%	100,00%	25,00%	100,00%	100,00%	75,00%
Spk_Area	50,00%	100,00%	25,00%	75,00%	75,00%	65,00%
Total general	60,71%	92,86%	30,36%	94,64%	82,14%	72,14%

Tabla 14: Resultados finales, Parámetros vs Métodos.

PARÁMETROS vs GRUPOS					
	Miopáticos	Crónicos	Subagudos	Normales	Total general
Amplitud	100,00%	80,00%	60,00%	80,00%	80,00%
Área	80,00%	100,00%	80,00%	40,00%	75,00%
Fases	80,00%	60,00%	80,00%	60,00%	70,00%
Turns_Accu	70,00%	50,00%	60,00%	60,00%	60,00%
Irregularidad	80,00%	80,00%	80,00%	80,00%	80,00%
Spk_Dur	60,00%	60,00%	80,00%	100,00%	75,00%
Spk_Area	80,00%	100,00%	80,00%	40,00%	75,00%
Total general	70,00%	72,86%	71,43%	74,29%	72,14%

Tabla 15: Resultados finales, Parámetros vs Grupos.

MÉTODOS vs GRUPOS						
	EA	MA	FCA	SW1	SW2	Total general
Miopáticos	57,14%	100,00%	21,43%	85,71%	85,71%	70,00%
Crónicos	57,14%	92,86%	35,71%	100,00%	78,57%	72,86%
Subagudos	64,29%	100,00%	14,29%	92,86%	85,71%	71,43%
Normales	64,29%	78,57%	50,00%	100,00%	78,57%	74,29%
Total general	60,71%	92,86%	30,36%	94,64%	82,14%	72,14%

Tabla 16: Resultados finales, Métodos vs Grupos.

DISCUSIÓN

Desde un contexto global se puede afirmar que los resultados obtenidos son satisfactorios. El método de extracción de la forma de onda del conjunto de potenciales de un PAUM que mejores resultados muestra, es el Sliding Window. Sin embargo, hay varios aspectos concretos que merecen más atención:

- a) Los métodos que menos errores cometen para las distintas familias patológicas son, por orden: el Sliding Window basado en la media (SW1), la mediana (MA) y el Sliding Window basado en la mediana (SW2). El SW1 (ver Tabla 16) ha sido mejor en los registros crónicos y en los normales. El siguiente método que ofrece los mejores resultados es MA que ha cometido menos errores con los miopáticos y los subagudos.

El SW2 no ha obtenido los mejores resultados en ninguna patología a pesar de ser el método que, por su valor cuadrático medio, más veces se ha utilizado como método de referencia en el análisis estadístico. En los casos en los que no ha sido como método a comparar, su valor cuadrático medio es muy próximo al utilizado como referencia. La gran dispersión de los datos que ofrece este método ha condicionado sus resultados, el caso más llamativo es el de la amplitud que ha obtenido un resultado muy alejado del deseado (ver Tabla 14).

- b) Para los registros normales, los de pacientes sanos, los mejores métodos son SW1 y SW2 lo que refuerza el artículo de 2016 de Malanda [7], que obtuvo resultados similares con otros registros de pacientes sanos.
- c) La construcción del Gold Standard puede ser una de las principales fuentes de error. La tendencia del técnico de seleccionar las trazas de mayor amplitud entre las más similares condiciona algunos parámetros como la amplitud o el área. En el caso del área también se ve críticamente afectado por la posición de los marcadores (ver Figuras 17 y 18). Si nos fijamos en los resultados de este parámetro en los registros Normales se observa un error del 40%, ver Tabla 15, que puede ser debido a la poca precisión al situar los marcadores temporales debido que en este tipo de registro abundan PAUMs con inicios y retornos asintóticos, en la Figura 16 se muestra un ejemplo.
- d) Como se ha comentado en la sección anterior los parámetros *amplitud*, *área* y *spike_area* están fuertemente correlacionados en todas las familias, lo que provoca que los errores que se cometen a lo largo de los distintos métodos varíen de la misma forma, Figura 19.
- e) El algoritmo FCA ha obtenido los peores resultados (ver Tabla 16) porque utiliza los cinco potenciales que más se parecen entre ellos, que pueden ser, o no, los que más se parezcan al Gold Standard. En la Figura 22 se muestra un ejemplo donde se presenta un PAUM del grupo de las neuropatías crónicas en el que se ve la diferencia entre la forma de onda obtenida por este método y el Gold Standard, además se indican los valores de las figuras de mérito obtenidas.
- f) Otra fuente de error importante en todos los métodos, aunque afecta en diferente forma y medida a cada uno, es el desalineamiento. En el ejemplo de la Figura 22 se muestra un registro con todos sus potenciales y los el PAUM que han obtenido los distintos métodos. Se observa como el Gold Standard se ve afectado con un suavizado

de los picos, que produce un aumento de los errores en amplitud, área, detección de turns y en el coeficiente de irregularidad.

- g) Existe cierto nivel de correlación entre el coeficiente de irregularidad y las fases y los turns. En la Figura 22 se muestra como varían los errores de estos parámetros.

El método que peor detecta los *turns* es EA que genera formas de ondas fuertemente condicionadas por todos los potenciales. La precisión en el reconocimiento de los turns se ve afectada principalmente por la detección de turns falso (turns por exceso, los obtenidos que no existen en el Gold Standard). En la Figura 21 se muestran señalados con flechas rojas los turns detectados por exceso por el método EA. En otros casos la detección de turns falsos puede ser debido a una forma de onda con un trazado muy irregular o con ruido.

- h) El parámetro EA, media aritmética, no ha tenido buenos resultados porque se basa en un enfoque muestra a muestra, y por lo tanto, es más vulnerable a contaminaciones de PAUMs secundarios, que provienen de unidades motoras diferentes. En la Figura 21 se muestra un ejemplo de como el PAUM del método EA se ve afectado por otros potenciales generando un trazado más sinuoso mientras que el SW1, que se apoya en las ventanas adyacentes, obtiene una forma de onda más similar al Gold Standard.

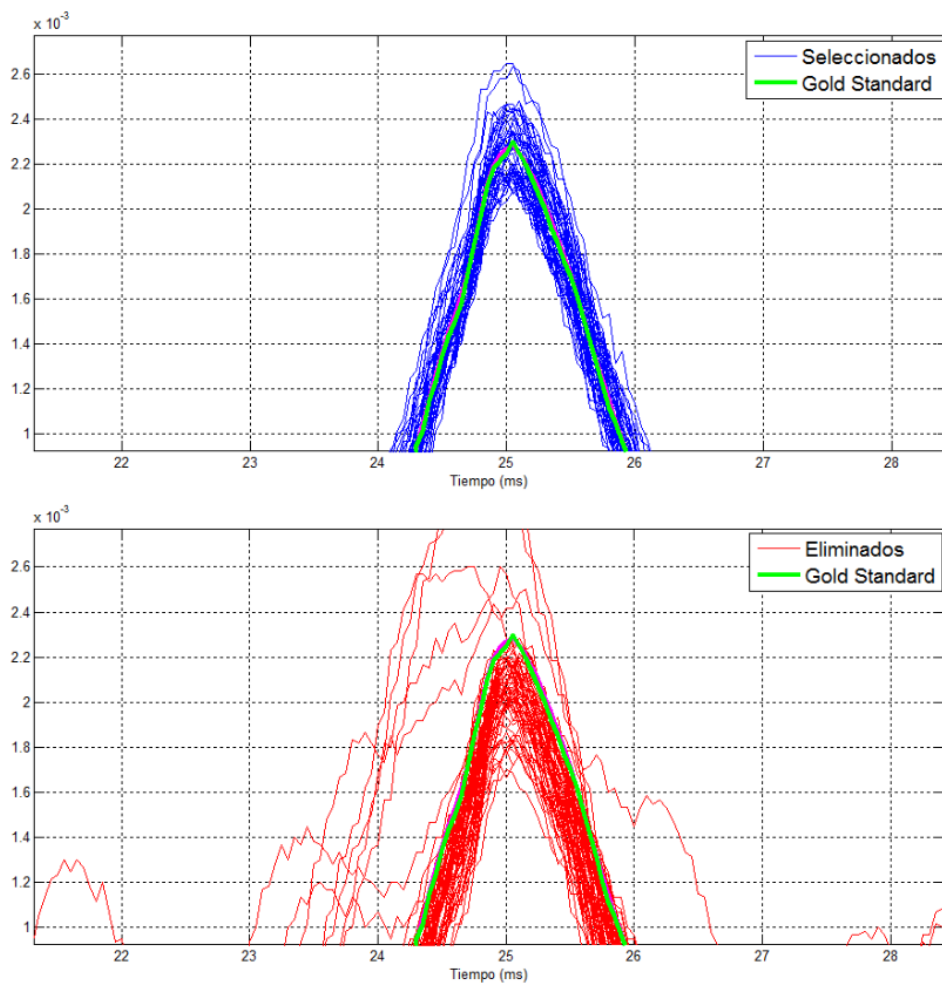


Figura 21: En la construcción del Gold Standard se tiende a seleccionar los potenciales de mayor amplitud.

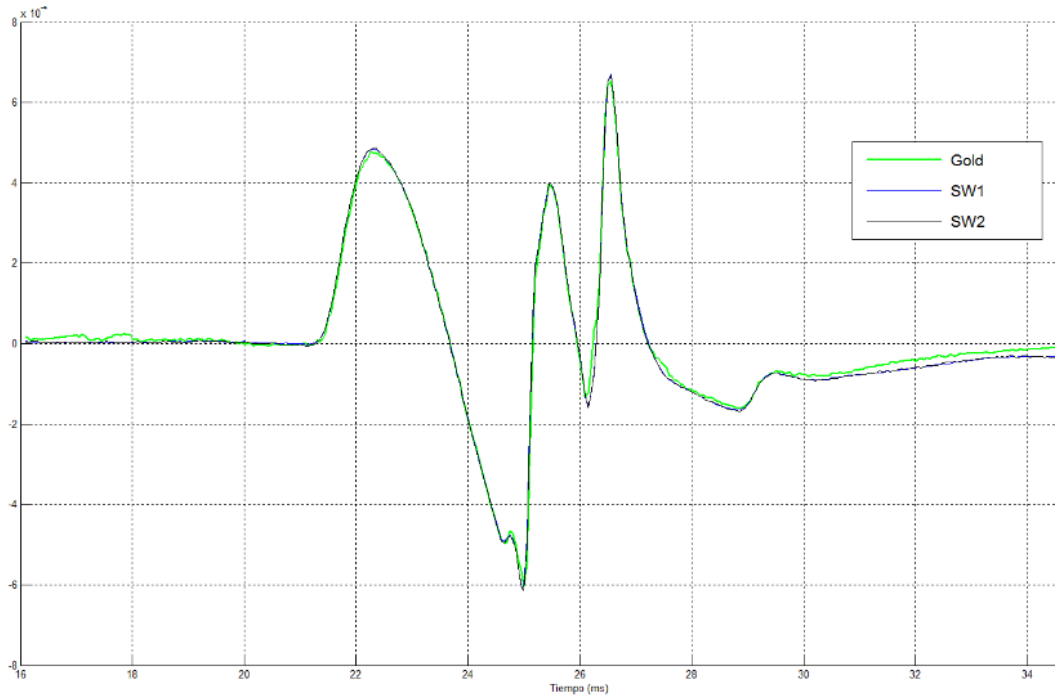


Figura 22: Unos marcadores, de inicio y fin mal situados pueden hacer que algún método cometa errores en el área. Notense, en los extremos de la señal, las diferencias entre las trazas de SW1 y SW2 con el Gold Standard.

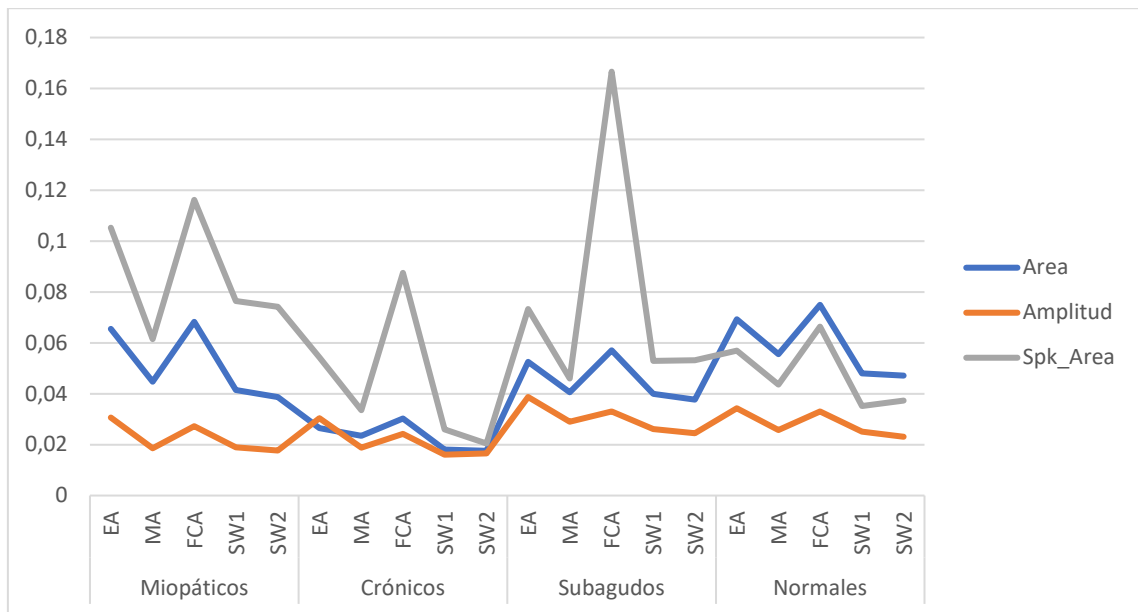


Figura 23: Variación de los errores de amplitud, área y spike_area a lo largo de los métodos y familias

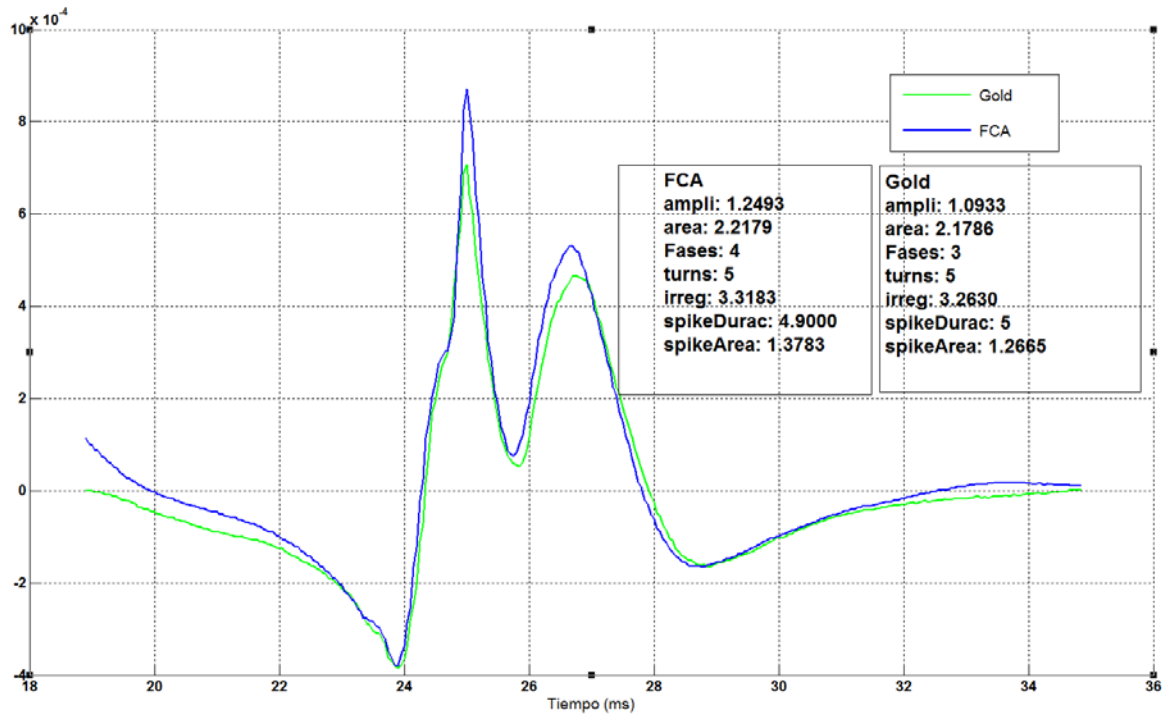


Figura 24: Ejemplo Five Closest Averaging

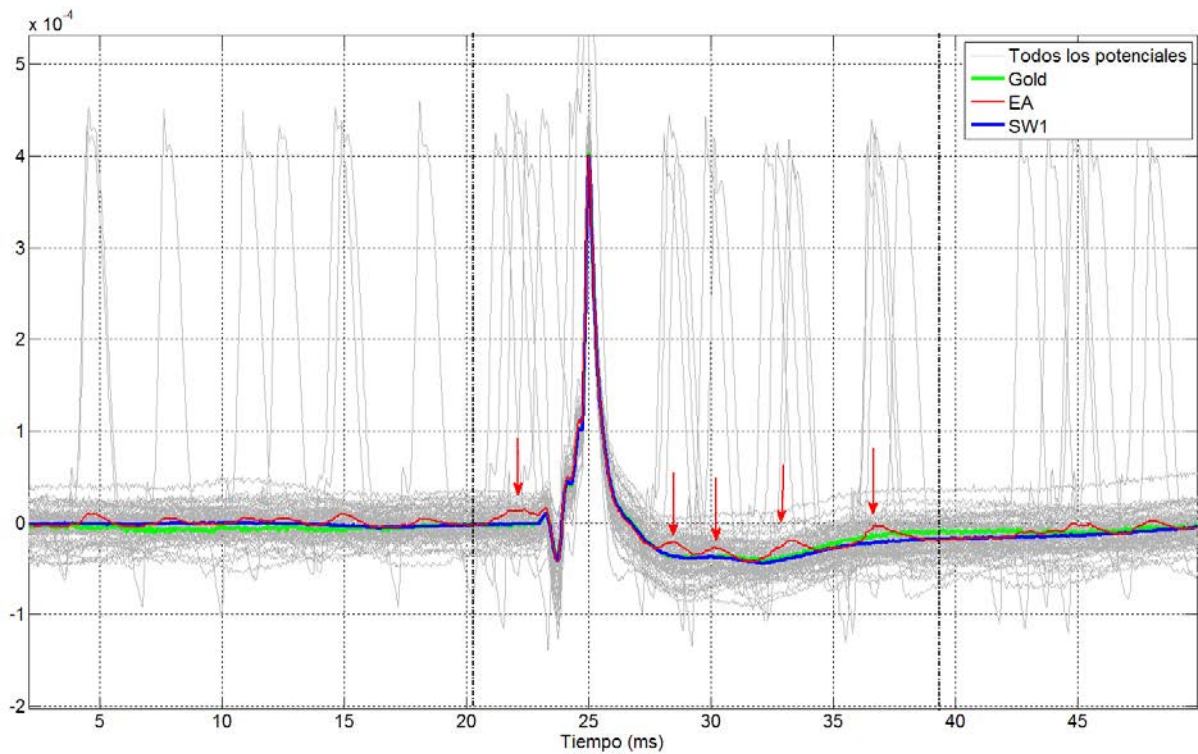


Figura 25: Ejemplo método EA.

