



UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA
INFORMÁTICA

Máster en Ingeniería de Sistemas y Control

**DESARROLLO DE UN PREDICTOR DE DISRUPCIONES
BASADO EN DETECCIÓN DE ANOMALÍAS**

DAVID ÁLVAREZ NUEVO

Dirigido por: Dr. JESÚS ANTONIO VEGA SÁNCHEZ

Co-director: Dr. SEBASTIÁN DORMIDO CANTO

Curso: 2014/2015 (Convocatoria de Septiembre)



MÁSTER EN INGENIERÍA DE SISTEMAS Y CONTROL

**DESARROLLO DE UN PREDICTOR DE DISRUPCIONES
BASADO EN DETECCIÓN DE ANOMALÍAS**

Proyecto de Fin de Máster de modalidad oferta específica propuesto por un
profesor

Realizado por: DAVID ÁLVAREZ NUEVO
Dirigido por: Dr. JESÚS ANTONIO VEGA SÁNCHEZ
Co-director: Dr. SEBASTIÁN DORMIDO CANTO



DESARROLLO DE UN PREDICTOR DE DISRUPCIONES BASADO EN DETECCIÓN DE ANOMALÍAS

Máster en Ingeniería de Sistemas y Control

Realizado por: DAVID ÁLVAREZ NUEVO

Dirigido por: Dr. JESÚS ANTONIO VEGA SÁNCHEZ

Co-director: Dr. SEBASTIÁN DORMIDO CANTO

Tribunal calificador:

Presidente: D./Da.
(Firma)

Secretario: D./Da.
(Firma)

Vocal: D./Da.
(Firma)

Fecha de lectura y defensa:
Calificación:



Autorización

Autorizamos a la Universidad Complutense y a la UNED a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firmado: David Álvarez Nuevo

Firma del alumno

A Lara...

Agradecimientos

En primer lugar, quiero agradecer al Dr. Sebastián Dormido Canto permitirme realizar este proyecto con él. Desde el primer día me ofreció toda su confianza para poder llevarlo a cabo. Me brindó la posibilidad de conocer el CIEMAT y el gran trabajo que realizan.

Junto con el Dr. Jesús A. Vega Sánchez, con su trabajo, me permitieron ver la gran cantidad de posibilidades laborables que ofrece el mundo de la ingeniería y la ciencia.

Dar las gracias, al Dr. Jesús A. Vega Sánchez, director del centro de recopilación de datos de fusión y tutor del proyecto, por permitirme desarrollar mi proyecto fin de máster bajo el marco de la fusión nuclear, hasta ahora totalmente desconocida por mí. Gracias por todas sus orientaciones, correcciones, tutorías y soporte. Su gran dedicación a la investigación han creado en mí todo un referente para futuros planes de trabajo y dedicación.

A todas las personas que han estado alrededor mía, que de algún modo han participado en mi vida y han influido en mi ánimo para poder finalizar esta etapa.

En especial, a mi familia, por su comprensión y apoyo en todo momento. Que me hace ver que el aprendizaje no tiene fin.

Resumen

Las interrupciones en dispositivos Tokamak son eventos muy peligrosos para la integridad de los propios dispositivos de fusión. Su extrema complejidad y su dependencia de fenómenos altamente no lineales, ha impedido hasta la fecha el desarrollo de una teoría satisfactoria. Por lo tanto, se utilizan métodos para predecir la ocurrencia de una interrupción y poder aplicar medidas de mitigación. Los dispositivos de fusión de siguiente generación (como ITER o DEMO) no podrán soportar muchas interrupciones. Por lo tanto, es necesario desarrollar predictores fiables que, con muy pocas muestras de entrenamiento, que alcancen una alta tasa de aciertos en la detección de interrupciones.

Actualmente la base de datos de cualquier dispositivo se compone de millones de señales. La mayoría de estas señales son de evolución temporal, con un número de muestras entre 10.000 y varios millones. Por ejemplo, en el tokamak JET (con una duración de descarga de 40 s), se han realizado más de 70.000 disparos y el número de señales que se adquieren en cada descarga puede ser superior a 10.000.

Por lo expuesto y dada la gran cantidad de datos que se dispone, es útil el uso de técnicas de reconocimiento de cambios en flujos de datos en el análisis en tiempo real como off-line. Este proyecto consiste en el desarrollo de una aplicación que detecte la aparición de una interrupción y active una alarma para poder proceder a parar la descarga. Para ello, el algoritmo se basará en mecanismos deslizantes de ventanas de datos de la señal. Por medio de la transformada de wavelet obtendremos valores característicos de las ventanas de datos que serán procesados por técnicas de clusterización y así generar la señal de referencia. Además almacenarán los datos generados en cada una de las etapas para su posterior visualización.

Lista de palabras clave

Detector de interrupciones, Transformada de Wavelet, Distancia de Mahalanobis, cluster, predictor, coeficientes de aproximación

Índice general

	<i>pag.</i>
Capítulo 1. La Fusión nuclear	1
1.1. Introducción.....	1
1.2. Fusión Nuclear como solución energética del futuro.....	2
1.3. Reacciones nucleares.....	3
1.4. Plasma de fusión.....	5
1.5. Confinamiento magnético.....	6
1.6. Dispositivos de fusión por confinamiento magnético.....	7
1.6.1. <u>Stellarator</u>	7
1.6.2. <u>Tokamak</u>	8
1.6.3. <u>Obtención de datos</u>	11
 Capítulo 2. Análisis diferido de Disrupciones	 14
2.1. Introducción.....	14
2.2. Momentos y tipos de disrupciones.....	15
2.2.1. <u>Momentos de las disrupciones</u>	15
2.2.2. <u>Tipos de disrupciones</u>	16
2.3. Naturaleza de los datos y técnica de clasificación.....	17
2.3.1. <u>Clasificación supervisada</u>	18
2.3.2. <u>Clasificación no supervisada</u>	18

Capítulo 3. Implementación de un sistema de predicción basado en una señal de bloqueo	20
3.1. Introducción y estado del arte.....	20
3.2. Detección de outlier aplicado a las interrupciones.....	22
3.3. Implementación del Predictor PBOD.....	24
3.3.1. <u>Introducción</u>	24
3.3.2. <u>Selección de las señales y uso de ventanas temporales</u>	24
3.3.3. <u>Transformada wavelet Haar para el análisis de señales</u>	27
3.4. Arquitectura del predictor.....	31
3.5. Técnicas de Clusterización.....	32
3.5.1. <u>Planteamiento general</u>	32
3.5.2. <u>Criterios de similitud. Similitud, divergencia y distancia</u>	34
3.5.3. <u>Criterios basados en distancias como indicadores de disparidad</u>	35
3.5.4. <u>Distancia Euclídea</u>	36
3.5.5. <u>Distancia Euclídea Normalizada</u>	37
3.5.6. <u>Distancia de Mahalanobis</u>	37
3.6. Aplicación del análisis Cluster en datos de fusión.....	38
3.7. Función de decisión.....	41
Capítulo 4. Desarrollo del entorno	44
4.1. Introducción.....	44
4.2. Ficheros de datos.....	44
4.3. Parametrización de las variables.....	45
4.4. Detalle del código desarrollado.....	46
4.4.1. <u>calculaCoeficientesDeAproximación</u>	46
4.4.2. <u>CalculaDistanciaMahalanobis</u>	47

4.4.3. <u>señalReferencia</u>	49
4.4.4. <u>Función principal</u>	49

Capítulo 5. Resultados obtenidos de la fase de entrenamiento del sistema de predicción **56**

5.1. Descripción de las descargas.....	56
5.2. Análisis de detección. Condición de disparo de la alarma.....	56
5.3. Análisis de descargas con ventanas de tiempo de 32 ms, 2 muestras de deslizamiento y nivel 4 de descomposición.....	56
5.3.1. <u>Descarga JPF 82460</u>	57
5.3.2. <u>Descarga JPF 82462</u>	61
5.3.3. <u>Descarga JPF 82463</u>	64
5.3.4. <u>Descarga JPF 82467</u>	68
5.3.5. <u>Descarga JPF 82468</u>	71
5.3.6. <u>Descarga JPF 82469</u>	73
5.3.7. <u>Descarga JPF 82471</u>	76
5.3.8. <u>Descarga JPF 82472</u>	81
5.3.9. <u>Descarga JPF 82474</u>	83
5.3.10. <u>Descarga JPF 82481</u>	86
5.4. Comparación de resultados utilizando distintos parámetros.....	90
5.4.1. <u>Descarga JPF 82460</u>	90
5.4.2. <u>Descarga JPF 82462</u>	92

Capítulo 6. Conclusiones y líneas de trabajo futuras	95
6.1. Conclusiones.....	95
6.2. Líneas de trabajo futuras.....	97
Capítulo 7. Bibliografía	99
Anexo I.....	102
Anexo II.....	111
Anexo III.....	116

Lista de figuras y de tablas

	<i>pag.</i>
<i>Figura 1.1. Representación de la reacción de fusión D-T</i>	4
<i>Figura 1.2. Secciones eficaces de las reacciones D-T, D-D, D-He³</i>	4
<i>Figura 1.3. Vista esquemática de un “Stellerator” (el TJ-II). En amarillo, se representa el plasma, en azul y celeste las bobinas y la cuarta parte de la cámara de vacío en gris</i>	7
<i>Figura 1.4. Vista esquemática de un Tokamak. La corriente que circula por el plasma es inducida mediante el transformador central. Esta corriente aporta la componente poloidal del campo magnético</i>	8
<i>Tabla 1.1 Tabla de parámetros generales del Tokamak JET</i>	9
<i>Figura 1.5. Ejemplo esquemático de una configuración con limitador y una con punto X. En la configuración con punto X la última superficie magnética no alcanza a entrar en contacto con el limitador</i>	11
<i>Figura 3.1. Evolución temporal de los valores medios de la corriente</i>	25
<i>Figura 3.2. Evolución temporal de la desviación estándar del espectro de Fourier</i>	25
<i>Figura 3.3. Evolución temporal de la descarga</i>	26
<i>Figura 3.4. Proceso de filtrado</i>	28
<i>Figura 3.5. Proceso de descomposición</i>	29
<i>Figura 3.6. Diagrama de dispersión de una descarga no disruptiva y una descarga disruptiva. (a) cluster compacto, (b) la presencia de outlier denotan la proximidad de una interrupción</i>	30
<i>Figura 3.7. Estructura del sistema PBOD</i>	32
<i>Figura 3.8. Representación bi-dimensional de los contornos equidistantes elípticos</i>	39
<i>Figura 3.9. Evolución temporal de la distancia de Mahalanobis de una descarga no disruptiva y una descarga disruptiva.</i>	39
<i>Figura 3.10. Línea continua roja: momento de la interrupción. Líneas rojas discontinuas: predicción del APODIS. Líneas azules discontinuas: predicción del LMP. Línea azul continua: predicción del PBOD</i>	40

<i>Tabla 3.1. Resultados de aplicar PBOD con distintos niveles de aproximación</i>	41
<i>Figura 3.11. Comparación entre LMP, APODIS y PBOD</i>	42
<i>Figura 4.1. Código de la función calculaCoeficientesDeAproximacion</i>	47
<i>Figura 4.2. Código de la función CalculaDistanciaDeMahalanobis</i>	48
<i>Figura 4.3. Código de la función senaReferencia</i>	49
<i>Figura 4.4. Código de la primera parte de la función main</i>	51
<i>Figura 4.5. Código de la segunda parte del main</i>	53
<i>Figura 5.1. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	57
<i>Figura 5.2. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	57
<i>Figura 5.3. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	58
<i>Figura 5.4. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	58
<i>Figura 5.5. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	59
<i>Figura 5.6. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$</i>	60
<i>Figura 5.6 bis. Evolución temporal de la señal de referencia aplicando zoom al momento de aparición de las inestabilidades</i>	60
<i>Figura 5.7. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	61
<i>Figura 5.8. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	61
<i>Figura 5.9. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	62
<i>Figura 5.10. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	63
<i>Figura 5.11. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	63
<i>Figura 5.12. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	64

<i>Figura 5.13. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	65
<i>Figura 5.14. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	65
<i>Figura 5.15. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	66
<i>Figura 5.16. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	66
<i>Figura 5.17. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$</i>	67
<i>Figura 5.17 bis. Evolución temporal de la señal de referencia aplicando zoom al momento de la aparición de inestabilidades</i>	68
<i>Figura 5.18. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	68
<i>Figura 5.19. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	69
<i>Figura 5.20. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	69
<i>Figura 5.21. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	70
<i>Figura 5.22. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	70
<i>Figura 5.23. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	71
<i>Figura 5.24. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	71
<i>Figura 5.25. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	72
<i>Figura 5.26. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	72
<i>Figura 5.27. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	73
<i>Figura 5.28. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	74

<i>Figura 5.29. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	74
<i>Figura 5.30. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	75
<i>Figura 5.31. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	75
<i>Figura 5.32. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	76
<i>Figura 5.33. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	77
<i>Figura 5.34. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	77
<i>Figura 5.35. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	78
<i>Figura 5.36. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	78
<i>Figura 5.37. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	79
<i>Figura 5.38. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$</i>	80
<i>Figura 5.38 bis. Evolución temporal de la señal de referencia aplicando zoom al momento de la aparición de interrupciones</i>	80
<i>Figura 5.39. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	81
<i>Figura 5.40. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	81
<i>Figura 5.41. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	82
<i>Figura 5.42. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	82
<i>Figura 5.43. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	83
<i>Figura 5.44. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	83

<i>Figura 5.45. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	84
<i>Figura 5.46. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	84
<i>Figura 5.47. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	85
<i>Figura 5.48. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	85
<i>Figura 5.49. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y al momento de la aparición de inestabilidades</i>	86
<i>Figura 5.50. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente</i>	87
<i>Figura 5.51. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés</i>	87
<i>Figura 5.52. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	88
<i>Figura 5.53. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	88
<i>Figura 5.54. Evolución temporal de la señal de referencia y valor del umbral de K_M</i>	89
<i>Figura 5.55. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$</i>	89
<i>Figura 5.56. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar,</i> <i>a) con una ventana temporal de 16 muestras con dos muestras de desplazamiento y un nivel 3 de descomposición</i> <i>b) no tiene diagrama de dispersión bi-dimensional puesto que el número de coeficientes es 4.</i>	90
<i>Figura 5.57. Evolución temporal de la distancia de Mahalanobis de la descarga</i> <i>a) para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición</i> <i>b) para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición</i>	91

<i>Figura 5.58. Evolución temporal de la señal de referencia</i>	91
a) <i>para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición</i>	
b) <i>para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición</i>	
<i>Figura 5.59. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar</i>	92
a) <i>con una ventana temporal de 16 muestras con dos muestras de desplazamiento y un nivel 3 de descomposición</i>	
b) <i>no tiene diagrama de dispersión bi-dimensional puesto que el número de coeficientes es 4.</i>	
<i>Figura 5.60. Evolución temporal de la distancia de Mahalanobis de la descarga</i>	92
a) <i>para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición</i>	
b) <i>para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición</i>	
<i>Figura 5.61. Evolución temporal de la señal de referencia</i>	93
a) <i>para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición</i>	
b) <i>para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición</i>	

CAPÍTULO 1

FUSIÓN NUCLEAR

1.1. Introducción

La producción y la utilización de la energía se han convertido en los últimos tiempos en objeto de discusión y preocupación por parte de la comunidad mundial. Estimaciones de la ONU indican que la población mundial alcanzará los 9400 millones de habitantes a mediados del siglo XXI. Este aumento inevitablemente generará un incremento de la demanda de energía. En los países desarrollados ya se están imponiendo criterios de ahorro energético. Sin embargo, el consumo en los países en vías de desarrollo es diez veces inferior que el de los países industrialmente avanzados. A medida que estos países mejoren su nivel de vida su demanda energética también aumentará con rapidez, y por lo tanto es previsible que la demanda de energía eléctrica no crezca al ritmo de la década anterior. Por ejemplo, en países como China o India, en los próximos años, alcanzarán una cifra de consumo cercana a la de los países occidentales, este impacto sobre el nivel de producción conllevaría una rápida disminución de las reservas mundiales de combustibles y dispararía la producción de contaminantes.

Los combustibles fósiles (carbón, petróleo, gas) son actualmente las principales fuentes de energía por su reducido coste. Estos recursos han sido masivamente explotados y las reservas existentes, principalmente de petróleo y gas, no garantizan su disponibilidad a largo plazo. Además, los residuos lanzados a la atmósfera durante su combustión tienen un impacto negativo en

el medio ambiente, en especial los gases de efecto invernadero y en particular el CO₂.

Las energías renovables constituyen una alternativa a corto plazo a las energías tradicionales, gracias a que la materia prima en la que se basan procede de una fuente inagotable. Además, su baja producción de contaminantes, restringida en la mayoría de los casos al proceso de conversión a formas de energía aprovechables por el ser humano, las hace muy atractivas. Sin embargo, también tienen desventajas importantes. Por ejemplo, su baja densidad, lo que dificulta su captación y concentración, su baja disponibilidad, su variabilidad asociada a los distintos periodos naturales, el elevado coste de implantación y explotación, el impacto ambiental visual o su baja eficiencia hacen que su uso sea aún limitado.

El enfoque nuclear se presenta como una solución poderosa ya que es capaz de producir grandes cantidades de energía. Esta puede ser generada mediante procesos de fisión nuclear, en las que se dividen núcleos de alto número másico, o de fusión nuclear, donde se juntan núcleos atómicos ligeros.

La energía producida mediante fisión, descubierta en 1939 por los investigadores alemanes Hahn y Strassmann, se genera como una reacción en cadena. Este tipo de reacciones, hoy en día, puede ser controlado con seguridad. Sin embargo, la extracción, enriquecimiento y utilización de materiales de alto peso atómico (como el uranio) para los procesos de fisión nuclear generan residuos radioactivos que tardan miles de años en perder su nocividad para el medioambiente. Debido a estas circunstancias, existen ciertos reparos en parte de la opinión pública para la producción de energía mediante la fisión.

Tanto productores como consumidores de energía se enfrentan a la urgente necesidad de ir a sistemas globales de energía más sostenibles y eficientes.

1.2. Fusión Nuclear como solución energética del futuro

La fusión nuclear aparece como una de las opciones más prometedoras en la producción de la energía, esta opción, propuesta ya desde los años 40, parece la solución perfecta a todos los problemas planteados.

La producción de energía por fusión utiliza como combustible los dos isótopos pesados del hidrógeno: el deuterio y el tritio. El primero se encuentra

en relativa abundancia en la naturaleza, como en el agua del mar, de donde podría extraerse de manera virtualmente ilimitada. En cuanto al tritio, es un elemento radiactivo de muy corta vida media, que puede producirse artificialmente a partir del litio. La energía obtenida mediante la unión de estos dos átomos es del orden de 4 veces mayor a la generada mediante la fisión, además, no tiene el riesgo de accidentes nucleares incontrolados, requiere un preciso y continuo control del combustible para la operación del dispositivo, y tampoco genera residuos radioactivos de alta actividad, ya que el período de semidesintegración del tritio es de 12 años.

Por otro lado, en el caso de la contaminación atmosférica, la fusión comparte la gran ventaja de las centrales de fisión. Es decir, al no existir combustión, el único gas que se vierte a la atmósfera es vapor de agua. El desarrollo exitoso de esta tecnología acabaría con la dependencia del mundo occidental de los combustibles fósiles, y también evitaría los problemas sociales que generan las centrales nucleares convencionales, dando a la sociedad una fuente de energía limpia, segura, de alta potencia y prácticamente inagotable.

1.3. Reacciones nucleares

La fusión es el proceso en el que dos núcleos de átomos ligeros (como el deuterio y el tritio) se unen formando elementos más pesados.

La energía generada es el resultado de la conversión de dos núcleos atómicos que se unen para formar uno de mayor masa atómica.

En la década de los 50 se formalizaron los primeros intentos para la producción de energía mediante fusión [Braams, 2002]. Para que los núcleos cargados positivamente sean capaces de superar las fuerzas electrostáticas de repulsión que existen entre ellos y se acerquen lo suficiente como para producir reacciones de fusión a un ritmo adecuado, hay que alcanzar temperaturas del orden de centenares de millones de grados (de aquí procede la denominación de fusión TERMO-nuclear). A estas temperaturas, el gas se encuentra en estado de plasma.

Las reacciones de fusión. Las reacciones de fusión más sencillas de realizar en la tierra son las que se consiguen a partir de isótopos del hidrógeno.

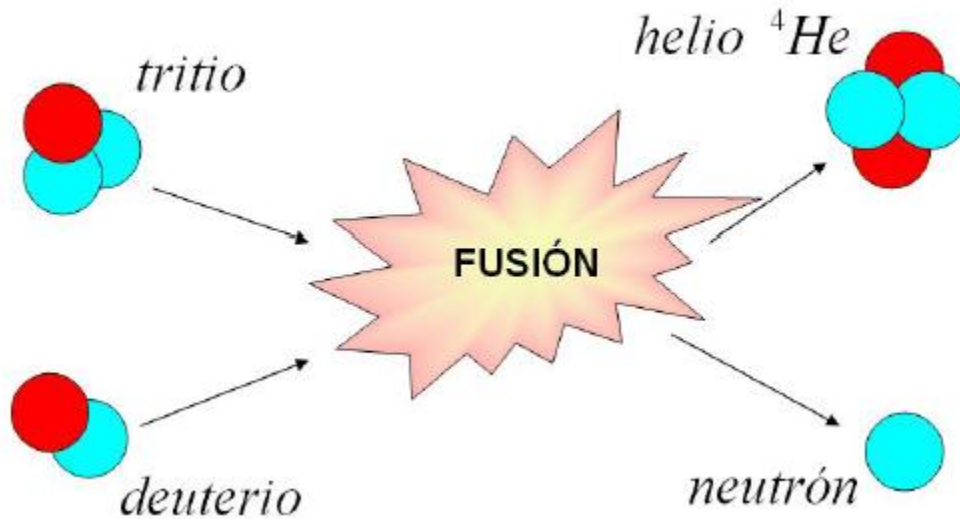


Figura 1.1. Representación de la reacción de fusión D-T

Un núcleo de deuterio se une a uno de tritio generando un núcleo de helio y un neutrón, liberando como consecuencia una cantidad de energía

Dentro de las posibles reacciones de fusión, las más atractivas son las denominadas como ciclo D-T (deuterio-tritio) por tener la mayor sección eficaz a menores temperaturas. En la figura 1.2, se puede observar que por debajo de 1 keV éstas tienen unos valores muy bajos, aunque crecen muy deprisa. En torno a temperaturas del orden de 10 keV (alrededor de 100 millones de grados), la reacción entre el D y T es varios órdenes de magnitud más probable que las restantes [Sheffield, 1994]. Por ello se ha elegido esta reacción para ser utilizada en la primera generación de reactores de fusión.

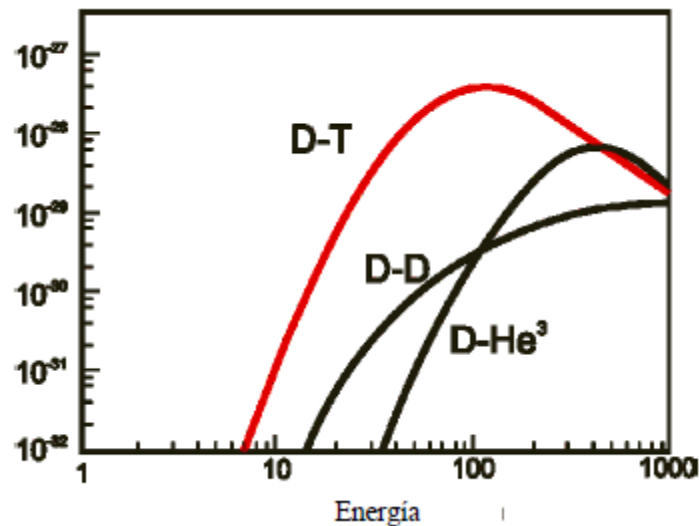
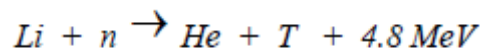


Figura 1.2. Secciones eficaces de las reacciones D-T, D-D, D-He³

El deuterio es un isótopo estable del hidrógeno presente en la naturaleza en una proporción de 1 por cada 6670 átomos de hidrógeno y su separación

isotópica es relativamente fácil. A pesar de esta pequeña proporción, la cantidad de deuterio contenida en los océanos sería prácticamente inagotable. El tritio es un isótopo inestable del hidrógeno, decae radiactivamente con una vida media de 12,36 años y no existe en la naturaleza, por lo tanto debe ser obtenido.

Ciclo del tritio. En la reacción D-T, el 80% de la energía la transporta el neutrón resultante, este neutrón puede servir para producir tritio al reaccionar con el litio según la ecuación:



Por lo tanto, si se rodea al recinto donde ocurren las reacciones de fusión de una envoltura de litio (llamado manto fértil), el reactor sólo necesitará una carga inicial de tritio, ya que posteriormente él mismo generará el tritio necesario. La fusión entre el *D-T* tiene, además de la complicación del manejo del tritio, otra dificultad añadida, que es la producción de un neutrón altamente energético capaz de modificar las propiedades mecánicas y radiactivas de los materiales estructurales del reactor. Desde este punto de vista, la fusión entre el *D-He³*, es mucho más atractiva, al ser una reacción que no produce neutrones, pero requiere temperaturas más altas. Además, el *He³* no existe en nuestro planeta, por lo que esta reacción no se considera actualmente en ningún escenario bajo estudio.

1.4. Plasma de fusión

Con el fin de superar la repulsión electrostática entre los núcleos que se quiere fusionar en condiciones de laboratorio, es necesario invertir mucha energía, esto se puede lograr mediante su calentamiento a temperaturas muy altas. En tales condiciones, la materia se encuentra en un estado gaseoso altamente ionizado y exhibe un comportamiento colectivo que llamamos plasma.

El gas se convierte en plasma cuando se calienta lo suficiente como para causar que los electrones queden libres del átomo. Las características de un plasma son muy diferentes de las de un gas ordinario. A diferencia de los gases ordinarios, los plasmas son buenos conductores que no solo son afectados significativamente por las fuerzas electromagnéticas, sino que también pueden llegar a generarlas. Esta autogeneración de fuerzas, como resultado de los movimientos colectivos de partículas cargadas y las corrientes eléctricas en el plasma, producen un comportamiento muy complejo que aún no se ha llegado a comprender en su totalidad. Comprender y describir los complejos sistemas dinámicos de los plasmas es un reto científico fundamental.

Para que un reactor de fusión sea energéticamente rentable, las reacciones deberán generar una energía significativamente superior a la aplicada para la creación y el mantenimiento del plasma. Según el criterio de Lawson la energía de las reacciones (en particular de las partículas) resultan suficientes para calentar el plasma sin necesidad de un aporte energético externo. Lawson definió inicialmente un umbral mínimo, mediante el triple producto de la densidad electrónica, la temperatura electrónica y el tiempo de confinamiento de la energía del plasma. Este triple producto, considerando un rendimiento del 33%, debe ser igual o superior a para que el plasma alcance la ignición.

1.5. Confinamiento magnético

Para poder calentar el combustible a la temperatura necesaria para que se produzcan suficientes reacciones de fusión, es necesario mantenerlo aislado, confinado en un cierto volumen y alejado del contacto con otros materiales. Hoy en día, los dos métodos investigados son el confinamiento inercial y el magnético.

El confinamiento inercial, se genera mediante el calentamiento y compresión de esferas de hidrógeno (las densidades alcanzadas son del orden de 10^{23} m^{-3} y las temperaturas de decenas de KeV) durante períodos muy cortos (nanosegundos). Para la compresión y el calentamiento de las esferas se requieren láseres de alta potencia, cuyo desarrollo y posible aplicación en estado continuo plantean problemas de difícil solución.

El confinamiento magnético, el plasma se confina mediante campos magnéticos que actúan creando una trampa magnética para las partículas cargadas del plasma. En las máquinas toroidales (que han sido las más desarrolladas), el campo magnético final se crea superponiendo a un campo magnético, según el eje del toro (toroidal), otro campo en dirección acimutal (poloidal).

En este sistema puramente toroidal, la curvatura y el gradiente del módulo de campo magnético provoca una deriva vertical en dirección opuesta para electrones e iones. Ésta crea un campo eléctrico que induce a su vez una deriva hacia el exterior en todo el plasma, por lo esta configuración toroidal es intrínsecamente inestable. Para evitar esta separación de cargas es necesario retorcer las líneas de campo mediante una componente magnética adicional (poloidal). Una sola línea de campo describe así una superficie de flujo. El transporte perpendicular al campo magnético B está restringido por la fuerza de Lorentz, y por tanto, en una misma superficie de flujo (dirección paralela a las líneas de campo) los parámetros del plasma se pueden considerar

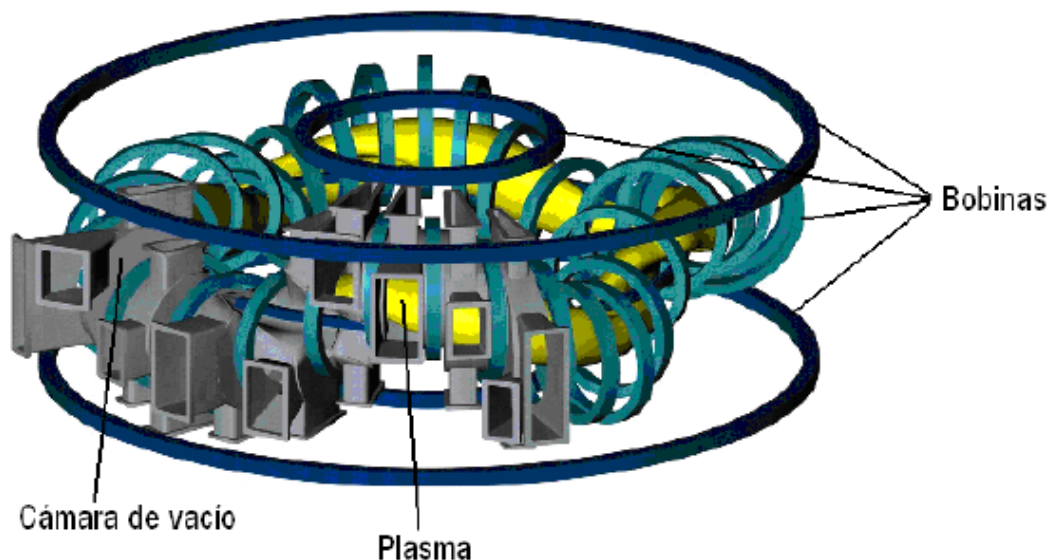
prácticamente constantes, mientras que las principales variaciones se producen en la dirección perpendicular a las líneas de campo magnético.

1.6. Dispositivos de fusión por confinamiento magnético

Las dos principales familias de dispositivos de fusión por confinamiento magnético, son el *STELLARATOR* y el *TOKAMAK*.

1.6.1. *Stellarator*

El Stellarator (ver *figura 1.3*) es un dispositivo toroidal de confinamiento magnético que fue propuesto por el astrofísico Lyman Spitzer en 1951. Los dispositivos de la clase Stellarator no tienen simetría toroidal, y los campos magnéticos necesarios para confinar el plasma son generados exclusivamente por bobinas externas, sin requerir una corriente eléctrica toroidal en el plasma. Esta característica hace que su confinamiento sea inherentemente estacionario, siempre que se mantengan las corrientes en las bobinas externas. Salvando estas diferencias, la física de los plasmas producidos en Stellarators y Tokamaks es similar. Debido a que el diseño y la construcción de los Stellarators es más complicado, su desarrollo ha ido siempre una generación por detrás de los Tokamaks, no alcanzando el rendimiento de estos últimos.



*Figura 1.3. Vista esquemática de un "Stellarator" (el TJ-II)
En amarillo, se representa el plasma, en azul y celeste las bobinas y la cuarta parte de la cámara de vacío en gris*

1.6.2. Tokamak

La configuración Tokamak (ver *figura 1.4*) fue propuesta por dos científicos rusos, Tamm y Sakharov. El nombre se deriva de las palabras rusas cuyo significado puede traducirse como “cámara toroidal con campo magnético”. La estructura de campo magnético se genera mediante bobinas toroidales y la componente poloidal la produce una corriente que se hace circular por el plasma. Esta corriente se induce mediante un transformador central.

