



**DETECCIÓN DEL EFECTO DE BOMBEO EN UN COMPRESOR CENTRÍFUGO  
DE UNA UNIDAD DE CRACKING CATALÍTICO.**

**OSCAR JAVIER ESCOBAR MORA**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERIAS FISICOMECANICAS  
ESCUELA DE INGENIERIAS ELECTRICA, ELECTRONICA Y DE  
TELECOMUNICACIONES  
BUCARAMANGA**

**2008**



**DETECCIÓN DEL EFECTO DE BOMBEO EN UN COMPRESOR CENTRÍFUGO  
DE UNA UNIDAD DE CRACKING CATALÍTICO.**

**OSCAR JAVIER ESCOBAR MORA**

Trabajo de Investigación presentado como requisito parcial para optar el título de  
Ingeniero Electrónico

**DIRECTOR:**

**JULIO AUGUSTO GELVEZ FIGUEREDO, MPE**

**CODIRECTOR:**

**CARLOS FERNANDO AGUDELO GOMEZ, Dr. (C)**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER**

**FACULTAD DE INGENIERIAS FISICOMECHANICAS**

**ESCUELA DE INGENIERIAS ELECTRICA, ELECTRONICA Y DE  
TELECOMUNICACIONES**

**BUCARAMANGA**

**2008**

## **DEDICATORIA**

*• A Dios, por ser mi guía y acompañarme en cada etapa de mi vida; por confiarme talentos y brindarme la oportunidad de ponerlos al servicio de los demás.*

*• A mis padres, por su ejemplo, confianza y apoyo incondicional. Es de ellos este logro, pues son la base sobre lo cual se cimienta lo que soy.*

*• A mi novia, por su amor sincero, fuente de inspiración y aliento para seguir adelante.*

*• A mis familiares y amigos, por compartir conmigo logros y fracasos con el mismo beneplácito.*

***Oscar Javier Escobar Mora***

## AGRADECIMIENTOS

Especial agradecimiento a la Universidad Industrial de Santander, a la escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones, por ser miembros activos de mi proceso de formación y facilitar los espacios y las instancias para el desarrollo de este trabajo de grado. A mi director, MPE. Julio Augusto Gélvez Figueredo, de igual manera que a mi codirector, Dr. (c) Carlos Fernando Agudelo Gómez, por permitir que su conocimiento y experiencia me orientaran hacia la consecución de éste logro. Al instituto Colombiano del Petróleo, ICP, por su apoyo y colaboración para el planteamiento y posterior desarrollo de este trabajo de investigación.

El autor.

## CONTENIDO

Dedicatoria	v
Agradecimientos	vi
Lista de tablas	viii
Lista de Figuras	ix
Lista de Anexos	x
Resumen	xi
Abstract	xii
Introducción	1
1. El compresor centrífugo	3
1.1. Breve recuento histórico	4
1.2. Partes del compresor centrífugo	5
1.3. Generalidades	9
1.4. El efecto Bombeo ( <i>Surge</i> )	11
2. Contexto real de la planta	15
2.1. Especificaciones del compresor	15
2.2. Sistema de automatización de la planta	16
2.3. Sensores	19
2.4. Históricos de la máquina	21
3. Métodos de detección del efecto bombeo	23
3.1. Sistemas preventivos	24
3.2. Sistemas que responden a las características incipientes	27
3.2.1. Método de cálculo de la función de auto correlación	28
3.2.2. Método del cálculo de la dispersión de la señal	29
4. Resultados	31
4.1. Desarrollo del software	31
4.1.1. Algoritmo de cálculo de la función de autocorrelación	31
4.1.2. Algoritmo de cálculo del nivel de dispersión	34
4.1.3. Interfaz gráfica	36
4.2. Pruebas realizadas y análisis de resultados	42
4.2.1. Método basado en el cálculo de la función de autocorrelación	43
4.2.2. Método basado en el cálculo del nivel de dispersión	52
5. Conclusiones	58
5.1. Conclusiones	58
5.2. Sugerencias	60
5.3. Evaluación de objetivos	61
Bibliografía	64
Anexos	65

## LISTA DE TABLAS

Tabla I. Valores en estado estable de los datos y la F.A.C.	48
Tabla II. Resultados estimados para el parámetro de detección.	50
Tabla III. Resultados de detección del fenómeno del bombeo mediante el método de la función de autocorrelación.	51
Tabla IV. Valores máximos de la Función de Dispersión en estado estable, para los sensores ubicados en el compresor.	54
Tabla V. Valores de umbral obtenidos durante el análisis en estado estable.	55
Tabla VI. Resultados de la detección del fenómeno del bombeo mediante el método del cálculo de la dispersión.	56
Tabla VII. Cuadro comparativo de los dos métodos.	57

## LISTA DE FIGURAS

Figura 1. Compresor centrífugo.	3
Figura 2. Cubierta del compresor.	5
Figura 3. Diafragma.	6
Figura 4. Partes del rotor.	7
Figura 5. Sellos.	8
Figura 6. Vista del eje de un compresor.	9
Figura 7. Curvas características del compresor centrífugo.	12
Figura 8. Ciclo característico del efecto de bombeo.	13
Figura 9. Sistema de automatización ADVANT OCS.	18
Figura 10. Aspecto físico de la MasterPiece 200/1.	19
Figura 11. Distribución de los sensores de vibración y desplazamiento en la máquina.	20
Figura 12. Métodos Anti efecto bombeo más utilizados.	24
Figura 13. Esquema típico anti efecto bombeo.	25
Figura 14. Límite de bombeo bajo diversos sistemas de referencia.	26
Figura 15. Resultados del algoritmo para dos variables diferentes.	30
Figura 16. Diagrama de flujo de la función de autocorrelación.	32
Figura 17. Diagrama de flujo del método de la función de autocorrelación.	33
Figura 18. Diagrama de flujo de la función media.	34
Figura 19. Diagrama de flujo de la función varianza.	35
Figura 20. Diagrama de flujo del algoritmo del cálculo del nivel de dispersión.	36
Figura 21. Pasos para cargar los datos históricos al programa.	37
Figura 22. Entorno de procesamiento de datos.	38
Figura 23. Resultado típico del entorno de procesamiento.	39
Figura 24. Resultados gráficos de aplicar el algoritmo basado en el cálculo de la autocorrelación, a los datos históricos del día del fallo.	44
Figura 25. Resultados gráficos de aplicar el algoritmo basado en el cálculo de la autocorrelación, a los datos históricos del día del fallo.	44
Figura 26. Efecto de la variación del número de secciones sobre la función de autocorrelación.	46
Figura 27. Ejemplo del proceso de cálculo del rango de variación de la función de autocorrelación y de los datos, en estado cuasi-estable.	47
Figura 28. Resultado obtenido al procesar el sensor FC-ZI-5003 con el algoritmo de la función de autocorrelación.	50
Figura 29. Cálculo de los valores extremos de la función de dispersión en estado cuasi-estable.	53
Figura 30. Resultado de procesar el sensor FC-VI 5001 con el algoritmo de la dispersión.	54

## LISTA DE ANEXOS

Anexo A. Código de programación de la función de autocorrelación	65
Anexo B. Código de programación del algoritmo de la función media	65
Anexo C. Código de programación del algoritmo de la función media	65
Anexo D. Código de programación del algoritmo de la función de Autocorrelación	66
Anexo E. Código de programación del algoritmo del método del cálculo de la dispersión	66
Anexo F. Código de programación de la interfaz para la adquisición de los datos históricos	66
Anexo G. Código de programación de la interfaz para el procesamiento de los datos históricos	68

## RESUMEN

**TITULO:** DETECCIÓN DEL EFECTO DE BOMBEO EN UN COMPRESOR CENTRÍFUGO DE UNA UNIDAD DE CRACKING CATALÍTICO<sup>1</sup>.

**AUTOR:** ESCOBAR, Oscar J.<sup>2</sup>

**PALABRAS CLAVE:** Compresor centrífugo, efecto bombeo, detección, dispersión, autocorrelación.

### DESCRIPCION:

Este trabajo de grado presenta el estudio de investigación realizado sobre el compresor centrífugo, FC-501 de la unidad de cracking catalítico de la refinería de Cartagena S.A., con el objetivo de plantear una solución viable al problema del efecto bombeo presente en la máquina.