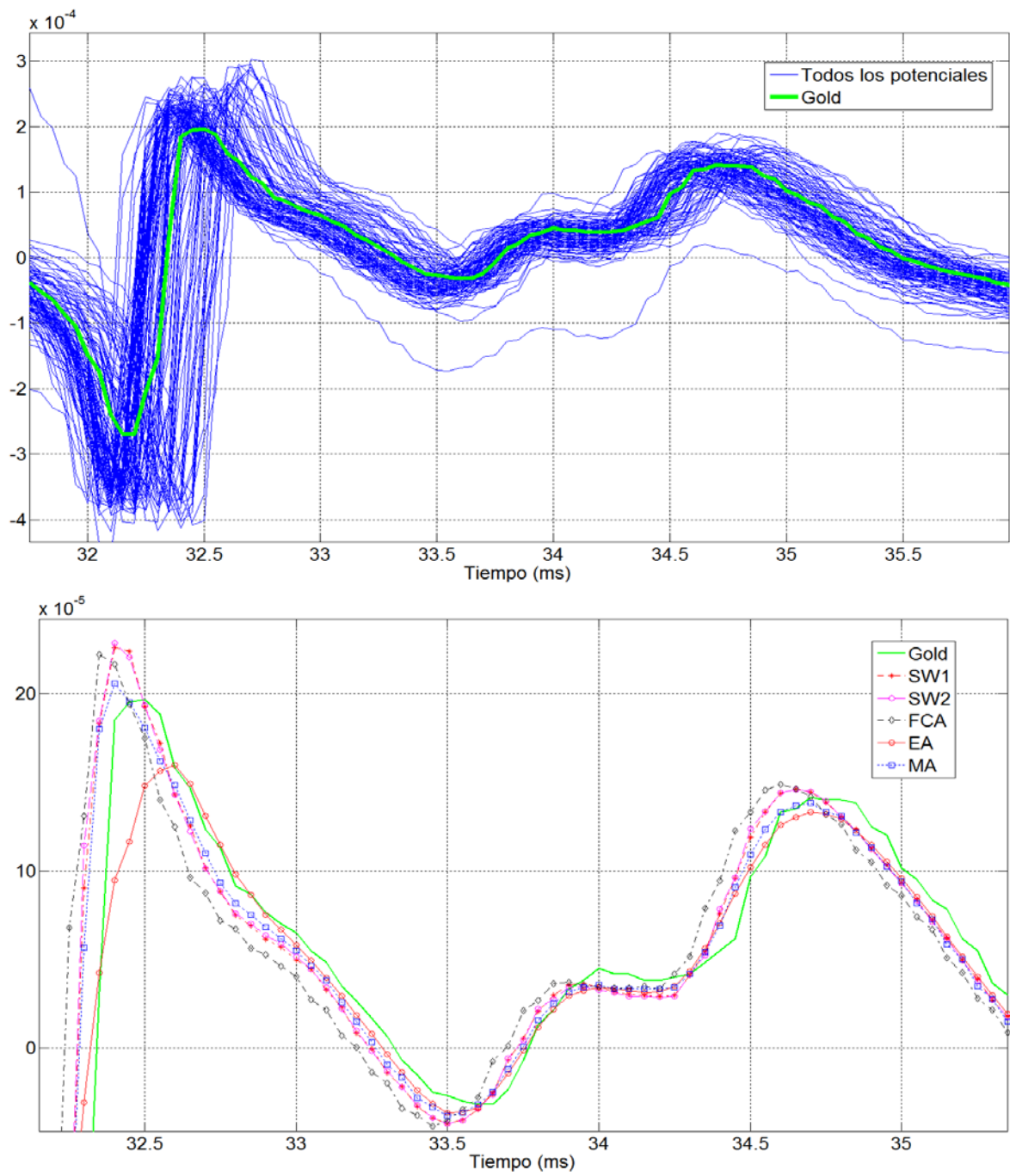


Figura 26: Desalineamiento de los potenciales en los registros.

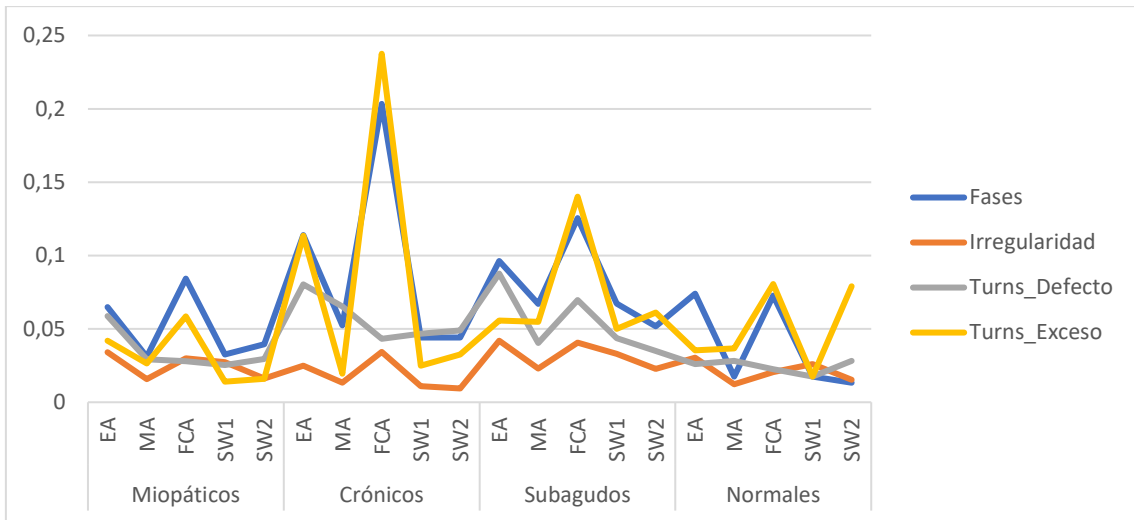


Figura 27: Variación de los errores de los parámetros fases, turns y coeficiente de irregularidad

CONCLUSIONES

En este estudio se ha encontrado que el algoritmo basado en una selección de segmentos por ventanas deslizantes, Sliding Window (SW), es significativamente mejor que los más utilizados en la extracción de curvas representativas de trenes de PAUMs: la media (EA), la mediana (MA) y el Five Closest Average (FCA) diseñado por Nendedkar y Sanders [6].

El método SW obtiene menos errores en registros de pacientes sanos, como ya demostró Malanda [7] y que se confirma de nuevo en este estudio. Además, obtiene muy buenos resultados en todas las patologías, destacando en las neuropatías crónicas que suponen una dificultad añadida por la complejidad de sus potenciales de acción.

Desde el punto de vista de las figuras de mérito, el método SW obtiene menos errores en todos los errores, con la excepción del spike_area que el mejor método ha sido MA.

En el caso de los Turns, tanto por exceso como por defecto, y fijándonos en las distribuciones de sus datos resultaría interesante realizar el análisis estadístico con herramientas no paramétricas como un test de Wilcoxon para muestras pareadas.

Después de analizar los errores de todas las señales se observa que la obtención de un buen Gold Standard es un aspecto importante ya que puede condicionar los resultados de los distintos métodos. Además, construir un buen Gold Standard y definir los instantes de inicio y fin, es una tarea complicada de realizar y que requiere de herramientas potentes, cómodas y versátiles como el PAUM_Analyzer.

Finalmente, esta herramienta aún tiene rango de mejora. Durante la eliminación de potenciales, añadir las opciones de deshacer y rehacer la eliminación aportaría una gran potencia a la aplicación. Por otro lado, puede darse el caso que se quieran conocer los parámetros del Gold Standard que se visualice en cada momento. Si se añade una opción que los calcule automáticamente dotaría la aplicación de una gran funcionalidad.

REFERENCIAS

- [1] Javier Rodriguez, Javier Navallas, Armando Malanda. *Teaching a master student how to model the electrical potentials produced by the muscle*. International Conference on Engineering Education ICEE-2010.
http://www.ineer.org/Events/ICEE2010/papers/W16A/Paper_1101_1279.pdf
- [2] Rafael Barea Navarro, *Electromiografía*, Universidad de Alcalá.
<http://www.bioingenieria.edu.ar/academica/catedras/bioingenieria2/archivos/apuntes/tema%205%20-%20electromiografia.pdf>
- [3] Ignacio Rodriguez-Carreño. *Cancellation of Baseline Fluctuation in Electromyographic Signals and Measurement of Motor Unit Action Potential Duration based on Wavelet Transforms*, Universidad Pública de Navarra, 2006.
- [4] E. Zalewska. *Evaluation of MUAP shape irregularity--a new concept of quantification*. 1995.
<https://www.ncbi.nlm.nih.gov/pubmed/7790018>
- [5] Albert Selva. *Miopatías inflamatorias. Dermatomiositis, polimiositis y miositis con cuerpos de inclusión*. 2008. Reumatol Clin.
<http://www.reumatologiaclinica.org/es/miopatias-inflamatorias-dermatomiositis-polimiositis-miositis/articulo/s1699258x08724641/>
- [6] Nandedkar, S.D. & Sanders, D.B. *Median averaging of electromyographic motor unit action potentials*. 1989. Med. Biol. Eng. Comput.
- [7] A. Malanda, I. Rodriguez-Carreño, J. Navallas, L. Gila. *Sliding window averaging for the extraction of representative waveforms from motor unit action potential trains*. 2016. Biomedical Signal Processing and Control.
<http://www.sciencedirect.com/science/article/pii/S1746809416000045>
- [8] Stålberg Erik. *Quantitative Motor Unit Potential Analysis*. 1996, *Journal of Clinical Neurophysiology*.
http://journals.lww.com/clinicalneurophys/Fulltext/1996/09000/Quantitative_Motor_Unit_Potential_Analysis.4.aspx
- [9] Zalewska Ewa. *Effectiveness of motor unit potentials classification using various parameters and indexes*. 2000. *Clinical Neurophysiology*,
<http://www.sciencedirect.com/science/article/pii/S1388245700003229>
- [10] A. Malanda, J. Navallas, J. Rodriguez-Falces, I. Rodriguez-Carreño. *Averaging methods for extracting representative waveforms from motor unit action potential trains*. 2015. *Journal of Electromyography and Kinesiology*.
<http://www.sciencedirect.com/science/article/pii/S1050641115000759>
- [11] Ewa Zalewska, Katarzyna Rowinska-Marcinska. *Quantitative motor unit potential shape analysis — application to real signals*. 1995. *Electroencephalography and Clinical Neurophysiology*.
-

[12] E. Stalberg, S. Andreassen, W. Trojaborg. *Quantitative analysis of individual motor unit potentials: a proposition for standardized terminology and criteria for measurement*. 1986. Journal of clinical neurophysiology
<https://www.ncbi.nlm.nih.gov/pubmed/3332279>

ANEXOS

Manual de Usuario

Guía Rápida

MUAP_Analyzer

Archivo

1APBSTEIN_Z03.paum.mat

Resumen de señales

Totales: 82
Actuales: 15 (18.29%)
Eliminada: 67 (81.7%)

M1= 21.05
M2= 31.15

Marcadores
 Eliminar
 Media
 Mediana
 Ver Marcadores

[_]
[_ _]

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46

Marcar todos

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46

15

1. Axes 1: Muestra las líneas actuales.
2. Axes 2: Muestra las líneas eliminadas.
3. Axes 3: Muestra las curvas media y mediana, eje sin interacciones.
4. Línea a eliminar del Axes 1, haciendo clic con el botón izquierdo. Se envía al Axes 2.
5. Línea resaltada en el Axes 1, haciendo clic con el botón derecho
6. Línea a recuperar del Axes 2, haciendo clic con el botón izquierdo. Se envía al Axes1.
7. Línea resaltada en el Axes 2, haciendo clic con el botón derecho
8. Modo:
 - a. Eliminar (por defecto): Permite las interacciones con las líneas.
 - b. Marcadores: Activa los marcadores (núm. 9) y su interacción.
9. Marcadores:
 - a. En rojo, se mueven haciendo clic y arrastrando, solo Axes 1.
 - b. En granate, fijados y sin interacción.
10. Cuadros de texto que se actualizan con los marcadores.
11. Mostrar u ocultar las curvas las curvas media y/o mediana en al Axes 1.
12. Hacer zoom rápido en los Axes 1 y 2, solo cuando existen marcadores.
13. Resetear zoom.
14. Tabla con los índices de las señales. Haciendo clic en un checkbox elimina o recupera la línea.
15. Eliminar o recuperar todas las líneas.
16. Panel a modo resumen de las señales. Se actualiza con cada interacción.
17. Archivo, funciones: guardar, resetear y cargar registro.

GUIA COMPLETA

Introducción

La finalidad de este software es la gestión de las señales y registros de una base de datos de electromiografía. El usuario puede trabajar de manera fluida con todos los registros quitando y añadiendo señales de un mismo registro, almacenar o descartar los resultados y cambiar de registro, todo ello de una forma amigable, dinámica y fluida.

Además de poder trabajar con bases de datos, este software también ofrece la posibilidad de trabajar con ficheros o registros importados manualmente.

Para ello se hace uso de las funciones de interfaces gráficas disponibles en cualquier versión de Matlab.

Mecanismo para acceder al sistema

Para lanzar la aplicación, es necesario situarse en la carpeta raíz de la aplicación y ejecutar el comando *"PAUM_Analyzer"* como se muestra en la Figura 1.

Elementos visuales

En esta sección se resumen los elementos visuales de este software ordenados según el habitual orden de aparición:

- **Pantalla de selección de registro:** Como indica su nombre en la esquina superior izquierda, en este cuadro de dialogo se pide al usuario que seleccione un fichero `'mat'`, Figura 2.
- **Ventana principal:** En la Figura 3, se muestra un ejemplo que contiene la mayoría de los elementos visuales. Ventanas donde se muestran las señales, barras de tareas, menús, distintos tipos de botones que activan funciones y tablas que recogen información.
- **Marcadores de anotación:** Los marcadores rojos se pueden mover haciendo clic con el boton izquierdo del ratón y arrastrando, el parte superior, en color celeste, se indica el indice del marcador, Figura 4. El primero, es el que indica el inicio del PAUM y el segundo el fin.
- **Menu:** Menu que permite cargar un registro, gestionarlo o cargar una base de datos, Figura 5.
- **Barra de heramientas:** Dispone de tres opciones, el funcionamiento es el normal en Matlab, pan, zoom in y zoom out, Figura 6.
- **Botones:** Arriba RadioButtons, con distintos colores CheckButtons y los PushButtons debajo y arriba a la derecha, Figura 7.
- **Panel de resumen:** Panel sin interacción meramente informativo que se actualiza según las señales que se muestren en cada momento, Figura 8.

- Dialogo de Workspace: Aparece cuando se quiere cargar un registro desde el Workspace, si la carga se realiza con éxito, aparece un cuadro de confirmación. Tal y como se muestra en la Figura 9.
- Tabla de señales: Tabla que permite activas y desactivar señales, las muestra en tiempo real y se actualiza con cada modificación, Figura 10.
- Modo gestion de bases de datos: Permite trabajar con las bases de datos de forma dinámica, al elegir esta opción se añade un panel con opciones para cargar y guardar los registros de forma más sencilla.

Funcionamiento

El modo registro único modo es el más común y funciona como base para los demás modos, es decir, que todas las funciones están disponibles en los demás modos a los que se les han añadido diversas funciones.

Modo Registro Único

Este software se inicia por defecto en este modo. Al ejecutar PAUM ANALYZER, se pide seleccionar un registro. La secuencia de las acciones es la que reflejan la Figura 1, la Figura 2 y la Figura 3.

Las funciones principales con las siguientes:

Función marcadores

Se activa haciendo clic en el Radiobutton *marcadores*. Al activar esta función aparecerán dos barras verticales de color rojo. Estas barras se pueden desplazar horizontalmente a lo largo del eje temporal. Al hacer clic en cada una de ellas, se activará en su parte superior una etiqueta de color azul celeste que indica que marcador es. El primer marcador actualizar de forma dinámica el indicador correspondiente situados a la derecha.