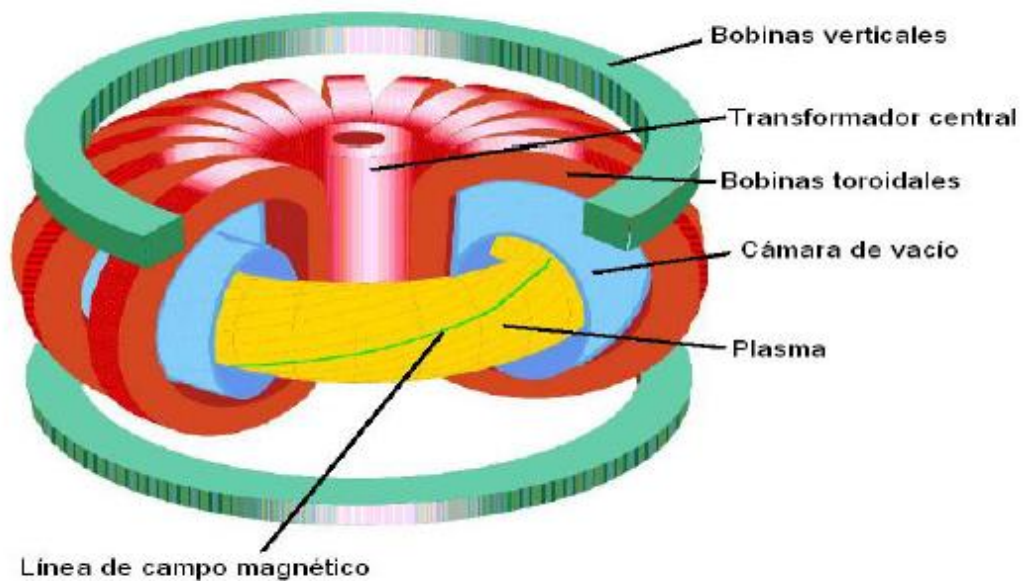


Figura 1.4. Vista esquemática de un Tokamak

La corriente que circula por el plasma es inducida mediante el transformador central. Esta corriente aporta la componente poloidal del campo magnético

Hoy en día la configuración Tokamak es la más difundida, es el dispositivo con el que se han conseguido los mejores resultados en el campo de la investigación en fusión nuclear controlada. Es simétrico con respecto al eje central, con bobinas externas alrededor del toro que generan un campo magnético toroidal muy alto (varios teslas).

Desde el punto de vista de ingeniería, presenta menos complicaciones para su construcción que un “Stellarator”. Sin embargo, una de sus principales desventajas se debe a pérdidas abruptas e inevitables de la energía del plasma (que ocurren sólo en este tipo de configuraciones), llamadas **disrupciones**.

Para el desarrollo del proyecto se utilizará señales generadas por este tipo de dispositivo. En concreto se utiliza el JET (“*Joint European Torus*”) es el dispositivo de fusión por confinamiento magnético más grande del mundo. Su operación comenzó en 1983 y fue la primera instalación capaz de producir una cantidad significativa de energía mediante fusión controlada (cerca de 2 MW) en la campaña de deuterio-tritio de 1991. Consta de 32 bobinas en forma de D

que se encuentran equi-espaciadas alrededor de la máquina. Estas bobinas son utilizadas para generar la componente toroidal de campo magnético. El transformador utilizado para inducir la corriente del plasma y a su vez generar la componente poloidal del campo situado en el centro del dispositivo (Figura 1.4).

Debido a la necesidad de un transformador para generar la corriente del plasma, la operación de JET (y la de todos los Tokamaks) es pulsada. Estos pulsos pueden generarse en JET con una frecuencia de uno cada veinte minutos y cada uno de ellos puede durar hasta sesenta segundos. El plasma queda confinado en una cámara de vacío en forma de toroide cuyo radio mayor es de 2.96 metros, teniendo las secciones transversales unas dimensiones alto/ancho de 4.2 metros y 2.5 metros respectivamente. Los parámetros generales del diseño original de JET se detallan en la Tabla 1.1.

Radio mayor del plasma	2.96m
Radio menor del plasma	2.10m(vert)-1.25m(horiz)
Longitud temporal de pulso (flat-top ⁹)	20s
Peso del núcleo de hierro	2800t
Potencia de las bobinas de campo Toroidal	380MW
Campo magnético toroidal (en el eje del plasma)	3.4T
Corriente del plasma	3.2MA (Plasma circular) 4.8MA (Plasma con forma D)
Potencia adicional de calentamiento	25MW

Tabla 1.1 Tabla de parámetros generales del Tokamak JET

Además del calentamiento óhmico⁸ y del NBI, el dispositivo consta de otros sistemas de calentamiento: Radio Frecuencia Ciclotrónica Iónica (ICRF, del inglés Ion Cyclotron Radio Frequency) y Corriente Híbrida Menor (LHCD, del inglés Lower Hybrid Current Drive).

Aunque el sistema LHCD puede tener un efecto de calentamiento ineficiente, es útil al generar en el plasma corrientes de varios MA. Las ondas electromagnéticas producen frecuencias resonantes mediante la modulación de un haz electrónico. En JET existen 24 tubos klystrons instalados en 6 módulos independientes. La onda electromagnética se transmite a la antena LHCD por un complejo sistema de guías de onda. Estas guías de onda consisten en

conductores metálicos huecos con secciones del mismo tamaño que la amplitud de la onda transmitida. La antena debe ser montada directamente en la pared interior del dispositivo y estar tan cerca como sea posible del plasma.

El calentamiento por ICRF se utiliza en la mayoría de los experimentos de JET. Es resonante con la segunda frecuencia armónica de giro de los iones más comunes en plasmas de JET (deuterio) o con la frecuencia de giro de otras especies (como el tritio o el helio). La potencia total del sistema ICRF de JET alcanza los 32 MW aunque en la práctica suele aplicarse sólo una parte de ella. Estas potencias son enormes si se las compara, por ejemplo, con una emisora de televisión, donde un transmisor de 50 kW ya es considerado más que suficiente. Las líneas de transmisión que conducen las ondas desde los generadores hasta el Tokamak son cables coaxiales de bajas pérdidas. Estos cables son conductores en forma de tubo metálico que contienen en el centro, aislado, otro conductor. Este tipo de cables se utiliza en la mayoría de las transmisiones a altas frecuencias, como en el caso de las señales satelitales de televisión que son capturadas por una antena parabólica y que mediante cables coaxiales conducen las señales a un televisor. Sin embargo en JET, debido a las altas potencias del sistema, ha sido necesario ensanchar cientos de metros de estos cables hasta los 20 cm de diámetro, confiriéndoles una apariencia de tubería. Las líneas de transmisión terminan en 4 antenas ICRF instaladas en la pared interior de la máquina. Las ondas electromagnéticas ICRF no se pueden propagar a través de la cámara de vacío, ya que su longitud de onda es demasiado grande, por lo que la antena debe ser ubicada lo más cerca posible del plasma.

Como se explicó anteriormente, en todos los Tokamaks la componente poloidal del campo magnético se genera mediante la corriente del plasma y la toroidal mediante bobinas externas. La superficie magnética puede delimitarse o bien mediante una estructura sólida (placas llamadas limitadores) o estar completamente definidas mediante una configuración llamada punto X (*X-point configuration*), tal como puede apreciarse en la figura. 1.5. El principal inconveniente de los limitadores es que son una fuente de impurezas que influyen negativamente en el plasma. Por otro lado, en plasmas elongados se genera una superficie magnética particular llamada separatriz (*separatrix*) que divide dos topologías magnéticas diferentes. Dentro de la separatriz se forman las superficies magnéticas cerradas y anidadas requeridas para un buen confinamiento. Fuera de la separatriz las superficies se encuentran abiertas. Una configuración con punto X es más estable comparada con las resultantes del uso de limitadores pero el volumen del plasma obtenido es menor. En general está aceptado que la configuración de punto X es la más adecuada para los Tokamaks y es la que actualmente se utiliza en JET.

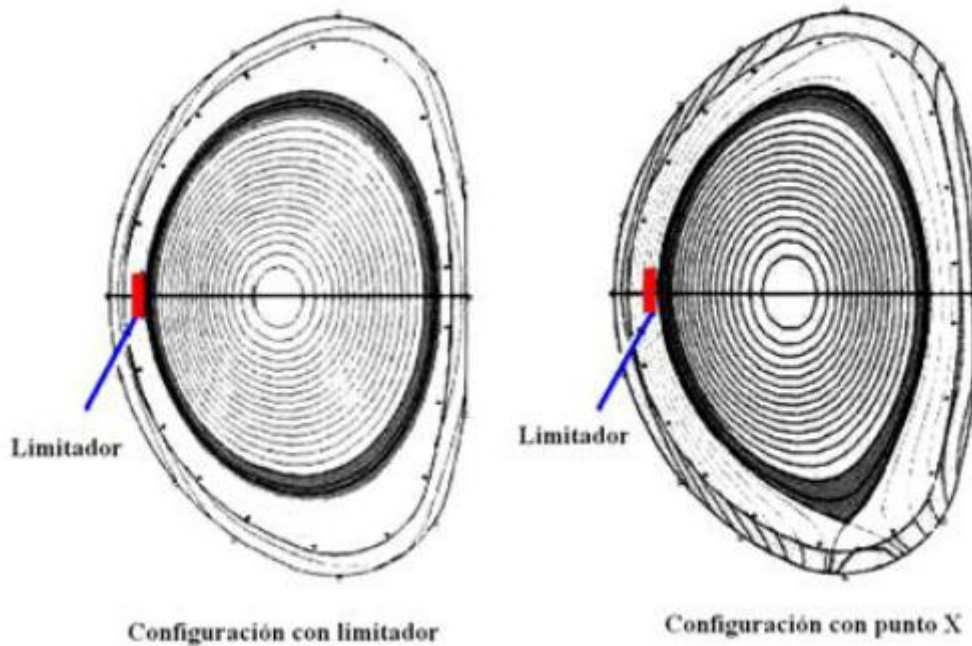


Figura 1.5. Ejemplo esquemático de una configuración con limitador y una con punto X. En la configuración con punto X la última superficie magnética no alcanza a entrar en contacto con el limitador

1.6.3. Obtención de datos

Los dispositivos desarrollados hasta el momento tienen como principales objetivos el estudio de los fenómenos físicos que ocurren en el plasma así como acercarse a las condiciones de ignición. El seguimiento de la evolución del plasma no es una tarea sencilla. Tal como se ha mencionado con anterioridad, el plasma puede encontrarse a temperaturas de decenas de KeV (unas diez mil veces la temperatura de la superficie del sol) y está confinado dentro de una cámara de vacío a cuyo interior es difícil acceder. Para la inspección del plasma y su evolución se requieren sistemas especiales de medida, llamados diagnósticos en la nomenclatura habitual de fusión. Los sistemas de diagnóstico son instrumentos especialmente desarrollados o adaptados para este tipo de complejas mediciones. Cada dispositivo para el estudio de la fusión dispone de un conjunto de diagnósticos que posibiliten determinar las características principales de los plasmas producidos.

Actualmente en JET se utilizan unos 80 sistemas de diagnósticos. Además, existen alrededor de 20 más que en proceso de construcción. Algunos de los más comúnmente utilizados en JET se resumen en la siguiente lista:

- Bobinas magnéticas: proveen mediciones de campo magnético (incluyendo la detección de modos magneto hidrodinámicos), corriente, inductancia y energía del plasma.

- Dispersión Thompson: miden temperatura y densidad electrónica, así como perfiles de densidad.
- Cámaras ultra-rápidas de luz visible.
- Sistemas de interferometría: miden densidad integrada de línea.
- Antenas ECE (Electron Cyclotron Emission): miden temperatura y densidad electrónica.
- Espectrómetros de luz visible, ultravioleta y rayos X: miden temperaturas y densidades.

CAPÍTULO 2

ANÁLISIS DIFERIDO DE DISRUPCIONES EN TOKAMAK

2.1. Introducción

La configuración Tokamak se presenta como una seria opción para el desarrollo de un futuro reactor. Sin embargo, se ve sometida con frecuencia a pérdidas abruptas del confinamiento del plasma llamadas disrupciones [Schuller, 1995]. Estas repentinas inestabilidades que causan el fin de la descarga, además de afectar a la continuidad de la operación del dispositivo, pueden constituir un serio factor de riesgo para su integridad. Durante una disrupción, en períodos de tiempo del orden de milisegundos, el plasma excede sus límites operacionales con la resultante pérdida de energía y corriente. Durante la primera fase de la disrupción es posible que se transfieran altísimas cargas térmicas a los componentes de la primera pared de la máquina. Posteriormente, se generan grandes corrientes que son inducidas en la cámara de vacío y en las estructuras aledañas, provocando fuerzas capaces de causar daños considerables. Actualmente, su aparición durante la operación es inevitable, especialmente en configuraciones de alto rendimiento. La necesidad de mitigar las consecuencias de este evento físico mediante su temprana detección es esencial, con el tiempo, se planea obtener plasmas con mayores corrientes y de tales energías que una sola disrupción podría provocar daños severos en varias partes del dispositivo, desde la erosión de los componentes expuestos al plasma (aquellos que recubren internamente la cámara de vacío y

resto de sistemas próximos a los flujos de partículas y energía procedentes del plasma) a daños en su estructura.

La caracterización física de las disrupciones para su posible predicción y control es extremadamente compleja, debido a la gran cantidad de variables involucradas en el fenómeno y la relación altamente no lineal entre ellas. Desafortunadamente, solamente existen modelos teóricos parciales que no son capaces de tratar las disrupciones de manera fiable. Debido a esta falta de teoría desde primeros principios, habitualmente se utilizan modelos estadísticos de aprendizaje para su predicción.

Algunas disrupciones ocurren considerablemente más rápido que otras y por lo tanto, como se hablará a continuación, pueden clasificarse según su tipo.

2.2. Momentos y tipos de disrupciones

2.2.1. Momentos de las disrupciones

Las disrupciones pueden ocurrir repentinamente, sin haber tenido comportamientos anormales obvios o después de periodos del orden de cientos de milisegundos, en los cuales, aparecen diferentes inestabilidades que se potencian hasta terminar abruptamente la descarga. De todas maneras, está generalmente aceptado que el fenómeno evoluciona en una serie de fases o momentos.

La primera fase se la llama “*pre-precursor*” o “*evento inicial*”. Allí la energía radiada o la densidad se incrementan debido a anomalías externas (como por ejemplo fallos mecánicos o de la operación) o internas (comportamientos imprevistos) del plasma, aunque esas variaciones suelen ser sutiles y por lo tanto la posibilidad de que sean detectadas es mínima.

Durante la “*fase de precursores*”, las irregularidades iniciadas en la fase anterior alcanzan un punto crítico determinado por la aparición de inestabilidades crecientes y detectables MHD (Magneto Hidrodinámicas). La importancia de esta fase para una detección temprana es fundamental: el fenómeno comienza a ser evidente y además, si es detectado con la suficiente rapidez, es posible iniciar acciones de control para evitarlo o al menos atenuar su efecto. Teóricamente estas inestabilidades comienzan como islas magnéticas que rotan alrededor del eje magnético del plasma y que crecen de tamaño exponencialmente en pocas decenas de milisegundos hasta llevar al plasma a la disrupción.

A la tercera fase se la conoce como "*fase rápida*". En esta, en tiempos del orden de milisegundos, el perfil radial de corriente se aplana y la temperatura en el centro del plasma cae con rapidez.

Y por último está la "fase de apagado". Durante el "*apagado térmico*" la temperatura electrónica cae bruscamente dejando al plasma en un estado altamente resistivo. El calentamiento óhmico resultante llega a ser del orden del Gigawatt. El aumento de la resistividad deriva parte de la corriente toroidal, hacia la cámara de vacío. Estas corrientes inducen momentos de gran magnitud en las estructuras colindantes a la primera pared, las cuales sumadas a las cargas térmicas transferidas en escalas de tiempo menores a 1 ms pueden causar daños severos al dispositivo.

2.2.2. Tipos de interrupciones

Aún no se conoce con exactitud el motivo por las que aparecen las interrupciones, sin embargo, existen diferencias en su desarrollo inicial y en las fases finales (en las que se pierde la energía) en virtud de las cuales si podemos establecer una clasificación dependiendo de su evolución. A continuación, mencionamos los tipos de interrupciones más comunes.

- Interrupciones de **límite de densidad**: se pueden describir como una inestabilidad que comienza cuando la densidad se eleva más allá de un umbral o cuando existen impurezas en el plasma. La energía radiada crece y consecuentemente se puede medir una caída de la temperatura. Además, el perfil de corriente se encoje como consecuencia del aumento de la resistencia del borde del plasma. Por propiedades inductivas del plasma, el *apagado de corriente* produce corrientes que circulan en el plasma frío fuera de la última superficie magnética cerrada transfiriendo grandes fuerzas a la cámara de vacío.
- Interrupciones de **mode lock**: son inestabilidades MHD que se producen debido a gradientes de temperatura o campos eléctricos externos produciendo islas magnéticas, las cuales crecen hasta terminar en una interrupción.
- Interrupciones **high radiated power**: producidas por impurezas en el plasma. La evolución de tales interrupciones se asemeja a las de límite de densidad pero con la diferencia que el incremento de energía radiada se produce ligeramente antes que el aumento en la densidad.

- Las interrupciones de **límite vertical** o **desplazamiento vertical**: ocurren normalmente en configuraciones de plasmas elongados. Muy rara vez el sistema de control permite este tipo de condiciones y por lo tanto en general se producen cuando los campos de estabilización vertical se desactivan expresamente para experimentos puntuales.
- También transiciones de regímenes de alto a bajo confinamiento (“**H-mode/L-mode disruptions**”) a altas densidades pueden desarrollar inestabilidades causantes de interrupciones.
- Las interrupciones de **Alta Beta** están vinculadas a incrementos de la presión del plasma que desencadenan una inestabilidad MHD. De hecho, beta es un parámetro Magneto Hidrodinámico (MHD) que mide la presión de confinamiento y por lo tanto se la considera directamente relacionada con los límites de estabilidad. En tokamak estos límites no se alcanzan en condiciones de operación normal y por lo tanto este tipo de interrupciones no es frecuente.
- Finalmente, las ITB (“**Internal Transport Barriers**”), son modos que suelen aparecer en las configuraciones etiquetadas como “*escenarios avanzados*” y su detección es particularmente complicada dado el breve margen entre la fase de *precursor* y la de *apagado de corriente*.

En este trabajo no se va a clasificar los tipos de interrupciones, si se ha creído conveniente detallar las principales características de cara a futuros trabajos.

2.3. Naturaleza de los datos y técnicas de clasificación

Los dispositivos de fusión generan una gran cantidad de datos en cada descarga, estos se obtienen a través de diagnósticos y se almacenan en bases de datos para un tratamiento diferido. Normalmente es imposible su análisis en tiempo real debido a la gran cantidad de datos y a la corta duración de las descargas (alrededor de 30 s en el Tokamak JET). Cada descarga adquiere el mayor número de datos posible para tratar de conocer las propiedades del plasma con la mayor precisión posible. El Tokamak JET ha llegado a adquirir hasta 10GB de datos en una descarga y su base de datos ocupa unos 100 Tbytes. Teniendo en cuenta esta gran cantidad de datos almacenada, resulta imposible a efectos prácticos analizar toda la información. Para facilitar estas labores, es necesario utilizar técnicas inteligentes de acceso a datos, de creación automática de bases de datos, y de generación de modelos basados

en aprendizaje automático. Como parte del aprendizaje automático, existen técnicas de clasificación que permiten agrupar muestras de acuerdo a ciertos criterios. Dependiendo de si se conocen o no las clases donde se van a ir clasificando las muestras, en estos casos se habla de clasificación supervisada o no supervisada respectivamente.

Estas técnicas de clasificación permiten obtener características en grupos de muestras con el fin de detectar comportamientos anómalos en algunas de ellas.

2.3.1. Clasificación supervisada

Este tipo de clasificación cuenta con un conocimiento a priori, es decir, se parte de modelos ya clasificados (objetos agrupados que tienen características comunes) para la tarea de clasificar un objeto dentro de una clase. Se pueden diferenciar dos fases:

- Una primera fase de entrenamiento donde a partir de unos datos de entrada se genera el modelo.
- La segunda fase, llamada fase de pruebas, el sistema de clasificación debe ser capaz de estimar la clase de nuevos objetos (diferentes a los utilizados en la fase de entrenamiento).

2.3.2. Clasificación no supervisada

A diferencia de la clasificación supervisada no se posee un conocimiento a priori de las categorías involucradas en el proceso de clasificación. A la clasificación no supervisada se la suele llamar también *clustering*.

En este tipo de clasificación, se dispone de muestras que tienen un conjunto de características, de las que no se sabe a qué clase o categoría pertenecen, siendo la finalidad el descubrimiento de grupos de muestras cuyas características afines permitan separar las diferentes clases.

En el siguiente apartado se explicará las técnicas de *clustering* para ver cómo se pueden aplicar en la detección de interrupciones en los dispositivos de fusión

CAPÍTULO 3

IMPLEMENTACIÓN DEL SISTEMA DE PREDICCIÓN BASADO EN UNA SEÑAL DE BLOQUEO

3.1. Introducción y estado del arte

Durante la primera fase de la disrupción es posible que se transfieran altísimas cargas térmicas a los componentes de la primera pared de la máquina, posteriormente, se generan grandes corrientes que son inducidas en la cámara de vacío y en las estructuras aledañas, provocando fuerzas capaces de causar daños considerables. A medida que estas inestabilidades empiezan aparecer la amplitud de las señales que se usan para detectarlas comienza a crecer. Rutinariamente, se utilizan estos valores de amplitud como un método para detectar las disrupciones causadas por las inestabilidades. Sin embargo, se ha demostrado que los predictores basados en métodos de aprendizaje en máquina generales (Máquinas de Vectores Soporte o Predictores de Venn entre otros) obtienen mejores resultados (en términos de tasa de éxito y los tiempos de advertencia) que simples predicciones basadas en una señal de bloqueo que se activará cuando se sobrepasa un umbral. Por ejemplo, en JET, el sistema de predicción, *Advanced Predictor Of DISruptions (APODIS)*, supera la capacidad de predicción del predictor Locked Mode (LMP), basado en un criterio de umbrales, la tasa de éxito Apodis y el tiempo medio de advertencia son alrededor del 82% y 274 ms, respectivamente. Las cantidades equivalentes

para el LMP son 67% y 255 ms. La razón de esta diferencia es el exhaustivo proceso de formación *APODIS*. Este está basado en una amplia base de datos y realizado con máquinas vectores soporte. Este trabajo se desarrolló con un sistema de predicción cuya arquitectura consiste en dos capas. En la primera capa se entrenan una serie de sistemas de clasificación para analizar secuencialmente los experimentos simulando el proceso de tiempo real. Como resultado, cada uno de los clasificadores provee una predicción de la inminencia o no de una interrupción. La determinación de activar o no una alarma depende de la combinación de los valores de salida de los clasificadores de primera capa. Esta combinación se realiza por otro clasificador en una segunda capa (llamado función de decisión) que también está basado en Máquinas de soporte de vectores. Esta segunda capa del sistema predictor es un aspecto completamente original y una de las causas de la mejora de los resultados obtenidos con respecto a otros modelos anteriores.

Tratando de evitar el uso de gran cantidad de descargas en el proceso de entrenamiento, recientemente, se ha desarrollado una nueva alternativa en la predicción de interrupciones que "parte desde cero". Son predictores con altas tasas de aprendizaje cuyo proceso de aprendizaje comienza con la primera interrupción. Estos sistemas de predicción están capacitados después de cada alarma fallida añadir descargas de interrupciones y no interrupciones a la base de datos. Modifican su conducta durante su ejecución, atendiendo a los cambios que se producen en su entorno o en el propio programa. En JET con el ILW, estos predictores adaptativos muestran tasas de éxito alrededor del 83% y los tiempos promedio de advertencia son de 244 ms.

Una opción más avanzada para la predicción de la interrupción sería el uso de predictores inteligentes que comienzan su proceso de aprendizaje con cada nueva descarga y sin la necesidad de la información anterior de las descargas anteriores.

En este trabajo se pretende modelar un sistema de detección de interrupciones basado en una señal de modo de bloqueo que no necesita el conocimiento de estados anteriores para su propósito. El sistema de predicción se basa en el reconocimiento automático de los cambios (detecciones anómalas) que se producen en las señales de entrada a través de la identificación de los valores atípicos (outlier) en el flujo de datos. Debido a esta razón, a este tipo de sistema de predicción se le llama predictor basado en la detección de valores atípicos (PBOD "**P**redictor **B**ased on **O**utlier **D**etection").

Como veremos más adelante, en PBOD, las muestras se procesan en ventanas de tiempo de un tamaño variable, estos valores constituyen el flujo de datos a analizar secuencialmente. Cerca de una interrupción, el modelo

detectará un cambio en el flujo de datos, momento en el que se activará una alarma.

En el apartado 3.2 se explica los conceptos de la detección de outlier aplicadas a las interrupciones.

En la sección 3.3, en una primera parte se habla del tipo de señales que se utilizan y de la obtención de las ventanas temporales a las que se les aplicará mecanismos de deslizamiento y comprensión para obtener los valores representativos. A continuación se explica la técnica de la transformada de wavelet Haar como solución para la compresión de las muestras.

En la sección 3.4 se da una visión general de la arquitectura del sistema de predicción, formado por tres capas. La primera capa, pre-procesado de las señales y obtención de los datos. Segunda capa, donde se aplican técnicas de clasificación para la detección de cambios en el flujo de datos. Y la última capa, denominada función de decisión, que digamos que en este de predictores no adquiere la misma importancia que en otros, debido a que la activación de la alarma se basa en el nivel de una única señal.

En la sección 3.5, se explica la técnica de clasificación utilizada, denominada clusterización, ya que se basa en el análisis de clusters (agrupamientos de datos).

En la sección 3.6 se ejemplifica como aplicar esta técnica.

Y en la sección 3.7 se muestran los resultados obtenidos de la aplicación de un PBOD a varias descargas en JET.

3.2. Detección de outlier aplicado a las interrupciones

En un entorno general de transmisión de datos en línea, estos se observan y se tratan de forma secuencial y una decisión sobre la identificación de cualquier cambio en los datos tiene que hacerse en tiempo real. Cualquier anomalía en los datos puede conllevar información interesante acerca del proceso.

Para poder determinar el momento en el que aparece cada interrupción, el sistema detector de outlier tiene que conocer en cada nueva descarga la evolución del comportamiento seguro. Es importante señalar que la aparición de descargas bajo diferentes escenarios puede generar diferentes clases de

disrupciones en el plasma. Por lo tanto, en principio, la evolución segura se tiene que aprender en cada descarga.

Un punto esencial en la aplicación de la detección de outlier para poder reconocer la próxima disrupción es estar seguro de que el cambio en los datos corresponde a un evento disruptivo. De lo contrario, se dispararía una gran cantidad de falsas alarmas y por consiguiente sería imposible la producción de escenarios de plasma interesante. Por lo tanto, el sistema de predicción deberá generar una señal de referencia que de algún modo pueda diferenciar cuando se trata de una disrupción y cuando es un valor anómalo de la señal original. En una primera aproximación, esta señal será la única que se usará para implementar el predictor basado en la detección de anomalías.

En este punto, es necesario hablar de los requisitos que debe cumplir el sistema de predicción de disrupciones basado en la detección de outlier. A continuación se exponen unas especificaciones que debe cumplir:

- En primer lugar, es importante tener en cuenta que los datos secuenciales se leen sólo una vez. La característica principal de un flujo de datos es que los datos entran uno a uno. Una vez que el dato ha sido tratado se desecha sin la posibilidad de recuperarlo posteriormente. Esta condición puede ser relajada en casos en que se necesiten en su conjunto para la toma de alguna decisión
- Otro punto que debe cumplir, es que el tiempo entre que aparece una alarma buena y su detección tiene que ser mínimo.
- En tercer lugar, hay que señalar que el número de alarmas perdidas o falsas alarmas debe ser mínimo.
- Y por último, pero no menos importante, el flujo de datos debe ser manejado de manera eficiente desde el punto de vista computacional, este punto será crucial para una posible implementación y ejecución en tiempo real.

3.3. Implementación del Predictor PBOD

3.3.1. Introducción

Hay dos factores principales en el desarrollo del sistema de predicción PBOD: el criterio para el reconocimiento de la interrupción y la resolución temporal de las predicciones. El primero de ellos es la clave del método para mantener bajas tasas de falsas alarmas, bajas tasas de alarmas perdidas, bajas tasas de alarmar prematuras, bajas tasas de detecciones tardías y altas tasas de alarmas validas. El segundo factor juega un papel central para activar una alarma tan pronto como sea posible y, por lo tanto, para lograr mayores tiempos de respuesta a las alarmas.

Decir que el PBOD tiene que tener mejor rendimiento que el predictor estándar, basado solo en la activación de la alarma cuando la amplitud de la señal de referencia cruza un umbral. Tras la experiencia obtenida en la selección de características en otros sistemas de predicción, PBOD también utiliza las ventanas de tiempo para extraer la información relevante para el desarrollo del predictor.

3.3.2. Selección de las señales y uso de ventanas temporales

No todas de las señales que se generan en los dispositivos de fusión contienen información útil para el estudio del fenómeno de las disrupciones. Es imprescindible seleccionar aquellas que aporten gran cantidad de información relevante y con la menor cantidad de datos redundantes. Estas señales serán las que se pre-procesarán y se someterán a técnicas de clasificación para así obtener la señal de referencia que marcará el nivel de disparo de las alarmas.

Para la fase de entrenamiento del predictor y el posterior test del mismo en futuros trabajos, se han utilizado los datos de dos señales obtenidas con los métodos de diagnostico del Tokamak JET.

- Por un lado, tenemos la corriente del plasma de las diferentes descargas, en concreto, es el valor medio de dicha corriente cada 32 ms. Estos valores están registrados en los archivos con extensión 01 muestreados cada milisegundo de tiempo. Se utilizaran para definir la franja de tiempo de la descarga. Estos instantes vendrán determinados cuando el valor de la señal alcance unos límites. A continuación se

muestra una evolución temporal tipo de esta señal, donde se puede apreciar la forma y el rango de valores que toma.

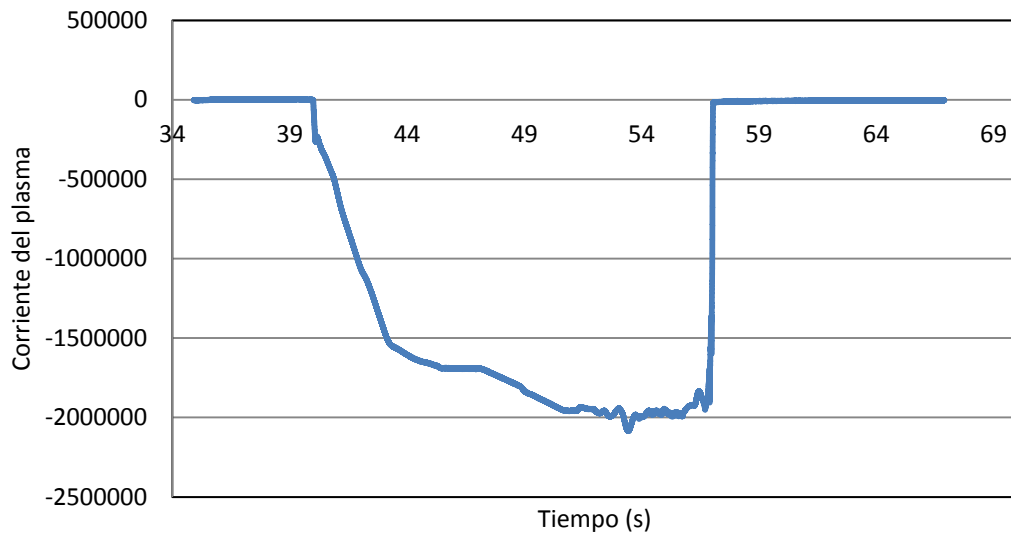
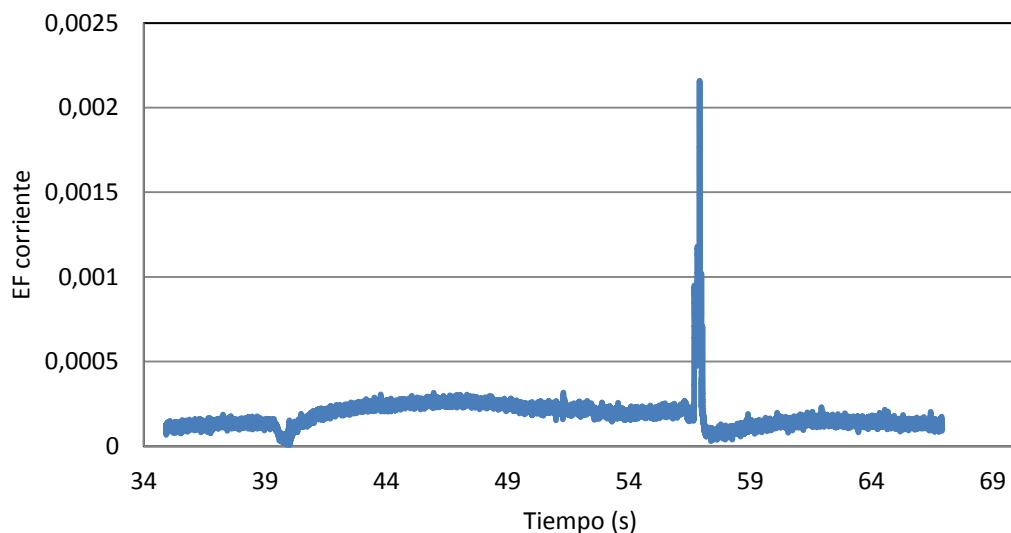


Figura 3.1. Evolución temporal de los valores medios de la corriente

- Y por otro lado, contamos con los archivos con extensión *02* también muestreados con una frecuencia de muestreo de un milisegundo, y contienen las desviaciones estándar de los espectros de Fourier de dicha corriente (sin considerar la componente continua) en las mismas ventanas de 32 ms en las que se determinan los valores medios. Son las señales que procesa el sistema de predicción para la detección de si ha habido o no una interrupción. En la siguiente figura se muestra el aspecto de una de ellas.