La propuesta toma elementos teóricos desarrollados por tesis doctorales de varias universidades, así como las características más relevantes del contexto de la máquina, a fin de establecer una solución que se acomode a estas condiciones y que brinde como resultado la detección oportuna del efecto de bombeo. Esta solución se halla en el campo de la probabilidad, condensada en dos métodos: El método basado en el cálculo de la función de autocorrelación y el método basado en el cálculo de la dispersión. Estos dos procedimientos son puestos a prueba con la ayuda de una herramienta software, que procesa los datos históricos de la máquina durante momentos críticos que facilitan el análisis de los resultados.

Con base en los resultados arrojados por diversas pruebas de desempeño de cada uno de los algoritmos, se determina la superioridad del método basado en la dispersión bajo las condiciones y restricciones definidas por los datos disponibles. Además, se comprueba la viabilidad de una implementación online de los métodos, pero, se sugiere realizar pruebas adicionales antes de llevarla a cabo.

---

<sup>1</sup>Trabajo de Grado.

<sup>2</sup> Facultad de Ingenierías Físico Mecánicas, Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Director: MPE, Julio Augusto Gélvez Figueredo. Codirector: Dr. (c) Carlos Fernando Agudelo Gómez.

## ABSTRACT

**TITLE:** SURGE DETECTION OF A CENTRIFUGAL COMPRESSOR ON A CATALYTIC CRACKING UNIT<sup>3</sup>.

**AUTHOR:** ESCOBAR, Oscar J.<sup>4</sup>

**KEYWORDS:** Centrifugal compressor, surge, detection, dispersion, correlation function.

### DESCRIPTION:

This degree work presents the investigation research applied to the centrifugal compressor, FC-501 from the Refinería de Cartagena S.A. catalytic cracking unit, in order to develop a viable solution for the machine's surge.

This proposal takes theoretical elements developed by doctoral thesis from many universities, as well as the most relevant characteristics about the machine context, in order to establish a solution according to these conditions and that brings as a result the surge's opportune detection. That solution is found on the probabilistic field, condensed on two methods: The method based on the autocorrelation function calculation and the method based on the dispersion function calculation. These two procedures are tested with the support of a software tool that processes the machine's historical data of critical moments which facilitates the results' analysis.

Taking into account the results from a variety of performance tests applied to each algorithm, the superiority of the method based on the dispersion calculation is determined, under the conditions and restrictions defined by the available data. Also, the methods' online implementation viability is proved, but, additional proves realization is suggested before it might be implemented.

---

<sup>3</sup> Engineer's degree work.

<sup>4</sup> Faculty of Physic-Mechanical Engineers. Electric, Electronic and Telecommunication school. Director: MPE, Julio Augusto Gélvez Figueredo. Co-director: Dr. (c) Carlos Fernando Agudelo Gómez.



## INTRODUCCION

El correcto funcionamiento de todos los componentes de un proceso industrial, más concretamente de las máquinas involucradas, garantiza la eficiencia del mismo, viéndose reflejada en disminución de costos de operación y mayores utilidades. Con miras a alcanzar este objetivo se deben establecer condiciones de trabajo donde la integridad de la máquina sea garantizada y de igual manera, plantear mecanismos de control que eviten que ésta entre en condiciones de operación inadecuadas, que contribuyen al desgaste de la máquina o inclusive a la destrucción de la misma.

Es así como los fenómenos relacionados con el correcto desempeño de la máquina cobran relevancia, y más aún, aquellos que modifican su funcionamiento y cuyo efecto es perceptible con cierta frecuencia. Este es el caso del efecto bombeo, *surge*, en los compresores centrífugos, el cual se caracteriza por oscilaciones en el flujo y la presión, tanto por encima como por debajo del punto de operación, sometiendo a grandes esfuerzos a la máquina misma y a las tuberías. Esta situación se presenta en un compresor de la unidad de cracking catalítico de la Refinería de Cartagena S.A., donde es utilizado como inyector de aire, del cual, gracias a sensores de vibración, velocidad, desplazamiento, caudal y presión instalados, se poseen datos históricos de instantes de tiempo donde se presume el fenómeno ha incidido. Con base en esta información y en el estudio de las características del efecto bombeo, se plantean dos técnicas basadas en el procesamiento digital de las señales de los sensores, auto-correlación de los datos y dispersión de los mismos, a fin de identificar el momento en que surge el fenómeno, con miras a generar una alarma que permita tomar acciones de control efectivas que eviten una operación inadecuada del compresor.

El contenido de este trabajo busca mantener un orden que le permita al lector situarse en el contexto del problema, para posteriormente comprender los análisis y resultados obtenidos. Es por esto que, en el primer capítulo se aborda al compresor centrífugo, su historia, así como, aspectos conceptuales, con el objetivo de entender las leyes que lo gobiernan, sus características principales y su funcionamiento; de esta manera, se establece la relación entre el fenómeno del bombeo y el contexto sobre el cual se desarrolla. Además, se describe el fenómeno de interés desde varias ópticas, mostrando resultados de otros estudios que han intentado modelarlo y caracterizarlo, siendo estos el punto de partida para el estudio particular del efecto bombeo, presente en el compresor centrífugo de la



refinería de Cartagena. En el segundo capítulo, se presenta el contexto práctico del fenómeno a analizar, es decir, las especificaciones del compresor, los sensores instalados en el mismo (disposición física), el sistema de adquisición de datos y los datos que se poseen para el estudio. El tercer capítulo plantea los métodos que se proponen como solución, sus características, antecedentes y el desarrollo particular para implementarlos como solución del problema. En el cuarto capítulo se encuentran plasmados los resultados de la experiencia de desarrollo del algoritmo de identificación, su aplicación, así como los inconvenientes y soluciones que se hicieron presentes durante el proceso. Finalmente, en el quinto capítulo se evalúan los objetivos planteados en el plan de trabajo, se desarrollan las conclusiones del proceso y se elaboran sugerencias para la implementación del algoritmo.



## CAPITULO 1.

### EL COMPRESOR CENTRÍFUGO

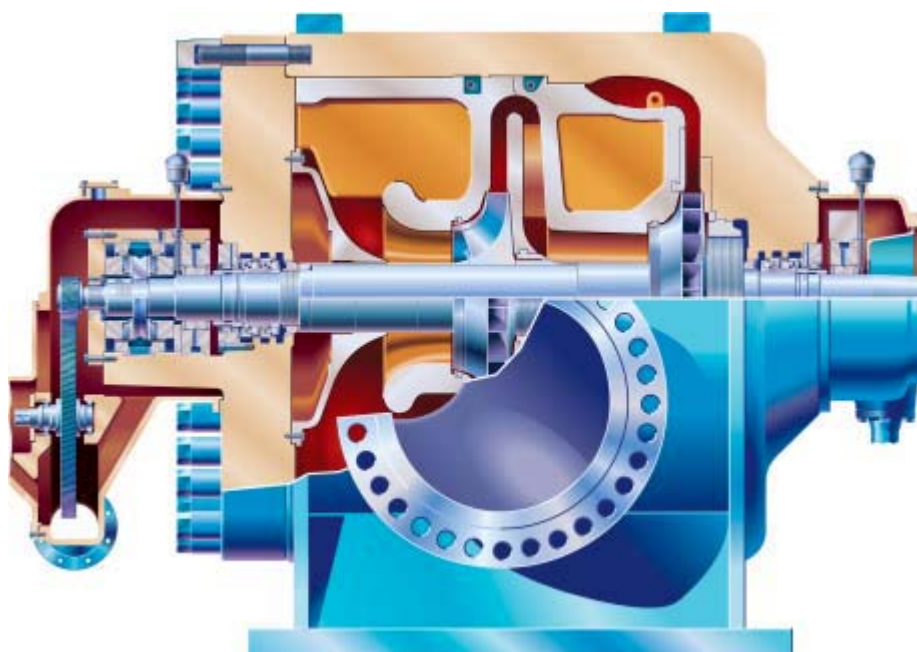


Figura 1. Compresor centrífugo. Fuente: Tomado de [2]

Es una turbo-máquina que transforma energía: Mecánica (producida en el rotor) a cantidad de movimiento en el fluido que pasa a través del compresor, el cual a su vez la transforma en energía cinética para finalmente llegar al difusor donde se convierte en energía potencial de presión. El desplazamiento del fluido es esencialmente radial y se aprovecha la fuerza centrífuga para impulsarlo desde los álabes hacia el difusor, gracias a lo cual el compresor recibe su nombre. Es utilizado para elevar presión de diversos gases, vapores o la mezcla de ambos, además, son de construcción sencilla, requieren bajo mantenimiento y permite un funcionamiento continuo durante largos periodos, razón por la cual son de gran importancia en la industria petroquímica.