Como se ilustra en la Figura 14, esta función solo está activada en el primer eje, el que está situado arriba y a la derecha.

Función eliminar

Esta función nunca se ejecuta a la vez que la función marcadores, por lo que los marcadores de anotación temporal, estarán desactivados, pero se pueden visualizar al pulsar el botón "Ver Marcadores", estos aparecerán en color granate y en todos los ejes. Se pueden mostrar u ocultar.

En esta función entra en juego el ratón y sus botones izquierdo y derecho, cuyas funciones son diferentes en el eje superior y en el inferior izquierdo. El eje inferior derecho no tiene ninguna función, ya que está a modo de resumen.

En el eje superior, se visualiza el tren de pulsos final, el que se pretende anotar. Para eliminar las trazas no deseadas, basta con identificar dicha traza y hacer clic con el botón izquierdo. Automáticamente desaparecerá de este eje y se añadirá al eje

inferior. En la Figura 15 se muestra un ejemplo con una línea que acaba de ser eliminada.

En el caso de que no se esté seguro acerca de si eliminar una traza, el botón derecho del ratón permite resaltar en color amarillo la línea pulsada y con un grosor mayor que el resto.

En el eje inferior, se muestran en rojo las señales descartadas. De forma análoga al eje superior, el botón izquierdo permite recuperar la línea seleccionada, que es añadida de nuevo al eje superior. El botón derecho del ratón permite resaltar la línea pulsada, en color gris y con un grosor de línea mayor. En la Figura 16 aparece un ejemplo con líneas resaltadas en ambos ejes.

Con cada cambio se actualizan los siguientes elementos:

- Tabla de la izquierda con el índice de las trazas
- El panel de resumen
- Las curvas media y mediana en los tres ejes.

Otras funciones

- Líneas media y mediana: Se pueden activar y desactivar, la última que se habilite se situará encima del resto de líneas. Solo se pueden ocultar las líneas media y mediana del eje superior.
- Ajuste de ventana: Los botones *pushbuttons* situados en el centro de la interfaz gráfica, Figura 7, permiten ajustar los límites de los ejes. El botón superior hace zoom ajustando el eje horizontal a los marcadores temporales, si los hay, y el eje vertical al máximo y mínimo del conjunto de señales que se muestran en el momento de pulsar el botón, Figura 17.

El *pushbutton* inferior, restablece los límites de los ejes, mostrando las señales enteras.

- Ver marcadores: Esta opción solo se puede utilizar cuando la aplicación se encuentra en el modo eliminar, permite mostrar en color granate y ocultar los marcadores temporales en todos los ejes.
- Tabla de señales: Desde la tabla de señales, Figura 10, se pueden activar o desactivar las trazas que se deseen. Se pueden activar o desactivar todas las trazas, muy útil si se desea resetear sin recurrir a la función Reset y perder los marcadores.

Además, esta función es la única forma de eliminar o recuperar las trazas mientras se está ejecutando la función de anotar marcadores.

Modo Base De Datos

Muy útil cuando se quiere anotar una gran cantidad de registros sin tener que buscar uno a uno cada uno de los registros. Este modo busca todos los archivos '.mat' de la carpeta indicada que no sean archivos de resultados, es decir, que empiecen por 'Res_'.

Este modo se ejecuta desde la barra de herramientas como indica la Figura 5.

Las funciones para gestionar una base de datos son:

- Cambiar carpeta: Abre una ventana y solicita que se seleccione una nueva carpeta. No se guardarán los cambios del registro actual, Figura 11
- Cargar selección: Se puede cargar el registro que se haya seleccionado previamente en la lista. No se guardarán los cambios del registro actual.
- Siguiente: carga el siguiente registro sin guardar los cambios en el registro actual.
- Guardar y siguiente: Guarda los cambios en el registro actual y carga el siguiente registro.

Modo Variable

En el caso de que se quiera anotar un registro que no esté en un fichero '.mat', se puede cargar un registro, en forma de variable, desde el Workspace de Matlab.

Para cargar una variable, desde la barra de herramientas, hacer clic en Archivo> Cargar>Cargar desde el WS.

Aparecerá un dialogo, Figura 9, que recoge todas las variables del Workspace, una vez seleccionada la variable deseada, aparecerá un mensaje que confirma el éxito de la carga.

Opción Frecuencia de Muestreo

Si se carga una variable o si el archivo cargado no incluye una frecuencia de muestreo, se ofrece la posibilidad de introducirla manualmente desde la barra de herramientas, como indica la Figura 18.

Cargar

Para cargar un nuevo fichero '.mat' desde cualquier directorio. No guarda los cambios del registro actual. Si se desea guardar hay que pulsar la opción de guardar en la barra de herramientas.

Reset

Función alojada en el menú Archivo de la barra de herramientas. Elimina los cambios en los marcadores temporales, señales eliminadas, panel de resumen, tabla de señales, curvas media y mediana y retorna a los últimos cambios guardados o bien a cero si no los hubiera.

Guardar

Guarda el archivo de resultados sobrescribiéndolo en caso de existir. Para guardar es necesario haber introducido los dos marcadores temporales, en caso contrario surgirá un mensaje de error avisando del mismo. En caso de que el registro actual haya sido cargado como variable desde el Workspace, se solicitará al usuario, mediante un cuadro de dialogo, que introduzca un nombre para poder almacenarlo.

Salir y Guardar

Al cerrar la aplicación siempre aparecerá el cuadro de dialogo de la Figura 19.

Listado de Imágenes

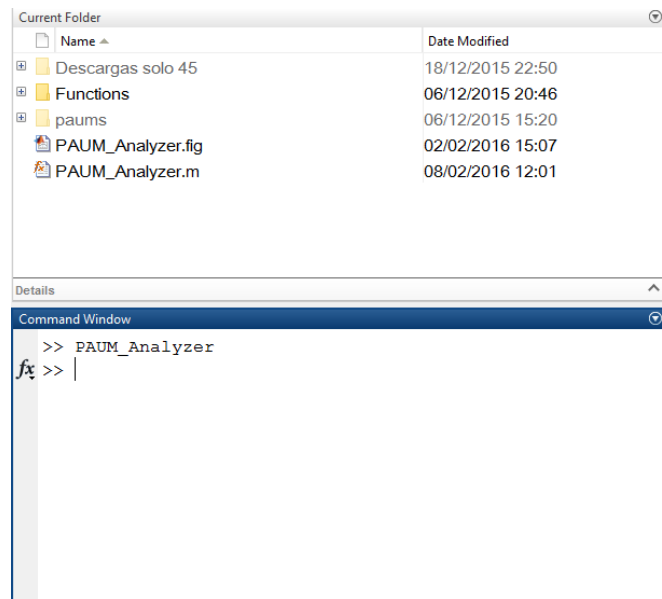


Figura 1: Lanzar la aplicación

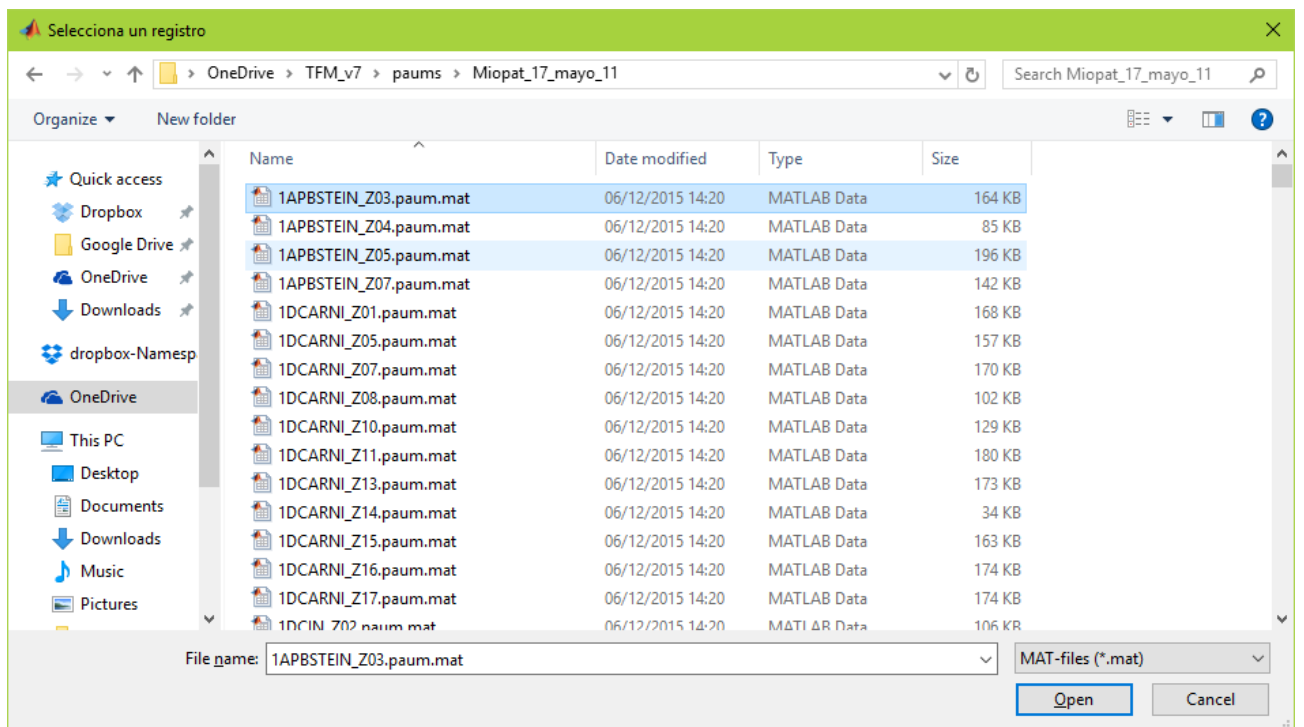


Figura 2: Seleccionar registro



Figura 3: Ventana principal

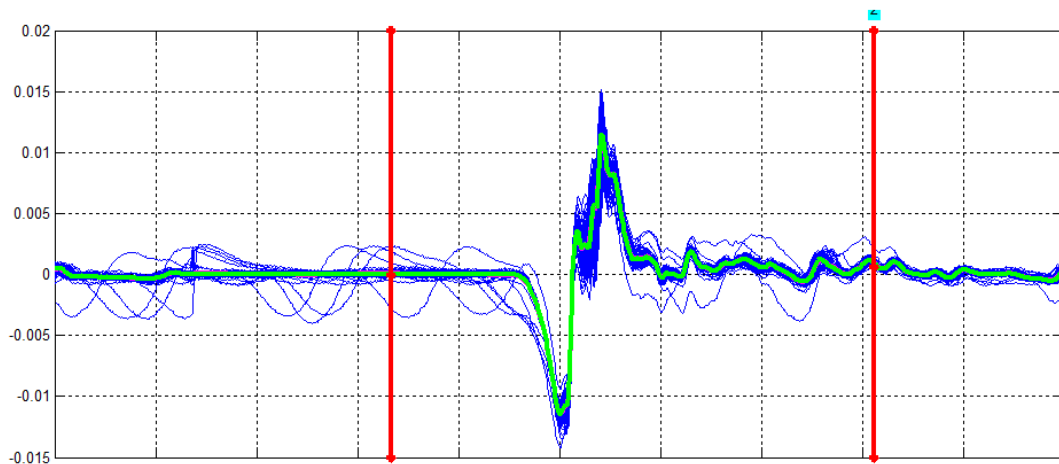


Figura 4: Marcadores para anotar el inicio y el fin del PAUM

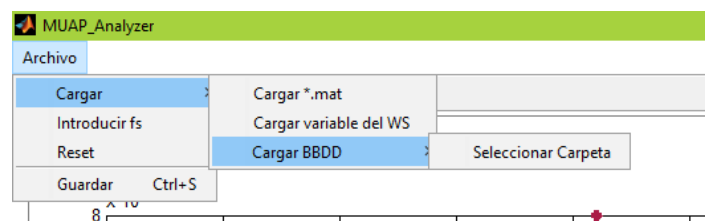


Figura 5: Menú con sus opciones desplegadas.

Archivo

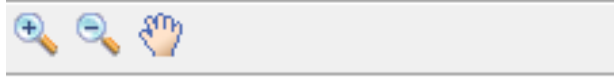


Figura 6: Barra de herramientas

<input type="radio"/> Marcadores	M1= 21.05
<input checked="" type="radio"/> Eliminar	M2= 31.15
<input checked="" type="checkbox"/> Media	<input checked="" type="checkbox"/> Ver Marcadores
<input checked="" type="checkbox"/> Mediana	
[_]	
[____]	

Figura 7: Botones, diferentes funciones.

Resumen de señales	
Totales:	82
Actuales:	14 (17.07%)
Eliminadas	68 (82.9%)

Figura 8: Panel de resumen

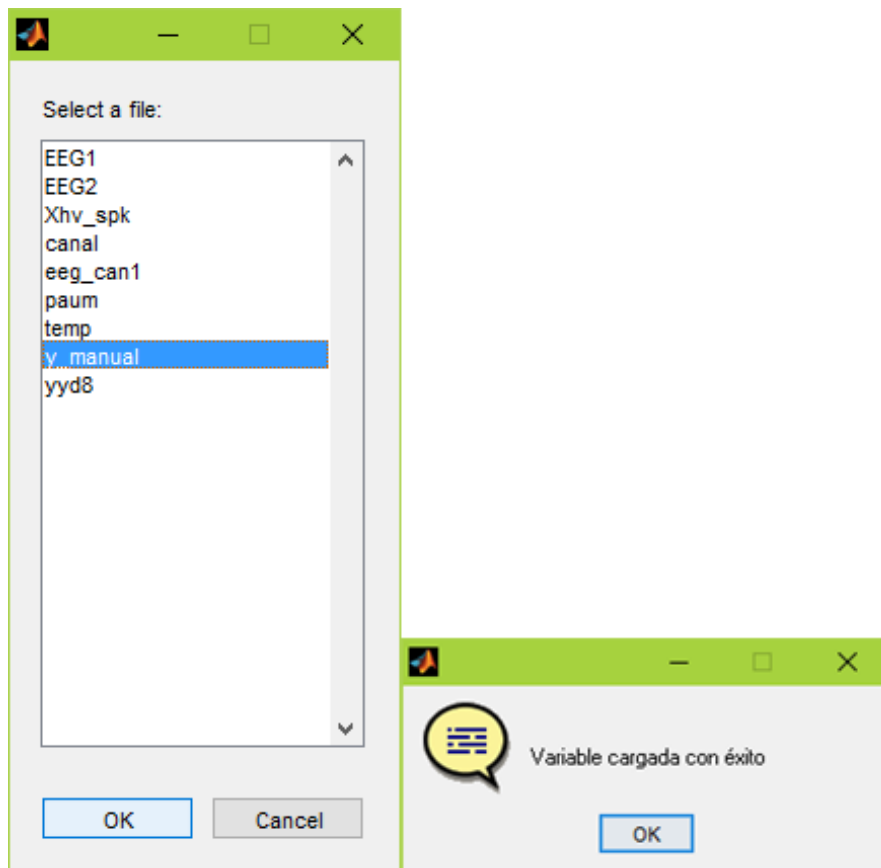


Figura 9: Cargar variable desde el Workspace

15	<input type="checkbox"/>
16	<input type="checkbox"/>
17	<input type="checkbox"/>
18	<input type="checkbox"/>
19	<input type="checkbox"/>
20	<input type="checkbox"/>
21	<input type="checkbox"/>
22	<input type="checkbox"/>
23	<input type="checkbox"/>
24	<input type="checkbox"/>
25	<input type="checkbox"/>
26	<input type="checkbox"/>
27	<input type="checkbox"/>
28	<input type="checkbox"/>
29	<input type="checkbox"/>
30	<input type="checkbox"/>
31	<input type="checkbox"/>
32	<input type="checkbox"/>
33	<input type="checkbox"/>
34	<input type="checkbox"/>
35	<input type="checkbox"/>
36	<input type="checkbox"/>
37	<input type="checkbox"/>
38	<input type="checkbox"/>
39	<input type="checkbox"/>
40	<input type="checkbox"/>
41	<input type="checkbox"/>
42	<input type="checkbox"/>
43	<input type="checkbox"/>
44	<input type="checkbox"/>