.Figura 3.2. Evolución temporal de la desviación estándar del espectro de Fourier

En la Figura 3.3 se puede observar cómo la evolución temporal total de una descarga, puede representarse como la concatenación de vectores de características, cada uno de ellos perteneciente a distintas ventanas temporales, representados por rectángulos.

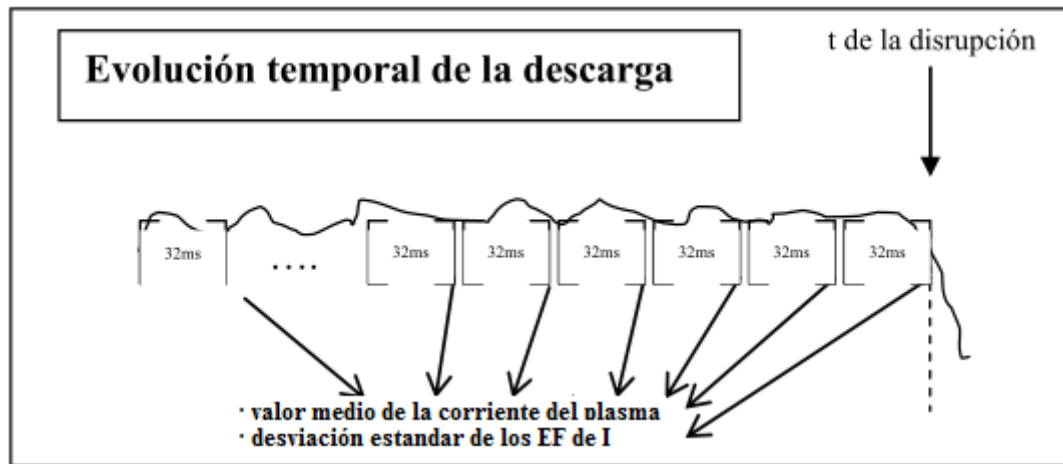


Figura 3.3. Evolución temporal de la descarga

Para el pre-procesado se agruparon los valores de la señal en ventanas temporales de tamaño variables. El tamaño de estas ventanas es un factor a ser determinado, que deberá ser siempre potencias de dos. La información contenida en estas ventanas se comprime en un número reducido de componentes por medio de la transformada wavelet Haar (coeficientes de aproximación). La transformada wavelet permite retener tanto el tiempo como la información de frecuencia de la señal. Cada nivel de descomposición de la transformación de Haar reduce en un factor de 2 el número de puntos en la señal inicial. Por lo tanto, para una ventana formada por 32 datos, la aplicación de la transformada de Haar en los niveles de descomposición 1, 2, 3 y 4 comprimirían la información en 16, 8, 4 y 2 muestras respectivamente.

Una vez formada la primera ventana de datos, se aplica un mecanismo de deslizamiento para ir formando las consecutivas ventanas. Este mecanismo de deslizamiento consiste en ir desplazando las ventanas de datos un número variable que podrá ser elegido. Esto significa que, suponiendo que el índice de desplazamiento es de dos muestras, después del pre-procesamiento (hallar los coeficientes de aproximación de la transformada Haar) de la primera ventana temporal (muestras 1-n), la segunda ventana se compondrá de las muestras 3 - n+2, la tercera 5 - n+4 y así sucesivamente, siendo n el tamaño de la ventana. Por ejemplo, suponemos que definimos un tamaño de la ventana de 32 muestras, la primera ventana estaría formada por las muestras 1-32, la segunda 3-34, la tercera 5-36 y así hasta finalizar. En otras palabras, cada 2 muestras de la señal se procesan las últimos 32.

3.3.3. Transformada wavelet para el análisis de señales

La transformada Wavelet es una función matemática que permite obtener datos de una señal seccionándolos en pequeños componentes en escala frecuencia-tiempo y poder analizarlos en forma separada. En comparación con la transformada de Fourier, la Wavelet permite trabajar sobre ondas no estacionarias, con discontinuidades o picos.

En ingeniería se representan las señales ya sea en función del tiempo o de la frecuencia. Las técnicas para el análisis de las señales, desde hace tiempo se basan en los métodos de análisis espectrales en los que se descompone una onda compleja en una suma lineal de componentes de onda más elementales que en el caso del análisis con Fourier son ondas senos y cosenos. Para muchas señales, el análisis de Fourier es muy útil ya que el contenido frecuencial de la señal es de gran importancia, pero también tiene un serio inconveniente. Cuando se produce una transformación al dominio de la frecuencia, se pierde la información del dominio del tiempo. Cuando vemos la transformada de Fourier de una señal, es imposible decir cuando tuvo lugar un determinado evento y por lo tanto difícil predecir una interrupción.

Si las propiedades de la señal no cambian mucho a lo largo del tiempo, es decir, si estamos ante una señal estacionaria, este inconveniente no tiene importancia. Sin embargo, las señales más interesantes contienen características no estacionarias y transitorias como puedan ser cambios abruptos y comienzos y finales de eventos. Esas características son a menudo las partes más importantes de la señal, y el análisis de Fourier no es adecuado para detectarlos.

El análisis wavelet permite el uso de intervalos largos en el tiempo allí donde necesitamos información de baja frecuencia más precisa, y regiones más pequeñas donde queramos información de alta frecuencia. Se debe tener en cuenta, que el análisis wavelet no usa regiones tiempo-frecuencia, sino regiones escala-tiempo. El análisis wavelet es capaz de revelar aspectos de la señal que otras técnicas de análisis de señal ignoran, aspectos como tendencias, puntos de ruptura, discontinuidades. Además, ya que se logra un punto de vista diferente, el análisis wavelet puede a menudo comprimir o reducir el ruido de una señal sin una degradación apreciable.

En el análisis de wavelet la señal se divide en versiones desplazadas y escaladas de la wavelet original (o wavelet madre). Con sólo mirar la forma de los wavelets y las ondas sinusoidales, podremos intuir que las señales con cambios abruptos se pueden analizar mejor con una wavelet irregular que con una senoide suavizada producto del análisis de Fourier.

En la transformada de wavelet discreta (TWD) el cálculo de los coeficientes en cada uno de los niveles conlleva un gran esfuerzo computacional, generando una gran cantidad de datos. Por lo tanto, es lógico plantearse el elegir sólo un nivel de descomposición y mantenerlo para todos los cálculos. De hecho, es aconsejable elegir tamaño de ventanas potencia de dos, de esta forma el análisis será mucho más eficiente sin perder precisión.

En el análisis wavelet se habla de coeficientes de aproximación que son las componentes de baja frecuencia de la señal y coeficientes de detalle, que son los componentes de alta frecuencia.

Para que la aplicación de la teoría de wavelet para la detección de interrupciones sea útil, es necesario que se disponga de algoritmos rápidos para poder encontrar los coeficientes. En la figura 3.4 se puede ver el proceso de filtrado, en su nivel más básico.

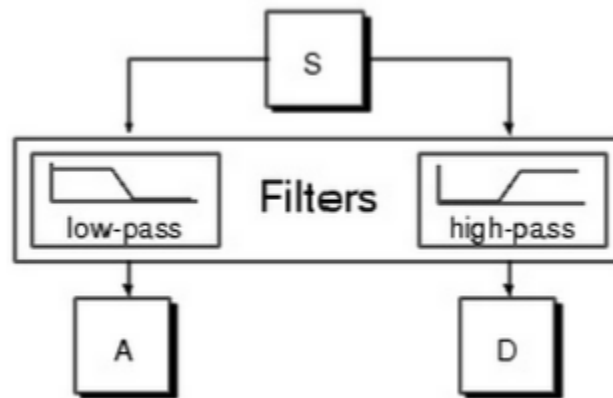


Figura 3.4. Proceso de filtrado

Donde S es la señal original que se hace pasar por dos filtros (paso alto y paso bajo), obteniéndose con ello dos señales denominadas aproximación (A) y detalle (D) de la señal, siendo los coeficientes de la señal detalle (cD_i) los que constituyen principalmente el ruido de alta frecuencia, por lo que los coeficientes de aproximación (cA_i) contienen menos ruido de los que tenía la señal original.

Cada filtrado dará como resultado una disminución a la mitad el número de muestras. Como se ve en la figura 3.5, un nivel de descomposición 3 nos daría que la señal original S será $S = cA_1 + cA_2 + cA_3 + cD_3$.

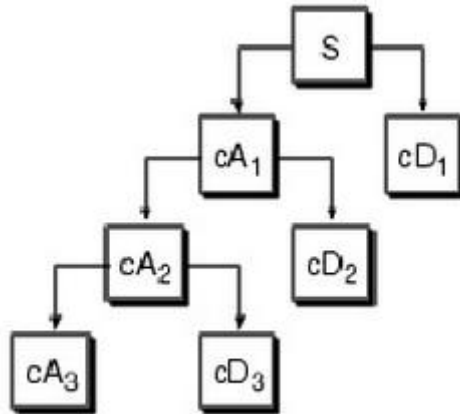


Figura 3.5. Proceso de descomposición

Transformada de wavelet Haar y cálculo de los coeficientes de aproximación de cada ventana

La transformada de Haar es la wavelet más sencilla. Para tener una idea de cómo funciona la wavelet y como hallar sus coeficientes usaremos un ejemplo sencillo. Supongamos que tenemos una señal de una dimensión con 8 valores,

$$[9 \ 7 \ 3 \ 5]$$

Esta señal se puede representar en la base Haar calculando la transformada de wavelet. Para ello, promediamos los valores que están juntos por parejas, resultando una señal de menor resolución. En este primer nivel de descomposición, se obtienen los coeficientes de aproximación de nivel 1.

$$[8 \ 4]$$

Es evidente que parte de la información se pierde en este proceso de promediado. Para recuperar los valores originales a partir de los coeficientes de aproximación se calculan los coeficientes de detalle, que captan la información que falta. El primer coeficiente de detalle es 1 y el segundo -1, con estos valores se podría recomponer la señal madre.

Por lo tanto, se ha descompuesto la señal original en una resolución más baja, reduciendo a la mitad el número de muestras. Repitiendo este proceso de forma recursiva se obtiene la siguiente descomposición total:

Resolución	Coef Aproximación	Coef Detalle
1er Nivel	[9 7 3 5]	[1 -1]
2o Nivel	[8 4]	[2]

Por último, la transformada de wavelet aplicando el criterio de Haar de la señal original quedaría definida en su nivel mayor de descomposición por el coeficiente de aproximación de nivel 2 junto a los coeficientes de detalle. Así, la transformada de wavelet de la señal original viene dada por:

$$[6 2 1 -1]$$

La figura 3.6, muestra un diagrama de dispersión de la señal de referencian, en el espacio bidimensional definido por la transformada de Haar en el nivel 4 de descomposición. Cada punto representa los coeficientes calculados en cada ventana de datos (ventana de 32 datos). Los puntos se trazan cada 2 datos mediante la aplicación del mecanismo de ventana deslizando descrito anteriormente.

En la figura (a) correspondiente a una descarga no disruptiva, se puede observar que los datos forman un cluster compacto. En cambio, en la figura (b) donde se representa una descarga disruptiva, se ve como los datos aparecen agrupados en la fase no disruptiva de la descarga y alejados del centro del cluster cuando la interrupción se está acercando. De acuerdo con esto, el punto rojo será el primero valor atípico en el flujo de datos y su reconocimiento determina cuándo activar una alarma.

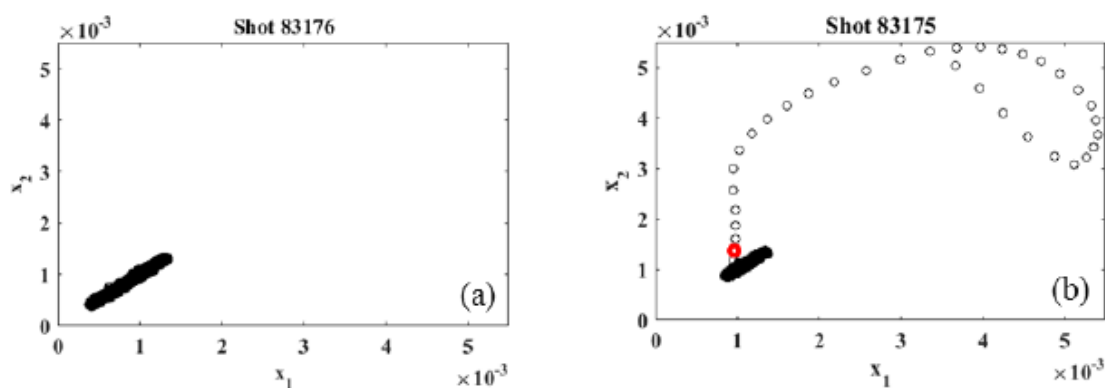


Figura 3.6. Diagrama de dispersión de una descarga no disruptiva y una descarga disruptiva. (a) cluster compacto, (b) la presencia de outlier denotan la proximidad de una interrupción

3.4. Arquitectura del predictor

La arquitectura general del sistema de predicción propuesto está compuesta por tres capas:

- La primera capa, donde se ha visto en el apartado, se realiza un pre-procesado de los datos. Se calculan los coeficientes de aproximación de las ventanas consecutivas de la señal.
- La segunda capa, analiza los resultados de la primera.
- Y la tercera capa (llamada función de decisión) donde se evalúa los datos y en base a ello se activa o no una alarma.

El predictor final puede ser entonces esquematizado en tres capas (Figura 3.7). Como vimos anteriormente, en la primera de ellas, utilizando la transformada de Haar se calculan los coeficientes de aproximación de las ventanas de datos formadas por los mecanismos de deslizamiento. En la segunda capa, como veremos más adelante, se analizará los resultados obtenidos. Para ello, se hará uso de criterios basados en distancias como indicadores de disparidad. Con cada uno de estos resultados se generará una señal de referencia que determine si se ha detectado o no comportamiento disruptivo. Y la tercera capa, detallada en la siguiente sección y llamada función de decisión (FD), encargada de disparar la alarma en caso de detección de una interrupción.

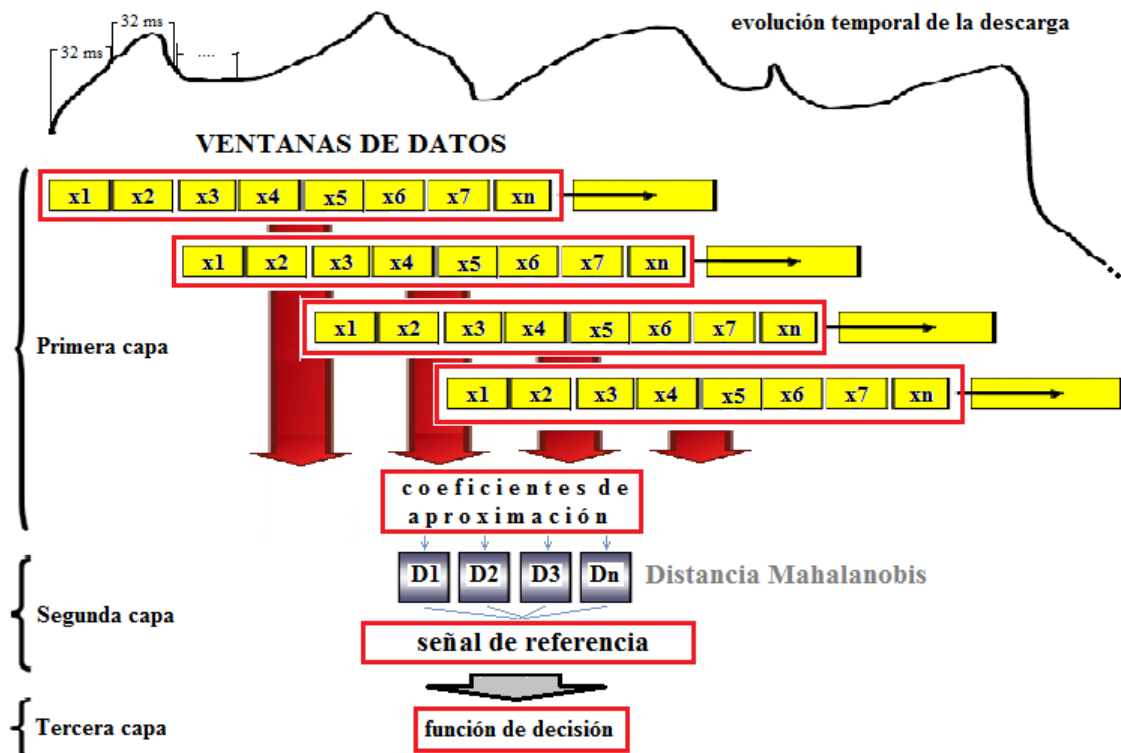


Figura 3.7. Estructura del sistema PBOD

3.5. Técnicas de Clusterización

3.5.1. Planteamiento general

El análisis cluster es un conjunto de técnicas multivariantes utilizadas para clasificar a un conjunto de individuos en grupos homogéneos.

Pertenece, al igual que otras tipologías y que el análisis discriminante al conjunto de técnicas que tiene por objetivo la clasificación de los individuos. La diferencia fundamental entre el análisis cluster y el discriminante reside en que en el análisis cluster los grupos son desconocidos a priori y son precisamente lo que queremos determinar; mientras que en el análisis discriminante, los grupos son conocidos y lo que pretendemos es saber en qué medida las variables disponibles nos discriminan esos grupos y nos pueden ayudar a clasificar o asignar los individuos en/a los grupos dados.

Así pues, el objetivo es obtener clasificaciones (clusterings), teniendo, por lo tanto, el análisis un marcado carácter exploratorio.

Se trata, fundamentalmente, de resolver el siguiente problema: Dado un conjunto de individuos (de N elementos) caracterizados por la información de n

variables X_j , ($j = 1, 2, \dots, n$), nos planteamos el reto de ser capaces de clasificarlos de manera que los individuos pertenecientes a un grupo (cluster) (y siempre con respecto a la información disponible) sean tan similares entre sí como sea posible, siendo los distintos grupos entre ellos tan disimilares como sea posible.

Como puede comprenderse fácilmente el análisis cluster tiene una extraordinaria importancia en la investigación científica, en cualquier rama del saber. Téngase presente que la clasificación es uno de los objetivos fundamentales de la ciencia. Y en la medida en que el análisis cluster nos proporciona los medios técnicos para realizarla, se nos hará imprescindible en cualquier investigación.

Con el análisis cluster se pretende encontrar un conjunto de grupos a los que ir asignando los distintos individuos por algún criterio de homogeneidad. Por lo tanto, se hace imprescindible definir una medida de similitud o bien de divergencia para ir clasificando a los individuos en unos u otros grupos.

Además algunas consideraciones podrán tenerse en cuenta antes de comenzar el análisis:

Así, debemos plantearnos si vamos a comenzar la agrupación partiendo de algunos grupos ya establecidos o si, por el contrario, comenzaremos por considerar cada elemento individual como un cluster inicial que posteriormente iremos agrupando hasta obtener los clusters finales: Deberemos plantearnos la posibilidad de reasignaciones a lo largo del proceso. Igualmente deberemos establecer criterios para detener la agrupación y para llevarla a cabo. Básicamente, el análisis constará de un algoritmo de clasificación que nos permitirá la obtención de una o varias particiones, de acuerdo con los criterios establecidos

El proceso completo puede estructurarse de acuerdo con el siguiente esquema:

- Partimos de un conjunto de N individuos de los que se dispone de una información cifrada por un conjunto de n variables (una matriz de datos de N individuos \times n variables) .
- Establecemos un criterio de similitud para poder determinar: Una matriz de similitudes que nos permita relacionar la semejanza de los individuos entre sí (matriz de N individuos \times N individuos).
- Escogemos un algoritmo de clasificación para determinar la estructura de agrupación de los individuos.

- Especificamos esa estructura mediante diagramas arbóreos o dendogramas u otros gráficos.

Así pues, la clave de una buena clasificación pasará por:

a) Una buena selección de las variables que nos van a describir a los individuos. En este sentido puede ser necesario un análisis de datos previo. La posibilidad de combinar secuencialmente el análisis factorial y el análisis cluster puede ser provechosa.

b) Poner cuidado en el criterio de similitud a utilizar.

c) Seleccionar adecuadamente el algoritmo de clasificación.

3.5.2. Criterios de similitud. Similitud, divergencia y distancia

Una vez hemos hecho una adecuada selección de las variables a considerar, cada uno de los individuos sujetos al análisis nos vendrá representado por los valores que tomen estas variables en cada uno de ellos. Este es el punto de partida de la clasificación. Para clasificar adecuadamente los individuos deberemos determinar lo similares o disimilares (divergentes) que son entre sí, en función de lo diferentes que resulten ser sus representaciones en el espacio de las variables.

Para medir lo similares (o disimilares) que son los individuos existe una enorme cantidad de índices de similitud y de disparidad o divergencia. Todos ellos tienen propiedades y utilidades distintas y habrá que ser consciente de ellas para su correcta aplicación al caso que nos ocupe.

La mayor parte de estos índices serán o bien, indicadores basados en la distancia (considerando a los individuos como vectores en el espacio de las variables) (en este sentido un elevado valor de la distancia entre dos individuos nos indicará un alto grado de disparidad entre ellos); o bien, indicadores basados en coeficientes de correlación ; o bien basados en tablas de datos de posesión o no de una serie de atributos.

3.5.3 Criterios basados en distancias como indicadores de disparidad

Se da, en general, el nombre de distancia o disparidad entre dos individuos i y j a una medida, indicada por $d(i,j)$, que mide el grado de semejanza, o a mejor decir de desemejanza, entre ambos objetos o individuos, en relación a un cierto número de características cuantitativa y / o cualitativas. El valor de $d(i,j)$ es siempre un valor no negativo, y cuanto mayor sea este valor mayor será la diferencia entre los individuos i y j .

Toda distancia debe verificar, al menos, las siguientes propiedades:

$$(P.1) \quad d(i,j) > 0 \text{ (no negatividad)}$$

$$(P.2) \quad d(i,i) = 0$$

$$(P.3) \quad d(i,j) = d(j,i) \text{ (simetría)}$$

Diremos que una distancia es euclidiana cuando pueda encontrarse un espacio vectorial de dimensión igual o inferior a la dimensión del espacio de las variables en el que podamos representar a los individuos por puntos cuya distancia euclídea ordinaria coincida con la distancia utilizada.

Es decir si existe un espacio vectorial R^m , con $m < n$ (siendo n el número de variables consideradas para representar a los individuos) y dos puntos de ese espacio, P_i y P_j de coordenadas: $P_i = (P_{i1}, P_{i2}, \dots, P_{im})$ y $P_j = (P_{j1}, P_{j2}, \dots, P_{jm})$ verificándose que la distancia que estamos considerando entre los individuos i y j es igual a la distancia euclídea entre los puntos P_i y P_j en R_m ; esto es: Si $d(i,j) = \sqrt{(P_i - P_j)^2}$, diremos que la distancia $d(i,j)$ es euclidiana.

Cuando la distancia es euclidiana se verifica además que:

$$(P.4) \quad d(i,j) < d(i,t) + d(j,t) \text{ (desigualdad triangular)}$$

Existe una gran cantidad de distancias e indicadores de disparidad y no se puede disponer de una regla que nos permita definir una disparidad conveniente para todo tipo de análisis. A continuación se verá las más importantes.

Antes conviene recordar que los datos de partida del análisis son las observaciones de n variables y N individuos. Teniendo en cuenta esto, podemos representar a los individuos en el espacio de las variables de manera que representaremos al individuo i -simo por el vector:

$$W_i = \begin{pmatrix} x_{1i} \\ x_{2i} \\ \vdots \\ x_{ni} \end{pmatrix}; \text{ y al individuo } j\text{-ésimo: } W_j = \begin{pmatrix} x_{1j} \\ x_{2j} \\ \vdots \\ x_{nj} \end{pmatrix}$$

3.5.4. Distancia Euclídea

La distancia euclídea es la distancia más conocida y más sencilla de comprender, pues su definición coincide con el concepto más común de distancia.

Su expresión es la siguiente: $d(i,j) = (W_i - W_j)' (W_i - W_j)$

La distancia euclídea, a pesar de su sencillez de cálculo y de que verifica algunas propiedades interesantes tiene dos graves inconvenientes:

- El primero de ellos es que la euclídea es una distancia sensible a las unidades de medida de las variables: las diferencias entre los valores de variables medidas con valores altos contribuirán en mucha mayor medida que las diferencias entre los valores de las variables con valores bajos. Como consecuencia de ello, los cambios de escala determinarán, también, cambios en la distancia entre los individuos. Una posible vía de solución de este problema es la tipificación previa de las variables, o la utilización de la distancia euclídea normalizada
- El segundo inconveniente no se deriva directamente de la utilización de este tipo de distancia, sino de la naturaleza de las variables. Si las variables utilizadas están correlacionadas, estas variables nos darán una información, en gran medida redundante. Parte de las diferencias entre los valores individuales de algunas variables podrían explicarse por las diferencias en otras variables. Como consecuencia de ello la distancia euclídea inflará la disparidad o divergencia entre los individuos.

La solución a este problema pasa por analizar las componentes principales (que están correlacionadas) en vez de las variables originales. Otra posible solución es ponderar la contribución de cada par de variables con pesos inversamente proporcionales a las correlaciones, lo que nos lleva, como veremos a la utilización de la distancia de Mahalanobis.

La distancia euclídea será, en consecuencia, recomendable cuando las variables sean homogéneas y estén medidas en unidades similares y/o cuando se desconozca la matriz de varianzas.

3.5.5. Distancia Euclídea Normalizada

La distancia euclídea normalizada entre los individuos i y j la definiremos como: $d(i,j) = (W_i - W_j)' S^{-1} (W_i - W_j)$

Donde S es una matriz diagonal con las varianzas en su diagonal principal y ceros en el resto de sus elementos.

Obviamente S^{-1} será su inversa: la matriz diagonal que tendrá los valores recíprocos de las varianzas en su diagonal.

Utilizar como matriz de la forma cuadrática distancia la matriz S^{-1} , en vez de la matriz identidad, I , es, claramente, equivalente a utilizar como valores de partida los valores de las variables cambiados de escala a la desviación típica de las variables.

Empleando este tipo de distancia solventamos el inconveniente de los efectos de unidades de medida distintas de las variables y obtenemos una distancia que no dependerá de las unidades de medida.

Sin embargo, la alta correlación entre algunas variables puede seguir siendo un grave inconveniente.

3.5.6 Distancia de Mahalanobis

La distancia de Mahalanobis entre los individuos i y j la definimos por la expresión: $d(i,j) = (W_i - W_j)' V^{-1} (W_i - W_j)$

Donde la matriz asociada a la forma cuadrática V^{-1} es la inversa de la matriz de covarianzas V .

Esta distancia presenta las ventajosas propiedades de solventar los dos inconvenientes de la aplicación de la distancia euclídea: Por un lado es invariante ante los cambios de escala y no depende, por tanto de las unidades de medida.

En efecto: Si consideramos las variables originales x representadas por el vector de variables:

$X = \begin{pmatrix} x1 \\ x2 \\ \cdot \\ \cdot \\ xn \end{pmatrix}$ y consideramos su transformada lineal a otras nuevas variables, y,

representadas por el vector de variables: $Y = \begin{pmatrix} y1 \\ y2 \\ \cdot \\ \cdot \\ yn \end{pmatrix}$ que vendrá dada por la

relación $Y = CX$; la matriz de covarianzas de Y será: $V = C'VC$

En el espacio de las nuevas variables los individuos vendrán representados por un nuevo vector: $W^* = C'W$.

La distancia de Mahalanobis sobre las nuevas variables será entonces:

$$\begin{aligned} D(i, j) &= (W_i^* - W_j^*)' V^{-1} (W_i^* - W_j^*) \\ &= (W_i - W_j)' C [(C'V^{-1}C)] C'(W_i - W_j) = \\ &= (W_i - W_j) V^{-1} (W_i - W_j) \end{aligned}$$

Por otro lado al utilizar la matriz V , se consideran las correlaciones entre las variables y se corrige el efecto de la redundancia.

Es interesante hacer ahora dos observaciones finales más:

1) Si las variables están correlacionadas, la distancia de Mahalanobis coincide con la distancia euclídea normalizada.

En efecto: Si las variables están correlacionadas la matriz V coincide con la matriz S , y, por tanto, la inversa de V coincidirá con la inversa de S .

2) La distancia de Mahalanobis coincide con la distancia euclídea calculada sobre el espacio de las componentes principales.

3.6 Aplicación del análisis Cluster en datos de fusión

A lo largo de la experiencia obtenida en las incontables descargas, se ha podido ver que la tendencia general en todas las descargas JET muestran una covarianza positiva en los datos, esto hace que la distancia euclidiana no se pueda utilizar para determinar cuando un punto se encuentra fuera de la agrupación. Esto es una consecuencia de la falta de simetría circular en el

cluster y su uso implicaría la activación de una gran cantidad de falsa alarmas. En cambio, la distancia de Mahalanobis tiene en cuenta la matriz de covarianza de la población, esto supone, normalmente, que el contorno de los puntos formen una elipse y no un círculo (Figura 3.8). Por lo tanto, este tipo de distancia resulta una opción adecuada para identificar los valores atípicos en el diagrama de dispersión de la descarga definido por la transformada de Haar en uno de sus niveles (figura 3.6).

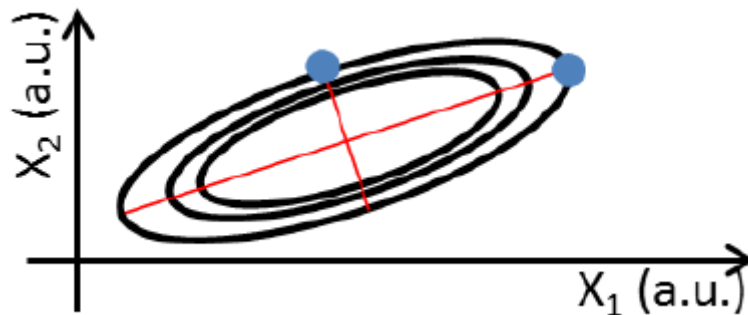


Figura 3.8. Representación bi-dimensional de los contornos equidistantes elípticos

Por lo tanto, los puntos azules tienen la misma distancia de Mahalanobis al centro.

En la figura 3.9 se puede ver cuál sería la evolución temporal de las distancias de Mahalanobis de cada nuevo punto de datos con respecto al centro del cluster formado por los anteriores datos de la descarga.

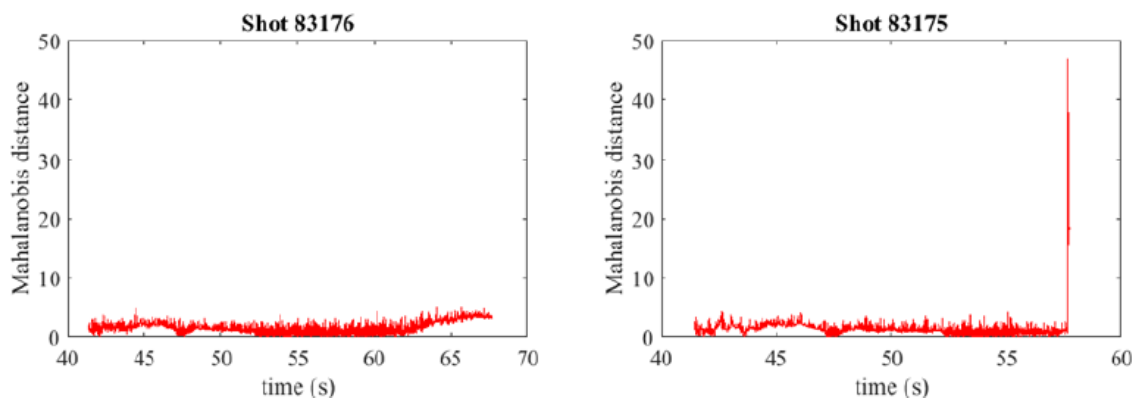


Figura 3.9. Evolución temporal de la distancia de Mahalanobis de una descarga no disruptiva y una descarga disruptiva.