## 1.1 Breve recuento histórico

Los primeros turbocompresores fueron desarrollados a comienzos de 1900. Estos eran originalmente manufacturados por las industrias de las turbinas a vapor, cuyo destino final era la ventilación de las minas demasiado profundas, tal y como era el caso de las minas de carbón. Por esa época el método de producción del impulsor, dependía de la forma en que era fabricado así como de la compañía productora. Todo esto se remonta a décadas antes de que la tecnología permitiera la construcción de compresores más eficientes. No fue sino a finales de la segunda guerra mundial cuando la inversión en tecnología sufrió un gran impulso y la construcción de compresores de alta velocidad fue posible. Entre 1947 y 1948 Ingersoll-Rand y Clark diseñaron el primer compresor centrífugo para la transmisión de gas. En 1952 "El Paso Natural Gas" (empresa norteamericana) se convirtió en la primera compañía en utilizar grandes compresores centrífugos manejados por turbina, para la transmisión de gas.

El relación de presión, a través del los primeros impulsores remachados, era de aproximadamente 1.2:1, lo que significaba que para aplicaciones comunes se debería contar con por los menos diez o doce etapas para alcanzar la presión requerida. Un impulsor moderno sencillo puede llegar a generar relaciones de presión tan altas como de 8:1.

A medida que la eficiencia de los compresores iba en aumento, de igual manera incrementaba la posibilidad de daños a causa del efecto bombeo. Era de vital importancia evitar el daño a las máquinas así como la interrupción del proceso de producción. Para satisfacer estas necesidades en aumento, los sistemas anti-bombeo evolucionaron.

Los primeros controles eran neumáticos. Monitoreaban la presión diferencial a través del compresor, contra el diferencial de presión en el instrumento de medida, señales que permitían dar ordenes de tipo neumático a una válvula de recirculación. Con eficiencia cada vez mayor, compresores más exigidos y aplicaciones más demandantes, fueron requeridos mejores controles anti efecto bombeo.

Inicialmente, los controles electrónicos eran modelos de los controles neumáticos. Eran más rápidos y menos complicados, más seguros y requerían de mantenimientos menos frecuentes que los de sus predecesores. Con la llegada del microprocesador, los sistemas anti efecto bombeo se tornaron algorítmicamente más complejos, los límites de bombeo eran modelados con polinomios y los



esquemas de control asimétricos fueron puestos en práctica. Los sistemas anti efecto bombeo evitan la aparición del fenómeno maximizando el rango de operación del compresor<sup>5</sup>.

## 1.2 Partes del compresor centrífugo

Pueden variar de acuerdo a la aplicación para la cual se ha diseñado el compresor, sin embargo, las partes principales son:

- Cubierta. Puede ser de división vertical u horizontal, y consiste en dos mitades unidas a lo largo del eje central. Todas las conexiones, como boquillas de carga o descarga, o tuberías de lubricación se ubican en una de las mitades de forma tal que la otra mitad se pueda desmontar con facilidad y brinde acceso a los componentes sin necesidad de desconectar los elementos que van a la planta. Puede ser de hierro fundido, acero fundido o templado u otros materiales, según la carga, temperatura de servicio, gas manejado o tamaño de la máquina. De igual manera, la presión a la que va a ser sometido el compresor, determina el grosor de la cubierta, así como la forma de asegurar las dos mitades.



Figura 2. Cubierta del compresor.

Fuente: Tomado de [2]

<sup>5</sup> Tomado de WHITE, Robert C, KURZ, Rainer, Surge Avoidance for Centrifugal compressors, Solar Turbines Inc, Estados Unidos.



- Diafragmas. Ubicados en la succión, en el intermedio o en la descarga, crean un camino junto con los componentes estacionarios. El diafragma de succión dirige el flujo hacia el ojo del impulsor y puede ser ajustado con aspas para optimizar el ángulo de entrada del flujo. El diafragma intermedio cumple una doble función, conformar el conducto del difusor donde la energía cinética es transformada en energía potencial de presión y ser la conexión hacia el ojo del siguiente impulsor, si es que existe. El diafragma de descarga, hace las veces de difusor del impulsor final y de espiral de descarga. Los diafragmas se fabrican en hierro fundido, acero o acero inoxidable. Sellos de fácil remoción en forma de laberinto, son instalados dentro del diafragma para evitar la inversión del flujo antes del impulsor y para evitar las pérdidas entre etapas.

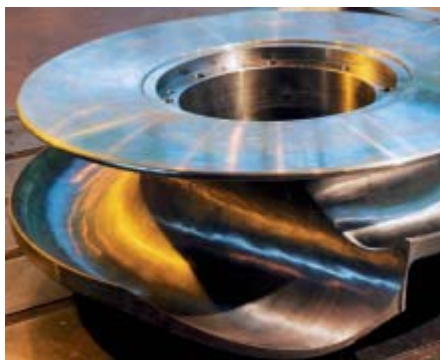


Figura 3. Diafragma. Fuente: Tomado de [2]

- Rotor. Lo conforman básicamente, el eje, el inductor, un impulsor con sus respectivas paletas, un disco, una cubierta y una masa de balanceo. Su diseño geométrico está pensado para cumplir con los requerimientos del flujo. Cada componente tiene una función determinada: El eje le permite al rotor girar libremente; en el inductor las paletas tienen el ángulo adecuado para que el fluido ingrese con la velocidad relativa paralela a la superficie de la paleta; El impulsor hace girar el fluido aproximadamente  $90^\circ$  para que comience a moverse en dirección radial, siguiendo el movimiento del rotor, lo cual produce un gran impulso tangencial; las paletas del impulsor generan presión sobre el fluido, la cual se reduce hasta anularse en el borde de fuga; la masa busca dar balanceo al rotor. Los impulsores se ensamblan a presión sobre el eje, y suelen fabricarse de acero fundido; las paletas que se fabrican inclinadas hacia atrás para mejorar el desempeño, se sueldan tanto al disco como a la cubierta o se apiñan al disco y después se sueldan a la cubierta.