Marcar todos

Figura 10: Tabla de líneas

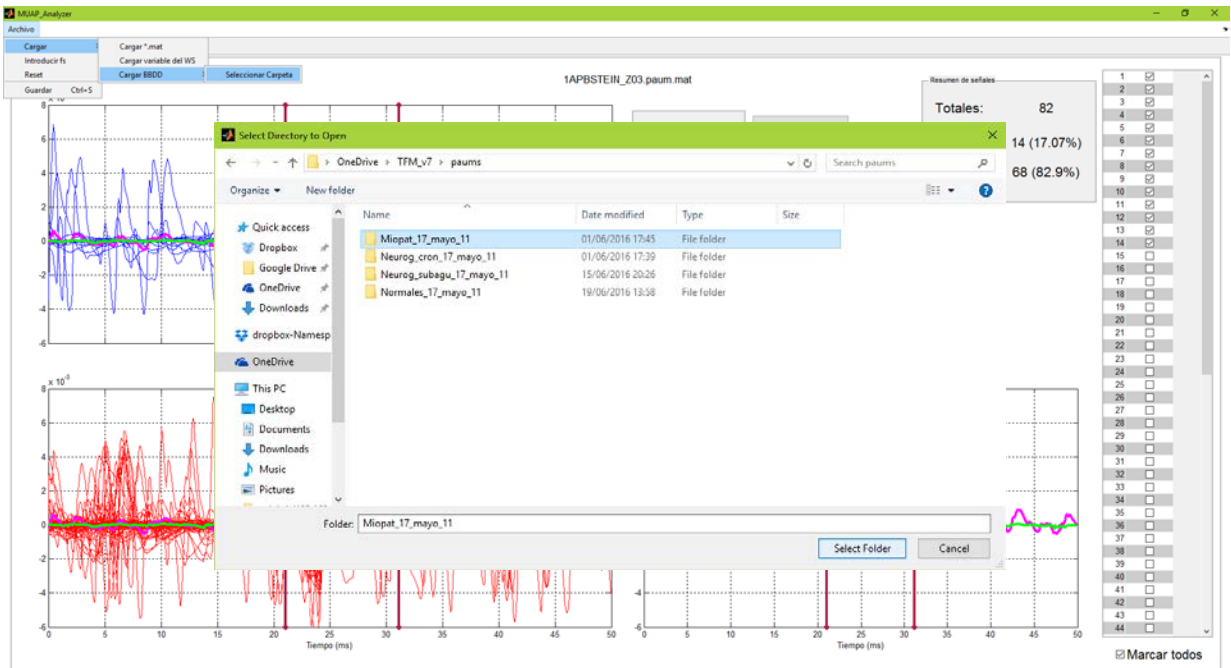


Figura 11: Modo BBDD, elegir carpeta

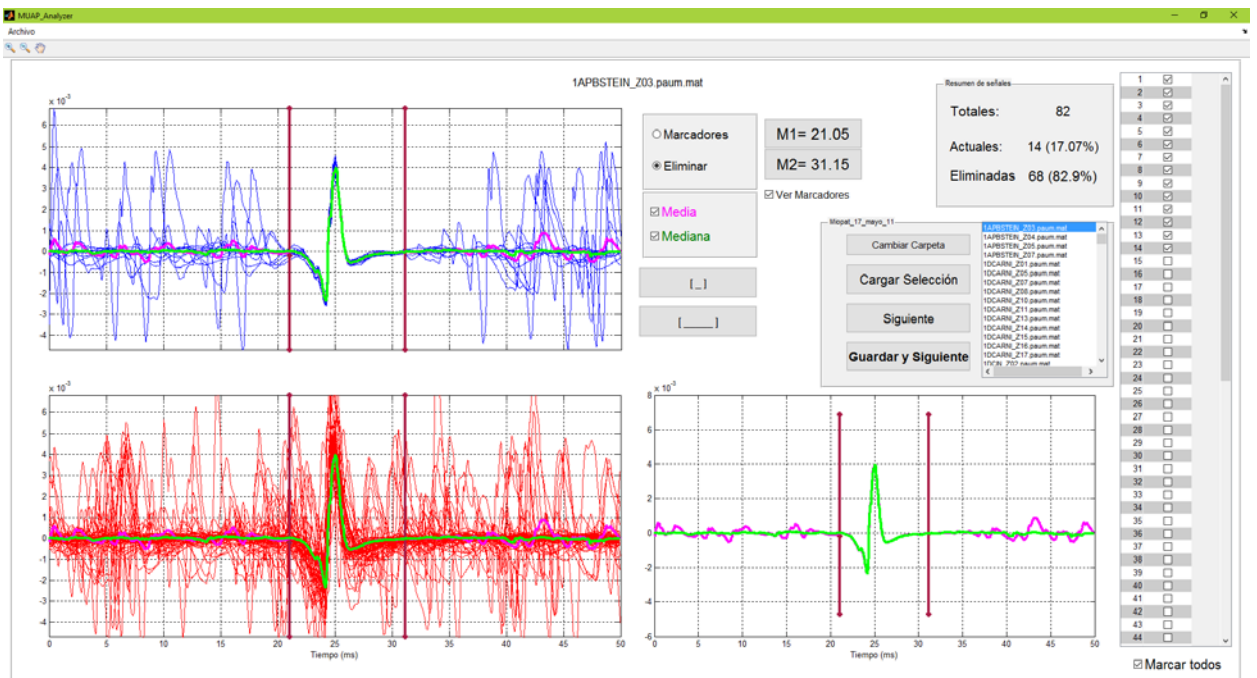


Figura 12: Pantalla principal en el modo BBDD

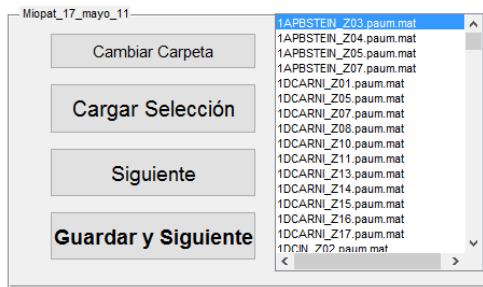


Figura 13: Panel de gestión de BBDD

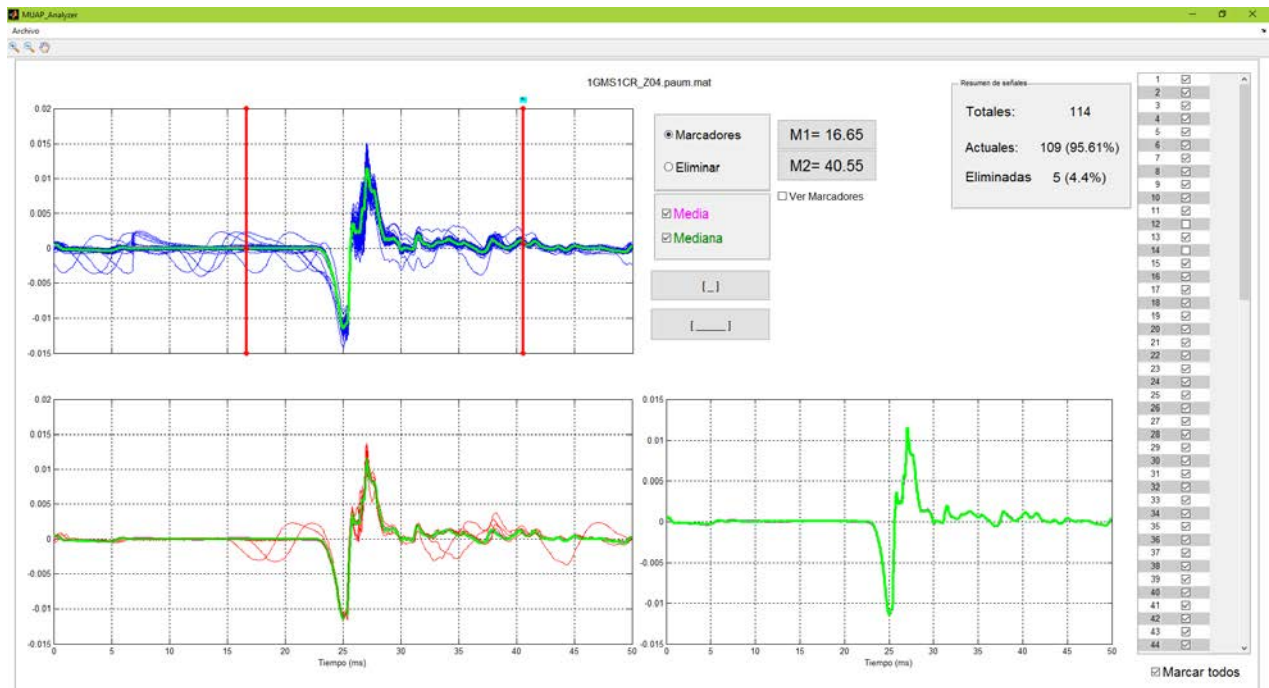


Figura 14: Modo Marcadores

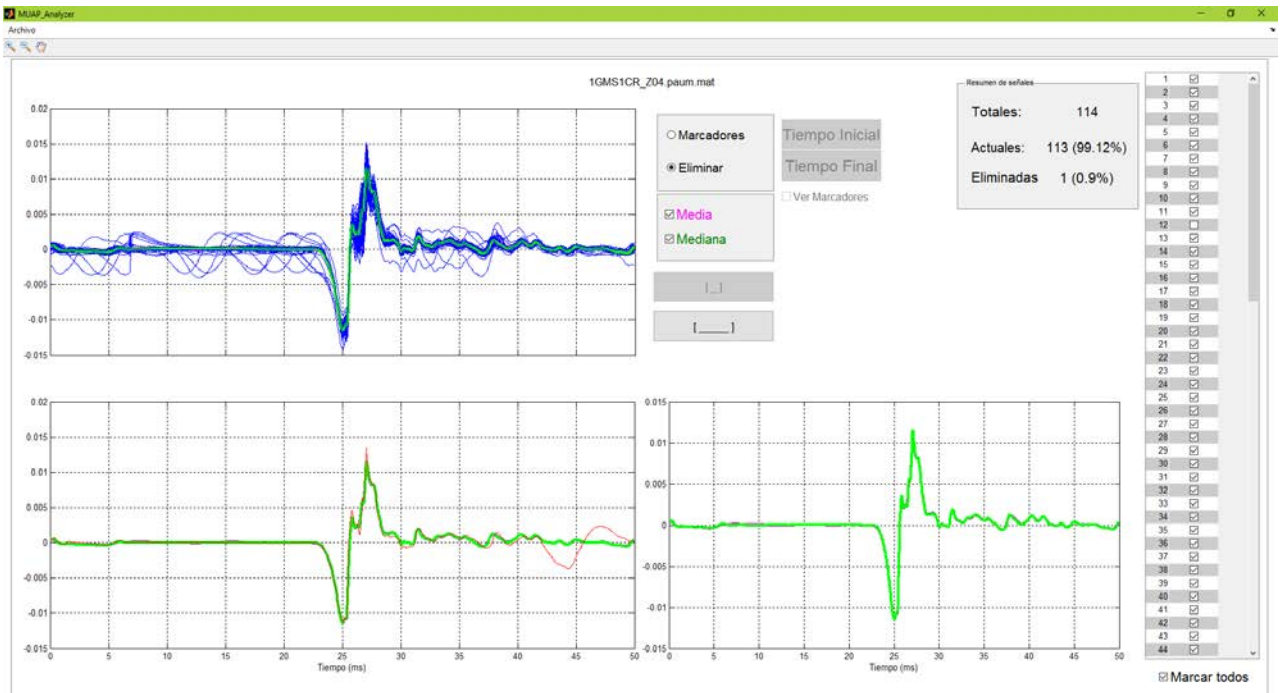


Figura 15: Línea eliminada



Figura 16: Líneas resaltadas

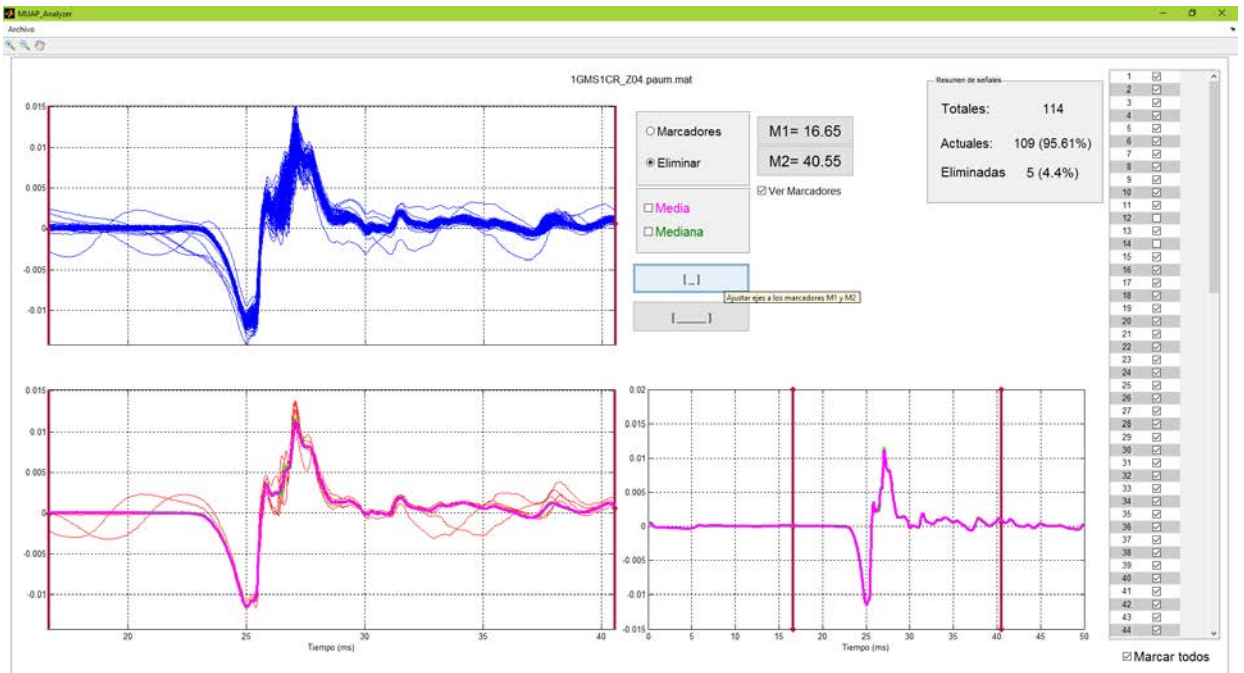


Figura 17: Zoom automático

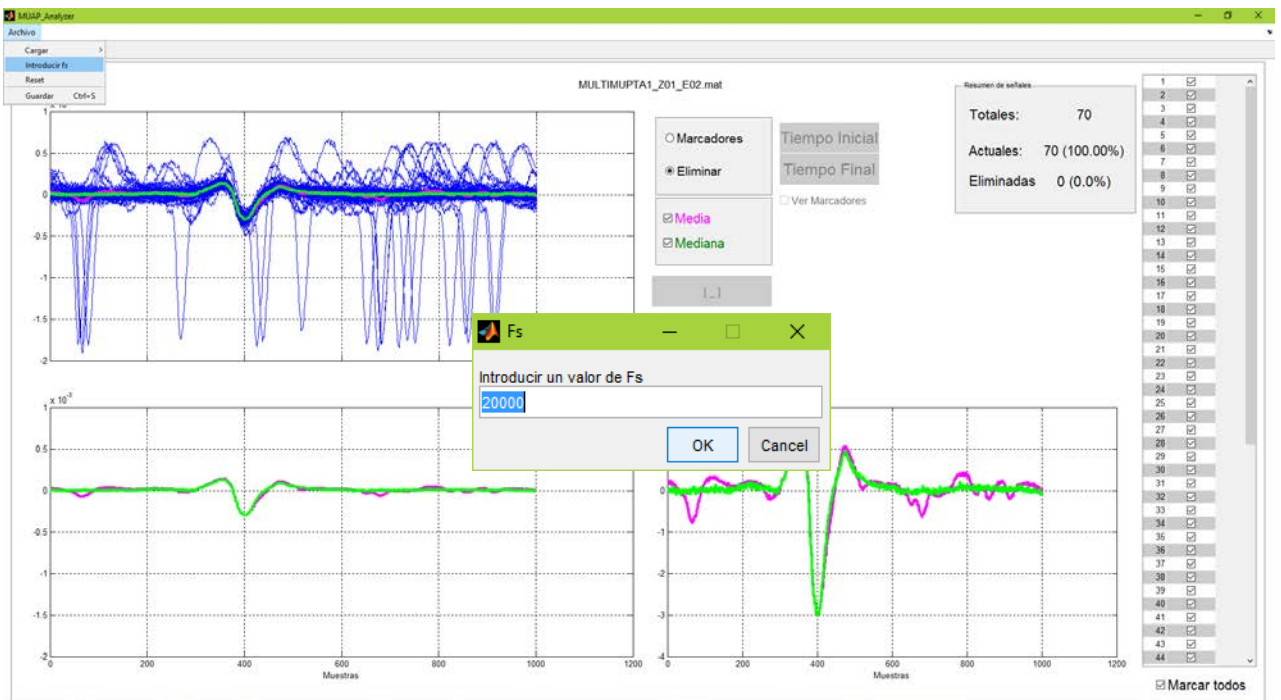


Figura 18: Introducir frecuencia de muestreo

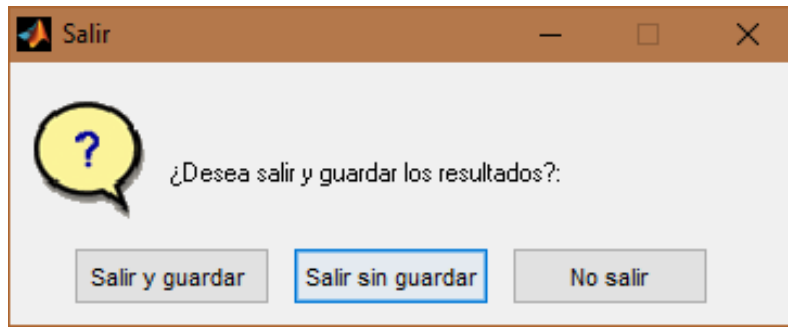


Figura 19: Diálogo de salida

MANUAL DEL PROGRAMADOR

1. Objeto de este documento

El objetivo de este manual del programador es dar a conocer al lector toda la información relacionada con la aplicación <PAUM_nalyzer> y la lógica con la que se ha desarrollado, la cual es propia del autor por lo cual he considerado documentarla.

Antes de continuar me gustaría aclarar que este documento no pretende ser un manual de uso de la aplicación, para ello se ha redactado otro documento. Tampoco pretende ser un curso de aprendizaje de cada una de las herramientas o funciones empleadas.

Para un mayor detalle acerca de cada una de las herramientas utilizadas, y su forma de operación y aplicación, se recomienda consultar los manuales respectivos de cada una de ellos.