La gráfica de la izquierda representa una descarga no disruptiva. Como puede verse, los valores de las distancia no presenta una desviación grande. En cambio, en la descarga de la derecha el gran incremento al final de la descarga identifica la aparición de un outlier.

En este punto, cabría la posibilidad de pensar en utilizar el valor de las distancias como señal de referencia para la detección de posibles interrupciones. Sin embargo, el umbral de las distancias de Mahalanobis para la activación de una alarma puede depender del tipo de escenario en el que estemos trabajando. Por lo tanto para evitar esto, el criterio para identificar los valores atípicos en cada momento t_p estará relacionado con la desviación estándar del conjunto de las distancias, y vendrá dado por la siguiente ecuación:

$$\left| \frac{D_{Mahalanobis}(t_p) - \text{mean}(D_{Mahalanobis}(t \leq t_p))}{\text{std}(D_{Mahalanobis}(t \leq t_p))} \right| > K_M \quad (3.1)$$

donde K_M es un valor fijado empíricamente y std es la desviación estándar de todas las distancias de $t \leq t_p$.

En la figura 3.10 (a) se muestra la evolución temporal de la señal de bloqueo (gráficas superior) y el factor outlier (gráficas inferior) obtenido aplicando la ecuación 3.1 para la descarga 83175. El PBOD dispara la alarma con mayor tiempo de advertencia que los predictores APODIS y LMP JET. En la figura 3.10 (b) correspondiente a una descarga diferente se puede ver como el predictor LMP no detecta la alarma y el PBOD reconoce el comportamiento perturbador con un tiempo de advertencia de 166 ms.

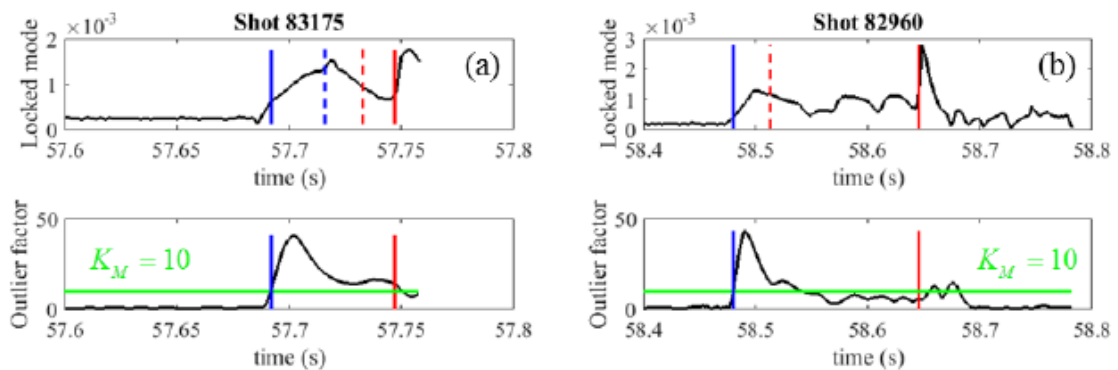


Figura 3.10. Línea continua roja: momento de la interrupción. Líneas rojas discontinuas: predicción del APODIS. Líneas azules discontinuas: predicción del LMP. Línea azul continua: predicción del PBOD

3.7. Función de decisión (FD)

En los distintos clasificadores basados en n valores de salida, no siempre estos valores coinciden en el carácter disruptivo/no disruptivo de la predicción, ya que algunos valores pueden detectar un comportamiento anómalo y al mismo tiempo otros no. En consecuencia, es necesario implementar una función que evalúe los resultados y que mediante ellos decida si disparar o no una alarma. En estos casos, el desarrollo de esta función de decisión es crucial para alcanzar la mayor tasa de reconocimiento posible.

En este PBOD, basado en una única señal de bloqueo no será necesaria la implementación de la FD ya que el disparo de la alarma dependerá del valor de la contaste K_M , si cabe destacar que es necesario una etapa de entrenamiento para poder determinar el valor de esta.

Experimentos realizados en otros PBOD, en una primera aproximación, han establecido empíricamente un valor de K_M igual 10, el cual proporciona resultados ligeramente mejores para JET. En la actualidad, se desconoce si este valor será el mismo para otros Tokamaks.

Resultados obtenidos en PBOD anteriores

A continuación se muestra una tabla (3.1) con los resultados obtenidos en trabajos anteriores con un detector de predicciones PBOD. Datos recogidos de la aplicación en JET a distintas descargas, tanto a no disruptivas descargas como a involuntarias interrupciones. La primera columna representa el número de coeficientes de cada ventana de tiempo (32 muestras) después de aplicar la transformada de wavelet Haar. En las columnas restantes se puede ver la tasa de falsas alarmas, alarmas perdidas, alarmas tardías, alarmas buenas y alarmas prematuras. Salvo en el caso en el que la comprensión de la ventana es de 16 coeficientes, todos los demás predictores son muy similares y muestran resultados muy prometedores.

Data compression	False alarms (%)	Missed alarms (%)	Tardy detections (%)	Valid alarms (%)	Premature alarms (%)
2	7.13	13.43	3.53	81.45	1.59
4	7.31	11.48	3.36	83.22	1.94
8	7.42	11.84	3.00	83.39	1.77
16	+18	12.37	3.71	81.80	2.12

Tabla 3.1. Resultados de aplicar PBOD con distintos niveles de aproximación

La figura 3.11 resume y compara los resultados de LMP, APODIS y PBOD para las mismas descargas. Los tiempos medios de advertencia para los predictores LMP, APODIS y PBOD son 255, 274 y 288 ms respectivamente.

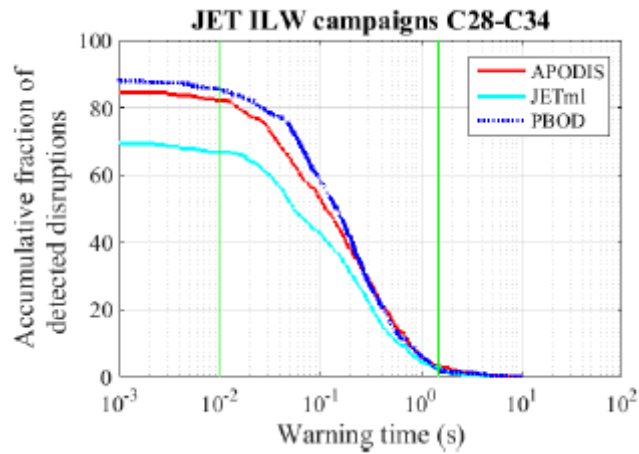


Figura 3.11. Comparación entre LMP, APODIS y PBOD

CAPÍTULO 4

DESARROLLO DEL ENTORNO

4.1. Introducción

La aplicación para la implementación del predictor de interrupciones basado en detección de anomalías se ha desarrollado en lenguaje C. Para la cual se ha utilizado **Dev-C++**, que es un entorno de desarrollo integrado para programar en lenguaje C/C++. Usa MinGW, que es una versión GCC como su compilador.

Para la elaboración del programa se ha hecho uso de memoria dinámica y gestión de ficheros. Se han creado dos librerías, `listas.h` y `matrices.h` para las operaciones con memoria dinámica y matrices.

4.2. Ficheros de datos

Como ya se ha comentado en otra sección, la aplicación utiliza dos ficheros de lectura y crea tres ficheros de datos. Como ficheros de lectura o entrada tenemos, los ficheros `JPF_XXXXX_01.txt` y `JPF_XXXXX_02.txt` que están formados por dos tipos de datos, una columna corresponde al tiempo y la otra al valor de la señal en ese instante. Es importante tenerlo en cuenta a la hora de hacer la lectura ya que los datos impares corresponden a instantes de tiempos y los datos pares a valores de la señal.

Los datos generados de la ejecución del predictor se guardarán en tres ficheros de texto diferentes *coeficientesAproximacion.txt*, guardará en columna los coeficientes calculados de cada ventana, *distanciaMahalanobis.txt*, donde aparecerán las distancias correspondientes a cada coeficiente y *señalReferencia.txt*, que como su nombre indican, guardará los valores de la señal de referencia.

4.3. Parametrización de las variables

Con la parametrización de las variables tenemos la posibilidad de modificar los aspectos puntuales del funcionamiento del sistema de una forma sencilla. Gran parte de la funcionalidad del sistema está basado en estas variables, que afectan de manera directa su comportamiento, por ello y puesto que una de las finalidades es la obtención de datos bajo diferentes marcos, conviene elaborar la aplicación de modo que el valor de las variables no afecte en el desarrollo de su ejecución.

Recordemos que las variables a parametrizar son: el tamaño de la ventana temporal, el desplazamiento y el nivel de descomposición.

- Con el tamaño de la ventana temporal, definimos el tamaño del vector donde se guardan el valor de las muestras de la señal. Este tamaño debe ser un número potencia de 2.
- El deslizamiento permite definir el número de muestras que tiene que desplazarse el vector de la señal para formar el nuevo vector que se va a procesar.
- Y por último, con el nivel de descomposición se define el grado de descomposición de la transformada de wavelet Haar que se va a aplicar, recordamos que cada nivel reduce en la mitad el tamaño de la muestras, por lo el valor del nivel debe ser un número entero tal que:

$$\text{número de muestras del vector} \geq 2^{(\text{nivel}+1)}$$

4.4. Detalle del código desarrollado

En este apartado se hace un repaso de los aspectos más importantes de la implementación realizada. Primero, revisaremos las principales funciones creadas y luego explicaremos las partes del main.

4.4.1. calculaCoeficientesDeAproximación

Como su nombre indica, calcula los coeficientes de aproximación de las ventanas temporales que se van formando secuencialmente. Recibe como parámetros de entrada *a* que es el número de coeficientes de aproximación que se calculan por cada ventana y que recordamos que depende del nivel de descomposición, y el array *x* que es el vector de tamaño MAX_MUESTRAS donde se almacenan los valores de la señal. La función devuelve como parámetro de salida el vector *y* donde se encuentran almacenados los valores de los coeficientes de aproximación.

Para cada NIVEL de descomposición, se calculan los coeficientes de aproximación como la suma de *sum1* más *sum2* dividido entre dos de todos los valores que forman el array de la señal o de los coeficientes que se hallan calculados anteriormente. Finalmente se forma el vector de coeficientes multiplicados por FACTOR raíz de 2 y por 1000.

```
1 void calculaCoeficientesDeAproximacion (int a, float x[], float *y)
2 {
3     int i, j;
4     int contador;
5     float sum1, sum2;
6
7     for (i=0; i<NIVEL; i++)
8     {
9         contador=0;
10        for (j=0; j<MAX_MUESTRAS/pow(2,i); j++)
11        {
12            sum1=sum2;
13            sum2=x[j];
14            if (j%2==1)
15            {
16                x[contador]=(sum1+sum2)/2;
17                contador++;
18            }
19        }
```

```

20  }
21  for (i=0; i<a; i++)
22      y[i]=x[i]*FACTOR*1000;           // datos en 10exp(-3),
23  }

```

Figura 4.1. Código de la función *calculaCoeficientesDeAproximacion*

4.4.2. CalculaDistanciaMahalanobis

Calcula la distancia de Mahalanobis que tiene cada nuevo punto respecto al centro de todos los demás. Señalar que todas las distancias son positivas. Debido a que para la primera distancia, el centro del cluster aún no está definido, esta distancia puede ser negativa. Es por ello que se añaden las tres líneas 43-46 al código de la función. La función recibe como parámetros de entradas, *líneas*, que es el número de coeficientes que llevamos calculados, y que será el número de líneas de la matriz formada por todos los coeficientes. *Columnas*, que es el número final de coeficientes que hay que hallar en cada ventana y que dependen del nivel de descomposición, y que corresponde al número de columnas de la matriz. *coeAprox* que es el vector que contiene el valor de los coeficientes de los que hay que calcular la distancia. Y finalmente, *lista*, puntero al inicio de la tabla donde se almacenan todos los coeficientes de las ventanas temporales calculados hasta ese momento.

Retorna el valor de la distancia calculada.

Para poder hallar la distancia, primero se genera la matriz de datos con todos los coeficientes, para ello, se recorre la tabla utilizando como índices el número de líneas y el número de columnas.

Recordemos que la función de la distancia de Mahalanobis viene dada por la distancia del punto al centroide de los datos y por la inversa de la matriz de covarianzas-varianzas. Por lo que el siguiente paso, será hallar cada uno de estos factores. Finalmente, se obtiene la distancia haciendo uso de la función *multiplicarMatrices*. El código de esta, así como el código de la función *MatrizInversa* y *MatrizCovarianza* se encuentra detallado en el apartado de anexos.

Devuelve el valor de la distancia.

```

1  float CalculaDistanciaMahalanobis(int lineas, int columnas, float coeAprox[],
2  elemento *lista)
3  {
4      float matrizDatos[lineas][columnas];
5      float centroide[columnas];
6      elemento *actual = lista;
7      int i, j;

```

```

8   float distanciaCentroide[1][columnas];
9   float transDistanciaCentroide[columnas][1];
10  float covarianza[columnas][columnas];
11  float inversa[columnas][columnas];
12  float Mahalanobis;
13  while(actual!=NULL)           // genera matriz de datos
14  {
15      for(i=lineas-1;i>=0;i--)
16      {
17          for(j=columnas-1;j>=0;j--)
18          {
19              matrizDatos[i][j]=actual->dato;
20              actual=actual->siguiente;
21          }
22      }
23  }
24  for (i=0;i<columnas;i++)      // calcula centroide datos
25  {
26      centroide[i]=0;
27      for (j=0;j<lineas;j++)
28      {
29          centroide[i]=centroide[i]+matrizDatos[j][i];
30      }
31      centroide[i]=centroide[i]/lineas;
32  }
33  for (j=0;j<columnas;j++)      // distancia al centroide
34  {
35      distanciaCentroide[1][j]=coeAprox[j]-centroide[j];
36      transDistanciaCentroide[j][1]=coeAprox[j]-centroide[j]; //transpuesta
37  }
38  MatrizCovarianza (lineas, columnas, matrizDatos, covarianza);
39  MatrizInversa (columnas, covarianza, inversa);
40  multiplicarMatrices (columnas, distanciaCentroide, inversa,
41  transDistanciaCentroide, &Mahalanobis);
42
43  if (Mahalanobis<0)           // solamente la primera distancia puede ser <0
44  {                             // puesto que el centro del cluster no está definido
45      Mahalanobis = Mahalanobis * (-1);
46  }
47
48  return (Mahalanobis);
49  }

```

Figura 4.2. Código de la función *CalculaDistanciaDeMahalanobis*

4.4.3. señalReferencia

Con la función `señalReferencia` se genera la señal de referencia que hará disparar la alarma. Es directamente proporcional a la diferencia de la distancia menos la media e inversamente proporcional a la desviación típica de de las distancias. Recibe como parámetros de entrada la *lista* con todas las distancias calculadas hasta ahora y la media de estas. Devuelve el valor de la señal.

```
1 float senalReferencia (elemento *lista, float distancia, float media)
2 {
3     float desviacion;
4     float valorSenal;
5
6     desviacionTípica (lista, media, &desviacion);
7     valorSenal =(distancia-media)/desviacion;
8
9     return (valorSenal);
10 }
```

Figura 4.3. Código de la función `señalReferencia`

El código de la función `desviacionTípica` se puede ver en el apartado de anexos.

4.4.4. Función principal

La función principal empieza con la declaración de los ficheros de trabajo, declaración e inicialización de memoria dinámica y con la declaración e inicialización de las variables de programa. Una vez definidas e inicializadas las variables, el resto del main, podemos estructurarlo en dos partes bien diferenciadas. Una primera parte que utiliza la señal de los ficheros de trabajo `_01` donde se define la ventana de tiempo que nos interesa de la señal de los ficheros `_02`, y una segunda parte, donde se implementa el predictor y se generan todos los datos para su posterior análisis.

De la observación de la señal `_01` de los ficheros `01`, sabemos que esta comienzan con un valor constante cercano a cero hasta que empiezan a descender llegando a alcanzar valores por debajo de `MAX_VENTANA1` (definido como constante), luego se mantienen más o menos constantes en valores muy negativos hasta el instante que empiezan a crecer llegando a superar el valor `MAX_VENTANA2` (definido también como constante), por lo tanto, la ventana de tiempo de la señal `_02` vendrá determinado por:

tiempo inicial: señal_01 < MAX_VENTANA1

tiempo final: señal_01 > MAX_VENTANA2

el código quedaría de la siguiente forma:

```
1  int main(void)
2  {
3
4  /* Declaración de ficheros de trabajo */
5  /*Declaración e inicialización de memoria dinámica */
6  /* Declaración e inicialización de variables de programa */
7
8  /* Apertura de ficheros de trabajo */
9
10 // DEFINICION DE LA VENTANA DE DATOS
11
12 fscanf(f_01, "%f", &valores);
13 contador_fichero++;
14
15 while (!feof(f_01))&&(!salir))
16 {
17     if (ts==1)          // se activa el primer time para la ventana
18     {
19         tiempo1 = valores; // tiempo inicio (ms) de la ventana
20         ts = 2;
21     }
22     if (ts==3)          // se activa el segundo time para la ventana
23     {
24         tiempo2 = valores; // tiempo final (ms) de la ventana
25         salir = 1;
26     }
27
28     if (contador_fichero%2==0)
29     {
30         if ((valores<MAX_VENTANA1)&&(ts==0))
31             ts = 1;
32
33         if ((valores>MAX_VENTANA2)&&(ts==2))
34             ts = 3;
35     }
36
37     fscanf(f_01, "%f", &valores);
38     contador_fichero++;
39 }
40 fclose (f_01);
41
```

```

42 printf("%f %f\n", tiempo1, tiempo2); // tiempo inicial y tiempo final de la
43                                     ventana de datos

```

Figura 4.4. Código de la primera parte de la función main

Una vez obtenido el tiempo inicial y el final se pasa a la implementación del predictor y obtención de los datos para su posterior análisis.

El código empieza iniciando el *contador_fichero* a cero, realizando la lectura del primer dato del fichero e iniciando el bucle que recorre todo el fichero_02. El proceso que se sigue es el siguiente.

- Mientras que los valores leídos no estén dentro del intervalo de tiempo (*tiempo1*, *tiempo2*) no se hace nada, sólo se incrementa el contador de fichero.
- Si el valor leído del fichero corresponde a un valor de la señal (no de tiempo) y está dentro de la ventana de tiempo entonces se procesa.
- Si se han leído menos valores que el tamaño máximo de la ventana, MAX_MUESTRAS, se forma la primera ventana, si no, se desplaza los valores de la ventana una posición para almacenar el último dato.
- Cuando se desplaza tantas muestras como indica DESPLAZAMIENTO se procesa la nueva ventana.
- Se calculan los coeficientes de aproximación de la nueva ventana en el nivel de descomposición que indica NIVEL. Se guardan en el fichero de trabajo **coeficientesAproximacion.txt** y se almacenan en la memoria dinámica.
- Se calcula la distancia de Mahalanobis de los coeficientes calculados respecto a los demás. Guardamos la distancia en el fichero de trabajo **distanciasMahalanobis.txt** y se almacena en la memoria dinámica.
- Por último, se guarda la señal de referencia en el fichero de trabajo **senalReferencia.txt**.

A continuación se muestra el código de la segunda parte.

```

1  contador_fichero=0;
2  fscanf(f_02, "%f", &valores); // lee primer valor del fichero
3  contador_fichero++;
4
5  while (!feof(f_02)) // mientras no se llegue al final del fichero
6  {
7      if ((contador_fichero%2==1)&&(valores>=tiempo1)&&
8          (valores<=tiempo2)) // lecturas impares, rango de tiempo calculado
9      {
10         fscanf(f_02, "%f", &valores); // lee primer valor del fichero

```

```

11     contador_fichero++;
12     if (contador_fichero%2==0)           // Lecturas pares, valores de amplitud
13     {
14         if (contador<=MAX_MUESTRAS)
15             amplitud[contador-1]=valores;
16         else
17         {
18             for (i=1; i<MAX_MUESTRAS; i++)           // desplazamos el vector
19                 amplitud[i-1]=amplitud[i];
20             amplitud[MAX_MUESTRAS-1]=valores; // se guarda el último valor leído
21             contador_desplazamiento++;
22         }
23
24     if (((contador>=MAX_MUESTRAS)&&
25         (contador_desplazamiento%DESPLAZAMIENTO==0))
26         ||(contador==MAX_MUESTRAS))
27     {
28         contador_coeficientes++;
29         calculaCoeficientesDeAproximacion (numeroCoeficientes, amplitud,
30     coeAprox);
31         for (i=0; i<numeroCoeficientes; i++)
32         {
33             anadir(&listaCoeficientes,coeAprox[i]);
34             fprintf(fcoeficientes, "%f ",coeAprox[i]);
35         }
36         fprintf(fcoeficientes, "\n");
37
38         if (contador_coeficientes>=numeroCoeficientes) // no se calcula
39                                                         mahalanobis
40                                                         hasta que el nº de
41                                                         distancias
42                                                         >= nº coeficientes
43         {
44             distanciaMahalanobis=CalculaDistanciaMahalanobis
45                                     (contador_coeficientes,
46                                     numeroCoeficientes, coeAprox,
47                                     listaCoeficientes);
48             fprintf(fdistancias, "%f",distanciaMahalanobis);
49             fprintf(fdistancias, "\n");
50             anadir(&listaDistancias,distanciaMahalanobis);
51
52             aux = aux+distanciaMahalanobis;
53             media = aux/contadorDistancias;
54             contadorDistancias++;
55             valorSenal = senalReferencia (listaDistancias,
56                                         distanciaMahalanobis,
57                                         media);           //detectaOutlier

```

```
58     fprintf(fsenaReferencia, "%f", valorSenal);
59     fprintf(fsenaReferencia, "\n");
60
61     /*if (valorSenal>UMBRAL)
62     {
63         alarmaON = 1;    // activar la alarma deshabilitado.
64                         // Fase entrenamiento nos conviene que llegue al
65                         // final de las descargas
66     }*/
67     }
68     /*4o if*/
69     contador++;
70     /*2o if*/
71     /*1er if*/
72
73     fscanf(f_02, "%f", &valores);
74     contador_fichero++;
75 }/*while*/
76
77 /* Se cierran los ficheros */
78
79 system("PAUSE");
80 return(0);
81
82 }/*fin main*/
```

Figura 4.5. Código de la segunda parte del main

Al final del código aparece deshabilitado la condición de disparo de la alarma ya que para la toma de datos nos conviene que las descargas lleguen hasta el final.

En el apartado de anexos se puede ver el código utilizado para manejar las estructuras dinámicas de datos así como el código del programa principal.

CAPÍTULO 5

RESULTADOS OBTENIDOS DE LA FASE DE ENTRENAMIENTO DEL SISTEMA DE PREDICCIÓN

En este apartado, vamos a aplicar el sistema de interrupción a una serie de descargas, con el fin de poder obtener una primera aproximación de cuál es su comportamiento. El objetivo es ir comparando la evolución de la señal de referencia que se va generando a medida que la descarga avanza.

5.1. Descripción de las descargas

Para la fase de entrenamiento del predictor se han escogido 10 descargas de la serie JPF_824xx que incluye descargas disruptivas como no disruptivas. Estas tienen una duración de alrededor de 60 segundos, aunque la ventana de tiempo para el análisis es de entorno a 30 segundos.

5.2. Análisis de detección. Condición de disparo de la alarma

En los ensayos que se van a mostrar en las siguientes secciones, se tomará un valor K_M (valor umbral de la señal de referencia) igual a 10. Recordemos que el valor de esta constante, dependerá del tipo de dispositivo de fusión, del sistema de interrupción empleado..., por lo que la precisión de los resultados estarán sujetos a una buena elección de esta. El disparo de la alarma se producirá en el momento que el valor de la señal de referencia sea mayor a K_M ,

$$\text{Condición disparo de la alarma: } V_{ref} > K_M$$

5.3. Análisis de descargas con ventanas de tiempo de 32 ms, 2 muestras de deslizamiento y nivel 4 de descomposición

Para la simulación del comportamiento del sistema de predicción y la obtención de datos, utilizaremos en las distintas descargas, ventanas de tiempo de de 32 ms con un desplazamiento de 2 ms, y un nivel de descomposición de 4. Por lo que la señal de referencia se generará a partir de conjuntos de datos formados por dos coeficientes de aproximación. A continuación se muestran los resultados obtenidos en cada una de las descargas.

5.3.1. Descarga JPF_82460

Descarga 82460, tiene una duración de tiempo de 34,913448 segundos a 99,90815. Se ha obtenido una ventana de tiempo que va de 41.336 a 57,02732 segundos.

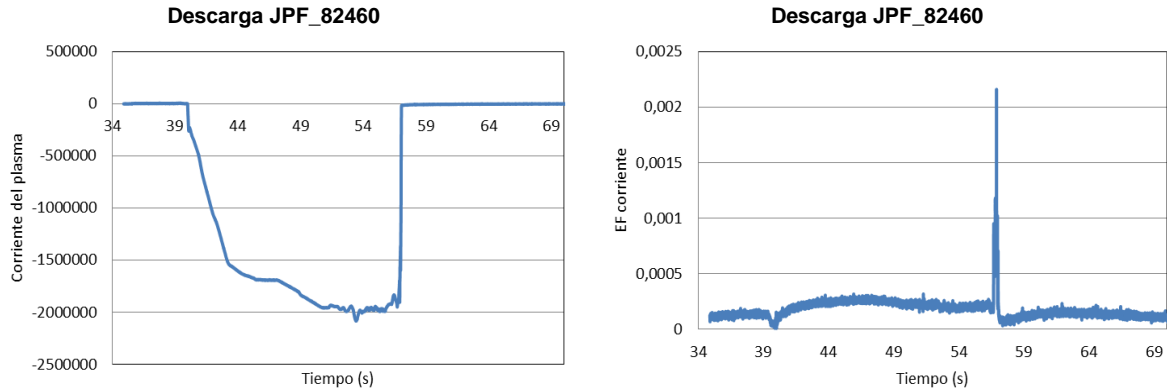


Figura 5.1. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

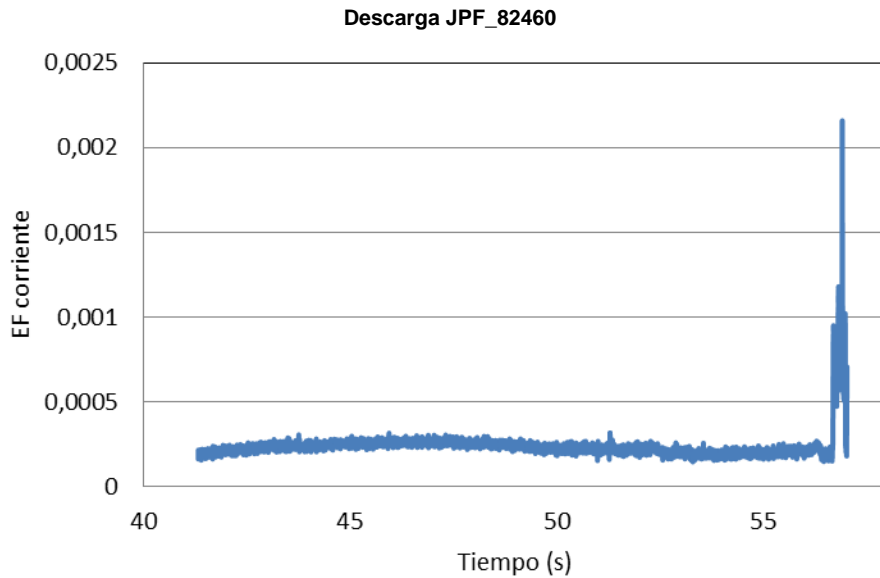


Figura 5.2. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

En la figura 5.2, en una primera observación se evidencia un comportamiento disruptivo, se puede ver que cerca del segundo 57 aparece una gran variación de los valores del EF de corriente de la señal.

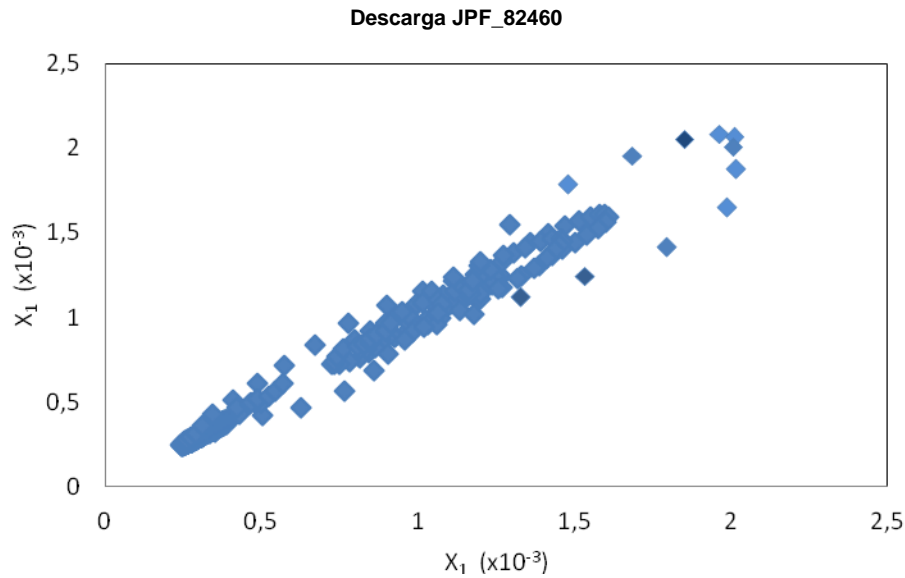


Figura 5.3. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

La figura muestra una covarianza positiva y una forma elíptica en los datos. Varios de los puntos presenta una distancia al centro del cluster mucho mayor que el resto de los valores.

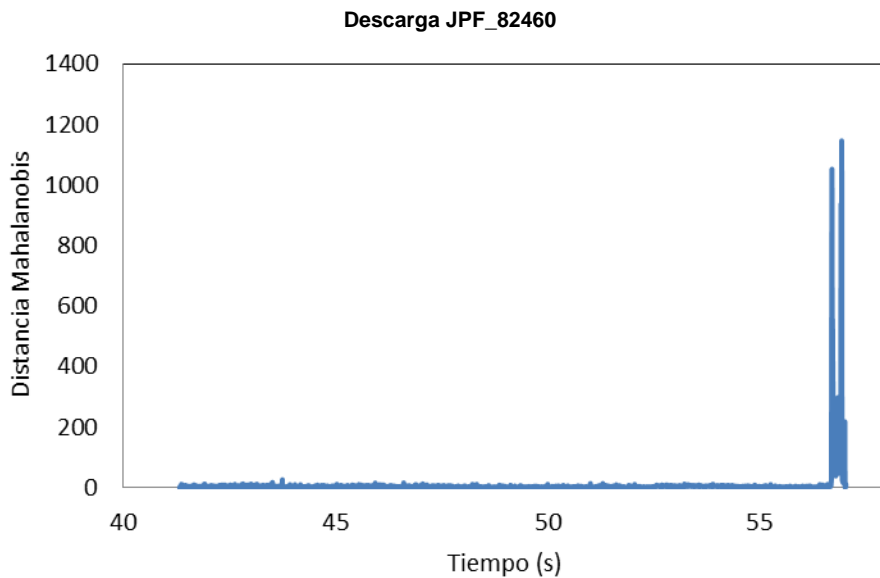


Figura 5.4. Evolución temporal de la distancia de Mahalanobis de la descarga

En la evolución de las distancias de Mahalanobis respecto al centro del cluster puede observarse que hay valores que están muy por encima de la desviación estándar, por lo que todo hace pensar que en las cercanías del segundo 57 se dispare la alarma, momento en el cuál comienzan a formarse la inestabilidades.

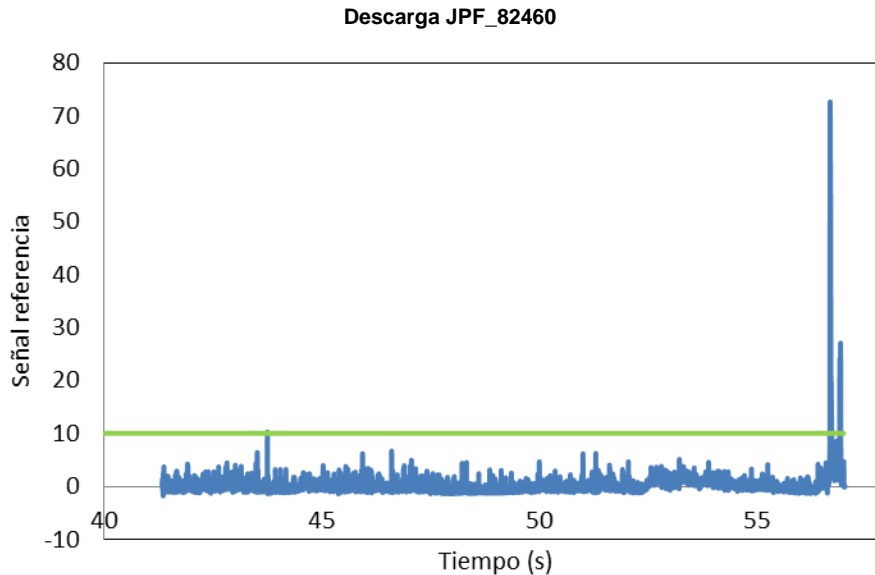


Figura 5.5. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

En la evolución temporal de la señal de referencia se identifica que entorno al segundo 57 de la descarga aparecen valores atípicos que están por encima del umbral de referencia. En cambio, haciendo un zoom de la señal de referencia, apreciamos que en el instante de tiempo $t = 43,79$ segundos, la señal alcanza un valor por encima de 10, lo que haría disparar una falsa alarma de interrupción, interrumpiendo la descarga y con la consiguiente pérdida de datos.