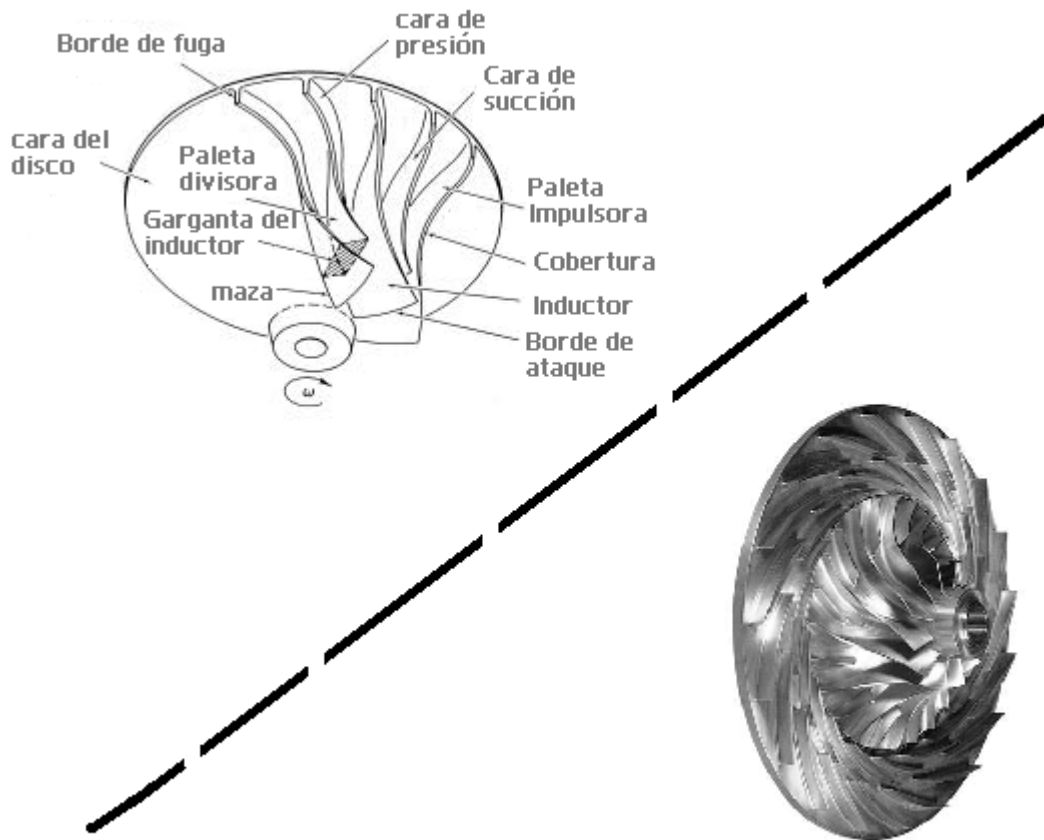


Figura 4. Partes del rotor. Fuente: autor

- ⇒ Sellos. Los sellos finales sobre el eje reducen o minimizan las fugas por la compresión del gas o la entrada de aire dentro de la cubierta del compresor. Dependiendo de la naturaleza del gas que va a ser comprimido y del grado de sellado que se quiera alcanzar se pueden utilizar diversos tipos de sellos.
  - Sellos de laberinto. Son utilizados cuando las propiedades o presión del gas permiten fugas mínimas. Son fabricados con aleaciones ligeras u otro tipo de material resistente a la corrosión y son fácilmente reemplazables. El número de dientes y el espaciado dependen de las condiciones de operación así como de la geometría. Cuando no se permiten fugas, con el caso de gases inflamables, este tipo de sellos se encuentran complementados con sistemas de extracción/inyección.



- Sellos de gas seco. El sellado es garantizado gracias a un cerrojo de gas creado por surcos trabajados a máquina dentro del sello rotativo ajustado al rotor.

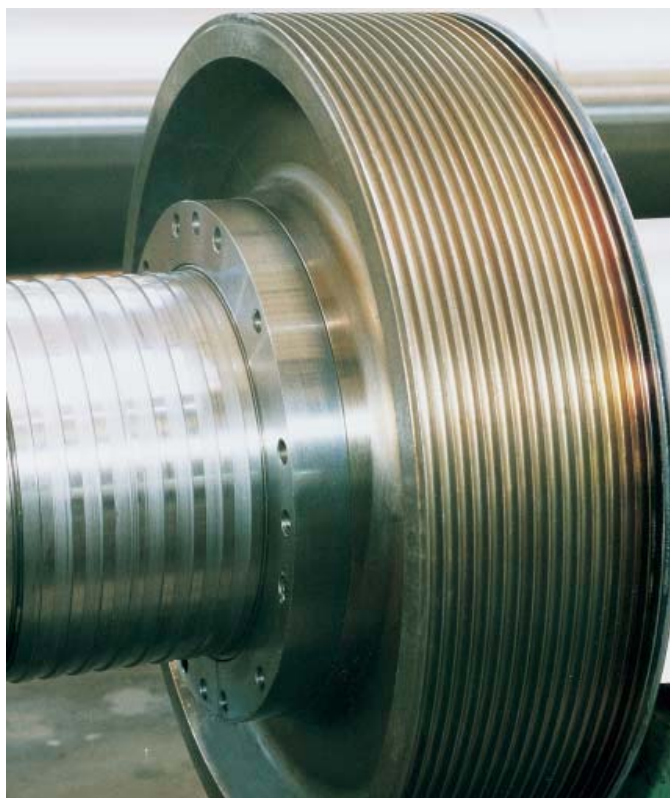


Figura 5. Sellos. Fuente: Tomado de [2]

- Rodamientos. Cumplen funciones de soporte y guía del eje. Existen varios tipos:
  - Hidrodinámicos. Pueden ser lisos (con soporte inclinado) o de empuje (con soporte inclinado y dispositivo ecualizador), los cuales vienen con celdas de carga para medir la presión a la que son sometidos, así como con termocuplas para registrar la temperatura.
  - Magnéticos. Operan bajo el principio de suspensión electromagnética. Aunque realizan las mismas funciones que los de tipo hidrodinámico presentan dos ventajas importantes: En primer lugar, reducen las pérdidas mecánicas gracias a la ausencia de fricción y en segundo lugar, se puede ajustar la posición axial y radial, así como la rigidez del rotor y las características de lubricación.

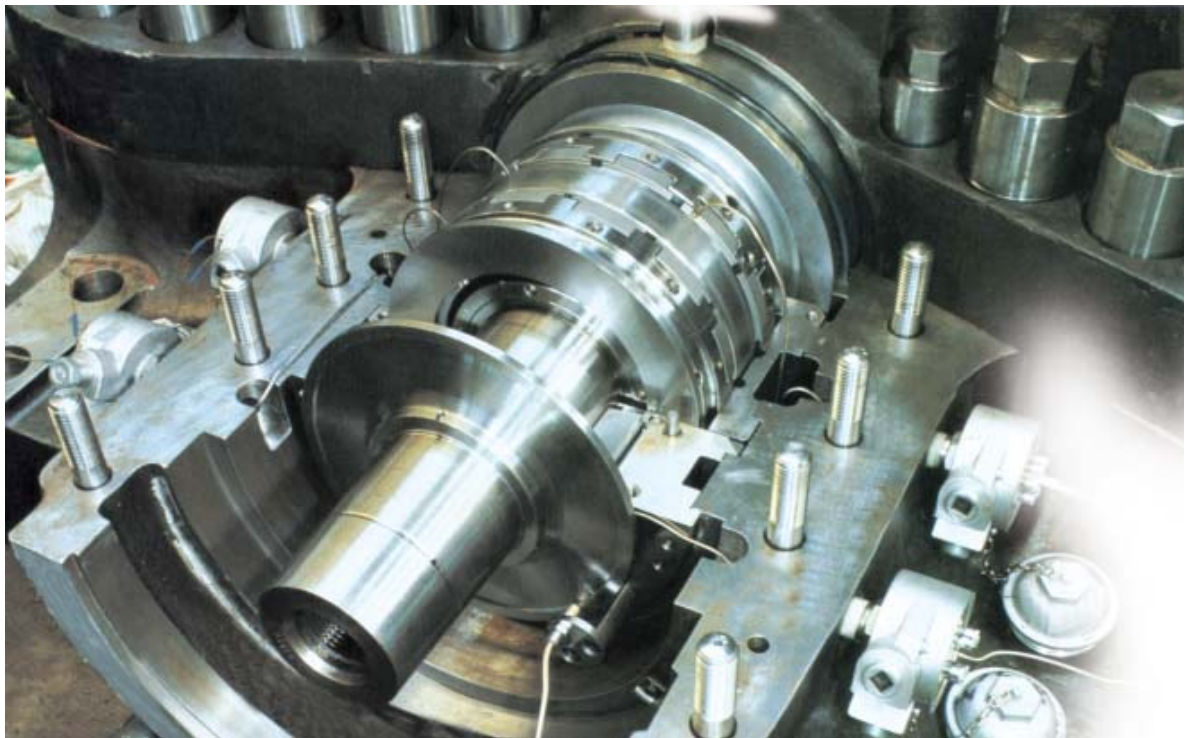


Figura 6. Vista del eje de un compresor. Fuente: Tomado de [2]

### 1.3. Generalidades

De acuerdo con [3], en un compresor centrífugo se produce presión del gas al aumentar la velocidad de éste cuando pasa a través del impulsor y luego recuperarla de forma controlada. La forma de la curva característica depende del diámetro externo del impulsor, el ángulo de inclinación de sus álabes y la forma del difusor. La mayor parte de los impulsores de la industria petroquímica son del tipo de inclinación hacia atrás, pues su curva característica exhibe una mayor pendiente permitiendo un control más sencillo. La velocidad en las puntas de un impulsor convencional suele ser de 800 a 900 (pie/s)<sup>6</sup>, lo cual indica que se podrán desarrollar hasta 9500 pie de carga, valor que dependerá del gas que se va a comprimir; en caso de requerir mayores valores, se implementan compresores con etapas múltiples, las cuales constan de varios impulsores conectados en serie.

Una característica importante de un compresor centrífugo, es el porcentaje de aumento de la presión el cual establece, desde el punto de operación hasta el punto de oscilación, cuanta disminución del flujo ocasiona inestabilidad en forma

<sup>6</sup> Se dan las unidades en el sistema inglés por la aplicación tan difundida en el contexto del problema.



de flujo pulsante, generando daños por sobrecalentamiento y/o falla de rodamientos.

Dadas las altas velocidades de los compresores, se establece un límite máximo para las vibraciones sobre el eje a fin de mantener balanceado el rotor. Este valor suele ser de dos milésimas de pulgada a cualquier velocidad.