2. Conceptos básicos

Antes de continuar con la lectura de este documento se recomienda el conocimiento avanzado de los siguientes aspectos de Matlab:

1. Funcionamiento y desarrollo de las Graphic User Interfaces(GUI)
2. Callbacks y su ejecución
3. Elementos gráficos y su jerarquía
4. Propiedades de cada tipo de elemento gráfico
5. Handles y punteros
6. Funciones: *set()*, *get()* y *findobj()*
7. Funciones: *setappdata*, *getappdata*, *isappdata* y *rmappdata*
8. Funciones básicas de Matlab: *plot*, *delete*, *flipud*, *cell2mat*,...

Si se desconoce alguna de las funciones mencionadas se recomienda el uso de la documentación que proporciona Matlab mediante el siguiente código:

```
doc <Nombre_de_la_funcion>
```

a) ENTORNOS GRÁFICOS EN MATLAB

Una Interfaz Gráfica de Usuario (GUI, por sus siglas en inglés que significan Graphical User Interface) consiste en la aplicación de una forma en la cual un usuario puede interactuar con una aplicación informática, realizando distintas funcionalidades de forma intuitiva y dinámica.

Como su nombre indica, se caracteriza por el desarrollo de elementos gráficos para poder brindar esta aptitud, utilizando objetos gráficos que reemplazan la introducción de líneas de código, realizando las funciones con tan solo pinchar un botón, seleccionando una función en un menú desplegable y ejecutando las aplicaciones en un entorno gráfico adecuado.

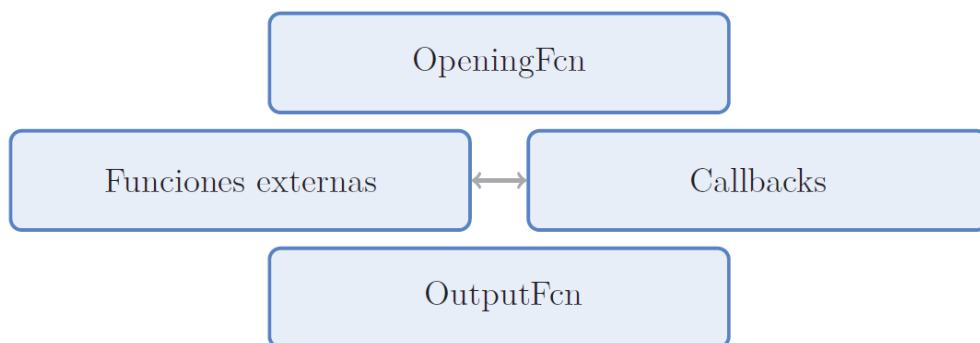
b) INTERFACES GRÁFICAS EN MATLAB

El entorno gráfico de usuario de la herramienta se ha generado utilizando la herramienta GUIDE de Matlab. Esta herramienta permite crear interfaces gráficas de manera rápida y sencilla. Para ello en primer lugar se van organizando sobre un entorno vacío todos los elementos que conformarán la ventana (cuadros de texto, botones, ventanas desplegables,...) y configurando sus diferentes propiedades: tamaño, valores iniciales, tipo de letra, etc.

Algunos de los elementos gráficos disponibles en la herramienta GUIDE que se han utilizado en la implementación de esta herramienta son los siguientes:

- **Casillas de verificación (checkboxes):** Permiten la selección simultánea de varias opciones, indicando la selección del elemento mediante una marca o tick.
- **Botones de acción (pushbuttons):** Al pulsarlos se ejecuta una acción determinada, como pasar a la siguiente ventana, visualizar un registro, iniciar un análisis, etc. Generalmente, sobre el propio botón se muestra una etiqueta de texto que indica la acción que se ejecutará al pulsarlo.
- **Cuadros de texto estático (static text):** Permiten mostrar una cadena de caracteres. Estos controles de texto estático se utilizan normalmente para etiquetar otros controles o indicar los valores asociados. El usuario no puede interactuar con ellos de forma directa.
- **Ejes (axes):** Proporcionan un espacio en el que se pueden representar gráficas.
- **Listas desplegables (pop-up menu):** Permiten seleccionar una opción de una lista. La primera pulsación sobre este elemento despliega un menú en el cual, tras una segunda pulsación, queda seleccionada la opción bajo el puntero del ratón.

El formato general de este fichero es el mostrado en la siguiente Figura:



Estructura de una GUI en Matlab

- **Callback:** Son las funciones en las que se define el comportamiento de los elementos que componen la ventana. GUIDE genera de forma automática una función Callback por cada elemento, añadiéndole a su nombre la cadena “_Callback”. Es en estas funciones donde se encuentra la mayor parte del código escrito...
- **Funciones externas:** Son funciones auxiliares creadas por el propio programador. Se puede llamar a estas funciones desde cualquiera de los otros 3 bloques anteriormente descritos.

Las funciones software de los diferentes bloques se comunican entre sí haciendo uso de una estructura llamada handles. Dicha estructura contiene un campo para identificar cada elemento de la ventana, y además el programador puede añadirle tantos campos como desee. Es necesario señalar que si dentro de un Callback se edita la estructura handles, hay que llamar a la función guidata antes de salir de la función; de lo contrario estos cambios se perderían.

Las funciones internas de las interfaces gráficas se comunican entre sí haciendo uso de una estructura llamada handles. Dicha estructura contiene un campo para identificar cada componente, y almacenar las propiedades de cada uno de estos. En este entorno gráfico, para el manejo de las señales y variables ajenas a la interfaz, se han utilizado las funciones setappdata y getappdata. Estas funciones permiten almacenar y recuperar datos en el espacio de trabajo del directorio raíz de Matlab.

c) TIPOS DE FICHEROS

Al guardar el trabajo, se generan dos archivos: uno con extensión “.fig”, que contiene la descripción de la ventana (elementos que contiene, propiedades de cada elemento, etc.) y otro con extensión .m, que permitirá controlar el comportamiento de la ventana.

3. Estructura del programa: Ficheros y registros

Para el correcto funcionamiento de esta aplicación son necesarios los siguientes archivos y que no se modifiquen sus nombres ni su ubicación:

- Archivos, en la carpeta raíz, intrínsecos a las GUI
 - PAUM_Analyzer.fig
 - PAUM_Analyzer.m
- Carpeta Functions, con los siguientes archivos:
 - my_isField_full.m
 - my_dualcursor2.m

Esta aplicación se ha diseñado para trabajar con una base de datos muy concreta. Se podrían utilizar otros registros siempre que su estructura sea similar y cumplan con los formatos que aquí se describen:

- El registro debe estar en un formato con el que Matlab pueda funcionar, en este caso ‘*.mat’.
- Los registros deben encontrarse en formato de matriz donde las señales deben estar en columnas, de forma que un registro con 42 señales de 1000 muestras cada una tenga una dimensión de 1000x42.
- El nombre del campo o variable que almacene dichas señales tiene que ser ‘X’ o ‘descargas’.
- Puede incluir o no la frecuencia de muestreo, en caso de incluirla tiene que ser en un campo o variable llamado ‘fs’.
- En caso de cargar una variable desde el Workspace de Matlab tiene que seguir las mismas condiciones, pero solo pueden cargarse variables en forma de matriz, nunca estructuras o tablas. En este caso la frecuencia de muestreo se introducirá a mano.

FICHEROS DE SALIDA

Los archivos de salida, es decir los datos que se guardan después de anotar cada registro son los siguientes:

- Nombre: 'Res_' más nombre original del archivo.
Ejemplo: "res_1DNOR_Z01.mat".
- Una estructura llamada "**result**", que incluye los siguientes campos:
 - **File**: nombre del fichero original.
 - **Paum_ini**: número de muestra que se corresponde con el inicio del paum, es decir el primer marcador temporal.
 - **Paum_fin**: segundo marcador temporal que indica el fin del paum, en muestras.
 - **Set_manual**: Índice de las señales elegidas por el usuario
 - **Mediana_curva**: curva mediana generada a partir de las señales seleccionadas en set_manual.

4. Estructura del programa: Diagramas de flujo

En este apartado se pretende ilustrar, mediante diagramas de flujo, el funcionamiento general de la aplicación. Incluyendo el inicio, las interacciones con el usuario, las funciones más importantes y la salida del programa.

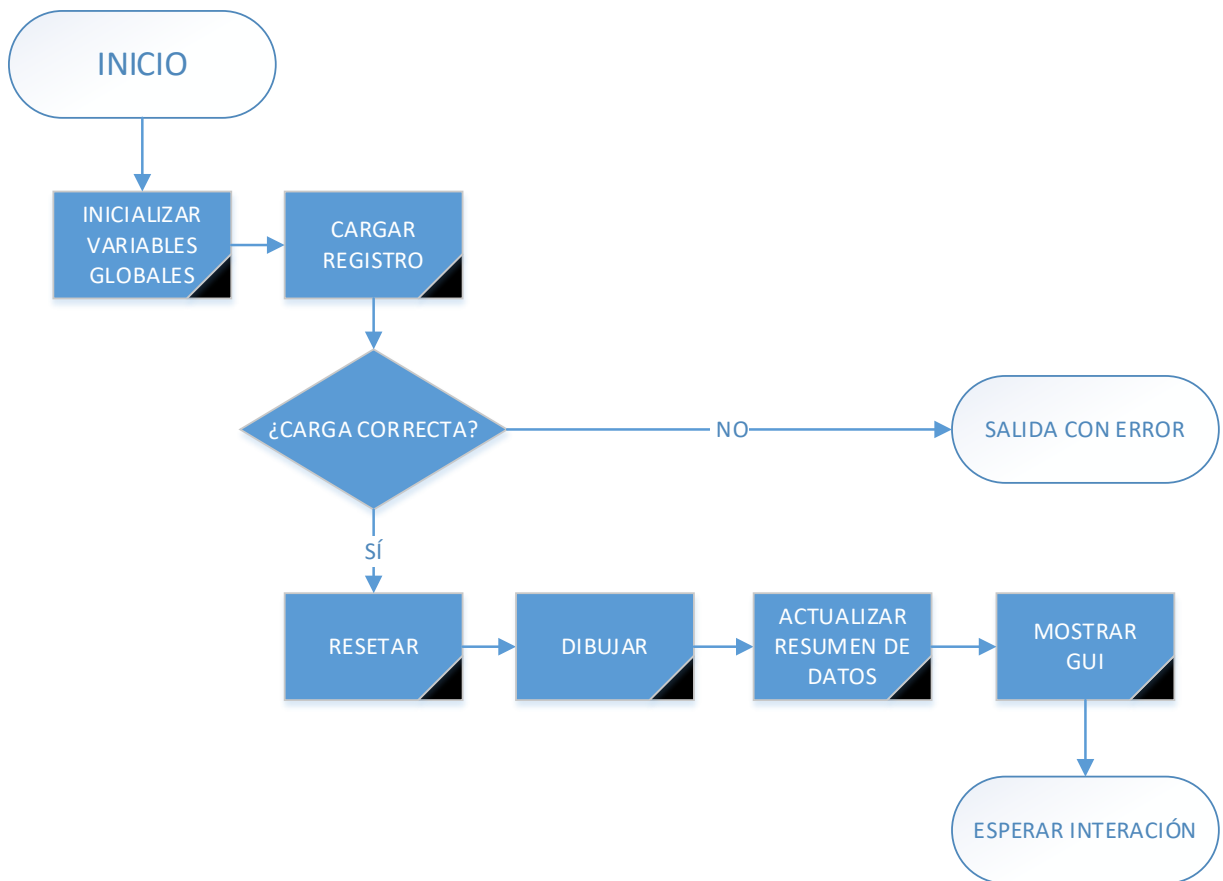
Antes de hacer continuar es necesario hacer hincapié en la forma con la que el programador maneja el flujo de datos. Para pasar los datos que son necesarios en la mayoría de funciones, que sea con la mayor fluidez y garantías posibles se ha optado por almacenarlos en la memoria local y volátil de Matlab a través de las funciones de la familia: `setappdata(0, 'Nombre', Nombre)`.

Las variables booleanas se almacenan en la estructura handles que se crea y almacena en la propia interfaz gráfica con cada ejecución.

a) INICIO

El inicio de esta interfaz gráfica solo se realiza una vez después de lanzarla desde la ventana de comandos. Cada interacción que realice el usuario, cada carga o reseteo de registro sigue caminos distintos.

En la primera parte del programa, se inicializa la GUI y se inicializan variables internas de la misma. Después le pide al usuario que indique un registro y si la carga es correcta, lo dibuja y muestra la GUI en pantalla.



b) POR FUNCIONES

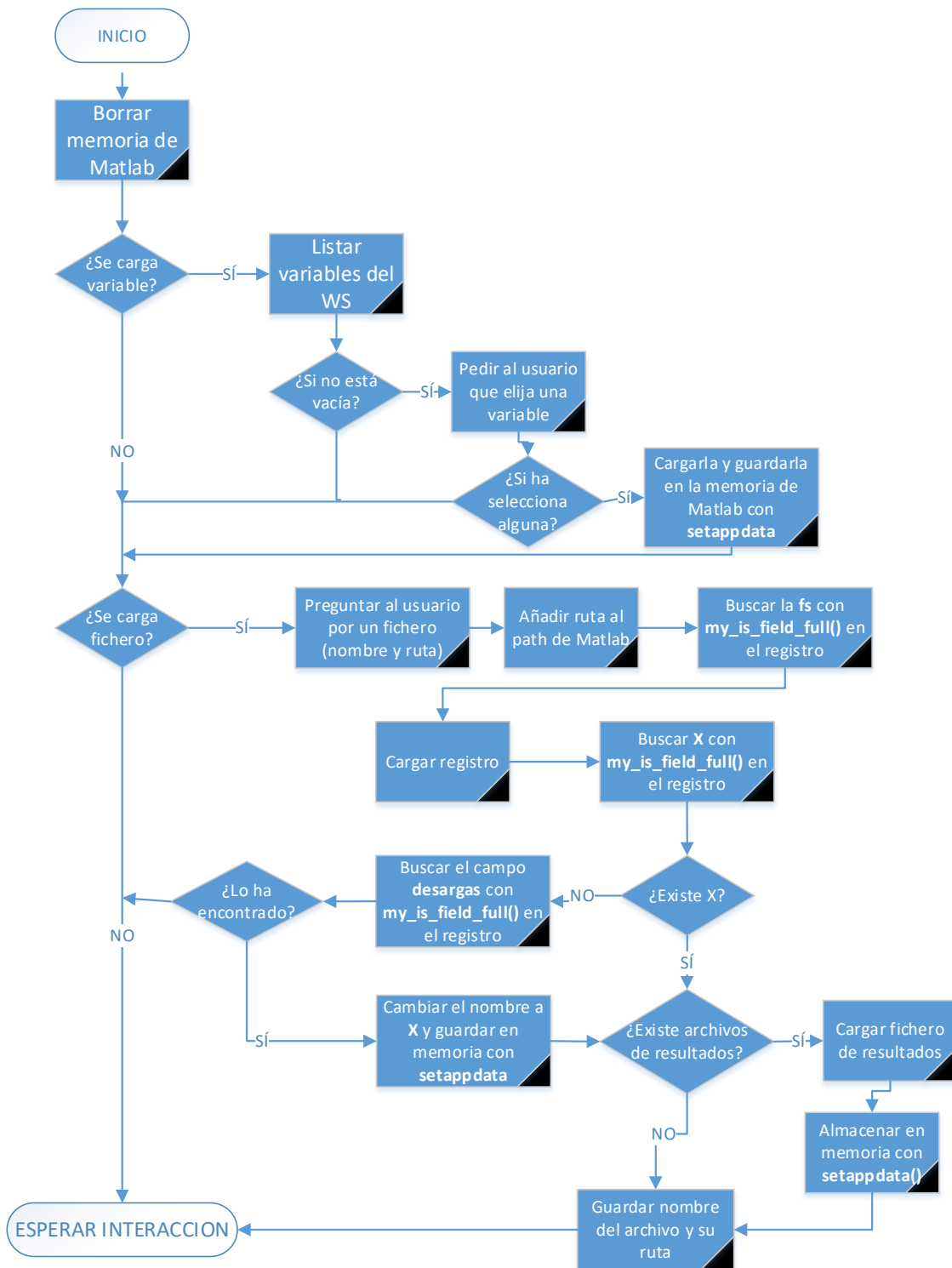
i. Cargar

En el primer inicio, solo se puede cargar un fichero. En cargas sucesivas se comprueba la variable `handles.carga_fichero`, 0 para cargar variables del Workspace y 1 para ficheros. Al iniciar esta función y para evitar errores por sobrescritura se borra la memoria utilizada en Matlab.

Si se carga desde el Workspace, lo primero es mostrar en pantalla un listado de las variables para que el usuario escoja y después se guarda en la memoria de Matlab. Si se carga un archivo, se le pide al usuario que cargue uno. Después, con la función `my_isField_full`, a la cual se le pasan como parámetro una estructura y un campo, se busca en todos los anidamientos posibles el campo 'fs' para guardarlo en la memoria de Matlab y cambiar la variable `handles.modos_fs`, que interviene en la función `Dibuja`.

De forma homologa se buscan los campos con las señales, en el proyecto original se usaban los nombres 'X' o 'descargas'. Posteriormente se guardan en la memoria de Matlab junto con su nombre y ruta.