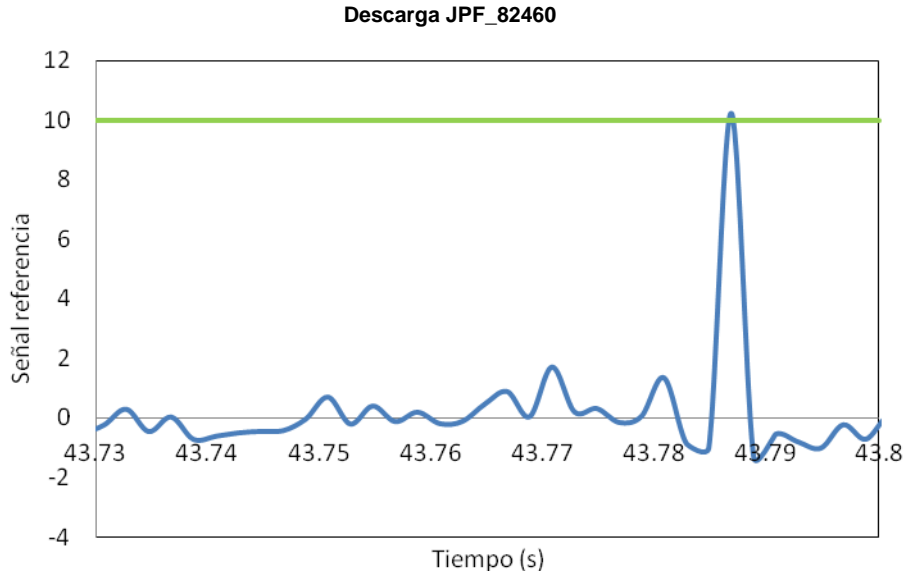


Figura 5.6. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$

Podemos decir entonces, que para un valor de K_M igual 10 se produce la aparición de una alarma prematura. Ya que el momento de la presencia de inestabilidades en la descarga ocurre en $t = 56$ segundos. En la siguiente figura se observa ese instante de tiempo con mayor precisión.

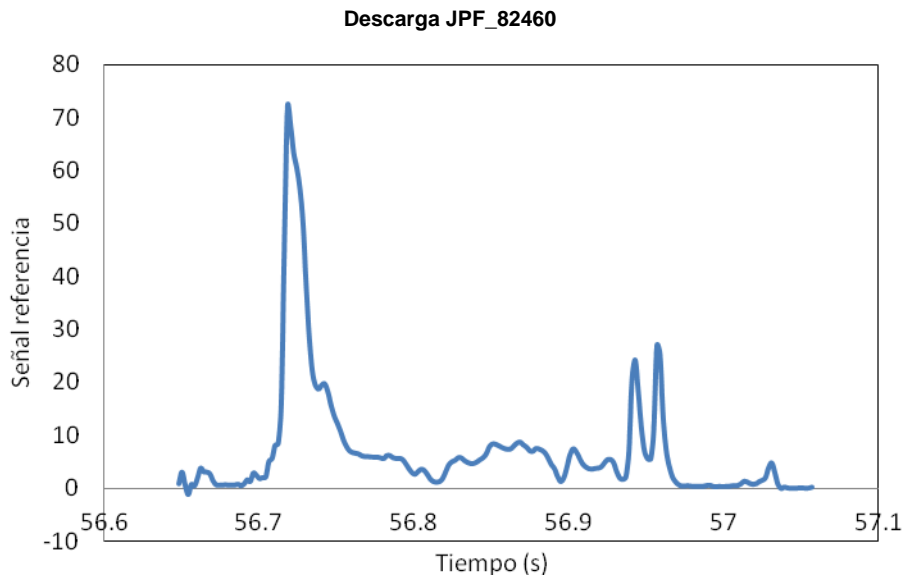


Figura 5.6 bis. Evolución temporal de la señal de referencia aplicando zoom al momento de aparición de las inestabilidades

5.3.2. Descarga JPF_82462

Descarga 82462, tiene una duración de tiempo de 34,737438 segundos a 95,402298. Se ha obtenido una ventana de tiempo que va de 41,483978 a 63,214558 segundos.

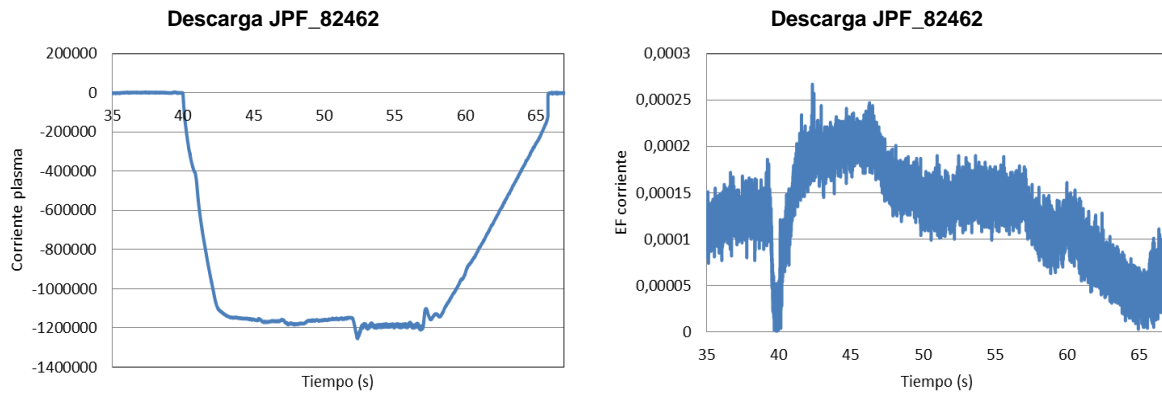


Figura 5.7. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

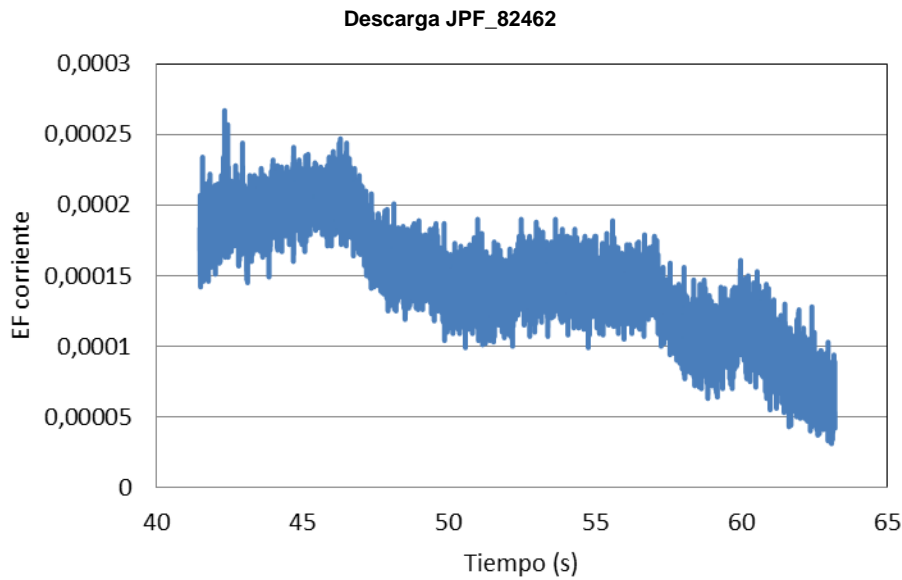


Figura 5.8. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

A simple vista, vemos que los valores del espectro de Fourier de la corriente no presentan gran variación. No existe la aparición de outlier. Estamos frente a un proceso de descarga no disruptiva.

Será importante ver cuál va a ser el comportamiento del sistema frente al cambio que se produce al inicio de la descarga, el predictor detectará un inicio de disrupción y por consiguiente activará una falsa alarma, o en cambio sabrá discernirlo como un valor anómalo.

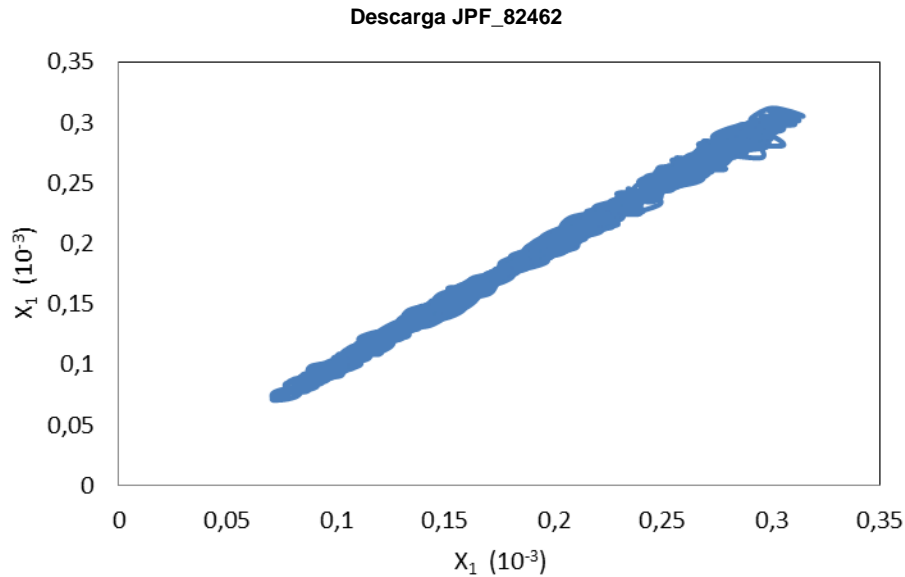


Figura 5.9. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

El diagrama de dispersión de los coeficientes de aproximación presenta un cluster compacto con varianza positiva y forma elíptica, típica de una señal sin cambios bruscos en el flujo de datos.

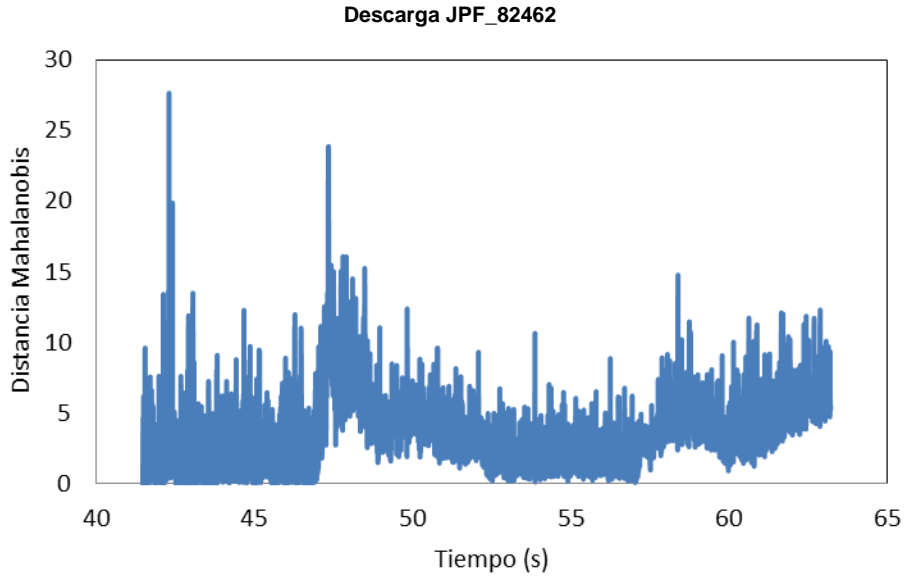


Figura 5.10. Evolución temporal de la distancia de Mahalanobis de la descarga

La representación temporal de las distancias muestra valores con distancias superiores a la media que pueden dar origen a la detención de la descarga.

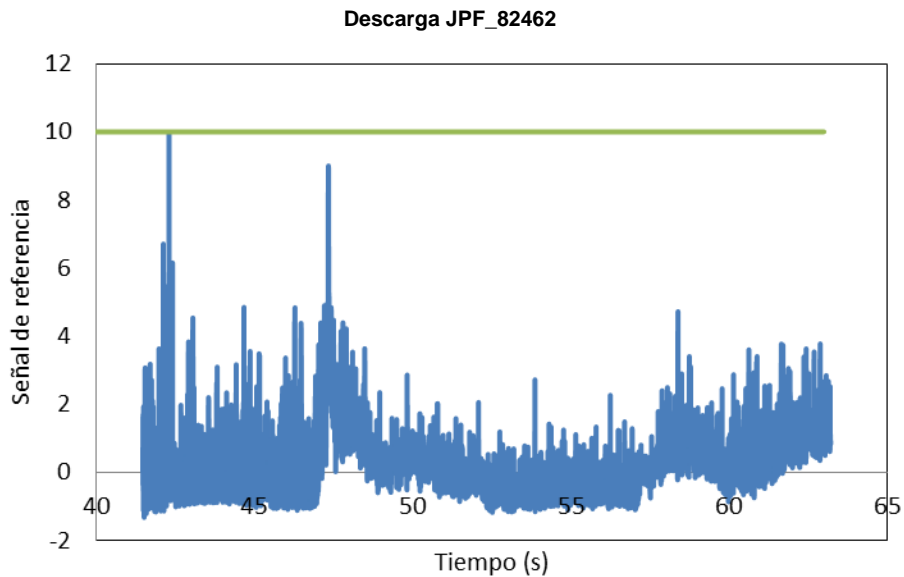


Figura 5.11. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

Finalmente, en la señal de referencia se observa que alrededor del segundo 41 aparece un valor alto pero sin llegar a rebasar el umbral para la detección.

Siendo el comportamiento correcto del sistema. Estamos frente a una descarga que ha resultado ser no disruptiva.

5.3.3. Descarga JPF_82463

Descarga 82463, tiene una duración de tiempo de 34,938068 segundos a 95,437073. Se ha obtenido una ventana de tiempo que va de 41,483631 a 54,669769 segundos.

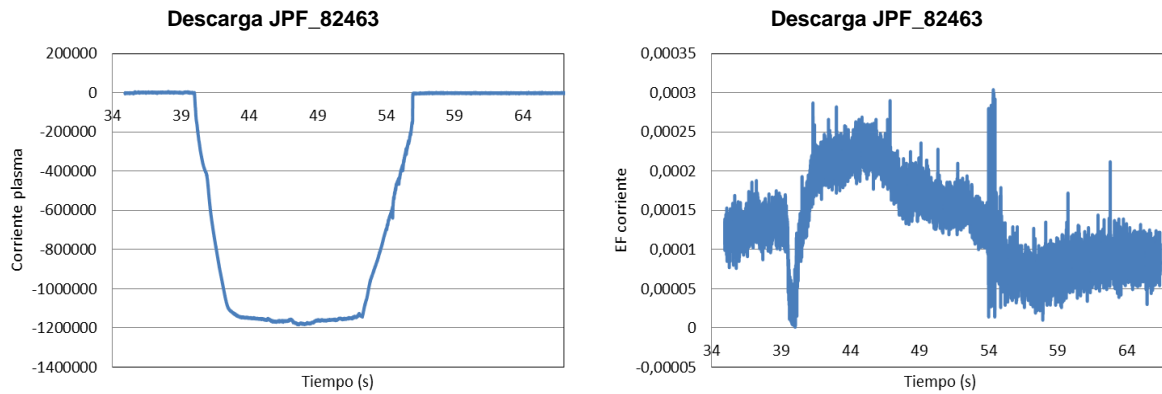


Figura 5.12. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

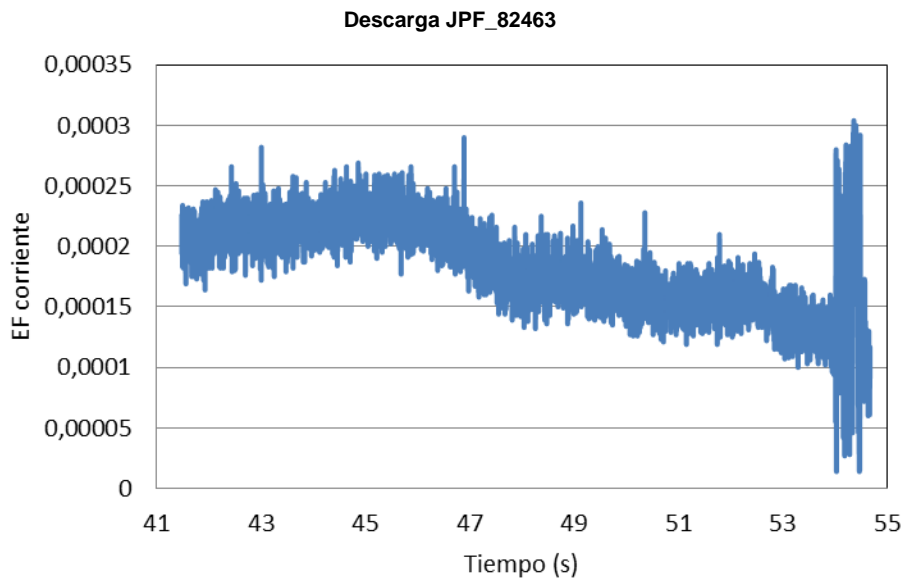


Figura 5.13. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

De la observación de la evolución de la señal, podemos deducir que estamos frente al caso de una descarga disruptiva. En los valores finales aparece gran variación en el flujo de datos. Será importante ver, cuál es el momento en el que el predictor dispara la alarma, ya que durante la descarga aparecen varios niveles no deseados de la señal.

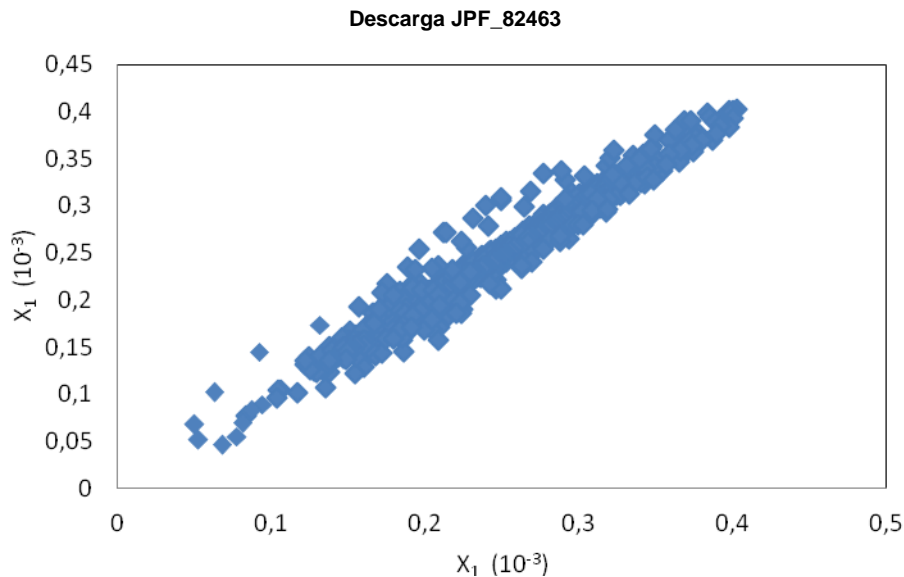


Figura 5.14. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

El diagrama de dispersión de los coeficientes de aproximación nos muestra, que efectivamente, existe varios puntos lo suficientemente lejos para asegurar que nos encontramos ante una descarga disruptiva. Como en los casos anteriores, se observa una covarianza positiva y con la forma elíptica típica de las descargas en JET.

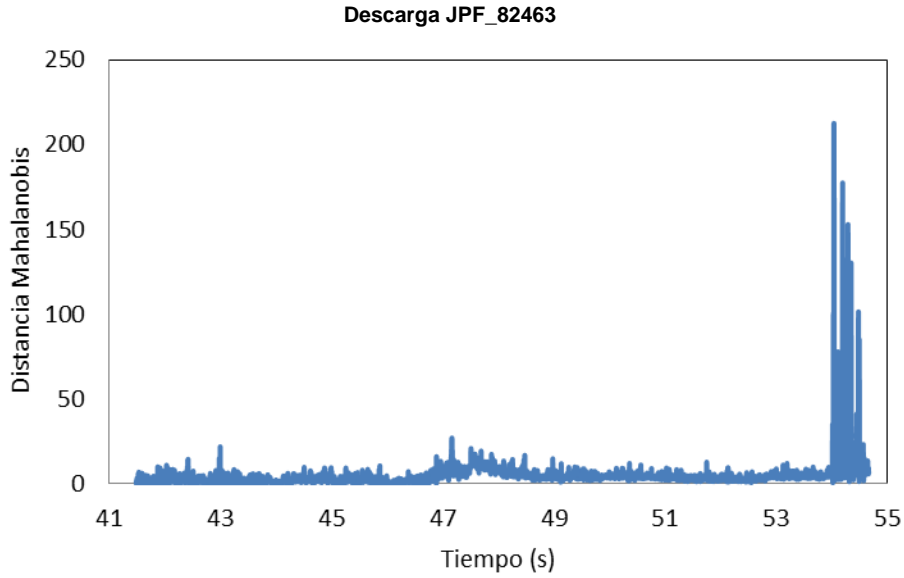


Figura 5.15. Evolución temporal de la distancia de Mahalanobis de la descarga

En la evolución en el tiempo de las distancias se pueden ver que el momento de la interrupción ocurre aproximadamente a los 54 segundos del inicio de la descarga, sin embargo, podemos apreciar que antes de llegar a ese tiempo aparecen algunos valores capaces de interrumpir la descarga.

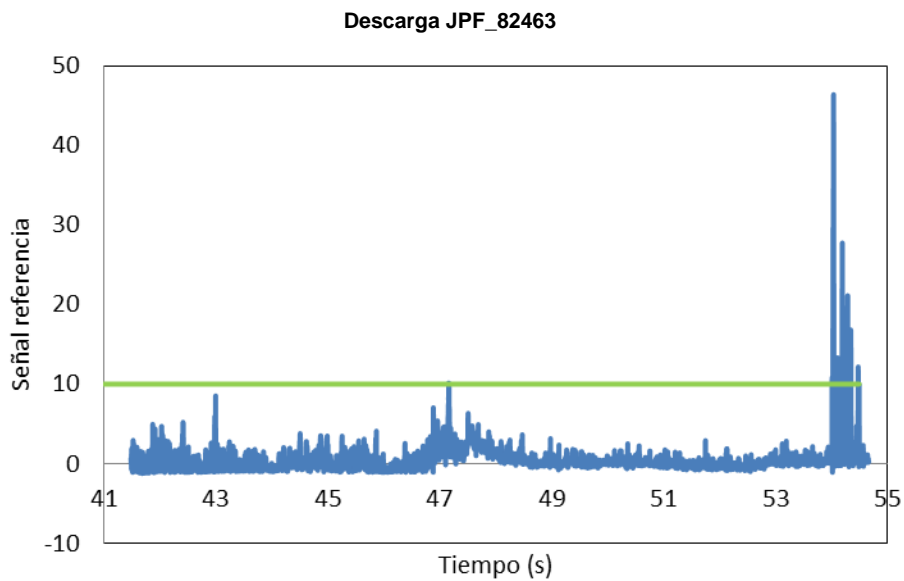


Figura 5.16. Evolución temporal de la señal de referencia y valor del umbral de $K_M=10$

Con la ayuda de un zoom en la señal de referencia desvelamos que estamos ante un caso nuevo de alarma prematura producida por un valor anómalo de la señal. Aparece en el instante de tiempo $t = 47,1$ segundos y alcanza un valor que sobrepasa por décimas el valor umbral. Si observamos, tanto la evolución de la señal como la evolución de la distancia vemos que este instante coincide con la aparición de un outlier en la señal.

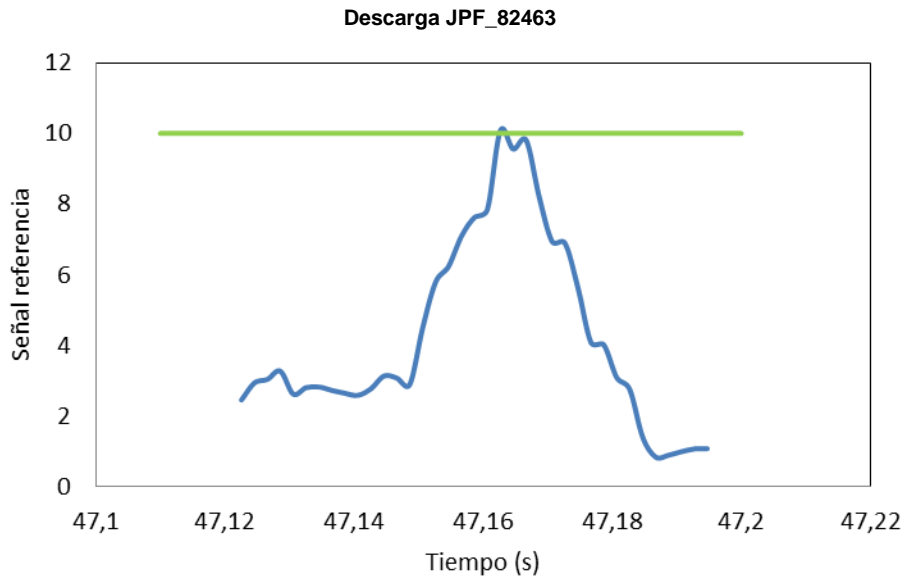


Figura 5.17. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$

En el momento de la disrupción, el valor de la señal de referencia supera el umbral varias veces en una décima de segundos. A continuación se muestra con más detalle el comportamiento de la señal de referencia en las proximidades de la disrupción.

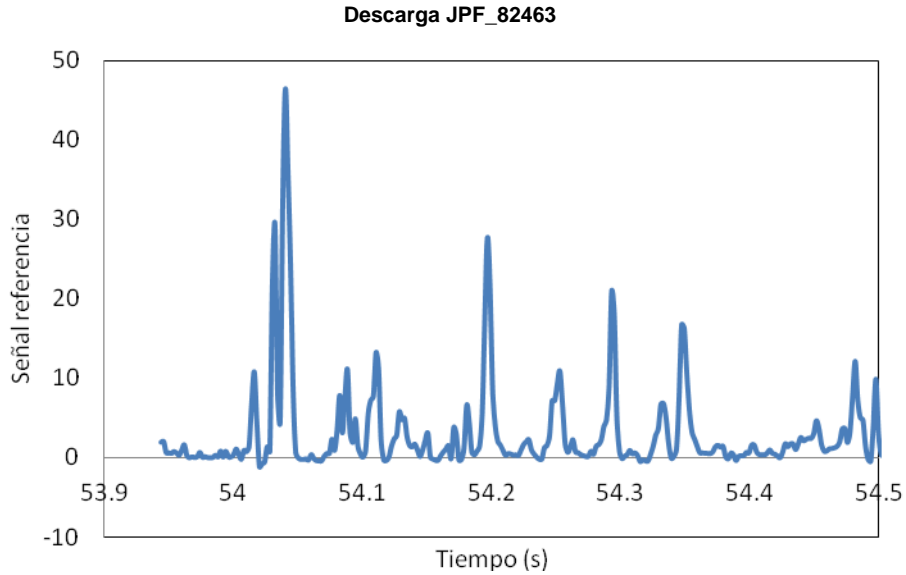


Figura 5.17 bis. Evolución temporal de la señal de referencia aplicando zoom al momento de la aparición de inestabilidades

5.3.4. Descarga JPF_82467

Descarga 82467, tiene una duración de tiempo de 34,890736 segundos a 95,625648. Se ha obtenido una ventana de tiempo que va de 41,488304 a 63,187889 segundos.

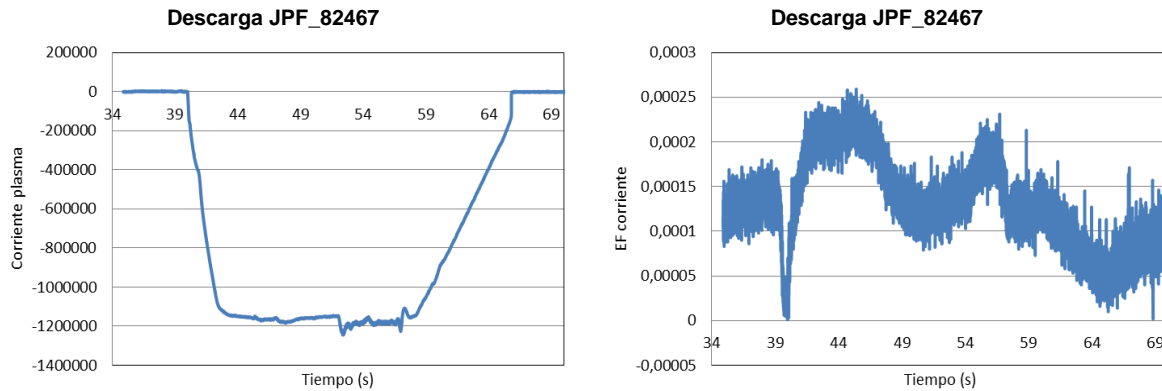


Figura 5.18. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

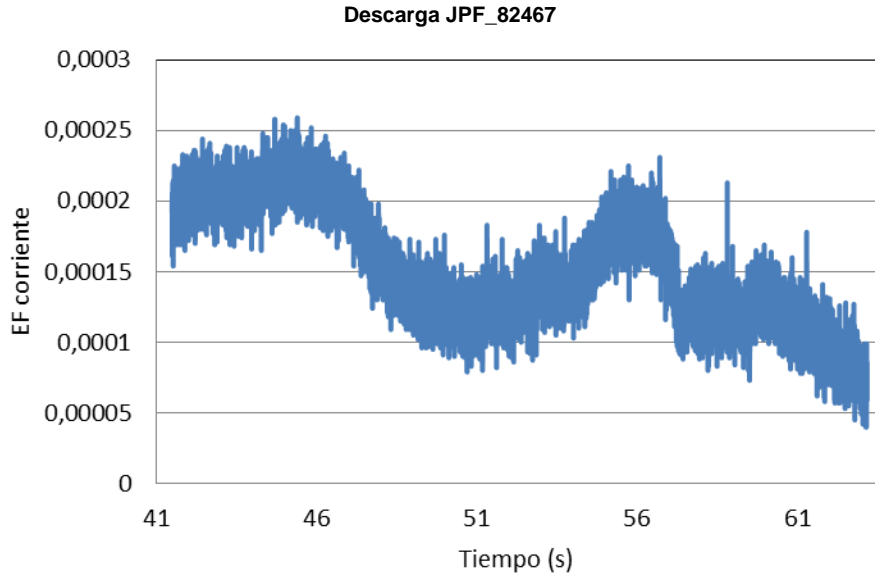


Figura 5.19. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

La evolución temporal de la señal evidencia un carácter no disruptivo. Habrá que ver cuál es la respuesta del sistema frente a los valores anómalos.

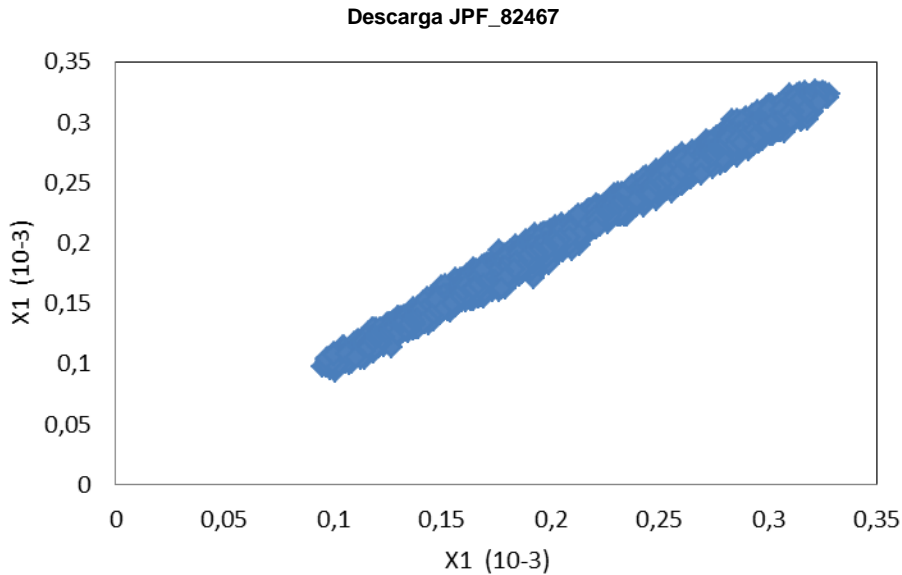


Figura 5.20. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

Los puntos hallados con la transformada de wavelet Haar presentan un clúster totalmente compacto con una tendencia positiva y con forma elíptica.