Entre las ventajas del compresor centrífugo se pueden mencionar:

- Capacidad de entregar gas libre de aceite.
- No existen piezas que se desgasten en la corriente del compresor, lo cual permite largos periodos sin mantenimiento (siempre y cuando los sistemas de lubricación y de sellos funcionen correctamente).
- Es económico pues, en el intervalo de 200 a 200.000 pie<sup>3</sup>/min, y según sea la relación de presión, se puede utilizar una sola unidad.
- Ofrece una variación amplia en el flujo con una variación pequeña en la carga.
- Se pueden obtener grandes volúmenes en un espacio pequeño.
- Se obtiene un flujo suave y libre de pulsaciones.

Entre las desventajas del compresor centrífugo se encuentran:

- Alta sensibilidad al peso molecular del gas que se comprime, lo cual ocasiona presiones de descarga muy altas o muy bajas.
- Se necesitan altas velocidades en las puntas para conseguir las presiones deseadas.
- Una caída de presión en el proceso puede ocasionar reducciones muy grandes en el volumen del compresor.
- El sistema de lubricación y de sellos es bastante complicado.

Desde un punto de vista termodinámico, el comportamiento del compresor se ve afectado por cambios, bien sea, del peso molecular, la razón de calores específicos, la presión o la temperatura tanto en la succión como en la descarga. Dichas variables pueden ubicar la operación de la máquina en un punto diferente del nominal sobre la curva de carga contra capacidad, dando por resultado un compresor que produce carga pero no presión. Sin embargo, aplicando las "leyes de afinidad" (ecu. 1), siendo N la velocidad en r.p.m., Q la capacidad en la entrada y H es la carga, se puede variar la velocidad del compresor para ajustar la característica del compresor con la salida requerida.

$$\frac{N_1}{N_2} = \frac{Q_1}{Q_2} = \frac{\sqrt{H_1}}{\sqrt{H_2}} \quad (\text{ecu. 1})$$



Cuando el manejador del compresor es de velocidad constante, se puede recurrir a tres métodos para controlar el compresor:

- Aspas de guía de la admisión. Aspas fijas de ajuste manual o automático en la entrada de la primera etapa que hacen que cambie el ángulo de aproximación del gas con relación al impulsor giratorio.
- Estrangulación de succión. Esto produce una presión de succión ligeramente más baja que la de diseño lo que se traduce en una carga total más elevada si la presión de descarga permanece constante.
- Estrangulación en la descarga. Con el flujo reducido, el compresor produce una carga mayor que la que necesita el proceso, por lo que se recurre a estranglarla antes de que llegue al equipo, cuyo efecto se refleja en ineficiencias, pues el caballaje para la compresión se desperdicia.

Todos los compresores dinámicos tienen un intervalo de capacidad limitado, a velocidad fija. Por debajo del valor mínimo (50-70% por debajo del nominal), el compresor tendrá oscilaciones, es decir inestabilidad. Es de vital importancia diseñar los sistemas de los compresores para evitar esta situación, lo cual se consigue con un control anti oscilaciones. El más sencillo, utilizado en compresores de aire, es una válvula de purga que se abre y deja salir el exceso de capacidad a la atmósfera cuando el flujo en el proceso es muy bajo. A veces también se utilizan controles más eficientes a base de válvulas de control de la succión. Cuando el gas comprimido no se puede desperdiciar, el control anti oscilación más común es el control de derivación, el cual devuelve el flujo indeseado a la fuente de succión.

#### **1.4 El efecto Bombeo (*Surge*)**

Dada la importancia del compresor en los procesos industriales, se requiere ejercer sobre él un control eficiente que garantice a su vez el correcto funcionamiento de la planta. Al diseñar el sistema de control se debe tener en cuenta el efecto "*surge*", el cual lleva la máquina a una condición de funcionamiento inestable caracterizada por fluctuaciones del flujo promedio a lo largo del sistema de compresión. La carga oscilatoria genera un gran impacto en los cojinetes y rodamientos que soportan el eje y pueden ocasionar efectos desastrosos en todo el sistema. Además, produce vibraciones que pueden percibirse auditivamente. Con relación a lo estudiado por [6], dependiendo del sistema de compresión y las condiciones de operación se distinguen diversas categorías de fluctuaciones del caudal y la presión:

- *Mild surge (bombeo leve)*. Durante el efecto, la frecuencia de oscilación es cercana a la frecuencia de resonancia de las tuberías y no ocurre inversión del



flujo. Dicha frecuencia es independiente de la velocidad de rotación y de las condiciones de operación.

- *Classic surge (Bombeo clásico)*. Se caracteriza por oscilación continua de la presión a una baja frecuencia, la cual es independiente de la velocidad de rotación y las condiciones de operación. Bajo esta situación la dinámica se vuelve no lineal lo que se traduce en armónicos de gran amplitud, presentando además por cortos intervalos de tiempo, la inversión del flujo.
- *Deep surge (Bombeo fuerte)*. También se encuentra caracterizado por oscilaciones prolongadas, tanto del flujo como de la presión, las cuales se hacen negativas para parte del ciclo. Durante el flujo inverso, el punto de operación se mueve a lo largo de una línea denominada característica de flujo inverso. Esta define la resistencia que oponen los álabes a la circulación en sentido inverso. Durante el semi-ciclo positivo el punto de operación se ubica cercano al punto de operación en estado estable. La frecuencia de oscilación se encuentra determinada por el vaciado y llenado del volumen conectado al compresor y por lo tanto de la configuración del sistema.
- *Modified surge (Bombeo modificado)*. Recibe este nombre cuando el efecto se presenta simultáneamente con *rotating stall*. Es entonces cuando el flujo es inestable y asimétrico respecto al eje.

La importancia de mantener el punto de operación de la máquina se puede entender a partir del análisis de la figura 7, en la cual se observa una familia de curvas características de un compresor centrífugo de velocidad variable, para cada velocidad de rotación.

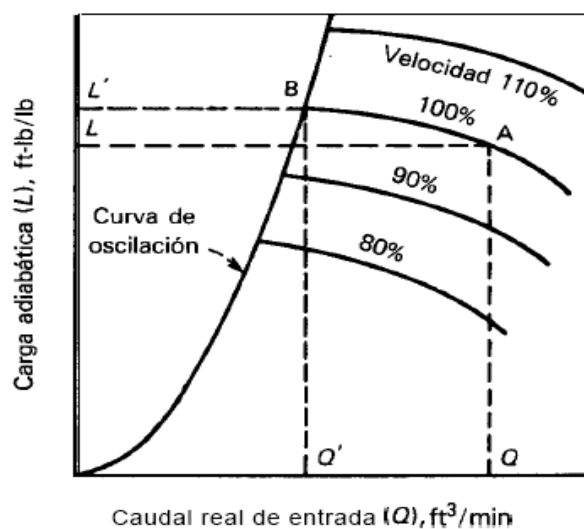


Figura 7. Curvas características del compresor centrífugo. Fuente: Tomado de [3]



Suponiendo que el compresor se encuentra trabajando en el punto A de la curva del 100% de la velocidad, el flujo de entrada es  $Q$  y la carga es  $L$ ; si por alguna razón el punto de operación se desplaza a B donde el flujo se reduce hasta  $Q'$  y la carga aumenta a  $L'$  que es la máxima que el compresor puede manejar a esta velocidad, el compresor se ubica en una región de operación inestable indicada por la curva de oscilación, hecho que se ve reflejado en pulsaciones rápidas en el flujo y la presión de descarga, ocasionando inversiones de alta frecuencia en el empuje axial sobre el árbol del compresor y daños mecánicos. Hay que resaltar que la curva de oscilación se encuentra definida para todas las velocidades.

En la figura 8 se puede observar un ciclo típico del efecto bombeo. Entre las características más destacables de la figura, se encuentra el incremento y posterior reducción de la presión, así como tasas de flujo negativo, lo que se interpreta como inversión del flujo. Además se puede observar la característica cíclica la oscilación, razón por la cual surge el concepto de frecuencia de oscilación.