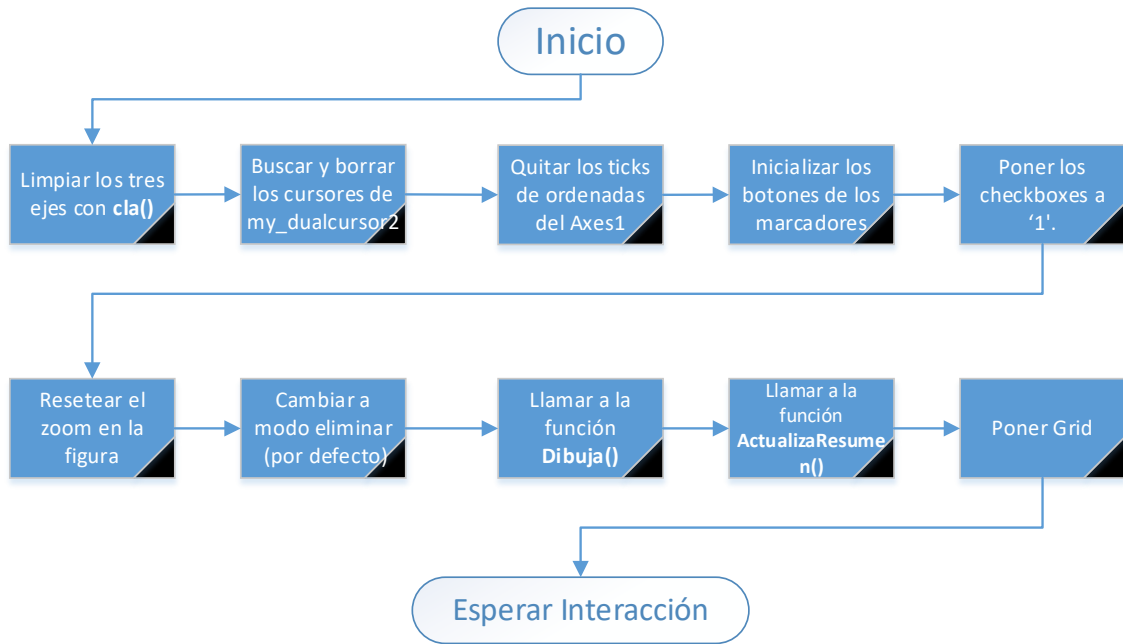
Si no se consiguen cargar datos válidos, la aplicación colapsa y muestra un error en la ventana de comandos.



ii. *Resetear*

Esta función se ejecuta en cada inicio de aplicación, en cada carga de registro y cada vez que el usuario pulse el botón de resetear en el menú.

El objetivo de esta función es volver al estado original en cada objeto para evitar errores de funcionamiento y de escritura. Se limpian los ejes, se borran los cursores y se devuelven los 'uicontrols' a su estado por defecto. Finalmente, se llama a la función *dibuja*, para que dibuje el registro cargado en la memoria local.



iii. Dibuja

Como su propio nombre indica, esta función se encarga de dibujar la matriz de datos X (se carga desde la memoria de Matlab), crear el eje de tiempos que corresponda y cargar los últimos datos guardados de este registro.

Primero se cargan los datos necesarios (X y fs) desde la memoria de Matlab. Aquí no hay ningún control de datos, si hay algún dato mal o su estructura no es la idónea, este software presentara un comportamiento inesperado.

Se comprueba la variable `handles.modo_fs`, si es 1 crea un eje de tiempos en función de la frecuencia de muestreo, sino crea un eje de muestras.

Después, en el `Axes1`, se dibuja con `plot` la matriz X, donde la última fila de la matriz es la última línea dibujada y la que queda encima del resto. Es decir que la matriz de handles de las líneas que maneja Matlab esta invertida respecto a nuestra matriz, por eso le damos la vuelta con la función `flipud`.

El control de las líneas dibujadas se realiza con la propiedad 'Visible' de la clase `Line`, de modo que cuando el usuario decida eliminar una línea, no se borra, simplemente se oculta.

En el `Axes2` se muestran las líneas eliminadas, de modo que en un inicio se copian todos los handles de las líneas del `Axes1` y se les modifica la propiedad 'Visible' a 'off'. Cada vez que se elimine una línea del `Axes1` se oculta y se muestra en el `Axes2`, funcionan de forma complementaria.

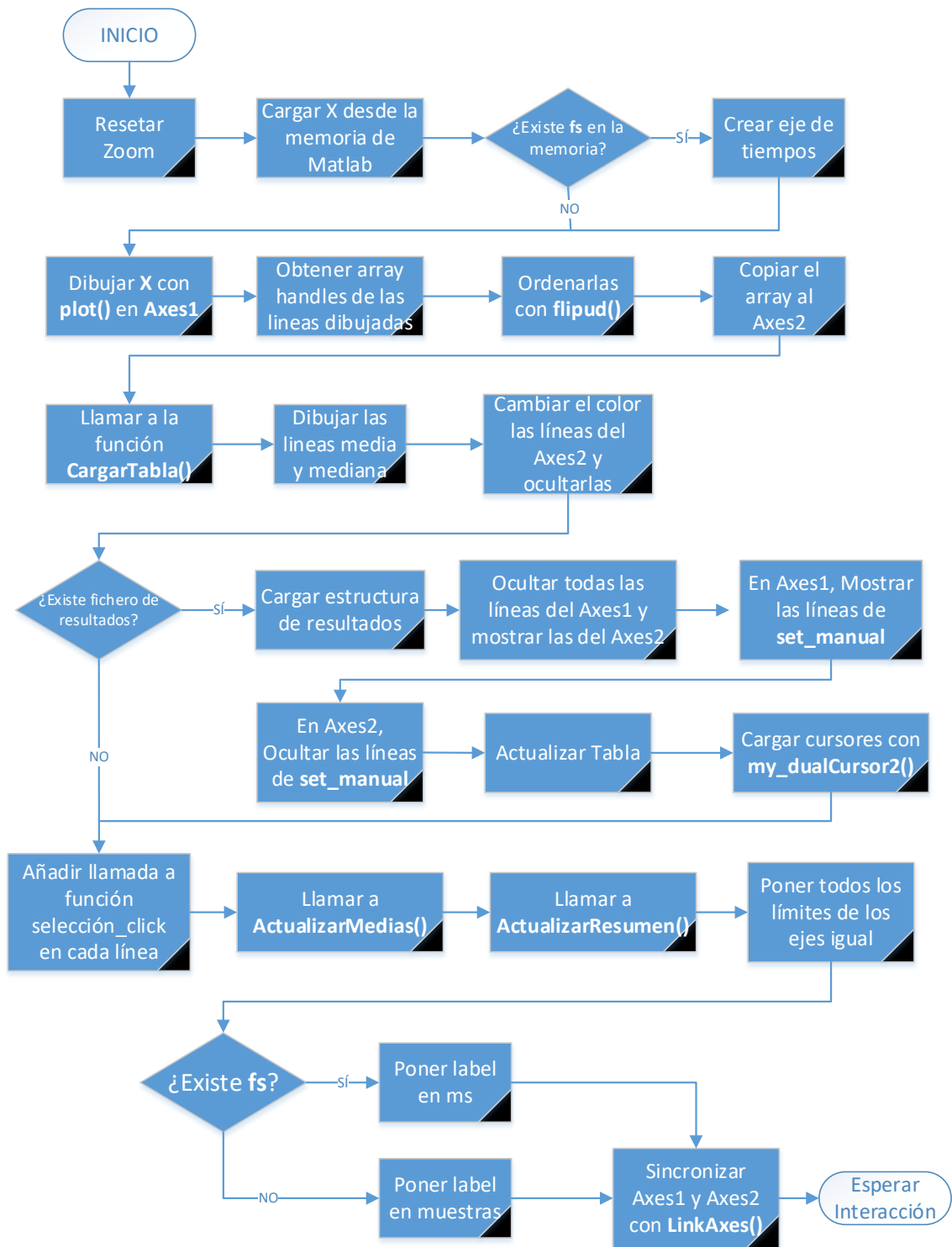
Después, se dibujan las líneas media y mediana, que se calculan a partir de las líneas visibles del `Axes1`. Estas líneas están siempre encima del resto y son similares en los tres ejes.

Se comprueba que exista un archivo de resultados, la estructura de su nombre tiene que ser 'res_' más el nombre de el mismo archivo. Si existe se carga, se lee el campo 'set_manual' que almacena las líneas que tienen que estar visibles. Se muestran y ocultan las líneas de forma similar a la anteriormente explicada. También se cargan los marcadores de tiempo (inicio y fin) almacenados, y se pasan por referencia a la función `my_dualcursor2`, que se encarga de situar las barras en su posición en el eje de abscisas.

A todas las líneas del `Axes1` y el `Axes2`, excepto las líneas media y mediana, se les completa la propiedad 'ButtonDownFcn' con una llamada a la función 'selection_click'. Esta propiedad llama a la función que se le indique cada vez que se haga clic en la línea.

Una vez están todas las líneas dibujadas y configuradas se actualizan el resto de objetos de la GUI que son la tabla de la derecha, las líneas medias y medianas y el panel de resumen.

Finalmente se sincronizan los `Axes1` y `2` con la función 'Linkaxes' para cada vez que se haga zoom o se mueva un eje, se comporten de forma similar.

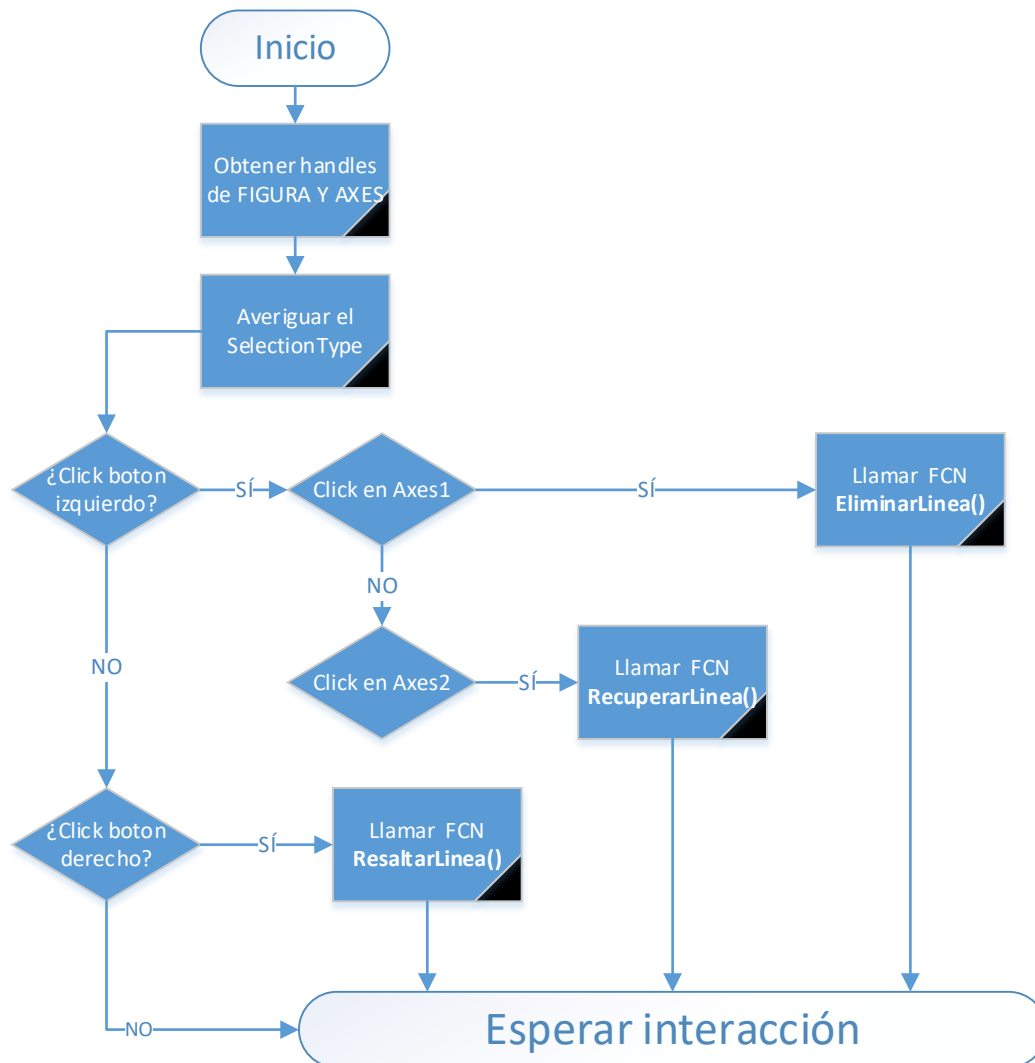


iv. *SelectionClic*

Esta función se programa a cada línea de los Axes1 y 2, excepto a las líneas media y mediana. Cada vez que se haga clic una línea, se ejecuta esta función que tiene como parámetro de entrada el handles (llamado hObject) de la propia línea.

El objetivo de esta función es hacer de control. Se ejecutan funciones distintas en función del botón del ratón pulsado y en el Axes en el que se encuentra la línea.

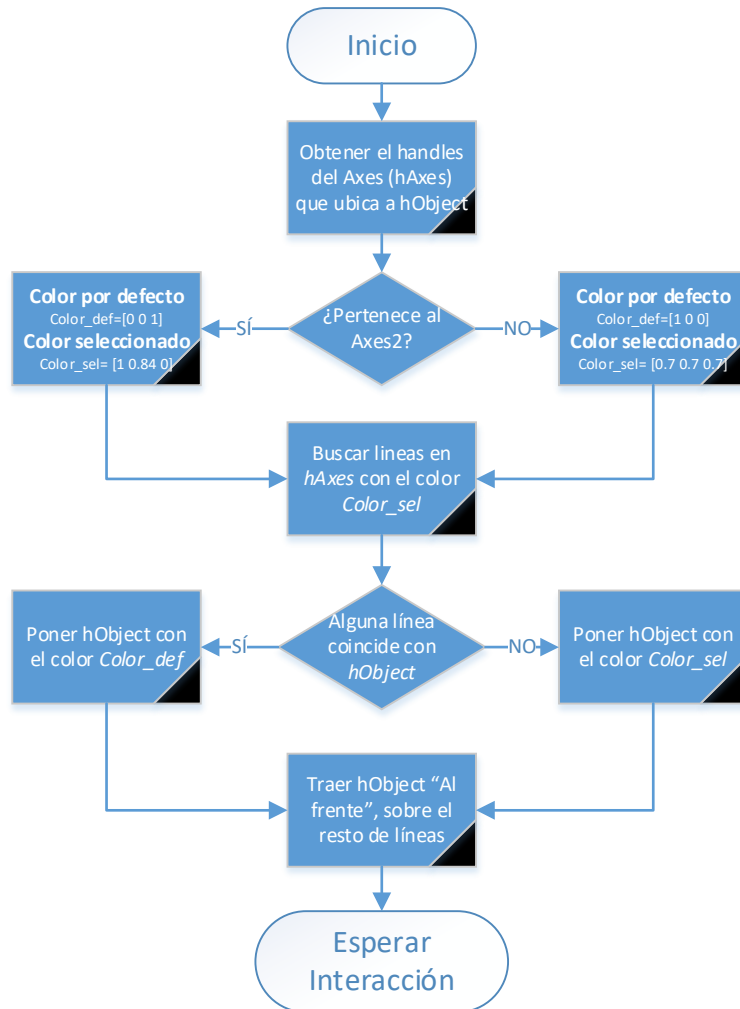
Si se pulsa el botón derecho se llama a la función de ResaltarLinea. Si por el contrario se hace clic con el botón izquierdo y en el Axes1 se llama a la función EliminarLinea, si es en el Axes2 a RecuperarLinea.



v. *ResaltarLinea*

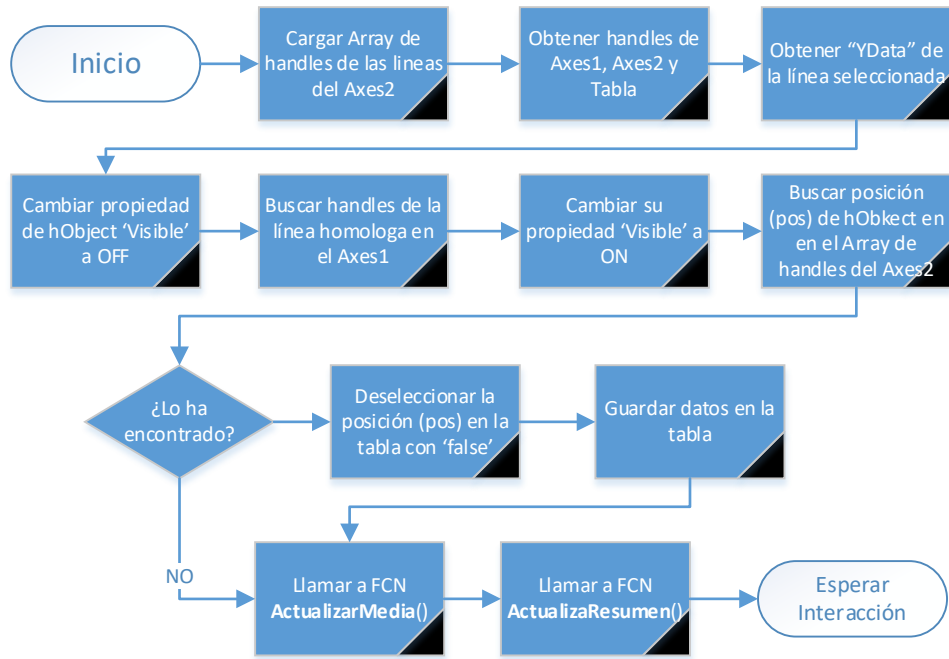
Esta función tiene como parámetro de entrada el handles(hObject) de la línea pulsada. Con el handles se averigua el Axes al que pertenece. En función del Axes se modifica el grosor y color de resaltado, amarillo para el Axes1 y gris para el Axes2.

Si la línea ya estaba resaltada se pone con su grosor y color original. Y en todos los casos se trae al frente, sobre el resto de líneas, para que se visualice mejor.