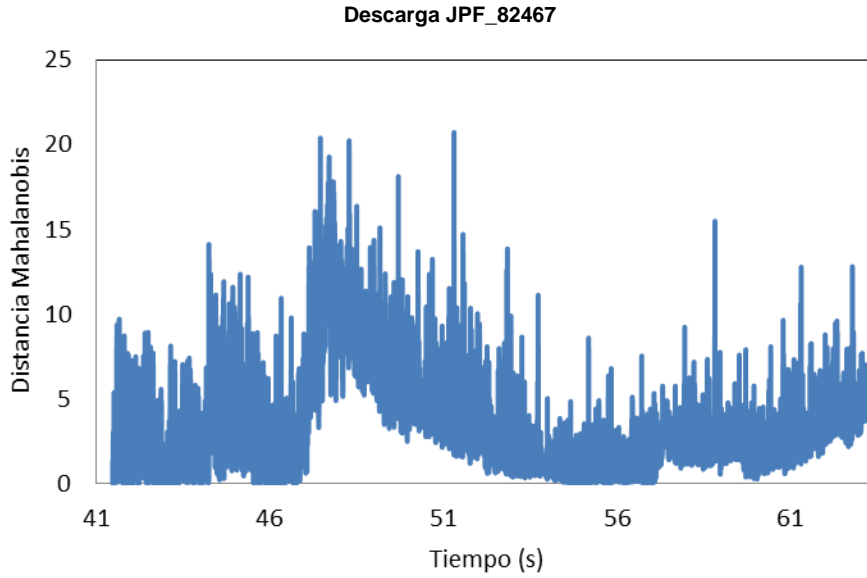


Figura 5.21. Evolución temporal de la distancia de Mahalanobis de la descarga

El conjunto de las distancias no presentan gran desviación en sus valores frente a la media, por lo que todo hace indicar que el comportamiento del predictor es el esperado.

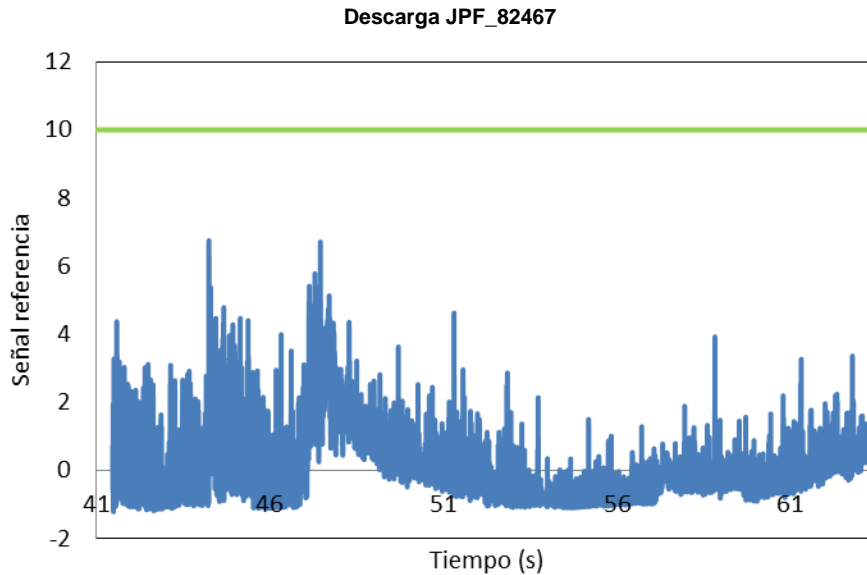


Figura 5.22. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

La señal de referencia denota que estamos en un caso de descarga no disruptiva, sin falsas alarmas y alarmas prematuras.

5.3.5. Descarga JPF_82468

Descarga 82468, tiene una duración de tiempo de 34,8762247 segundos a 95,554276. Se ha obtenido una ventana de tiempo que va de 41.481815 a 63,190399 segundos.

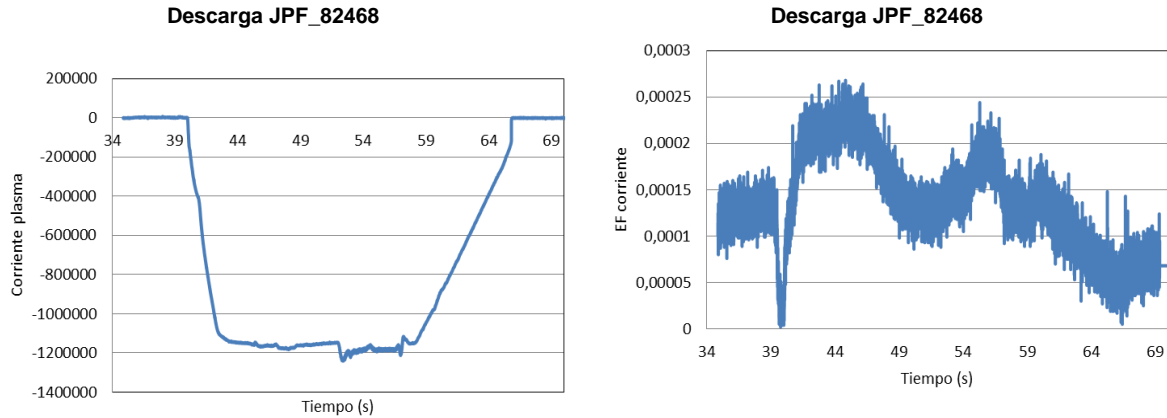


Figura 5.23. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

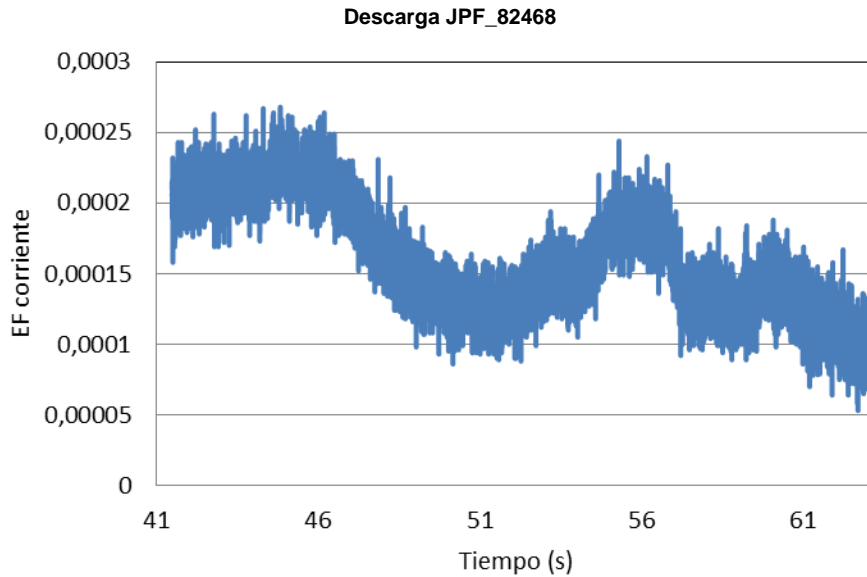


Figura 5.24. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

En la figura, se puede ver un comportamiento de la señal muy parecido al de la descarga 82667, por lo que no nos entretendremos en ella. Su evolución temporal no refleja cambios bruscos en el flujo de datos por lo que no evidencia un comportamiento disruptivo, tampoco existe la aparición de valores anómalos.

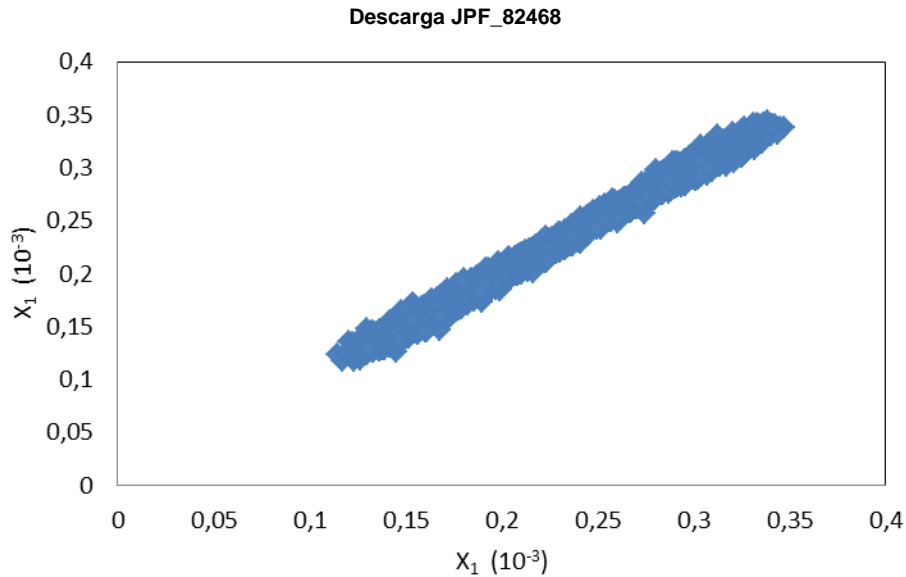


Figura 5.25. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

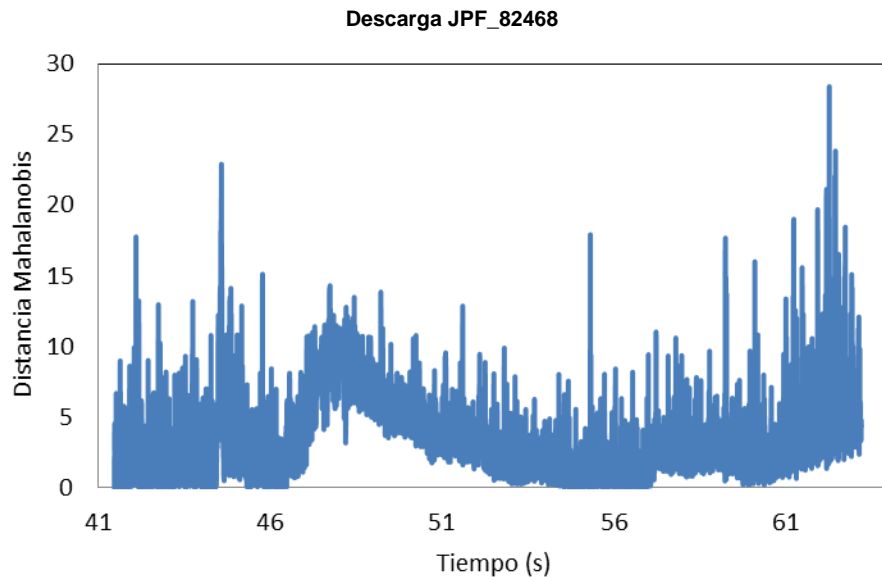


Figura 5.26. Evolución temporal de la distancia de Mahalanobis de la descarga

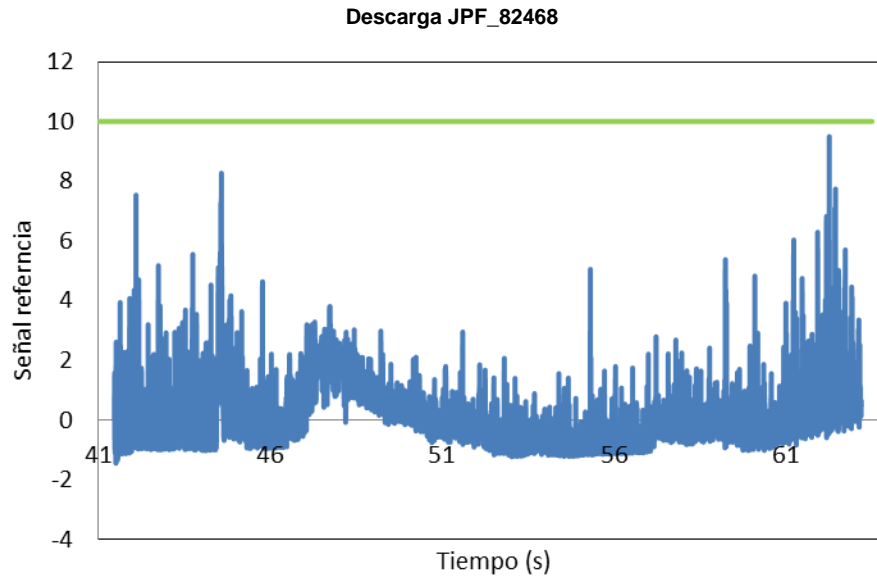


Figura 5.27. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

Finalmente, de la observación de la señal de referencia se puede confirmar que se trata de una descarga sin inestabilidades y que ha llegado a su fin ya que tampoco se ha disparado una falsa alarma.

5.3.6. Descarga JPF_82469

Descarga 82469, tiene una duración de tiempo de 34,774281 segundos a 95,229317. Se ha obtenido una ventana de tiempo que va de 41,485847 a 63,708427 segundos.

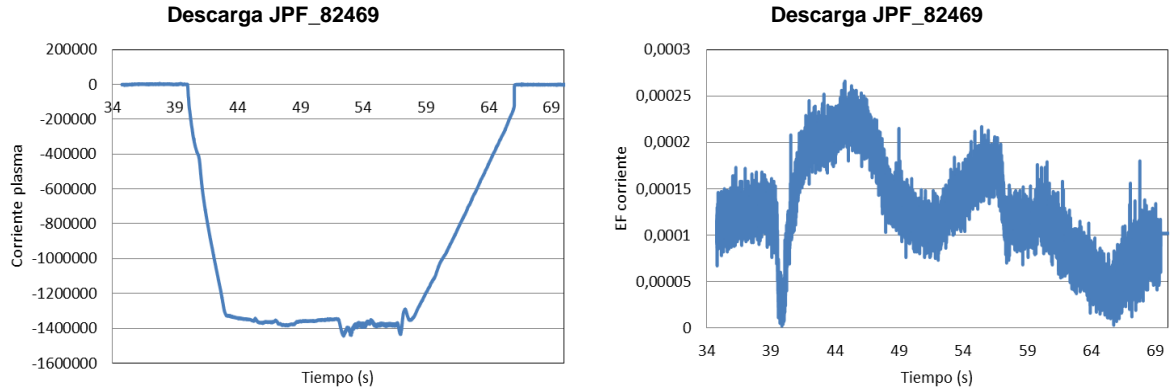


Figura 5.28. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

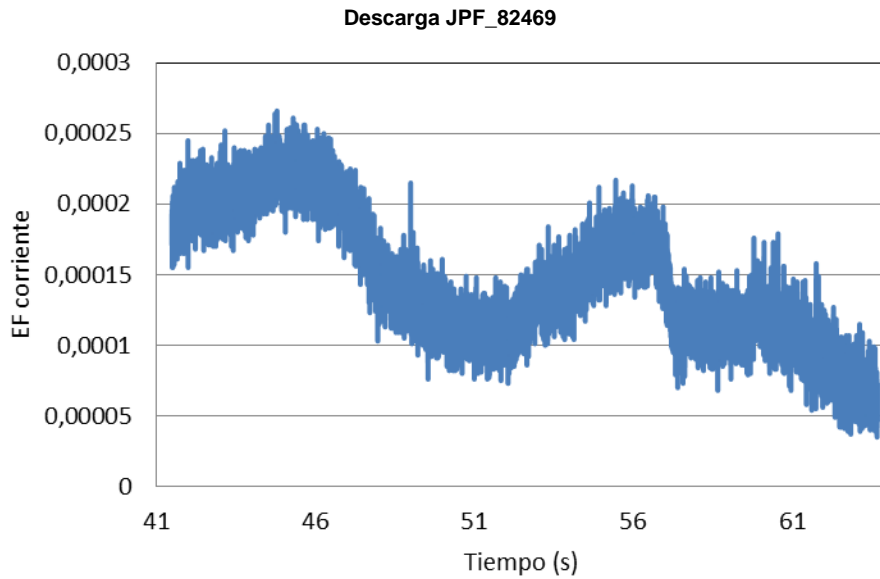


Figura 5.29. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

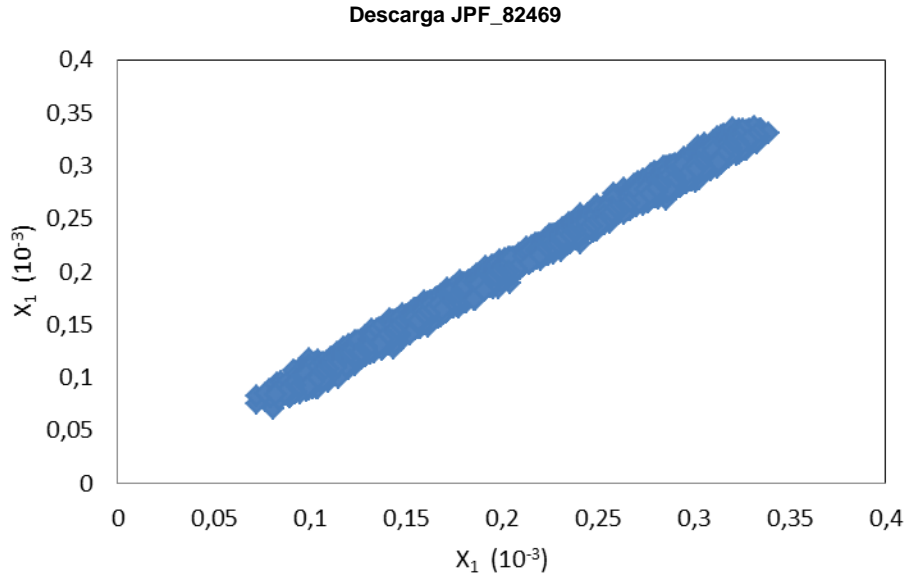


Figura 5.30. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

En cluster de datos totalmente compacto hace presagiar un carácter no disruptivo de la descarga. Estamos, entonces, en un caso igual al de los dos anteriores.

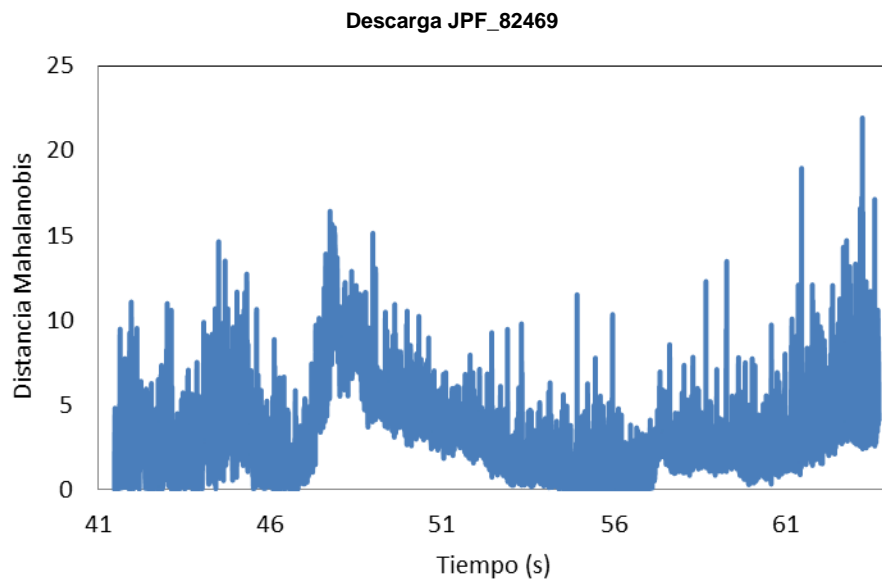


Figura 5.31. Evolución temporal de la distancia de Mahalanobis de la descarga

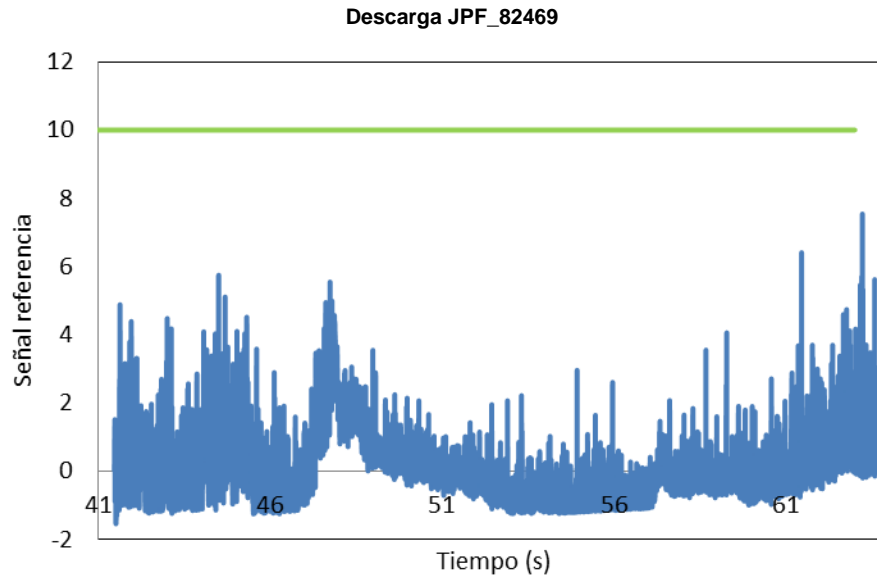


Figura 5.32. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

Los valores de la señal de referencia están por debajo del umbral de disparo, por lo que de nuevo, la descarga puede llegar al final de su ciclo debido a la ausencia de disparos de falsas alarmas.

5.3.7. Descarga JPF_82471

Descarga 82471, tiene una duración de tiempo de 34,787212 segundos a 95,928162. Se ha obtenido una ventana de tiempo que va de 41,477772 a 63,180344 segundos.

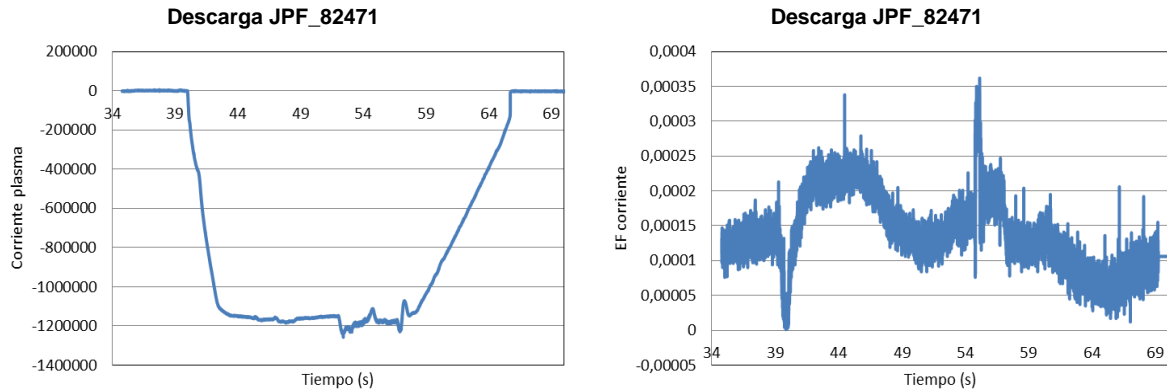


Figura 5.33. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

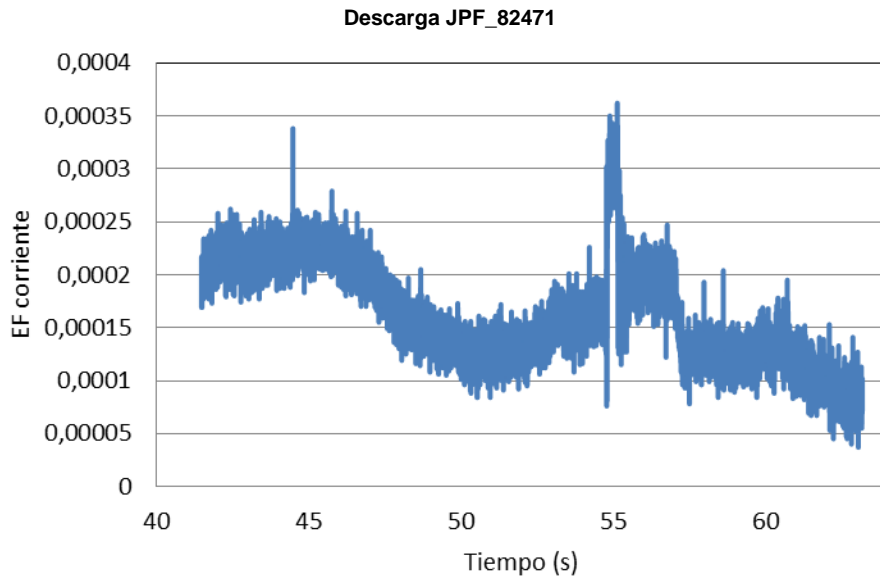


Figura 5.34. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

De la observación de la evolución temporal de la descarga se puede evidenciar un comportamiento disruptivo. Podríamos decir que el momento en que aparecen las inestabilidades ocurre a los 55 segundos de la descarga, sin embargo, se aprecia un gran outlier en el instante de tiempo $t = 44,5$ a continuación veremos cuál es el comportamiento del predictor ante ese valor.

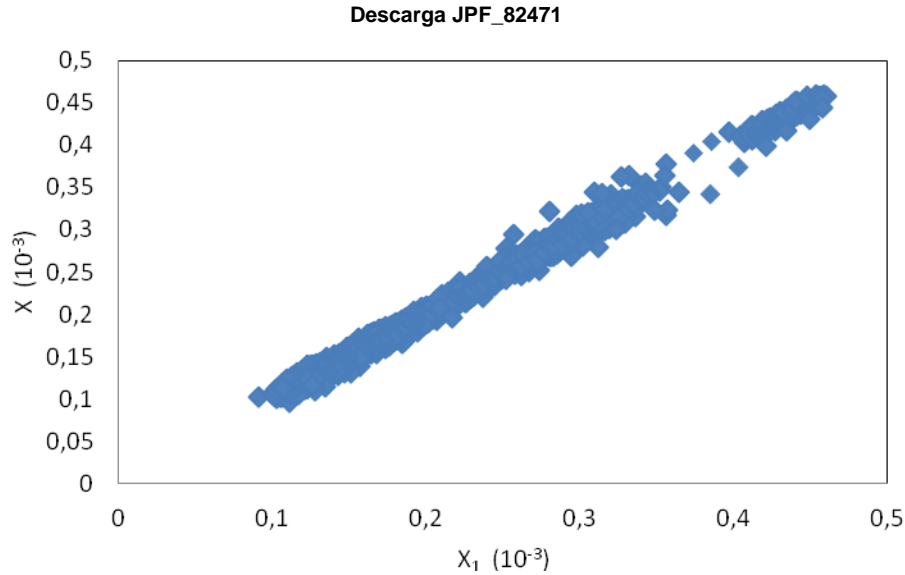


Figura 5.35. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

En el diagrama de dispersión se puede ver que aparecen gran cantidad de puntos que se encuentran separados del centro del cluster. El cluster presenta una tendencia positiva con forma elíptica.

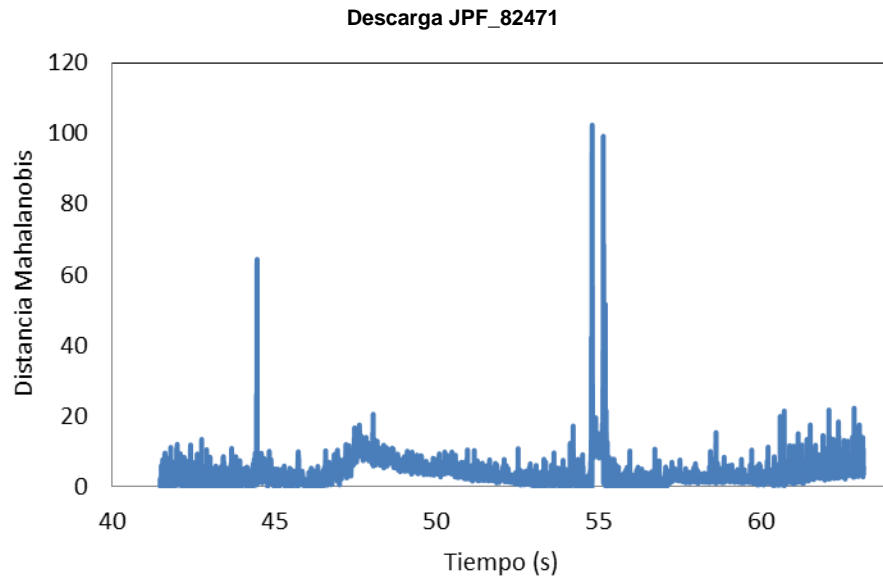


Figura 5.36. Evolución temporal de la distancia de Mahalanobis de la descarga

La evolución temporal de las distancias muestra la existencia de varios puntos alejados lo suficientemente para provocar el disparo de la alarma. En la

representación de la señal de referencia se observa que existen varios puntos por encima del umbral.

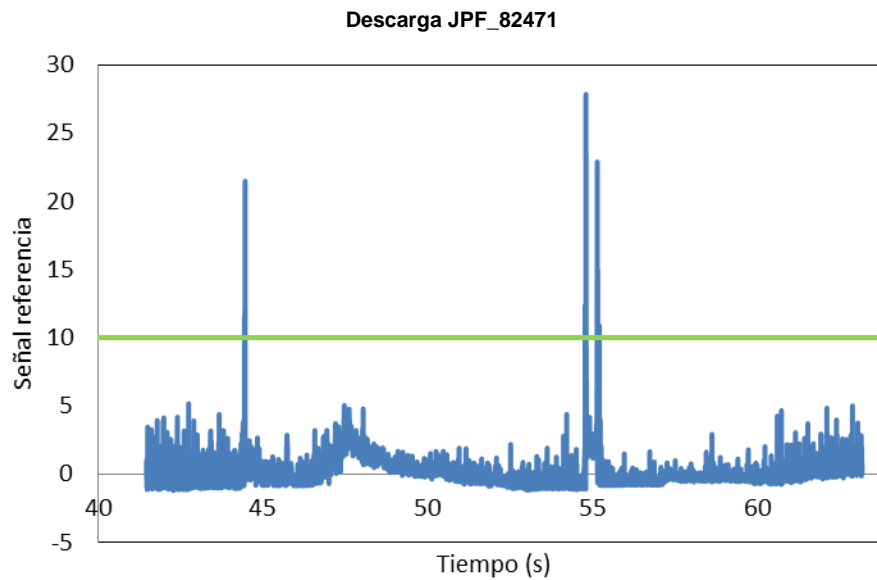


Figura 5.37. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

Como se puede apreciar, el valor del outlier es lo suficientemente grande como para disparar la alarma antes de tiempo, por consiguiente, estamos frente a un caso de alarma prematura. A continuación se muestra más detalladamente cuál es el comportamiento de la señal de referencia en el momento del disparo de la alarma y en el momento de la aparición de la interrupción.

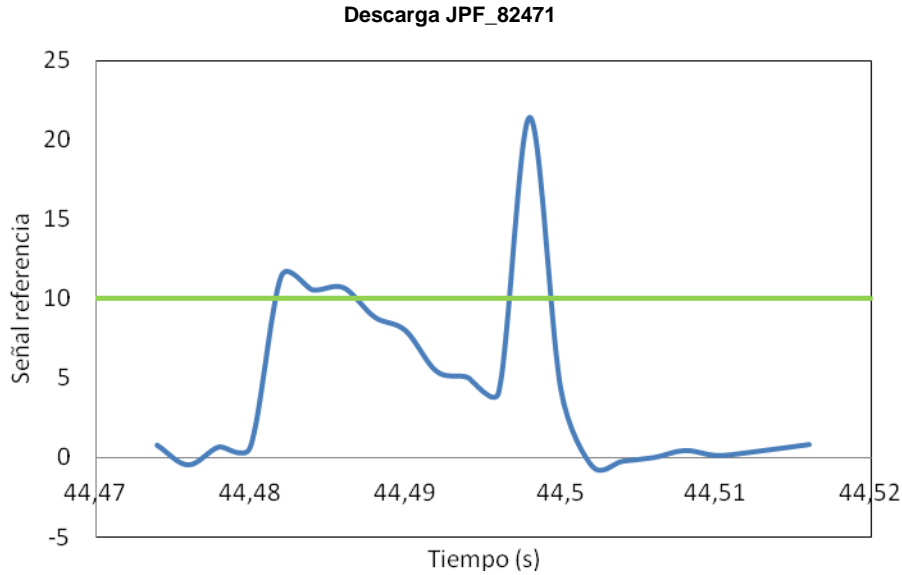


Figura 5.38. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$

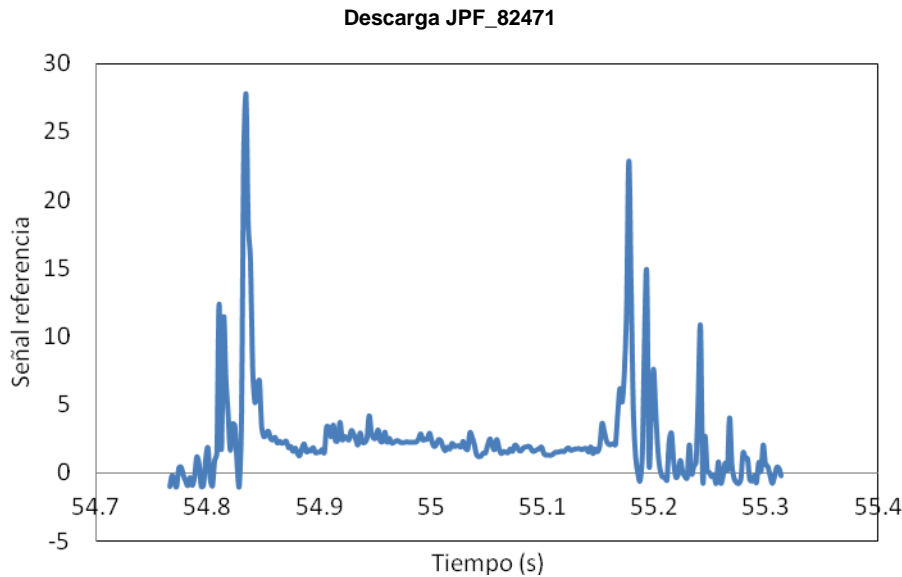


Figura 5.38bis. Evolución temporal de la señal de referencia aplicando zoom al momento de la aparición de interrupciones

El momento del disparo de la alarma se produce en $t = 44,48$ segundos, aunque las inestabilidades no empiezan a aparecer hasta los 54,8 segundos de la descarga.