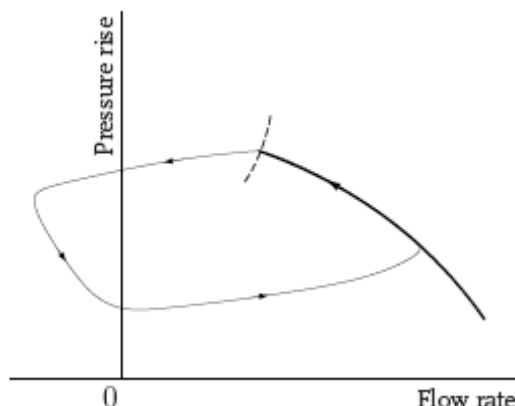


Figura 8. Ciclo característico del efecto bombeo. Fuente: Tomado de [5]

Tal y como se evidencia, la razón principal para que surja el fenómeno es la incapacidad de mantener un flujo mínimo circulante a través del compresor, lo que ubica al compresor en un punto de operación sobre la curva de oscilación. Entre las causas más comunes del efecto se pueden mencionar:

- Mal diseño de la máquina.
- Ensamble inapropiado.
- Restricciones en la succión o en la descarga.
- Obstrucción de los conductos internos.
- Irregularidades en las condiciones del proceso (cambio repentino de la carga).
- Pérdida inadvertida de velocidad.
- Cambios en la temperatura, presión y composición del gas.



A continuación, se va a realizar el desarrollo matemático, que busca sustentar el aporte de variables propias del proceso, en el surgimiento del efecto bombeo.

La carga producida en un compresor centrífugo es proporcional al cuadrado de la velocidad (ecu. 2), así mismo el caudal es proporcional a la velocidad (ecu. 3), de donde se obtiene la relación entre el caudal y la carga adiabática (ecu. 4).

$$L = C_1 \cdot N^2 \quad (\text{ecu. 2}) \quad Q = C_2 \cdot N \quad (\text{ecu. 3}) \quad L = C_3 \cdot Q^2 \quad (\text{ecu.4})$$

La relación entre la razón de compresión "Rc" (ecu. 5), donde P<sub>2</sub> es la presión de descarga y P<sub>1</sub> la presión de succión, y la carga, se encuentra determinada por (ecu. 6).

$$Rc = \frac{P_2}{P_1} \quad (\text{ecu. 5}) \quad Rc = \left(1 + \frac{Lm\phi}{1.645T_1Z_1}\right)^{\frac{1}{\phi}} \quad (\text{ecu. 6})$$

Donde, m es el peso molecular, φ la relación de calores específicos, T<sub>1</sub> la temperatura de succión y Z<sub>1</sub> el factor de supercompresibilidad. A partir de esta ecuación se puede observar como la temperatura, presión y características del gas pueden modificar la carga ubicando al compresor en un punto de operación inestable. Este modelo matemático, sustenta el desarrollo de la teoría de control a partir de la curva característica de la máquina el cual se aborda en capítulos posteriores.



## **CAPITULO 2.**

### **CONTEXTO REAL DE LA PLANTA**

En este capítulo se ambienta el entorno en el cual se encuentra operando el compresor centrífugo, objeto de estudio de este trabajo de grado. Es importante poder conocer, en la medida de lo posible, todas las variables propias de la situación que se va a resolver, para aterrizar el problema, conceptualizar el proceso, dar un mejor análisis a los resultados obtenidos, y establecer la validez y aplicabilidad de los mismos. Toda esta información cobra mayor relevancia, cuando se requiere establecer las condiciones de operación nominal de la máquina, y como estas se ven reflejadas en cada una de las variables físicas medibles a través de la instrumentación instalada, además, brinda una idea acerca de las constantes temporales, tanto del compresor como del proceso en el cual es utilizado, con miras a establecer acciones de control eficientes.

Entre las variables de interés se pueden mencionar, las características del compresor, la función que desempeña dentro del proceso, su rango de operación, de igual manera, los sensores instalados, su principio de operación, sus especificaciones técnicas, también los datos con los que se cuenta, la forma en que son adquiridos, posibles errores introducidos durante su obtención, entre muchas otras cuestiones que pueden aportar a la contextualización del problema. Estas y otras preguntas intentan ser resueltas durante el desarrollo de este capítulo,

#### **2.1. Especificaciones del compresor.**

Es utilizado como soplador principal de aire, en la unidad de cracking catalítico de la Refinería de Cartagena S.A., donde aporta en la combustión de coque en el Regenerador y alimenta la succión del soplador de control. Se encuentra referenciado con la sigla FC-501 y comprime aire de 14.7 – 44.7 PSIA (pounds per square inch absolute), lo que equivale a 0-30 PSIG (pounds per square inch gage). La toma de aire es vertical y presenta un diámetro de 52 pulgadas, la cual cuenta con una caperuza y una malla. La capacidad del compresor es de 62990 SCFM (standard cubic feet per minute).

Los cambios de flujo se obtienen al reajustar las válvulas de admisión de la turbina motriz. El elemento motor del soplador, es una turbina de condensación total que exhasta a un condensador de superficie. El aire comprimido pasa por un tubo de



30 pulgadas que lo lleva directamente a la entrada de aire primario y secundario del quemador auxiliar. Si el flujo cae por debajo de 38400 SCFM se activa la alarma de bajo flujo FLACV-527 y a su vez la FLACV-518 (carga al reactor), lo que activa el sistema de emergencia como protección del proceso.

Para arrancar el compresor, se pone en servicio la línea de vapor de 60 PSIG, etapa durante la cual se lleva a cabo revisión de escapes y corrección de los mismos. Luego, se activa el sistema de lubricación del soplador, calentando el aceite inicialmente con el serpentín de vapor de la consola de lubricación. A continuación, se pone en funcionamiento el *Barrier Device*, el agua refrigerante al condensador de superficie, el nivel de condensado y la circulación del mismo, el vapor a los sellos y eyectores, y se continua calentando la línea de vapor de 600 PSIG hasta el grifo de entrada a la turbina de entrada del Soplador Principal. Se pone en automático las bombas auxiliares de lubricación y condensado. Cuando la temperatura del vapor alcance un mínimo de 650 °F en el sensor de temperatura de entrada a la turbina, se arranca el soplador lentamente a la atmósfera con toda la válvula de mariposa abierta y los bloques de aire primario y secundario cerrados. Se deja girando la maquina a 1000 RPM durante 45 minutos, para luego aumentarla lentamente hasta 3510 RPM. Finalmente se lleva el flujo de aire hasta 52 KSCFM y se ajusta la presión en la descarga con la válvula de mariposa a la atmósfera, entre 15 y 20 PSIG.

Cuando se programa la operación de apagada, se tiene que ir reduciendo lentamente la carga del proceso, sin embargo, tras haber alcanzado aproximadamente la mitad de la carga nominal, se procede a apagar el compresor de un solo golpe, por medio de la activación de los sistemas de emergencia, lo cual permite la verificación del correcto funcionamiento de dicho sistema<sup>7</sup>.

## 2.2. Sistema de automatización de la planta

En esta sección se presenta una descripción somera del sistema de automatización implementado en la Refinería de Cartagena S.A., ADVANT OCS, con base en la información proporcionada por [1]. Es un sistema integrado, abierto y distribuido para la automatización industrial. Está conformado por una familia de unidades computarizadas y varias opciones de comunicación que incluyen, buses de campo, LAN's y WAN's. Esta combinación define un sistema de automatización poderoso,

<sup>7</sup> Tomado de Rojas R. y Castaño J, Manual de cracking. , versión 01, 2005, coordinación de Cracking, Refinería de Cartagena.



capaz de realizar tareas sencillas como el control de una máquina, o inclusive el control total de una planta. Las principales funcionalidades del sistema son:

- Control de procesos.
- Interacción con los operarios.
- Ingeniería.
- Manejo de la información.
- Control por lotes.
- Comunicación.

El sistema puede ser configurado de una gran variedad de formas, dependiendo del tamaño y complejidad del proceso. Inicialmente se pueden ubicar controladores independientes y estaciones de control locales para la supervisión y control de máquinas en secciones separadas. Después, se pueden interconectar todos los controladores a un operador central y a una estación de manejo de información (IMS) para configurar una red de control. Finalmente, varias redes de control pueden ser interconectadas dentro de una red completa de la planta que puede ser compartida por IMS centrales, ingeniería y estaciones de operador. El sistema puede ser complementado con la adición de hardware y software para aplicaciones específicas, como peso, variadores de velocidad, sistemas de seguridad y elementos para la medición de la calidad. La capacidad de expansión de sistema permite al usuario agregar nuevos equipos sin hacer obsoletos los existentes. La habilidad de auto-configurarse elimina la necesidad de editar las descripciones de la topología existente cuando se instalan nuevas estaciones garantizando así una reducción de costos. En la figura 9 se observa el sistema de automatización completo.