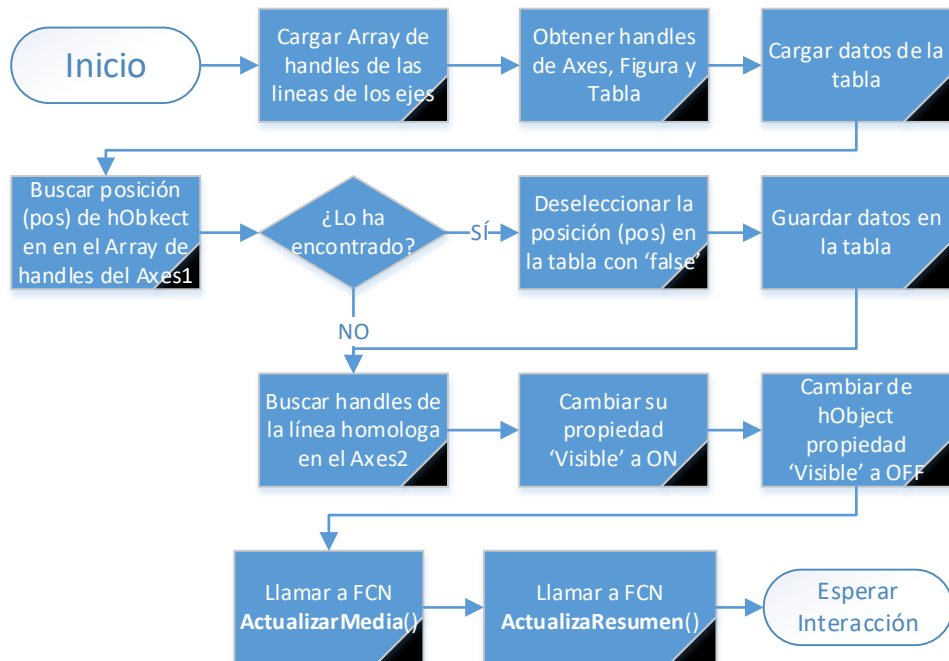
vi. *RecuperarLinea*

El objetivo de esta función es recuperar una línea eliminada, para ello se juega con la propiedad 'Visible' de la línea. Se muestra en el Axes1 y se oculta en el Axes2. Finalmente se actualizan las líneas medias y el resto de objetos visuales.



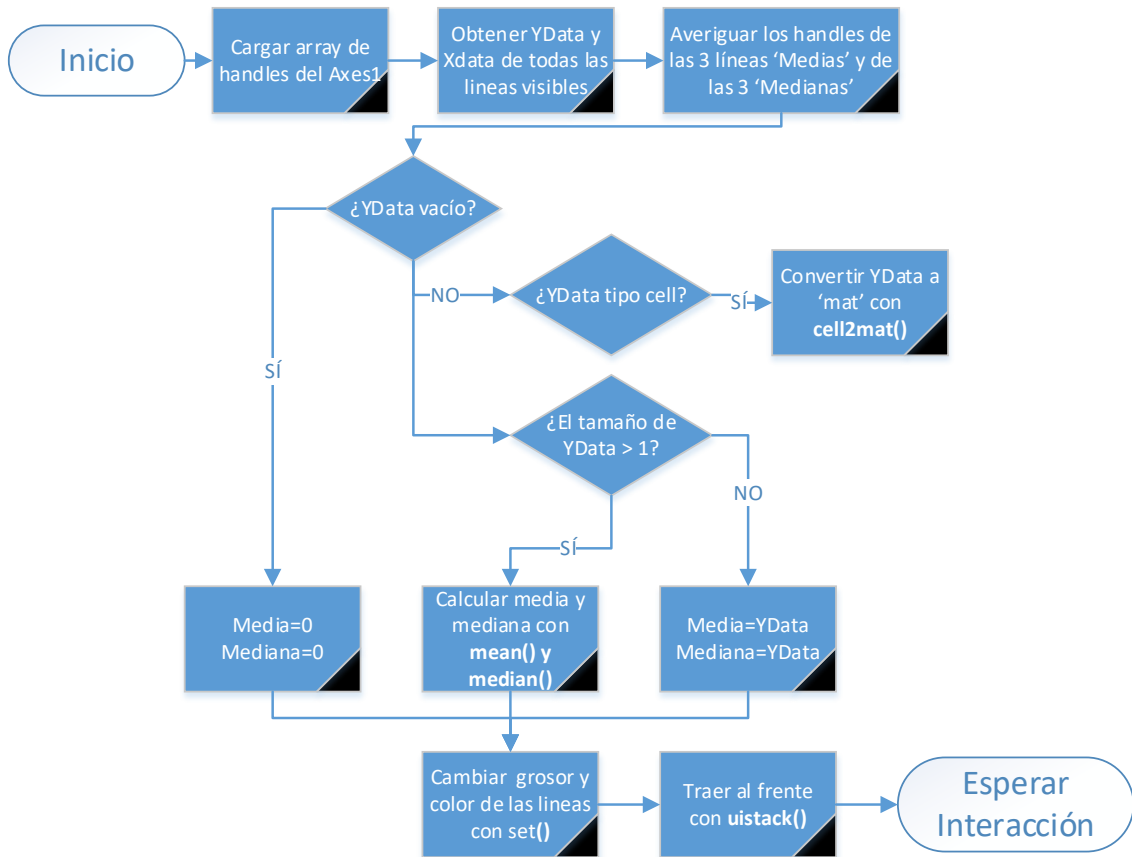
vii. EliminarLinea

El objetivo de esta función es eliminar la línea deseada. Se oculta la línea sobre la que se ha hecho clic (hObject), con la propiedad 'Visible'. Se busca la línea homologa del Axes2 para hacer visible con la función set. Finalmente se actualizan las líneas medias y el resto de objetos visuales.



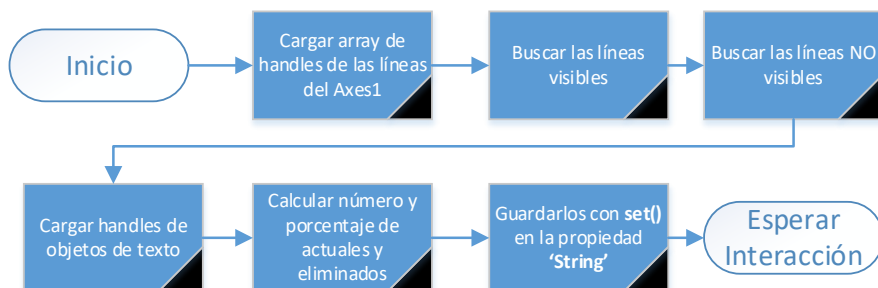
viii. ActualizarMedias

Cada vez que se muestren y oculten líneas, se llama a esta función para que actualice las curvas media y mediana. Para ello se carga la propiedad 'YData' de, únicamente, las líneas visibles del Axes1. El campo 'YData' es de tipo 'celda' siempre que su dimensión sea mayor que uno, por ello se realiza un control de flujo. Si está vacío, es decir, no ha líneas visibles, las curvas tienen media y mediana 0. Si hay solo una línea, el valor de la media y la mediana será el de esa misma línea. Si hay varias líneas se convierten a formato 'mat' y se calculan la media y la mediana.



ix. ActualizarResumen

El objetivo de esta función es actualizar el panel con el resumen de las líneas seleccionadas. Para ello se utiliza la función 'findobj' para buscar las líneas visibles y ocultas de los Axes1 y Axes2, respectivamente. Finalmente se calcula el número y el porcentaje. Se ejecuta con cada línea eliminada o recuperada.



x. *Guardar_res*

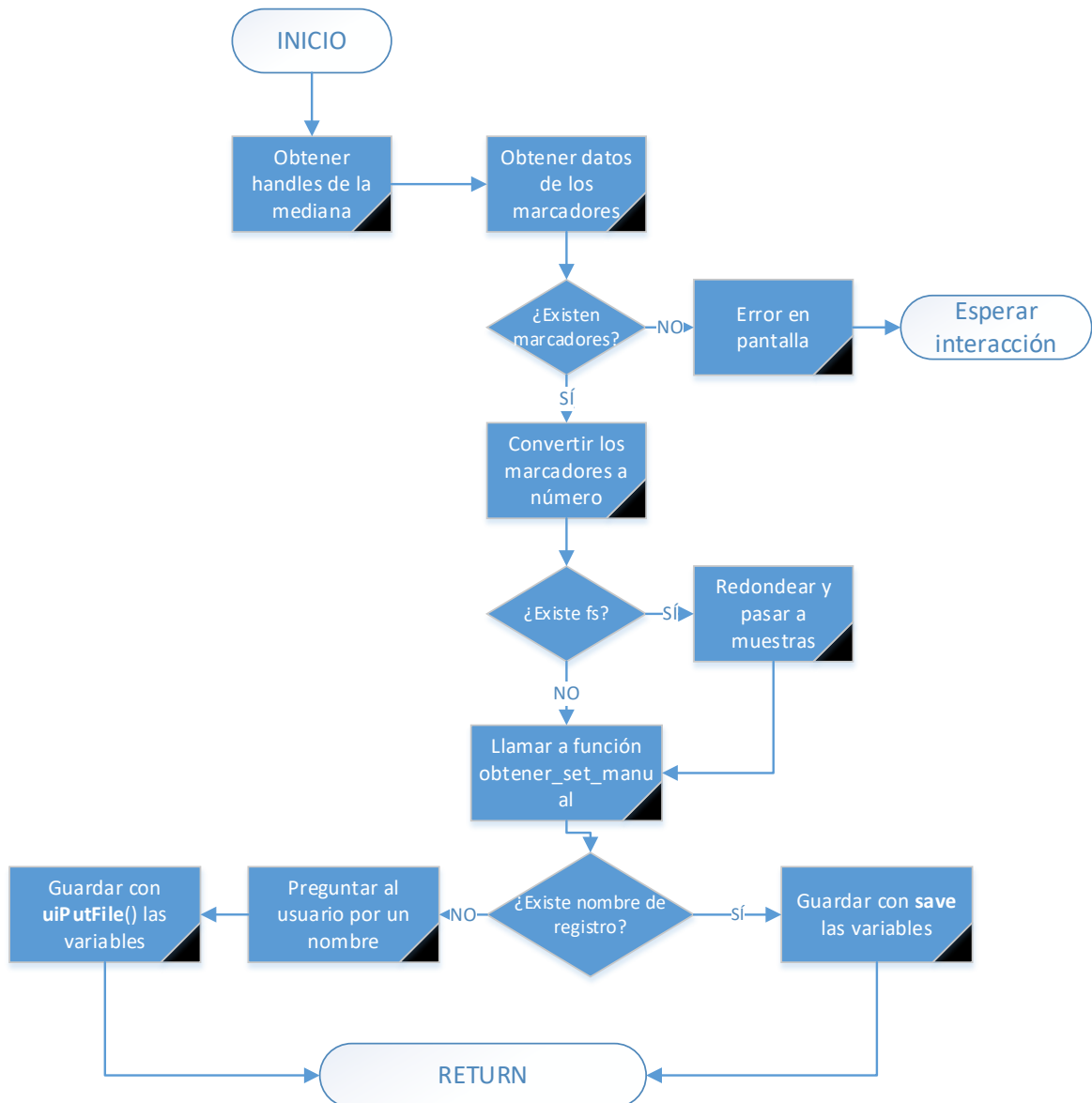
Esta función devuelve la variable 'val', que toma el valor 1 para un guardado correcto y 0 para un guardado incorrecto.

Tras obtener los handles necesarios, comprueba que se han introducido marcadores, en caso negativo muestra un error en pantalla y de devuelve 'val=0'.

Si existen se comprueba el valor de handles.modos_fs para convertir, o no, los valores de los marcadores a muestras.

Después se llama a la función obtener set_manual que devuelve los índices de las líneas visibles.

Finalmente se comprueba si handles.FileName tiene un nombre para el guardado, en caso contrario se le pide al usuario una ruta y un nombre para guardar el fichero, si el guardado es correcto devuelve 'val=1'. Los campos guardados son: FileName, set_manual, paum_ini, paum_fin y la curva mediana.



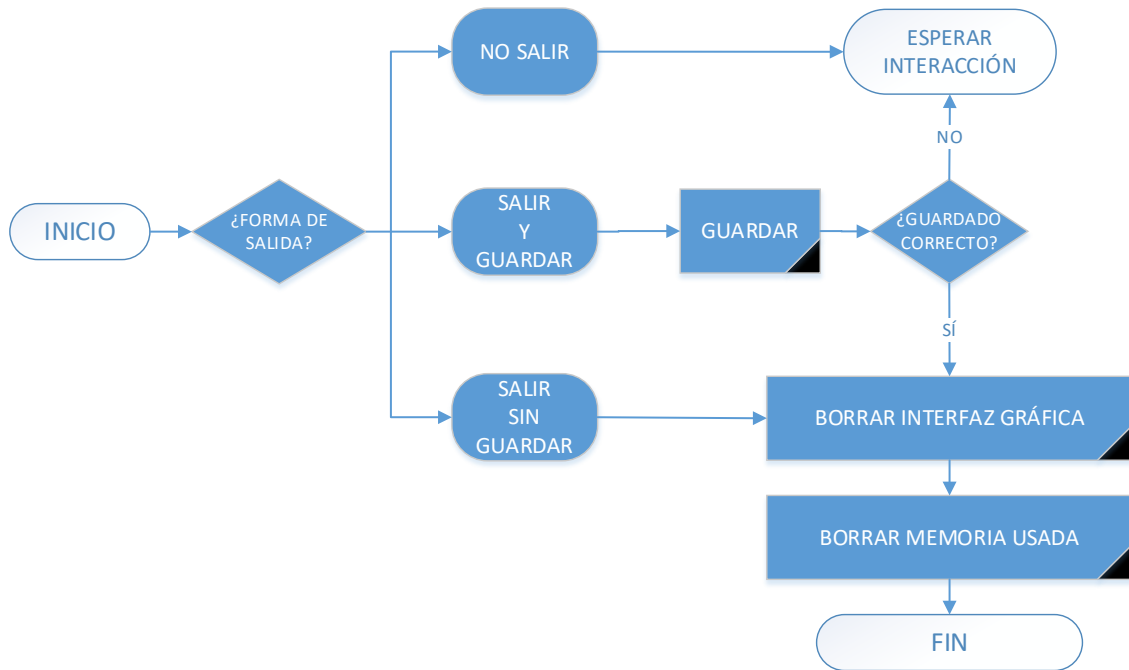
c) SALIDA

Al pulsar el botón de salir de la ventana (según sistema operativo y versión), se le pregunta al usuario que desea hacer.

En el caso de no querer salir, no se realiza ninguna acción.

Si desea salir guardando, se llama a la función guardar y se comprueba que se ha guardado bien. En caso contrario se lanza un mensaje de error.

Antes de salir se borra la interfaz gráfica, todos los handles y la memoria de matlab utilizada.



5. Funciones

A. CARGAR

```
function val=cargar(hObject, eventdata, handles)
b) %Funcion para cargar el registro
c) %Si carga val=1
d) %Si no carga val=0
e)
f) %Primero borramos las variables en el puntero 0 e inicializamos val
g) borrarAPPDATA;
h) val=0;
i)
j)
k) if handles.carga_fichero==0
l) %%Carga de variable
m)     %Listar variables del Workspace
n)     str = evalin('base','who');
o)     if ~isempty(str)
p)         [s,val] = listdlg('PromptString','Select a file:',...
q)             'SelectionMode','single',...
r)             'ListString',str);
s)     end
t)     if val==1
u)         %Almacenar variable seleccionada en 0
v)         X=evalin('base', str{s});
w)         setappdata(0,'X',X);
x)         val=1;
y)         handles.ruta=[];
z)         handles.FileName=[];
aa)
bb)     end
cc)
dd) elseif handles.carga_fichero==1
ee)     %%Carga de fichero
ff)     [FileName,ruta] = uigetfile('*.mat','Selecciona un registro');
gg)
hh)     if ~(FileName==0)
ii)         val=1;
jj)         addpath(ruta);
kk)
ll)         %Cargar registro en memoria local/funcion
mm)         carga=load(FileName,'-mat');
nn)         handles = guidata(hObject);
oo)         %Busca en toda la estructura el campo 'fs'
pp)         handles.modos_fs= my_isField_full (carga, 'fs');
qq)         guidata(hObject, handles);
rr)
ss)         %Busca en toda la estructura el campo 'X' o 'descargas'
tt)         %PARA BUSCAR LAS SEÑALES BAJO OTRO NOMBRE CAMBIAR 'X' POR 'NOMBRE_DESEADO'
uu)         result= my_isField_full (carga, 'X');%CAMBIAR PARA BUSCAR OPOR OTRO NOMBRE
vv)         if result==0%%SI en la estructura cargada no existe X, lo intenta con 'descargas'
ww)             my_isField_full (carga, 'descargas');
xx)             if isappdata(0,'descargas')
yy)                 setappdata(0,'X',getappdata(0,'descargas'));%Cambiar el nombre(a 'X') del
registro almacenado
zz)                 rmappdata(0,'descargas');
aaa)                 end
bbb)                 end
ccc)
ddd)                 %Cargar (si existe) el archivo de resultados
eee)                 if exist(['res_',FileName], 'file')==2
fff)                     load(['res_',FileName]);
ggg)                     setappdata(0,'result',result)
hhh)                 end
iii) % Almacenar en la estructura handles la ruta y el nombre del fichero
```

```

jjj)         handles.ruta=ruta;
kkk)         handles.FileName=FileName;
lll)         end
mmm) end
nnn) %Poner/Actualizar el nombre del registro en el statictext superior
ooo) set(handles.text_filename, 'String',handles.FileName);
ppp) handles.carga_fichero=1;
qqq) guidata(hObject, handles);