5.3.8. Descarga JPF_82472

Descarga 82472, tiene una duración de tiempo de 34,768211 segundos a 96,028831. Se ha obtenido una ventana de tiempo que va de 41.483753 a 56,64576 segundos.

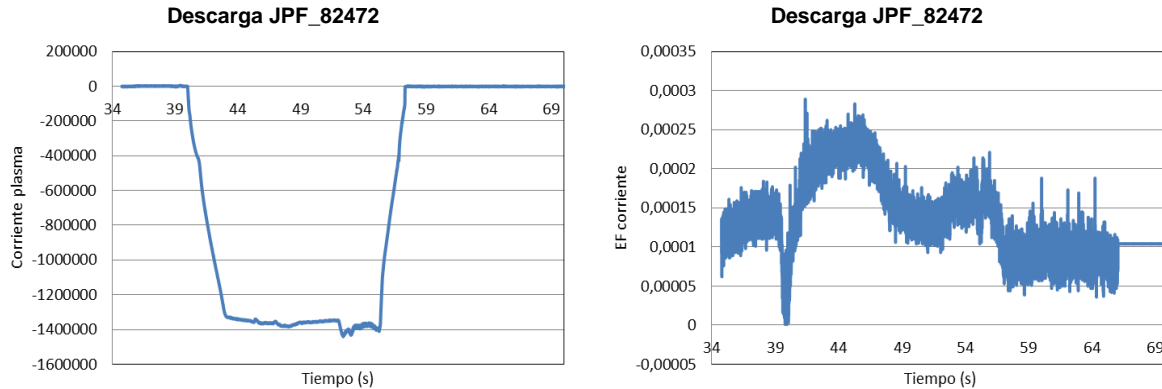


Figura 5.39. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

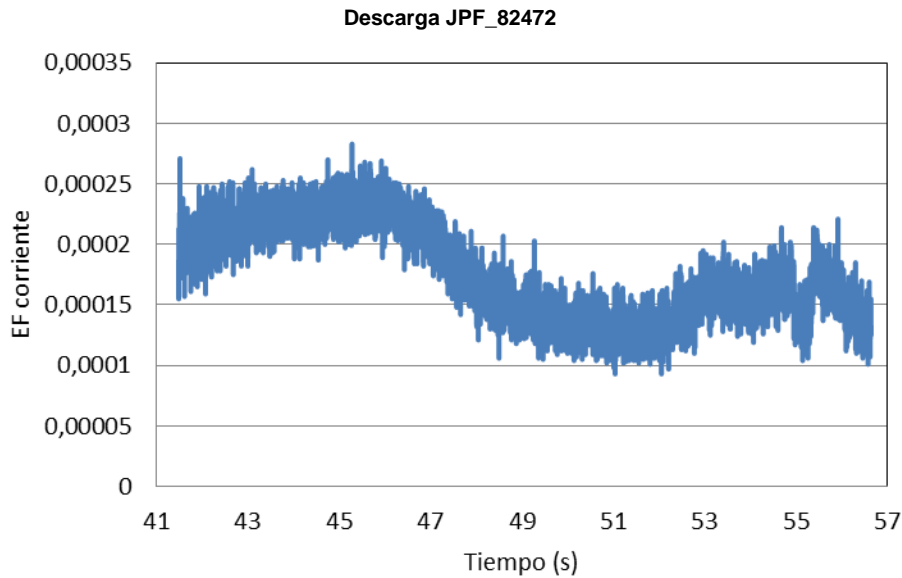


Figura 5.40. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

Descarga no disruptiva, no presenta inestabilidades en su evolución en el tiempo. Tampoco se observa la aparición outlier que pueden hacer disparar una

falsa alarma. El cluster de la representación de los coeficientes de aproximación muestra un grupo compacto con covarianza positiva y forma elíptica.

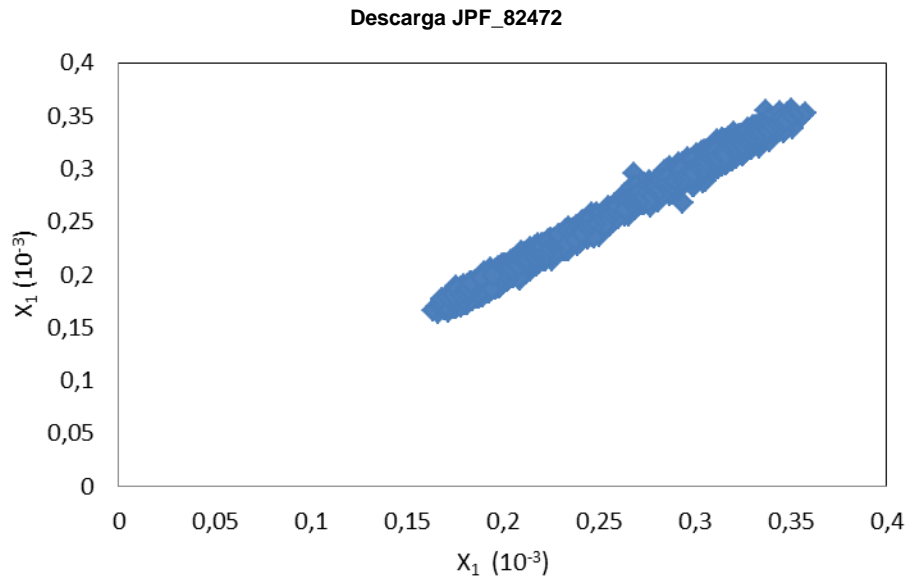


Figura 5.41. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

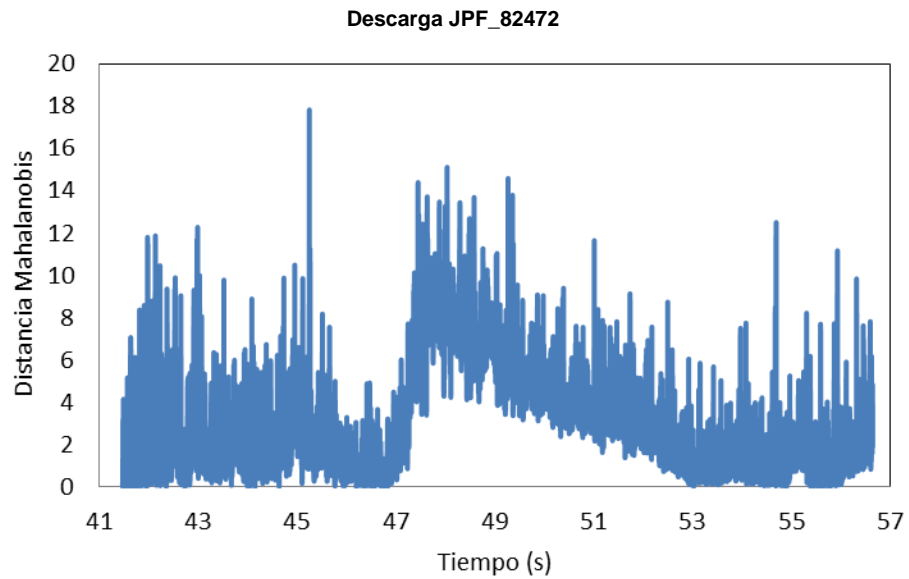


Figura 5.42. Evolución temporal de la distancia de Mahalanobis de la descarga

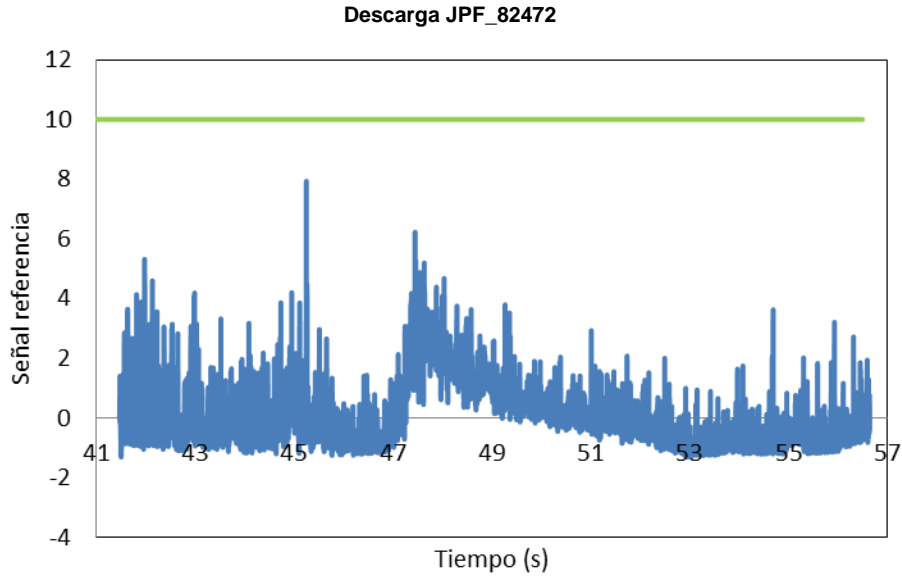


Figura 5.43. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

5.3.9. Descarga JPF_82474

Descarga 82474, tiene una duración de tiempo de 34,740376 segundos a 95,553329. Se ha obtenido una ventana de tiempo que va de 41,479935 a 63,714462 segundos.

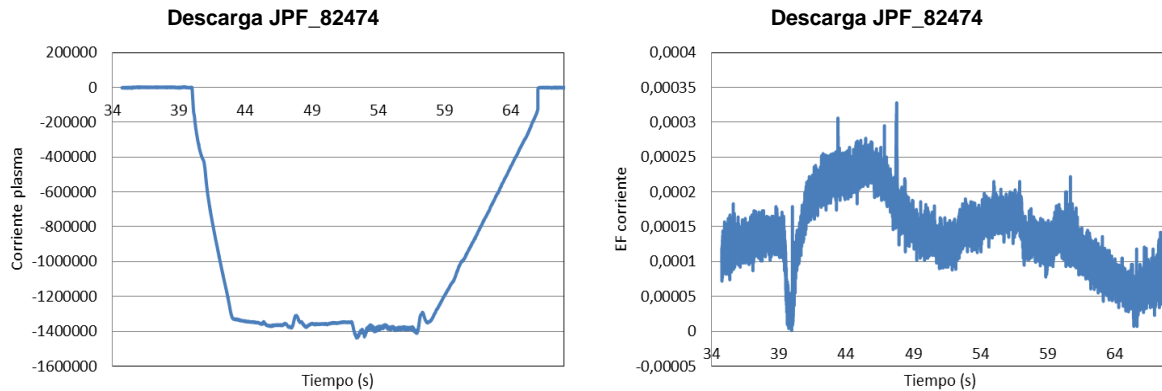


Figura 5.44. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

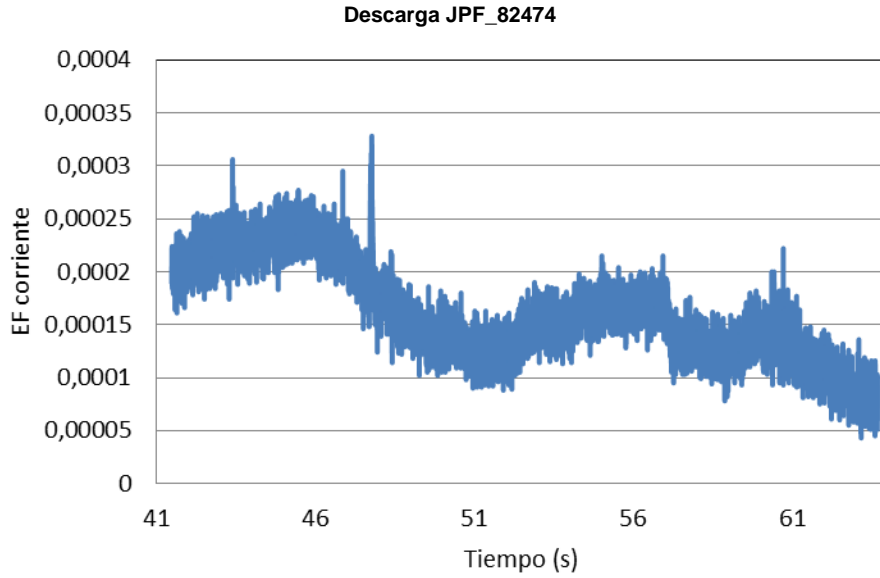


Figura 5.45. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

De la observación de la evolución temporal de la señal, claramente se deduce que en la cercanías del instante de tiempo $t = 46$ segundos, aparecen valores que harán disparar la alarma de disrupción.

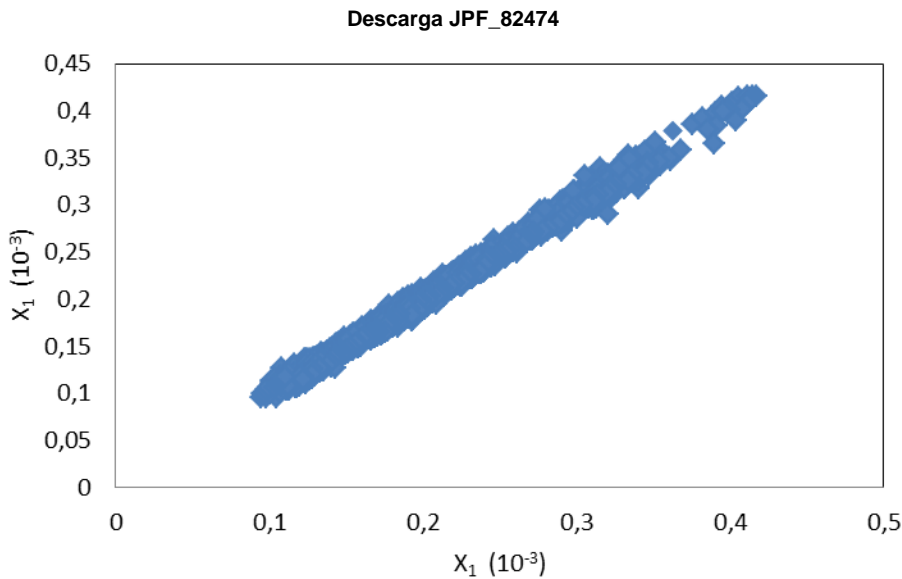


Figura 5.46. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

El diagrama de dispersión de la transformada de wavelet de la señal revela que existen varios valores separados del clúster principal. Estos puntos tienen una distancia algo superior a la media lo que hará que estén muy próximos al umbral de la señal de referencia.

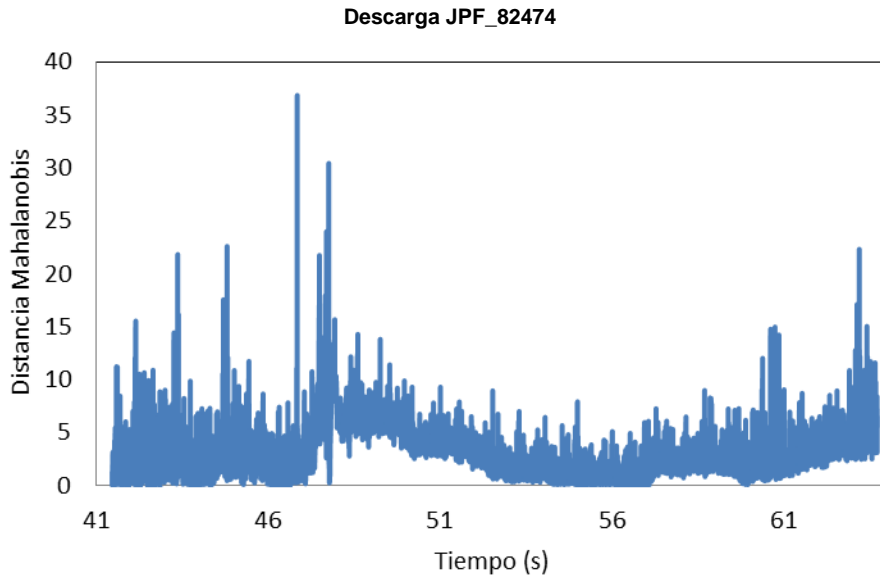


Figura 5.47. Evolución temporal de la distancia de Mahalanobis de la descarga

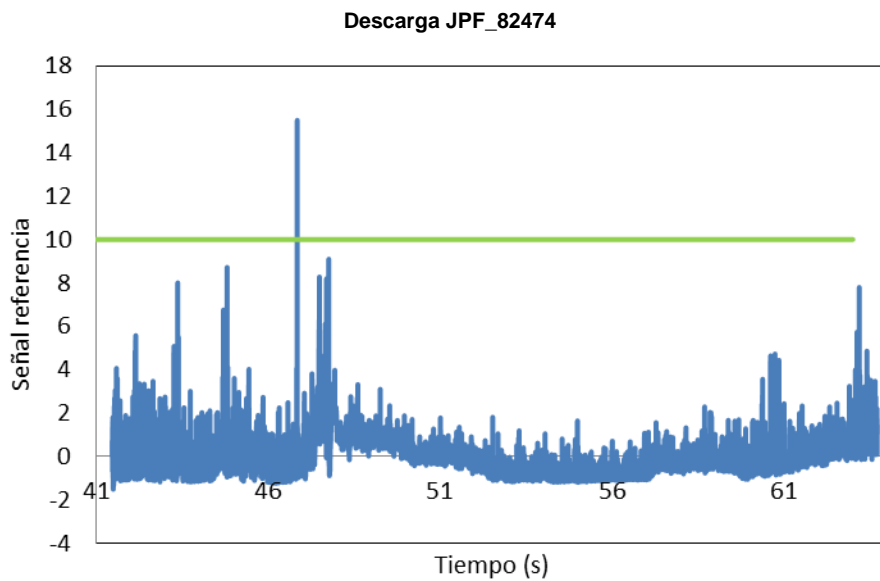


Figura 5.48. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

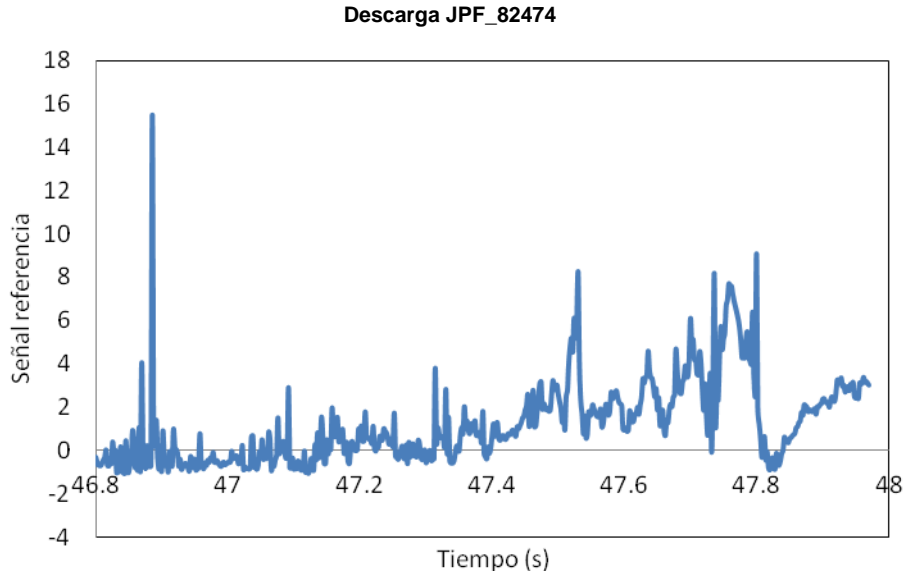


Figura 5.49. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y al momento de la aparición de inestabilidades

La alarma se dispara en el instante $t = 46,4$ seg, momento en el que la descarga se interrumpe, aunque la inestabilidades aparecen momentos después, estas, no son tan fuertes como la de los casos anteriores. Se podría decir entonces que estamos frente a un caso de falsa alarma, disparada por la aparición de un outlier de un par de ciclos de duración.

5.3.10. Descarga JPF_82481

Descarga 82481, tiene una duración de tiempo de 34,913448 segundos a 99,90815. Se ha obtenido una ventana de tiempo que va de 41.336 a 57,02732 segundos.

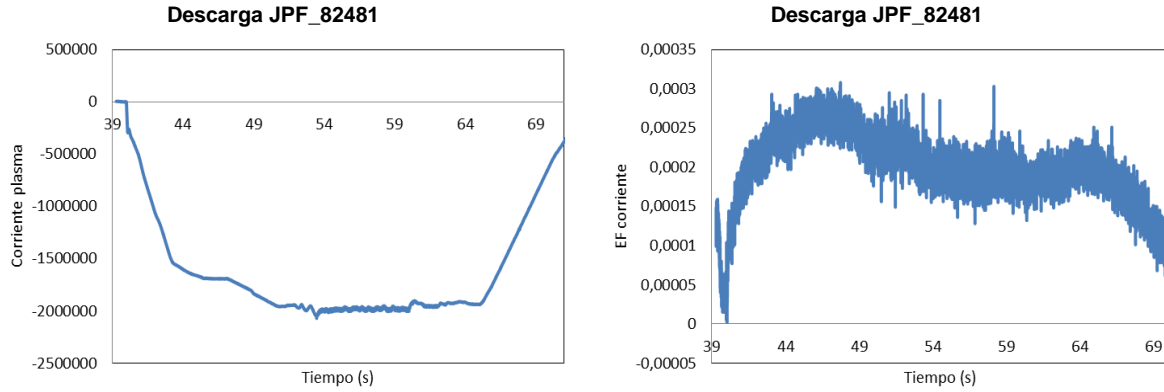


Figura 5.50. a) Evolución temporal corriente del plasma, b) Evolución temporal de la desviación estándar del espectro de Fourier de dicha corriente

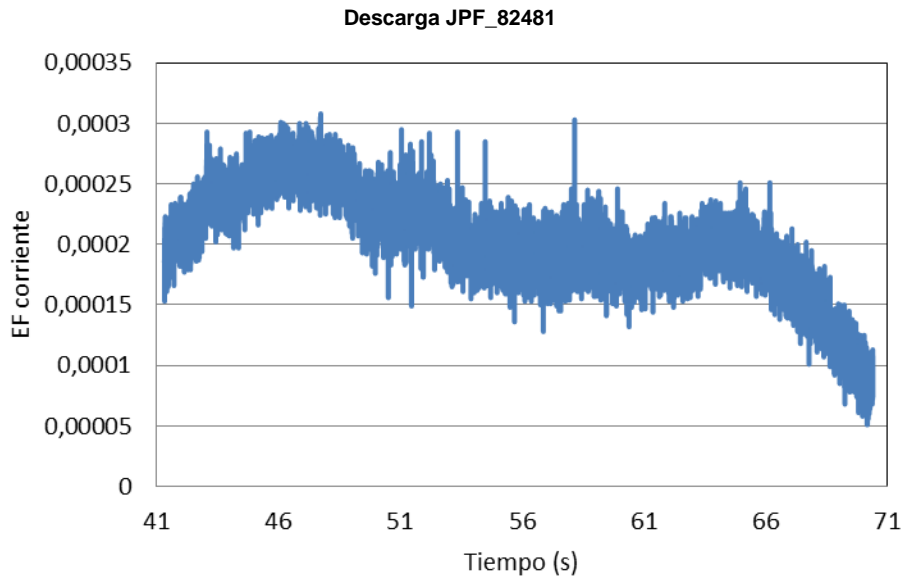


Figura 5.51. Evolución temporal de la desviación estándar del espectro de Fourier de la corriente comprendida entre el intervalo de tiempo de interés

En la evolución temporal se observa que la señal no presenta un carácter disruptivo. Otra cosa diferente es ver el comportamiento del sistema de predicción frente a la aparición de outlier en la señal.

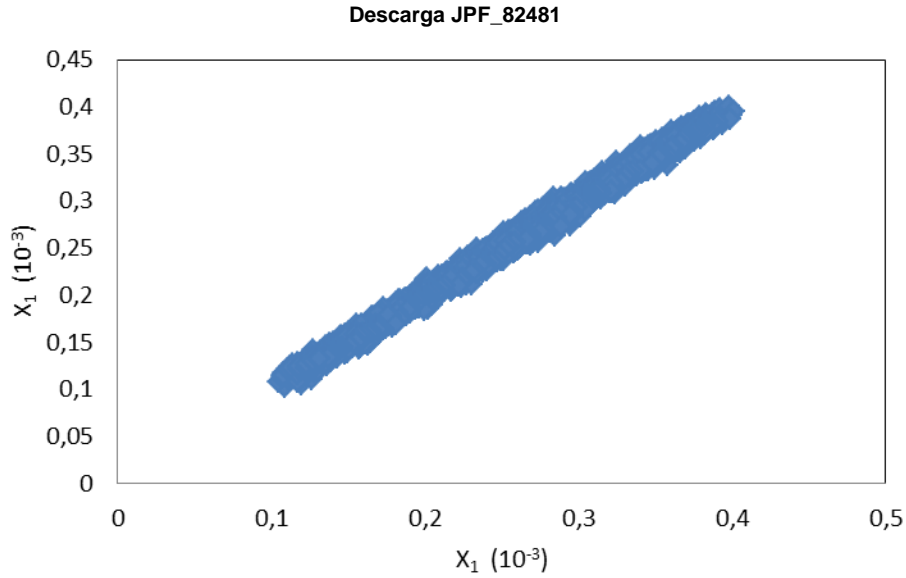


Figura 5.52. Diagrama de dispersión di-dimensional de la descarga aplicando la transformada de wavelet Haar

El cluster no denota la existencia de momentos disruptivos. Aparece como un conjunto de datos compacto con tendencia positiva y forma elíptica.

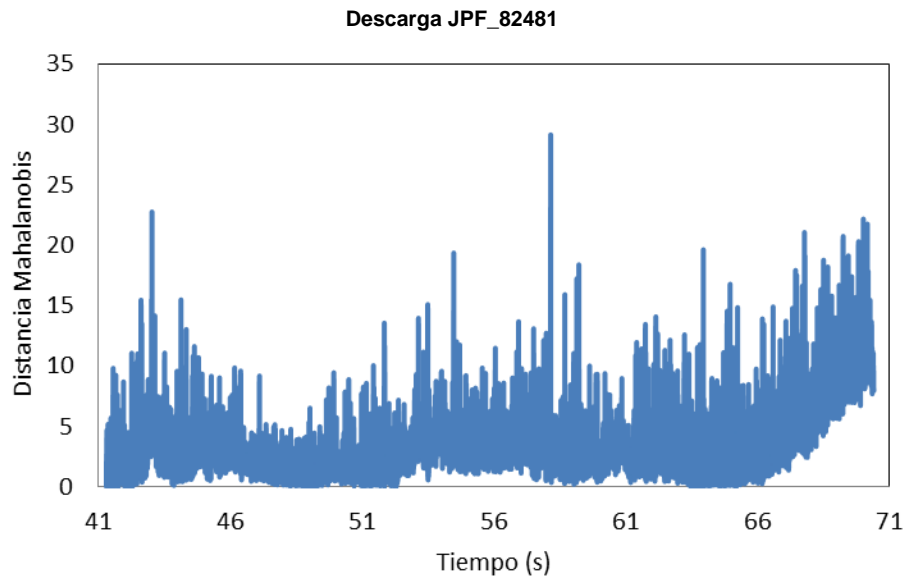


Figura 5.53. Evolución temporal de la distancia de Mahalanobis de la descarga

Una vez calculadas las distancias, se aprecia que en el instante $t = 58,1$ aparece un valor superior a la media de las distancias que coincide con el momento del mayor outlier.

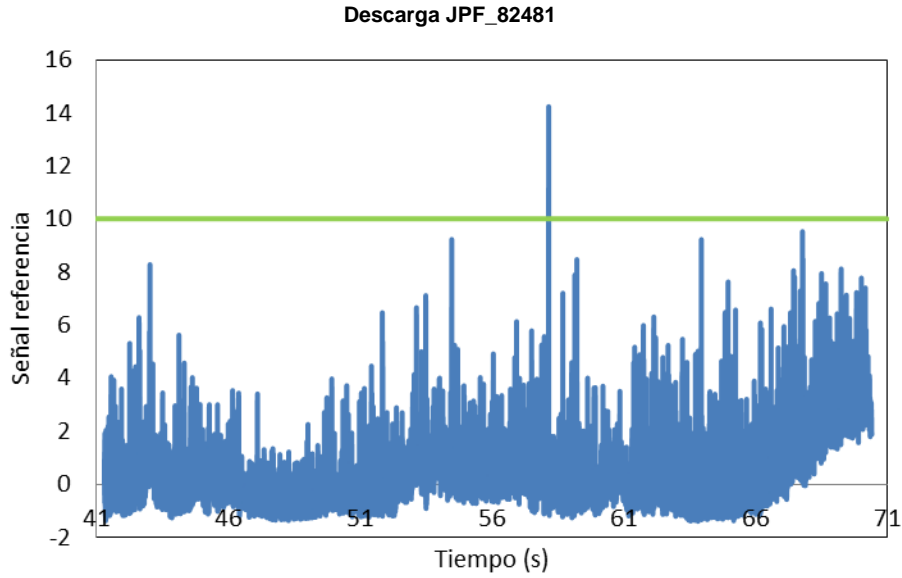


Figura 5.54. Evolución temporal de la señal de referencia y valor del umbral de $K_M = 10$

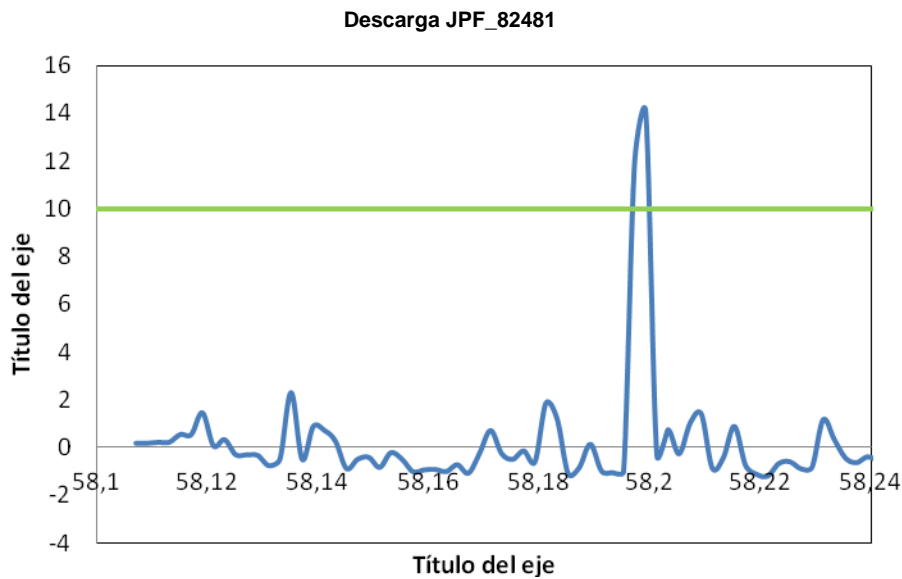


Figura 5.55. Evolución temporal de la señal de referencia aplicando zoom al momento de disparo de la alarma y valor del umbral de $K_M = 10$

Ayudándonos de un zoom, podemos confirmar que el valor outlier que hace saltar la alarma se encuentra en el instante $t = 58,16$ segundos. En este caso, en el que la descarga no es disruptiva y se ha disparado la alarma, podemos decir

que se trata de una falsa alarma y no una alarma prematura como ocurría en las descargas anteriores.