El manejo de la información es una de las mayores prestaciones del sistema de automatización. El manejo y producción de la información se lleva a cabo con protocolos industriales a través de los cuales se proveen plataformas abiertas para el uso de software propietario. Entre las ventajas del sistema de comunicación se pueden mencionar:

- Portabilidad garantizada por un ambiente de programación poderosa basada en estándares industriales.
- Integración de la información.
- Recolección de datos históricos, almacenamiento y recuperación.
- Reportes.
- Cálculos.
- Manejo de objetos y soporte para la definición de objetos propios.

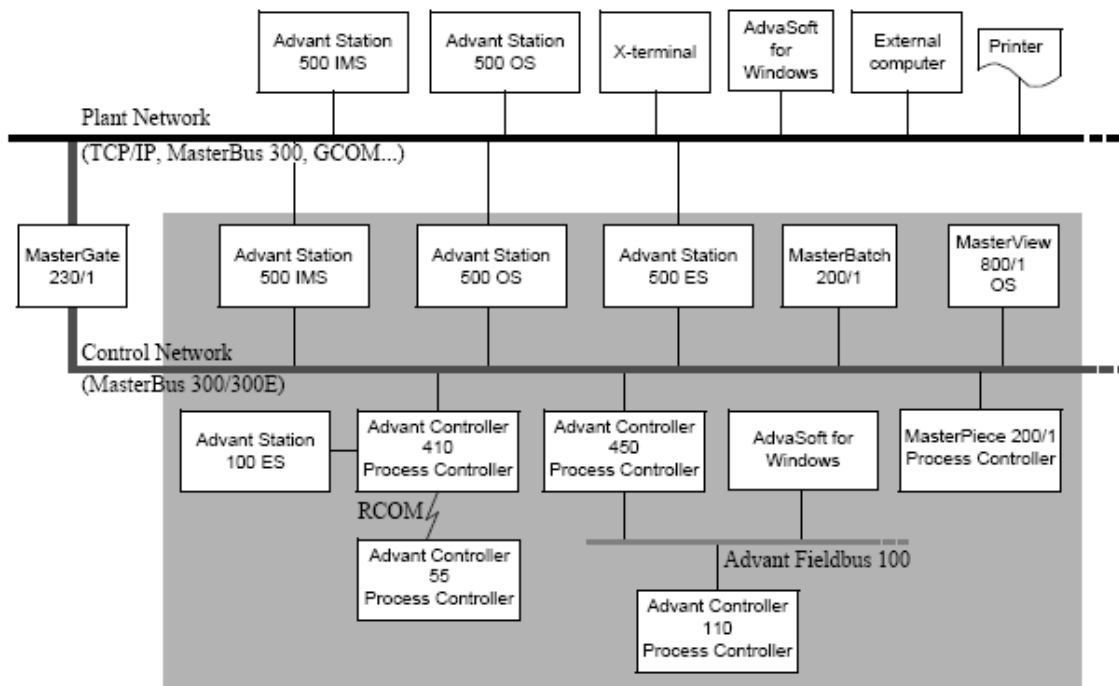


Figura 9. Sistema de automatización ADVANT OCS.

Fuente: Tomado de [1]

La comunicación en campo se lleva a cabo través de varios protocolos como son MasterFieldBus, AdvantFieldBus y RCOM/RCOM+, mientras que la comunicación con el exterior a través de estaciones de comunicación como EXCOM, MVI y GCOM. MasterFieldBus es un enlace de comunicación de lata velocidad, que conecta unidades de entrada/salida, MasterPiece51, MasterPiece 90, Advant Controller 110, SAMI (convertidores para manejadores de motores asíncronos) y TYRAK (convertidores para manejadores de motores DC) a la MasterPiece 200/1, mostrado en la figura10. La MasterPiece 200/1 es una estación de proceso para el control lógico y regulatorio, así como para la supervisión, que cuenta hasta con 4600 pines de entrada/salida. Puede ser utilizado independiente o como un controlador integrado en una sistema de control distribuido. Su capacidad de configuración cubre un amplio rango de funciones como control lógico y secuencial, manejo de texto y datos, aritmética, reportes, control de posición y de regulación, incluyendo el PID y el adaptativo auto-sintonizado. Su programación se lleva a cabo con el lenguaje de programación AMPL, que se basa en bloques funcionales con representación gráfica. Están disponibles una gran cantidad de interfaces para señales digitales y analógicas, con variedad de rangos y de



voltajes. Existen inclusive algunas unidades especiales para el manejo de termocuplas pt100, entradas impulso y control de posición. Un computador puede comunicarse con esta unidad mediante protocolo EXCOM, el cual está codificado en ASCII y manejando un esquema de petición/respuesta asíncrono.

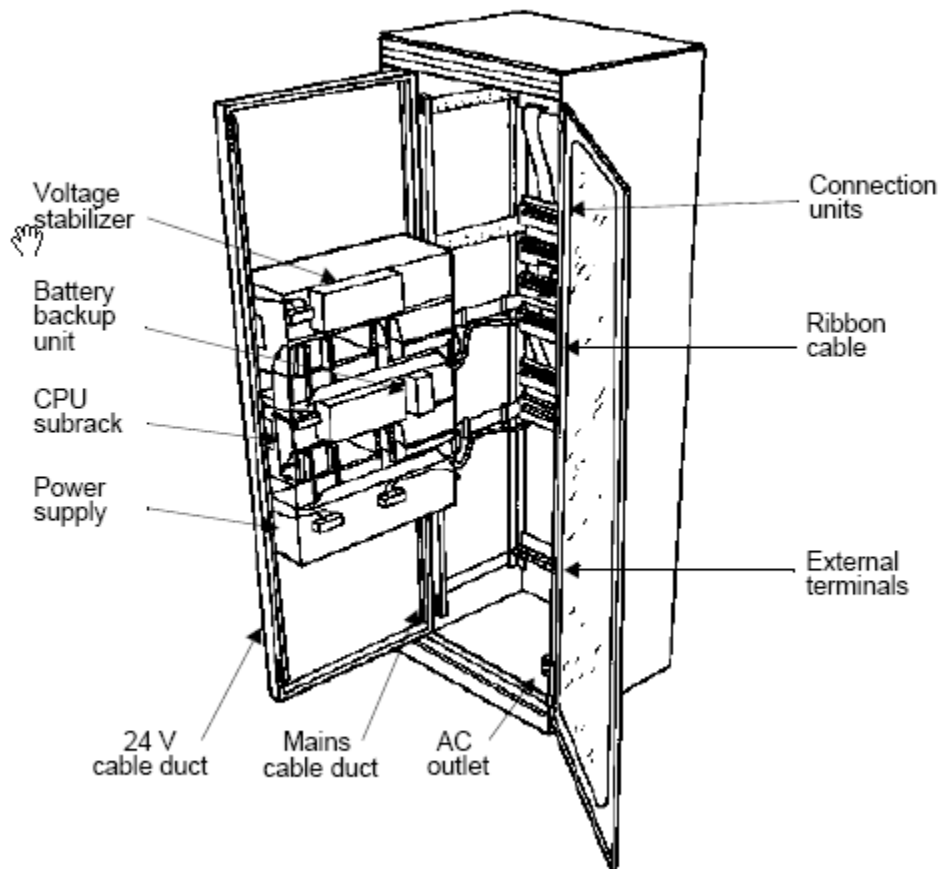


Figura 10. Aspecto físico de la MasterPiece 200/1. Fuente: Tomado de [1]

### 2.3. Sensores.

A continuación se presenta una lista de los sensores de los cuales se poseen datos históricos para su posterior procesamiento. La disposición física de los sensores de vibración y desplazamiento se indica en la figura 11, donde los números indicados en el listado de dichos sensores, corresponden a los números mencionados en la figura.