```

B. RESETEAR

```

function resetear(hObject, eventdata, handles)
1. %Funcion para resetear los botones y los axes
2. cla(handles.axes1)
3. cla(handles.axes2)
4. cla(handles.axes3)
5.
6. %Buscar y Borrar cursores(granates)
7. h_cursoros=findobj(handles.figure1, 'Tag', 'Copy_Cursor');
8. delete(h_cursoros);
9.
10. %Quitar los ticks de ordenadas axes1
11. set(handles.axes1, 'XTickLabel', '')
12.
13. %Inicializar botones de los marcadores
14. set(handles.push_t_ini, 'String', 'Tiempo Inicial', 'Enable', 'off');
15. set(handles.push_t_fin, 'String', 'Tiempo Final', 'Enable', 'off');
16. set(handles.push_ajustar_ejes, 'Enable', 'off');
17. set(handles.check_markers, 'Value', 0, 'Enable', 'off');
18.
19. %Poner los checkbox a 1
20. set(handles.check_media, 'Value', 1)
21. set(handles.check_mediana, 'Value', 1)
22. set(handles.check_all, 'Value', 1)
23.
24. %Reset Zoom
25. zoom(handles.figure1, 'Off');
26. zoom reset;
27. pan(handles.figure1, 'Off');
28.
29. %Modo eliminar(por defecto)
30. set(handles.uipanel2, 'SelectedObject', handles.radio_eliminar)
31.
32. guidata(hObject, handles);
33. dibuja(hObject, eventdata, handles);
34.
35. %Actualizar el panel de resumen de señales
36. ActualizarResumen;
37.
38. %Grid
39. grid(handles.axes1, 'on');
40. grid(handles.axes2, 'on');
41. grid(handles.axes3, 'on');
42. guidata(hObject, handles);

```

C. DIBUJA

```

function dibuja(hObject, eventdata, handles)
1. zoom reset;
2. X=getappdata(0, 'X');

3. %Crear eje de tiempo si hay valor fs
4. if isappdata(0, 'fs')
5. fs=getappdata(0, 'fs');
6. ejex=1000*(0:length(X)-1)/fs;%Eje en milisegundos
7. else

```

```

8. ejex=1:length(X);
9. end

10. %Limites
11. %Lo pongo redundante para que funcione bien al cambiar de registro
12. set(handles.axes1, 'XLim', [ejex(1) ejex(end)]);
13. set(handles.axes1, 'YLim', [1.15*min(X(:)) 1.15*max(X(:))]);
14. xlim([ejex(1) ejex(end)])
15. ylim([1.15*min(X(:)) 1.15*max(X(:))]);
16. zoom reset;

17. %Plotear y cambiar propiedades
18. plot(handles.axes1, ejex, X, 'b');
19. set(handles.axes1, 'XLimMode', 'auto', 'YLimMode', 'auto')

20. line_registros=findobj(handles.axes1, 'Type', 'line', '-and', 'Color', 'b');
21. line_registros=flipud(line_registros); %Invertir el orden
22. %Poner un Tag a cada linea con su indice
23. for i=1:length(line_registros)
24. set(line_registros(i), 'Tag', num2str(i));
25. end

26. %Copiar lineas para el axes2
27. line_registros_ax2=copyobj(line_registros, handles.axes2);
28. set(line_registros_ax2, 'Tag', 'line_ax2', 'Visible', 'off', 'Color', 'r');

29. %Dibujar curvas media y mediana para los 3 ejes con Tags distintos
30. h2=line('XData', ejex, 'YData', mean(X,2), 'Parent', handles.axes1);
31. h3=line('XData', ejex, 'YData', median(X,2), 'Parent', handles.axes1);

32. h4=line('XData', ejex, 'YData', mean(X,2), 'Parent', handles.axes2);
33. h5=line('XData', ejex, 'YData', median(X,2), 'Parent', handles.axes2);

34. h6=line('XData', ejex, 'YData', mean(X,2), 'Parent', handles.axes3);
35. h7=line('XData', ejex, 'YData', median(X,2), 'Parent', handles.axes3);

36. set(h2, 'Tag', 'Medial', 'Color', 'm', 'LineWidth', 3);
37. set(h3, 'Tag', 'Mediana1', 'Color', 'g', 'LineWidth', 3);

38. set(h4, 'Tag', 'Media2', 'Color', 'm', 'LineWidth', 3);
39. set(h5, 'Tag', 'Mediana2', 'Color', 'g', 'LineWidth', 3);

40. set(h6, 'Tag', 'Media3', 'Color', 'm', 'LineWidth', 3);
41. set(h7, 'Tag', 'Mediana3', 'Color', 'g', 'LineWidth', 3);

42. cargarTabla(handles, line_registros);

43. %Cargar si existen los paum almacenados en result
44. if isappdata(0, 'result')
45. result=getappdata(0, 'result');
46. pos=result.set_manual;
47. %Activar solo los 'set_manual' en el axes1 y desactivar en el axes2
48. set(line_registros, 'Visible', 'off', 'Color', 'b', 'LineWidth', .5);
49. set(line_registros(pos), 'Visible', 'on', 'LineWidth', .5);

50. set(line_registros_ax2, 'Visible', 'on', 'LineWidth', .5);
51. set(line_registros_ax2(pos), 'Visible', 'off', 'Color', 'b', 'LineWidth', .5)
52. %Modificar tabla
53. data=get(handles.tabla1, 'Data');

54. if ~isempty(pos)
55. data(:,2)={false};
56. for i=1:length(pos)
57. data{pos(i),2}=true;
58. end
59. set(handles.tabla1, 'Data', data);
60. end

```

```

61. %Cargar cursores/marcadores temporales(ON-->OFF de my_dualCursor2)
62. if handles.modos_fs==1
63. fs=getappdata(0,'fs');
64. ini=1000*result.paum_ini/fs;
65. fin=1000*result.paum_fin/fs;
66. end

67. my_dualcursor2([], [ini fin], 'r*', handles, handles.axes1)
68. set(handles.push_t_ini, 'Enable', 'on', 'Value', 0);
69. set(handles.push_t_fin, 'Enable', 'on', 'Value', 0);
70. set(handles.check_markers, 'Enable', 'on');
71. set(handles.input_fs, 'Enable', 'on');
72. set(handles.push_ajustar_ejes, 'Enable', 'on');
73. my_dualcursor2('off', [], [], [], handles.axes1)

74. %reset PAN y ZOOM
75. z = zoom(handles.figure1);
76. p = pan(handles.figure1);
77. set(z, 'ActionPostCallback', []);
78. set(p, 'ActionPostCallback', []);
79. set(handles.check_markers, 'Value', 1)
80. check_markers_Callback(hObject, eventdata, handles)

81. end

82. %Poner [propiedad] funcion al clicar las lineas del registro
83. set(line_registros, 'ButtonDownFcn', @seleccion_click);
84. set(line_registros_ax2, 'ButtonDownFcn', @seleccion_click);

85. %Almacenar los handles de los registros en 0
86. setappdata(0, 'line_registros', line_registros);
87. setappdata(0, 'line_registros_ax2', line_registros_ax2);

88. ActualizarResumen;
89. actualizarMedias()

90. %Poner todos los ejes iguales
91. xL=get(handles.axes1, 'XLim');
92. yL=get(handles.axes1, 'YLim');

93. set(handles.axes2, 'XLim', xL);
94. set(handles.axes2, 'YLim', yL);

95. set(handles.axes3, 'XLim', xL);
96. set(handles.axes3, 'YLim', yL);
97. set([handles.axes1 handles.axes2], 'XLimMode', 'manual', 'YLimMode', 'manual');

98. %Poner labels en tiempo o muestras
99. if exist('fs', 'var')
100.     xlabel(handles.axes2, 'Tiempo (ms)');
101.     xlabel(handles.axes3, 'Tiempo (ms)');
102. else
103.     xlabel(handles.axes2, 'Muestras');
104.     xlabel(handles.axes3, 'Muestras');
105. end

106. set(handles.axes3, 'XLimMode', 'auto', 'YLimMode', 'auto');
107. handles=guidata(hObject);
108. guidata(hObject, handles);

```

D. SELECCION_CLICK

```

function seleccion_click(hObject, dump)
1. hFig=ancestor(hObject, 'figure');
2. hax=ancestor(hObject, 'axes');
3. b = get(hFig, 'selectiontype');
4.

```

```

5. if strcmpi(b,'normal')%%Click izq
6.     %Depende de que axes sea la linea
7.     if strcmpi('axes1',get(hax,'Tag'))
8.         EliminarLinea(hObject,0);
9.     elseif strcmpi('axes2',get(hax,'Tag'))
10.        recuperarLinea(hObject,0);
11.    end
12. end
13. if strcmpi(b,'alt')%%Click dch
14.     resaltarLinea(hObject,0);
15. end

```

E. RESALTARLINEA

```

function resaltarLinea(hObject,eventdata)
1. %Funcion para resaltar la linea seleccionada(hObject) por el usuario
2. %Si la linea ya estaba seleccionada la vuelve a poner de su color
3. hAxes = ancestor(hObject,'axes');
4. %Establecer colores en funcion del axes que sea
5. if strcmp('axes2',get(hAxes,'Tag'))
6.     def_color='r';
7.     color_sel=[0.7 0.7 0.7];%RGB Gris
8. else
9.     color_sel=[1 0.8431 0];%RGB Amarillo/Dorado
10.    def_color='b';
11. end
12. h_lineSel=findobj(hAxes,'Type','Line','-and','Color',color_sel);
13. if h_lineSel==hObject
14.    set(hObject,'Color',def_color,'LineWidth',.5);
15. else
16.    lineas_axes=findobj(hAxes,'Type','Line','-and',{'Color','b','-or','Color',color_sel});
17.    set(lineas_axes,'Color',def_color,'LineWidth',.5);
18.    set(hObject,'Color',color_sel,'LineWidth',3);
19.    uistack(hObject,'top')
20. end

```

F. ELIMINARLINEA

```

function EliminarLinea(hObject,dump)
1. line_registros=getappdata(0,'line_registros'); %cargar handles de las lineas
2. %Buscar handles de la tabla y cargar los datos
3. hFig = ancestor(hObject,'figure');
4. ax2=findobj(hFig,'Tag','axes2');
5. ax1=ancestor(line_registros(1),'axes');
6. %Modificar tabla
7. hTabla=findobj(hFig,'Tag','tabla1');
8. data=get(hTabla,'Data');
9. pos=find(line_registros==hObject);
10. if ~isempty(pos)
11.    data{pos,2}=false;
12.    set(hTabla,'Data',data);
13. end
14. %Buscar handles de esta linea en el ax2
15. line_ax2=findobj(ax2,'YData',get(hObject,'YData'));
16. set(line_ax2,'Visible','on','LineWidth',.5)
17. %Guardar cambios
18. set(hObject,'Visible','off','Color','b','LineWidth',.5);
19. % axh = ancestor(hObject,'axes');
20. actualizarMedias;

```

21. ActualizarResumen;

G. RECUPERARLINEA

```
function recuperarLinea(hObject,eventdata)
1. % hObject handles de la linea del ax2
2. line_registros=getappdata(0,'line_registros');
3. line_registros_ax2=getappdata(0,'line_registros_ax2');
4. YData=get(hObject,'YData');
5. h_line_ax1=findobj(line_registros,'YData',YData);

6. set(h_line_ax1,'Visible','on')
7. set(hObject,'Visible','off')
8.
9. %Modificar tabla
10. %Buscar handles de la tabla y cargar los datos
11. hFig = ancestor(hObject,'figure');
12. hTabla=findobj(hFig,'Tag','tabla1');
13. data=get(hTabla,'Data');
14. pos=find(line_registros_ax2==hObject);
15. if ~isempty(pos)
16. data{pos,2}=true;
17. set(hTabla,'Data',data);
18. end
19.
20. actualizarMedias;
21. ActualizarResumen;
```

H. ACTUALIZARMEDIAS

```
1. function actualizarMedias()
2. %Funcion que actualiza la media y la mediana de los 3 axes
3. line_registros=getappdata(0,'line_registros');
4. YData=get(findobj(line_registros,'Visible','on'),'YData');
5. XData=get(line_registros(1),'XData');
6. h_media=findobj(gcf,'Tag','Medial','-or','Tag','Media2','-or','Tag','Media3');
7. h_mediana=findobj(gcf,'Tag','Mediana1','-or','Tag','Mediana2','-or','Tag','Mediana3');

8. if ~isempty(YData) %Si no hay ningun registro seleccionado YData esta vacio
9. if iscell(YData) % Si hay mas de una linea YData es una cell
    i. YData=cell2mat(YData)';
10. end
11. if min(size(YData))>1
    a. media=mean(YData,2)';
    b. mediana=median(YData,2)';
12. else
13. media=YData;
14. mediana=YData;
15. end
16. else %Si Ydata esta vacio(no hay ninguna linea) la media y la mediana son cero
17. media=zeros(1,length(XData));
18. mediana=zeros(1,length(XData));
19. end

20. %MEDIA(Magenta)
21. set(h_media,'Color','m','LineWidth',3,'YData',media);
22. %MEDIANA(Verde)
23. set(h_mediana,'Color','g','LineWidth',3,'YData',mediana);
24. for i=1:length(h_media)
25. uistack(h_media(i),'top');
26. end
27. for i=1:length(h_mediana)
28. uistack(h_mediana(i),'top');
29. end
30. %MEDIA(Magenta)
31. set(h_media,'Color','m','LineWidth',3,'YData',media);
```



```

32. %MEDIANA(Verde)
33. set(h_mediana, 'Color', 'g', 'LineWidth', 3, 'YData', mediana);
34. for i=1:length(h_mediana)
35.     uistack(h_mediana(i), 'top');
36. end
37. for i=1:length(h_mediana)
38.     uistack(h_mediana(i), 'top');
39. End

```

I. ACTUALIZARRESUMEN

```

function ActualizarResumen()
1. % Actualizar el panel de resumen de señales
2. % Primero se obtienen las señales totales, visibles y no visibles
3. % despues se buscan los handles de los statics text
4. % y se actualizan los valores con set
5. line_registros=getappdata(0, 'line_registros');
6.
7. actuales=findobj(line_registros, 'Type', 'line', '-and', 'Visible', 'on');
8.     if actuales==0
9.         actuales=[]; %Se pone en vacio([]) porque alguna versiones de Matlab devuelven 0,
           y length(0)=1;
10.     end
11. eliminados=findobj(line_registros, 'Type', 'line', '-and', 'Visible', 'off');
12.     if eliminados==0
13.         eliminados=[]; %Se pone en vacio([]) porque alguna versiones de Matlab devuelven 0,
           y length(0)=1;
14.     end
15.
16. hFig = ancestor(line_registros(1), 'figure');
17. h_total=findobj(hFig, 'Type', 'uicontrol', '-and', 'Tag', 'text_numTotal');
18. h_actual=findobj(hFig, 'Type', 'uicontrol', '-and', 'Tag', 'text_numActual');
19. h_elimina=findobj(hFig, 'Type', 'uicontrol', '-and', 'Tag', 'text_numEliminadas');
20.
21. %Calcular porcentajes
22. p_act=100*length(actuales)/length(line_registros);
23. str_act=num2str(p_act, '% 10.2f');
24.
25. p_elim=100*length(eliminados)/length(line_registros);
26. str_elim=num2str(p_elim, '% 10.1f');
27.
28. set(h_total, 'String', num2str(length(line_registros)));
29. set(h_actual, 'String', [num2str(length(actuales)) ' (' str_act '%)']);
30. set(h_elimina, 'String', [num2str(length(eliminados)) ' (' str_elim '%)']);

```

J. GUARDAR_RES

```

1. function val=guardar_res(handles)
2.     %VAL=1 GUARDADO
3.     %VAL=0 NO GUARDADO
4.     val=0;
5.     h_mediana=findobj(handles.axes3, 'Tag', 'Mediana3');
6.     result.mediana_curva=get(h_mediana, 'YData');
7.
8.     str_t_ini=get(handles.push_t_ini, 'String');
9.     str_t_fin=get(handles.push_t_fin, 'String');
10.    if strcmp(str_t_ini, 'Tiempo Inicial')==1 && strcmp(str_t_fin, 'Tiempo Final')==1
11.        %WARNDLG, NO HAY MARCADORES
12.        NO SE GUARDA
13.        val=0;
14.        errordlg('Para guardar, definir los Marcadores', 'Error');
15.    else
16.
17.        result.paum_ini=str2double(str_t_ini(5:end));
18.        result.paum_fin=str2double(str_t_fin(5:end));
19.
20.        if handles.modo_fs==1
21.            fs=getappdata(0, 'fs');

```

```

22.         result.paum_ini=floor(result.paum_ini*fs)/1000;%%Floor para muestras
enteras_____%%MIRAR SI ESTAN BIEN AJUSTADAS LAS MUESTRAS
23.         if result.paum_ini<=0
24.             result.paum_ini=1; %Muestra 0 no existe
25.         end
26.         result.paum_fin=floor(result.paum_fin*fs)/1000;%Dividido entre 1000
porque esta en (ms)
27.     end
28.
29.     result.set_manual=obtener_set_manual(); %Funcion que devuelve los indices
30.
31.     %Guardar
32.     if ischar(handles.FileName)
33.         result.file=handles.FileName;
34.         save([handles.ruta filesep 'res_' result.file],'result')
35.         val=1;
36.     else
37.         %Pedir Nombre al usuario
38.         answer=inputdlg('Introducir result.file:',...
39.             'Introducir el nombre del PAUM','X','on');
40.         %Seleccionar ruta para guardar
41.         result.file=answer{1};
42.         [FileName,PathName,FilterIndex] = uiputfile('.mat','Guardar
como...',result.file);
43.         val=1;
44.         if isequal(FileName,0) || isequal(PathName,0)
45.             val=0;
46.         end
47.     end
48. end

```