5.4. Comparación de resultados utilizando distintos parámetros

A continuación, y sin entrar en detalle, se muestran los resultados del comportamiento del sistema de predicción para distintos valores de sus parámetros. En concreto se han realizado simulaciones de comportamiento en dos descargas distintas, JPF_82460 descarga disruptiva y JPF_82462 descarga no disruptiva, para ventanas de 16 muestras con un desplazamiento de 2 muestras y un nivel de descomposición de 3, y para ventanas de 64 con un deslizamiento de 4 y nivel de descomposición de 4.

5.4.1. Descarga JPF 82460

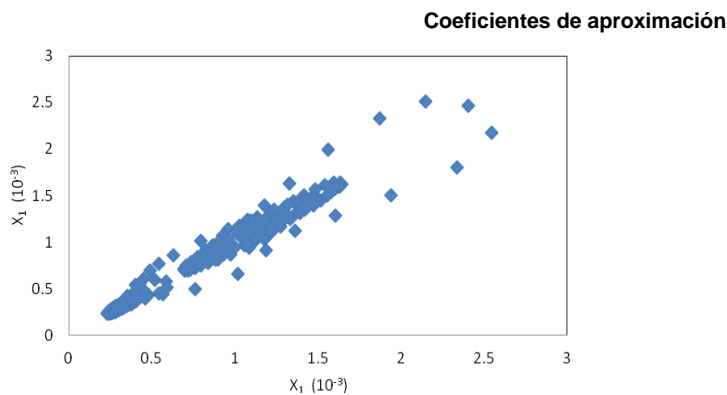


Figura 5.56. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar,

- con una ventana temporal de 16 muestras con dos muestras de desplazamiento y un nivel 3 de descomposición
- no tiene diagrama de dispersión bi-dimensional puesto que el número de coeficientes es 4.

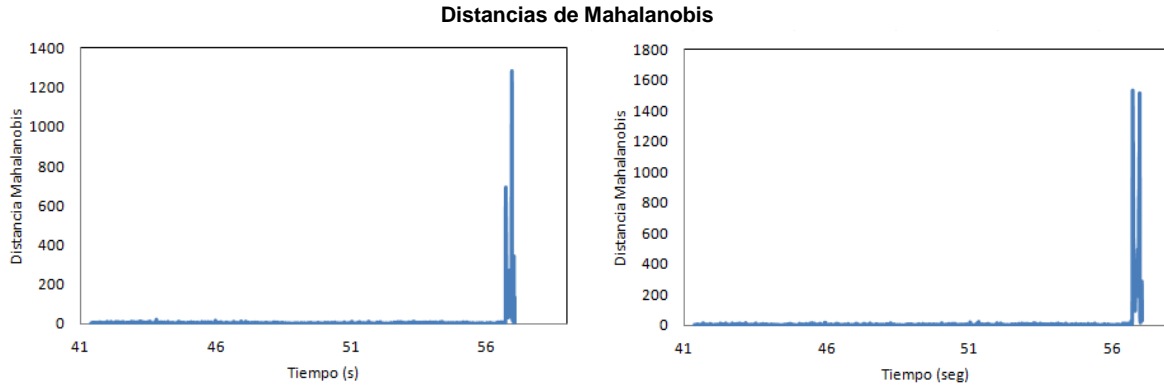


Figura 5.57. Evolución temporal de la distancia de Mahalanobis de la descarga
 a) *para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición*
 b) *para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición*

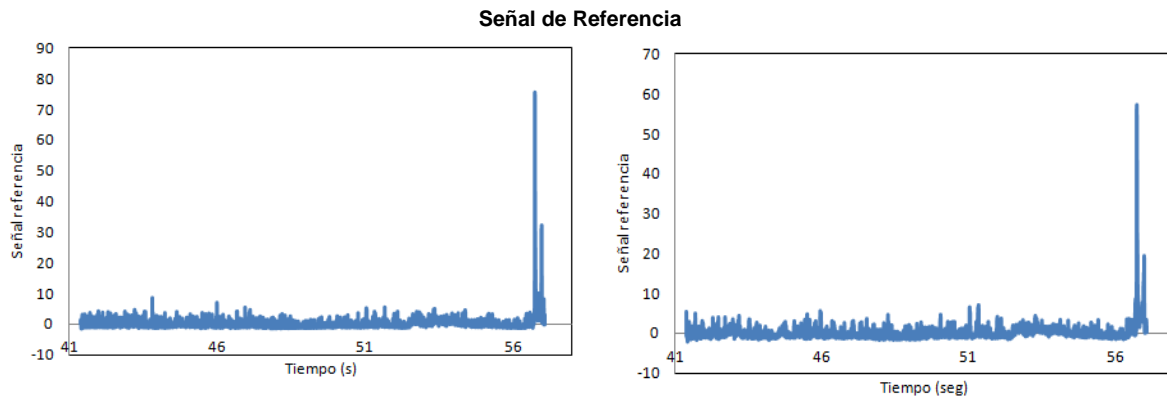


Figura 5.58. Evolución temporal de la señal de referencia
 c) *para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición*
 d) *para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición*

5.4.2. Descarga JPF_82462

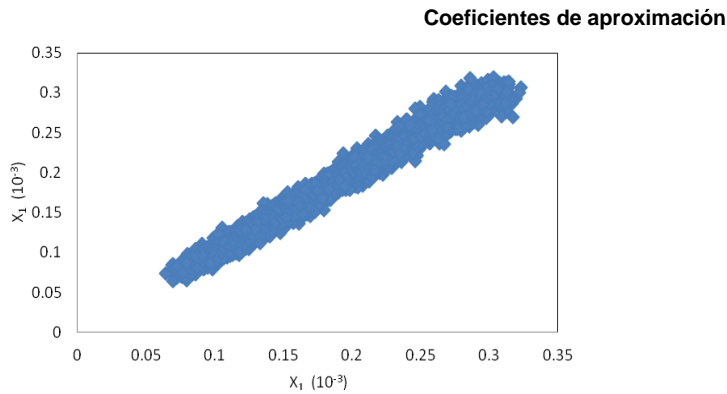


Figura 5.59. Diagrama de dispersión bi-dimensional de la descarga aplicando la transformada de wavelet Haar

- a) con una ventana temporal de 16 muestras con dos muestras de desplazamiento y un nivel 3 de descomposición
- b) no tiene diagrama de dispersión bi-dimensional puesto que el número de coeficientes es 4.

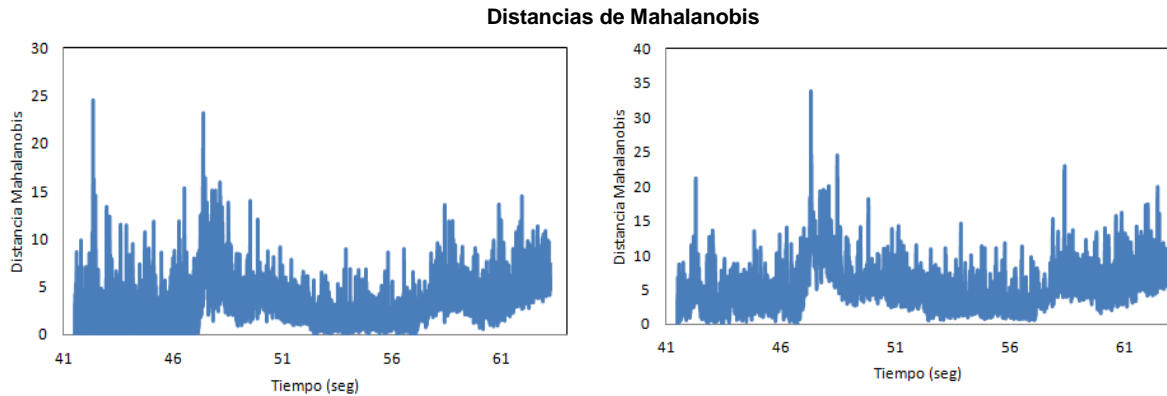


Figura 5.60. Evolución temporal de la distancia de Mahalanobis de la descarga

- a) para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición
- b) para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición

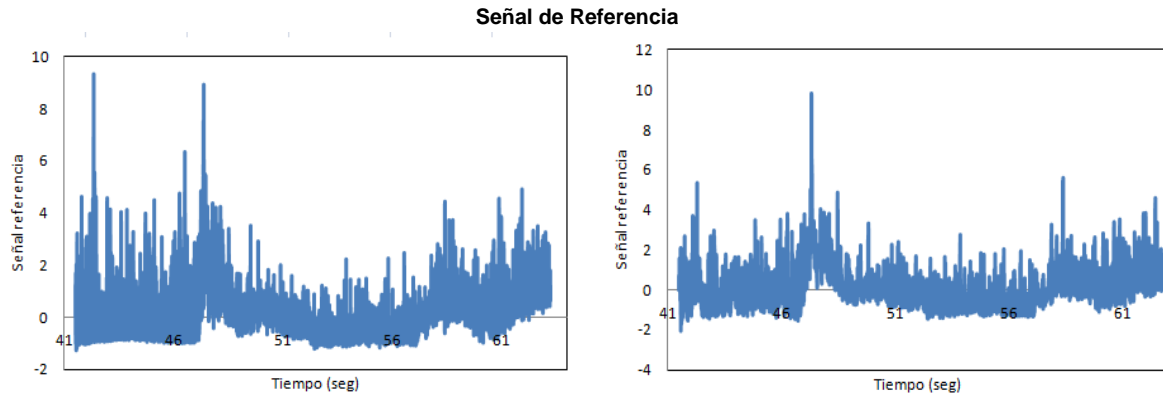


Figura 5.61. Evolución temporal de la señal de referencia
a) *para ventanas temporales de 16 muestras, con dos muestras de desplazamiento y un nivel 3 de descomposición*
b) *para ventanas temporales de 64 muestras, con 4 muestras de desplazamiento y un nivel 4 de descomposición*

En líneas generales, el comportamiento del predictor para distintos valores de sus parámetros es similar.

CAPÍTULO 6

CONCLUSIONES Y LÍNEAS DE TRABAJO FUTURAS

6.1. Conclusiones

En este proyecto se ha utilizado técnicas de clustering para la detección de un fenómeno físico asociado a la fusión nuclear como son las disrupciones. El sistema de predicción implementado se basa en la generación de una señal de referencia que hace que dispare una alarma cuando sobre pasa unos umbrales.

Para la fase de entrenamiento se han utilizado 10 descargas que han servido para evaluar su respuesta ante los cambios producidos en la señal original. De los resultados obtenidos podemos concluir:

- Se demuestra que el modelo propuesto es efectivo para la detección de cambios en flujos de datos procedentes de medidas de fenómenos físicos de fusión nuclear. Es altamente sensible a la aparición de outlier en la señal, esto tiene el inconveniente que el sistema es propenso a la activación de falsas alarmas durante el proceso de la descarga. Esto se solucionaría modelando la función de decisión.
- De las 10 descargas procesadas, suponiendo que K_M igual 10 (umbral de 10), resultaría que en 5 de ellas no se dispararía la alarma y en las otras 5

si. En las descargas que no se dispara no hay inestabilidades ni grandes valores anómalos. Tres de las descargas con inestabilidades la alarma se dispara prematuramente y en las otras dos aparece una falsa alarma. De la observación de estos casos, llegamos a la conclusión que cuando la alarma se dispara por un outlier en la señal, la función de referencia se mantiene por encima del umbral durante un instante de tiempo muy pequeño, apenas uno o dos milisegundos, por lo que uno de los puntos claves para detectar si se trata de una interrupción o un outlier será determinar si la señal de referencia permanece por encima del umbral durante un tiempo determinado que dependerá de la frecuencia de muestreo de la señal y del desplazamiento.

- Se comprobó la robustez del sistema con parámetros distintos. El comportamiento del sistema ante descargas disruptivas y no disruptivas es bueno. Se obtuvo que a medida que el tamaño de las ventanas y el deslizamiento sea mayor, la resolución del sistema baja, esto puede provocar que algunos de los outlier que aparecen en la señal no se detecten pero también provoca que el tiempo de reacción del sistema sea menor.
- Dejando a un lado si se trata de una falsa alarma o no, otra conclusión que se puede extraer es que el tiempo de respuesta del sistema es muy rápido. Por ejemplo:
 - En la descarga 82471 el primer valor que predice atípico aparece en el instante de tiempo $t = 44,478577$ segundos, y la detección se da a los 44,483 segundos.
 - En la descarga 82474 el primer valor que predice como atípico aparece en el instante $t = 46,862572$, y la detección se da a los 46,882 segundos.
 - En la descarga 82481 el primer valor que predice como atípico aparece en el instante $t = 58,16473$, y la detección se da a los 58,2 segundos.

A estos valores de milisegundos habrá que sumarles el tiempo computacional, que es fijo para cualquier clase de sistema de predicción.

- El modelo propuesto es de propósito general no necesitándose desarrollos específicos para cada aplicabilidad particular. No obstante es preciso realizar un preprocesado efectivo de los datos a analizar para elegir el valor de los parámetros y la función de decisión.

- Como ya se ha dicho, el valor de la función de decisión, K_M , es un valor experimental, que se debe de elegir tras la observación del análisis de cientos de descargas.

6.2. Líneas de trabajo futuras

Debido a la gran importancia y al valor añadido al proyecto, como primera línea de trabajo destacar la accesibilidad para la instalación del predictor programado en la red en tiempo real de JET.

El predictor utiliza métodos de clasificación que permiten la implantación del sistema en una red de tiempo real. Se ha visto que mediante el procesamiento secuencial de ventanas de tiempo es posible condensar la información relevante para distinguir si estas se encuentran cercanas en el tiempo a una interrupción. Con la implementación del sistema conseguimos la posibilidad de detectar la interrupciones en tiempo real.

Y en cuanto a líneas de trabajo futuro enfocadas al sistema de predicción de interrupción podemos reseñar las siguientes mejoras:

- Como ha quedado de manifiesto, es necesario el análisis de un mayor número de descargas cambiando el valor de los parámetros, para así recoger una mayor variabilidad en el tipo de variaciones a detectar por el modelo, lo cual mejorará aún más su fiabilidad. Ayudaría al modelado de la función de decisión para reducir la tasa de falsas alarmas o las alarmas prematuras.
- Detectar el tipo de interrupción: si al detectar un comportamiento disruptivo es posible, además, determinar el tipo de interrupción, se podrían obtener modelos de detección mucho más fiables.

CAPÍTULO 7

BIBLIOGRAFÍA

C.M. Braams y P.E. Stott. "Nuclear Fusion. Half a Century of Magnetic Confinement Fusion Research". *Plasma Phys. Control. Fusion* 44 1767. 2002

J. Sheffield. "The physics of magnetic fusion-reactors". *Rev. Mod. Phys.* 66, 1015. 1994,

J. R. Reitz, F. J. Milford y R. W. Christy, "Fundamentos de la Teoría Electromagnética". Addison-Wesley Iberoamericana, Wilmington. 1996

A. H. Boozer. "Theory of tokamak disruptions". *Physics of Plasmas* 19. 058101, 2012

Braams, C.M., and P.E. Stott. "Nuclear Fusion. Half a Century of Magnetic Confinement Fusion Research." *Plasma Physics and Controlled Fusion* 44, no. 8 (2002)

Cannas, B., A. Fanni, G. Sias, P. Sonato, M. K. Zedda, and JET-EFDA contributors. "Neural approaches to disruption prediction at JET." *31st EPS Conference on Plasma Phys.* London, 2004

Cannas, B., A. Fanni, P. Sonato, M. K. Zedda, and JET-EFDA contributors. "A prediction tool for real-time application in the disruption protection system at JET." *Nuclear Fusion* 47, no. 11 (2007)

J. Vega, R. Moreno, A. Pereira, S. dormido-Canto, A. Murari. Advanced Disruption Predictor Based on the Locked Mode signal: Application to JET

J. M. López, J. Vega, D. Alves, Member, IEEE, S. Dormido-Canto, A. Murari, J. M. Ramírez, R. Felton, M. Ruiz, and G. de Arcas. "Implementation of the Disruption Predictor APODIS in JET's Real-Time Network Using the MARTe Framework"

Giuseppe A. Rattá Gutiérrez. "Técnicas de Minería de datos aplicadas a fusión nuclear: Predicción en tiempo real y clasificación". PhD thesis, UNED - CIEMAT, 2010.

Jesús M. Ramírez Pérez. "Máquinas de vectores soporte en entornos de supercomputación: aplicada a fusión nuclear". PhD thesis UNED – CIEMAT, 2014

Eric J. Stollnitz Tony D. DeRose David H. Salesin. "Wavelets for Computer Graphics: A Primer Parte1"

Eduardo Cano, Rubén Salcedo y Gustavo Soto. "Análisis de principios y aplicaciones de la transformada de waveler"

<http://www.uv.es/ceaces/multivari/cluster/CLUSTER2.htm>. "Introducción al análisis cluster"

Carles M. Cuadras. "Distancias Estadísticas". Departament d'Estadística Universitat de Barcelona

Tolbox de Matlab. "Mahalanobis distance"

J Lejarza y I. Lejarza "Análisis de datos multidimensionales"

Anexo I. Código para el manejo de matrices

matrices.h

```
#ifndef _MATRICES_H_
#define _MATRICES_H_

/* FUNCIONES PROTOTIPOS */

void MatrizCovarianza (int N, int M, float matriz[][M], float
matrizCovarianza[][M]);
void MatrizInversa(int Dim, float Sist[][Dim], float Inv[][Dim]);
void multiplicarMatrices (int M, float matrizA[][M], float matrizB[][M], float
matrizC[][1],
                        float *resultado);
void desviacionTipica (elemento *lista, float, float *resultado);

#endif
```

matrices.c

```

#include <stdio.h>
#include <stdlib.h>
#include "lista.h"

/*+++++
CALCULA LA MATRIZ DE COVARIANZA
Parámetros de entrada:   N y M enteros, número de filas y de columnas
                        matriz
Parámetros de salida:   matrizCovarianza
+++++*/

void MatrizCovarianza (int N, int M, float matriz[][M], float matrizCovarianza[][M])
{
    float matrizDesviacion [N][M];
    int i, j, m, k;
    int sum = 0;
    float promedio[M];
    float varianza[M];

    for (i=0;i<M;i++)
        sum=sum+i;

    float matrizProducto[N+1][sum];

    for (i=0;i<M;i++)
    {
        promedio[i]=0;
        varianza[i]=0;

        for (j=0;j<N;j++)           // calcula el promedio de cada variable(columna)
            promedio[i]=promedio[i]+matriz[j][i];
        promedio[i]=promedio[i]/N;

        for (j=0;j<N;j++)
        {
            matrizDesviacion[j][i]=matriz[j][i]-promedio[i];           // matriz de desviación
            varianza[i]=varianza[i]+pow((matriz[j][i]-promedio[i]),2);     // calcula la
                                                                           varianza de cada variable
        }
    }
}

```

```

    }
    varianza[i]=varianza[i]/(N-1);
}

for (i=0;i<N;i++)
{
    m = 0;
    for (j=0;j<M;j++)
    {
        for (k=j+1;k<M;k++)
        {
            matrizProducto[i][m]=(matrizDesviacion[i][j])*(matrizDesviacion[i][k]);
            m++;
        }
    }
}

for (i=0; i<sum; i++)
{
    matrizProducto[N][i]=0;
    for (j=0; j<N; j++)
    {
        matrizProducto[N][i]=matrizProducto[N][i]+matrizProducto[j][i];
    }
    matrizProducto[N][i]=matrizProducto[N][i]/(N-1);
}

k=0;
for (i=0;i<M;i++) // matriz de covarianza
{
    for (j=i;j<M;j++)
    {
        if(i==j)
            matrizCovarianza[i][j]=varianza[i];
        else
        {
            matrizCovarianza[i][j]=matrizProducto[N][k];
            matrizCovarianza[j][i]=matrizProducto[N][k];
            k++;
        }
    }
}
}
}

```

```

/*+++++
CALCULA MATRIZ INVERSA DEL SIST
Parámetros de entrada:  Dim es el tamaño de la matriz cuadrada
                       Sist matriz
Parámetros de salida:  Inv matriz inversa
+++++*/

void MatrizInversa(int Dim, float Sist[][Dim], float Inv[][Dim])
{
  int NoCero,Col,C1,C2,A;
  float Pivote,V1,V2;

  /*Se inicializa la matriz inversa, como la matriz identidad:*/
  for(C1=0;C1<Dim;C1++)
    for(C2=0;C2<Dim;C2++)
      if (C1==C2)
        Inv[C1][C2]=1;
      else
        Inv[C1][C2]=0;

  for(Col=0;Col<Dim;Col++)
  {
    NoCero=0;
    A=Col;
    while(NoCero==0)
    {
      if(((Sist[A][Col]>0.0000001) || ((Sist[A][Col]<-0.0000001)))
      {
        NoCero=1;
      }
      else
        A++;
    }
    Pivote=Sist[A][Col];
    for(C1=0;C1<Dim;C1++)
    {
      V1=Sist[A][C1];
      Sist[A][C1]=Sist[Col][C1];
      Sist[Col][C1]=V1/Pivote;
      V2=Inv[A][C1];
      Inv[A][C1]=Inv[Col][C1];
      Inv[Col][C1]=V2/Pivote;
    }
    for(C2=Col+1;C2<Dim;C2++)
    {
      V1=Sist[C2][Col];

```

```
for(C1=0;C1<Dim;C1++)
{
  Sist[C2][C1]=Sist[C2][C1]-V1*Sist[Col][C1];
  Inv[C2][C1]=Inv[C2][C1]-V1*Inv[Col][C1];
}
}

/*Aqui ya esta triangularizada, con 1s en diagonal, ahora se diagonaliza*/
for(Col=Dim-1;Col>=0;Col--)
for(C1=(Col-1);C1>=0;C1--)
{
  V1=Sist[C1][Col];
  for(C2=0;C2<Dim;C2++)
  {
    Sist[C1][C2]=Sist[C1][C2]-V1*Sist[Col][C2];
    Inv[C1][C2]=Inv[C1][C2]-V1*Inv[Col][C2];
  }
}
}
```

```

/*+++++
MULTIPLICA LAS MATRICES A, B Y C.
Parámetros de entrada:  matrizA, matrizB, matrizC
                        M números de columnas
Parámetros de salida:  resultado, resultado de la multiplicación
+++++*/

```

```

void multiplicarMatrices (int M, float matrizA[][M], float matrizB[][M], float
matrizC[][1], float *resultado)
{
    //float matrizA[1][M]           //(1 x m)
    //float matrizB[M][M]           //(m x m)
    //float matrizC[M][1]           //(m x 1)
    float matrizResultado[1][M];    //(1 x m)
    *resultado=0;
    int j, k;

    for(j=0;j<M;j++)
    {
        matrizResultado[1][j]=0;
        for(k=0;k<M;k++)
        {
            matrizResultado[1][j]=matrizResultado[1][j]+(matrizA[1][k]*matrizB[k][j]);
        }
    }

    for(k=0;k<M;k++)
    {
        (*resultado) = (*resultado)+(matrizResultado[1][k]*matrizC[k][1]);
    }
}

```

```
/*+++++  
CALCULA LA DESVIACION TIPICA  
Parámetros de entrada:  lista  
                        media  
Parámetros de salida:  resultado  
+++++*/
```

```
void desviacionTipica (elemento *lista, float media, float *resultado)  
{  
    elemento *actual = lista;  
    int i=0;  
    *resultado=0;  
    while(actual!=NULL)  
    {  
        *resultado = *resultado + pow(actual->dato - media,2);  
        actual=actual->siguiente;  
        i++;  
    }  
    *resultado = pow(*resultado/(i-1),0.5);  
}
```


Anexo II. Código para el manejo de memoria dinámica

listas.h

```
#ifndef _LISTA_H_
#define _LISTA_H_

/* LISTA SIMPLEMENTE ENLAZADA. CADA ELEMENTO CONTIENE UN Nª FLOAT */

typedef struct datos elemento;
struct datos
{
    float dato;
    elemento *siguiente;
};

/* FUNCIONES PROTOTIPOS */

elemento *NuevoElemento(void);
void anadir(elemento **, float );
void visualizar(elemento *);
void borrarlista(elemento *);

#endif
```

listas.c

```
#include <stdio.h>
#include <stdlib.h>
#include "lista.h"

/* crea un elemento nuevo
devuelve el elemento creado */

elemento *NuevoElemento(void)
{
    elemento *q=(elemento*)malloc(sizeof(elemento));
    return(q);
}

/* añade un elemento nuevo en la lista creada
añade el elemento al principio de la lista (o final)
recibe el coeficiente calculado como parametro de entrada
y utiliza y devuelve la lista dinámica con el nodo introducido */

void anadir(elemento **lista, float valor)
{
    elemento *q;
    elemento *actual = *lista;

    q = NuevoElemento();
    q->dato = valor;
    q->siguiente = actual;
    actual = q;

    *lista = actual;
}

/* muestra por pantalla los coeficientes calculador
```

en orden de la lista */

```
void visualizar(elemento *lista)
{
    elemento *actual = lista;
    while(actual!=NULL)
    {
        printf("%f", actual->dato);
        actual=actual->siguiente;
        printf("\n");
    }
}
```

/* borra los elemento de la lista */

```
void borrarlista(elemento *lista)
{
    elemento *actual = lista;
    while(actual!=NULL)
    {
        lista=lista->siguiente;
        free(actual);
        actual=lista;
    }
}
```


Anexo III. Código del programa principal

principal.c

```
#include <stdio.h>
#include <stdlib.h>
#include "lista.h"
#include "matrices.h"

// CONSTANTES

#define MAX_VENTANA1 -750000
#define MAX_VENTANA2 -500000

#define MAX_MUESTRAS 32
#define DESPLAZAMIENTO 2
#define NIVEL 4

#define FACTOR 1.41421356 // factor raiz de 2
#define UMBRAL 10

//PROTOTIPOS

void calculaCoeficientesDeAproximacion (int a, float x[], float *y);
float CalculaDistanciaMahalanobis(int b, int a, float x[], elemento *);
float senalReferencia (elemento *lista, float distancia, float media);

/* FUNCION PRINCIPAL */

int main(void)
{
    FILE *f_01; // ficheros de trabajo
    FILE *f_02;

    FILE *fdistancias; // ficheros de resultados
    FILE *fcoeficientes;
    FILE *fsenalReferencia;

    elemento *listaCoeficientes = NULL;
    elemento *listaDistancias = NULL;

    float valores; // valores leidos del fichero */
    float amplitud[MAX_MUESTRAS]; /* array de los valores del tamaño de la
                                    ventana (potencia de 2) */
```

```

float media = 0;
float aux = 0;

float tiempo1, tiempo2;
int ts=0;
int salir = 0;
int contador=1;          /* cuenta los valores de amplitud leidos. Determinar
                          1a ventana. Determinar desplazamiento */
int contador_fichero=0;  /* cuenta valores leidos del fichero. Determinar si
                          es amplitud o tiempo */
int contador_desplazamiento=0; /* cuenta los desplazamientos. Determinar
                                cuando hay que calcular los coeficientes de aproximación */
int contador_coeficientes=0; /* cuenta conjunto de coeficientes calculados.
                                Uno por ventana */

int contadorDistancias=1;
int i, j;                /* contadores de los for */
int numeroCoeficientes; /* numero de coeficientes calculados. Dependerá
                          del nivel */
numeroCoeficientes = MAX_MUESTRAS/pow(2,NIVEL);
float coeAprox[numeroCoeficientes]; /* contiene el valor de los coeficientes
                                      calculados */

float distanciaMahalanobis=0;
float valorSenal;
//int alarmaON = 0;

// abrimos ficheros de trabajo
f_01 = fopen("c:/Users/deivid/Desktop/PFCMaster/descargas/
             JPF_82462_01.txt", "r");
if (f_01==NULL)
{
    printf ("Error al abrir fichero.txt");
    system("PAUSE");
    return -1;
}

f_02 = fopen("c:/Users/deivid/Desktop/PFC Master/descargas/
             JPF_82462_02.txt", "r");
if (f_02==NULL)
{
    printf ("Error al abrir fichero.txt");
    system("PAUSE");
    return -1;
}

fdistancias = fopen("c:/Users/deivid/Desktop/PFC Master/descargas/
                    distanciasMahalanobis.txt", "w");

```

```
if (fdistancias==NULL)
{
    printf ("Error al abrir fichero.txt");
    system("PAUSE");
    return -1;
}

fcoeficientes = fopen("c:/Users/deivid/Desktop/PFC Master/descargas/
                    coeficientesAproximacion.txt", "w");
if (fdistancias==NULL)
{
    printf ("Error al abrir fichero.txt");
    system("PAUSE");
    return -1;
}

fsenalReferencia = fopen("c:/Users/deivid/Desktop/PFC Master/descargas/
                        senalReferencia.txt", "w");
if (fdistancias==NULL)
{
    printf ("Error al abrir fichero.txt");
    system("PAUSE");
    return -1;
}

// DEFINICION DE LA VENTANA DE DATOS
fscanf(f_01, "%f", &valores);
contador_fichero++;

while (!(feof(f_01))&&!salir))
{
    if (ts==1)           // se activa el primer time para la ventana
    {
        tiempo1 = valores; // tiempo inicio (ms) de la ventana
        ts = 2;
    }
    if (ts==3)           // se activa el segundo time para la ventana
    {
        tiempo2 = valores; // tiempo final (ms) de la ventana
        salir = 1;
    }
    if (contador_fichero%2==0)
    {
        if ((valores<MAX_VENTANA1)&&(ts==0))
            ts = 1;
        if ((valores>MAX_VENTANA2)&&(ts==2))
            ts = 3;
    }
}
```

```

    }
    fscanf(f_01, "%f", &valores);
    contador_fichero++;
}
fclose (f_01);

printf("%f %f\n", tiempo1, tiempo2);    // tiempo inicial y tiempo final
                                        // de la ventana de datos

contador_fichero=0;
fscanf(f_02, "%f", &valores);    // lee primer valor del fichero
contador_fichero++;

while (!feof(f_02))    // mientras no se llegue al final del fichero
{
    if ((contador_fichero%2==1)&&
        (valores>=tiempo1)&&
        (valores<=tiempo2))    // lecturas impares, rengu de tiempo calculado
    {
        fscanf(f_02, "%f", &valores);    // lee primer valor del fichero
        contador_fichero++;
        if (contador_fichero%2==0)    // Lecturas pares, valores de
                                        // amplitud
        {
            if (contador<=MAX_MUESTRAS)
                amplitud[contador-1]=valores;
            else
            {
                for (i=1; i<MAX_MUESTRAS; i++) // desplazamos el vector
                    amplitud[i-1]=amplitud[i];
                amplitud[MAX_MUESTRAS-1]=valores; // se guarda el último valor leído
                contador_desplazamiento++;
            }
        }

        if (((contador>=MAX_MUESTRAS)&&
            (contador_desplazamiento%DESPLAZAMIENTO==0)) ||
            (contador==MAX_MUESTRAS))
        {
            contador_coeficientes++;
            calculaCoeficientesDeAproximacion (numeroCoeficientes, amplitud,
                                                coeAprox);
            for (i=0; i<numeroCoeficientes; i++)
            {
                anadir(&listaCoeficientes, coeAprox[i]);
                fprintf(fcoeficientes, "%f ", coeAprox[i]);
            }
            fprintf(fcoeficientes, "\n");
        }
    }
}

```

```

        if (contador_coeficientes>=numeroCoeficientes) // no se calcula
mahalanobis hasta que el nº de distancias >= nº coeficientes
        {
distanciaMahalanobis=CalculaDistanciaMahalanobis(contador_coeficientes,
                                                    numeroCoeficientes,
                                                    coeAprox,
                                                    listaCoeficientes);

        fprintf(fdistancias, "%f",distanciaMahalanobis);
        fprintf(fdistancias, "\n");
        anadir(&listaDistancias,distanciaMahalanobis);

        aux = aux+distanciaMahalanobis;
        media = aux/contadorDistancias;
        contadorDistancias++;
        valorSenal = senalReferencia (listaDistancias,
                                      distanciaMahalanobis,
                                      media); //detectaOutlier
        fprintf(fsenalReferencia, "%f",valorSenal);
        fprintf(fsenalReferencia, "\n");

        /*if (valorSenal>UMBRAL)
        {
            alarmaON = 1;    // activar la alarma deshabilitado.
                            // Fase entrenaiento nos conviene que llegue al
                            // final de las descargas
        }*/
    }
}/*4o if*/

    contador++;
}/*2o if*/
}/*1er if*/
fscanf(f_02, "%f", &valores);
contador_fichero++;
}/*while*/
fclose(fdistancias);    // se cierran los ficheros
fclose(f_02);
fclose(fcoeficientes);
borrarlista(listaCoeficientes); // se borra la memoria
borrarlista(listaDistancias);

system("PAUSE");
return(0);

}/*fin main*/

```