- FC-PI-5002 Presión Regenerador
- FF-FI-551 Carga al Reactor
- FC-PDI-503 Diferencial Succión - Descarga Soplador de Aire Principal
- FC-ZI-500R Posición Válvula Admisión de Vapor a Soplador
- FC-ZI-5003 Desplazamiento Axial lado Compresor del FC-C-501 **(1)**
- FC-ZI-5002 Desplazamiento Axial lado Turbina del FC-C-501 **(11)**
- FC-ZI-5001 Desplazamiento Axial lado Turbina del FC-C-501 **(10)**
- FC-VI-5008 Vibración Eje X Cola Compresor del FC-C-501 **(2)**
- FC-VI-5007 Vibración Eje Y Cola Compresor del FC-C-501 **(3)**
- FC-VI-5006 Vibración Eje X Acople Compresor del FC-C-501 **(4)**
- FC-VI-5005 Vibración Eje Y Acople Compresor del FC-C-501 **(5)**
- FC-VI-5004 Vibración Eje X Acople Turbina del FC-C-501 **(6)**
- FC-VI-5003 Vibración Eje Y Acople Turbina del FC-C-501 **(7)**
- FC-VI-5002 Vibración Eje X Lado Turbina del FC-C-501 **(8)**
- FC-VI-5001 Vibración Eje Y Lado Turbina del FC-C-501 **(9)**
- FC-SI-500A Velocidad Turbina del FC-C-501
- FC-FI-501 Aire Total Succión FC-C-501

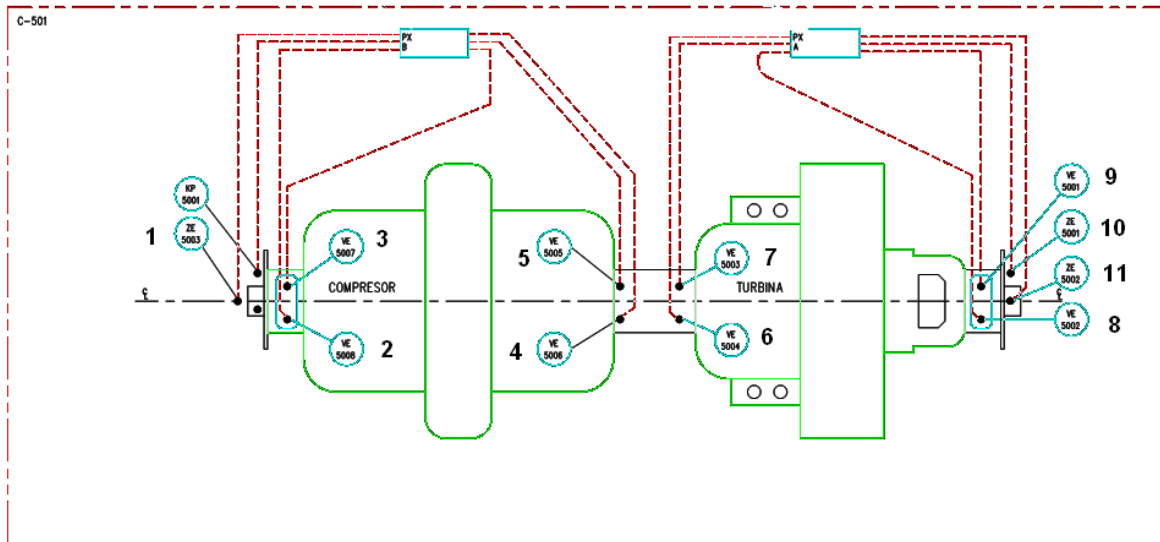


Figura 11. Distribución de los sensores de vibración y desplazamiento en la máquina.

Tal y como se comprueba en los datos históricos de estos sensores, el FC-VI-5003 y el FC-VI-5004, se encuentran fuera de servicio, por ende, su lectura siempre indicará cero.



## 2.4. Históricos de la máquina

Se encuentran compilados dentro de un archivo cuya extensión es “.xls”. Esto indica que para su tratamiento se requiere de una aplicación que se especialice en hojas de cálculo. La información presenta un esquema estandarizado lo cual permite el procesamiento de información compilada en paquetes cuya procedencia radique de diferentes fechas. Dentro de la hoja de cálculo, cuya distribución se entiende como una cuadrícula formada por filas y columnas, se encuentran ubicados los datos, cada uno ocupando una celda (intersección de una fila con una columna).

Se cuenta con tres bloques de datos, denominados ICP-datas2, ICP-datas3 y ICP-datas4, los cuales poseen información del estado de los sensores en tres fechas diferentes y con frecuencia de muestreo diferentes. ICP-datas2 corresponde al comportamiento de la máquina durante un día de operación normal del compresor. Se tienen datos de todos los sensores indicados en el numeral anterior con excepción de FC-FI-551 Y FC-PI-5002, muestreados a una frecuencia de de 0.33 Hz, es decir, hay un dato cada 3 segundos, de donde se deduce que se cuenta con 28000 muestras por sensor. ICP-datas3 presenta la información entre las 9:30 y las 11:00 am, momento donde ocurre una parada de emergencia, presuntamente a causa de la existencia del fenómeno del bombeo. Se posee información de todos los sensores, con una frecuencia de 0.5 Hz, lo cual indica que se tienen 2700 datos. ICP-datas 4 compila la información de la arrancada de la planta, entre las 3:00 y las 8:00 pm del mismo día de la parada de emergencia. Se posee información de todos los sensores con una frecuencia de muestreo de 0.5 Hz, para un total de 9000 datos. Como no se posee información de todos los sensores, en todos los paquetes de datos, se decide excluir los sensores faltantes, es decir, no se procesará la información correspondiente a FC--FI-551 Y FC-PI-5002, por considerarse insuficientes para las pruebas. Además, los sensores FC-VI-5003 y FC-VI-5004 no son tomados en cuenta pues siempre marcan cero, por lo tanto se procesará la información de trece sensores.

En la primera fila se encuentren los encabezados, donde la primera celda corresponde a la variable temporal, con un formato de día-mes-año-hora- minuto-segundo. A partir de la segunda columna se encuentra la información de los sensores, donde el encabezado corresponde a la referencia del instrumento, tal y como se enunció en el numeral anterior, y las filas, después de la segunda, presentan los valores numéricos resultado de la conversión en el instante de tiempo indicado en la primera celda de la fila. Los valores se encuentran en un formato de signo y número de seis dígitos, los cuales se distribuyen en el valor



entero y los restantes en el valor decimal. Por ejemplo, al número 32.65478 se le asignan dos dígitos para el valor entero 32, y los restantes cuatro dígitos se le asignan al valor 65478 lo que indica que se perderá el número 8 correspondiente al dígito menos significativo. Siempre se asigna primero el valor entero y posteriormente el valor decimal, pero el cero entero no cuenta como dígito, o sea, 0.645589 se podrá obtener tal cual, sin pérdida de ningún dígito.

Estos datos son los que van a ser sometidos al análisis a fin de detectar las características del efecto de bombeo en el compresor. La estructura en que fueron almacenados facilita su adquisición mediante Matlab, y permite el discernimiento de los datos correspondientes a cada sensor.



## CAPITULO 3.

### METODOS DE DETECCION DEL EFECTO BOMBEO

De acuerdo a lo mencionado en [4], los sistemas para la prevención del efecto bombeo se pueden dividir en dos grandes grupos:

- Preventivos, conocidos como ACS (Antisurge Control System). En este caso, el límite de bombeo es determinado experimentalmente o por cálculos teóricos y es dibujado en la curva dinámica del compresor; a su derecha (10% a 15% del incremento del flujo) se encuentra la línea de control. Si el punto de trabajo del compresor cae a la izquierda de la línea de control, la bomba o la válvula de recirculación se abre ligeramente evitando así que el compresor entre en operación inestable.
- Sistemas que responden a las características incipientes del efecto Bombeo. Distinguen características del comportamiento de las señales provenientes de los sensores, que determinan las condiciones de operación del compresor, tales como tasa de flujo, presión de sobrecarga, corriente del manejador eléctrico principal, entre otras, generando una señal discreta para forzar la apertura de la válvula de recirculación y/o la parada del compresor. Además del sistema antibombeo se puede instalar un indicador de operación en la región inestable como respuesta a fallos en el sistema de detección.

En la figura 12 se observa un esquema que busca ejemplificar las características de cada uno de estos grupos de sistemas, los cuales son implementados en un esquema de control realimentado. Allí se muestra la curva dinámica del compresor, elemento esencial del grupo de sistemas preventivos, la cual permite llevar a cabo acciones de control según la ubicación del punto de operación en una de las tres regiones de operación: Incondicionalmente estable (III), peligro (II) e inestable (I). También se presenta un resultado típico de las señales de los sensores y los marcadores utilizados por los métodos de respuesta a características incipientes del bombeo. Ambos utilizan principios diferentes para accionar la válvula de recirculación que garantizará un flujo mínimo cuando las condiciones del proceso inciten en el compresor el efecto bombeo. Las generalidades de los ACS, así como de los que responden a las características incipientes del efecto bombeo, haciendo énfasis en los métodos planteados como solución del problema que busca resolver esta tesis, serán ampliados a continuación.

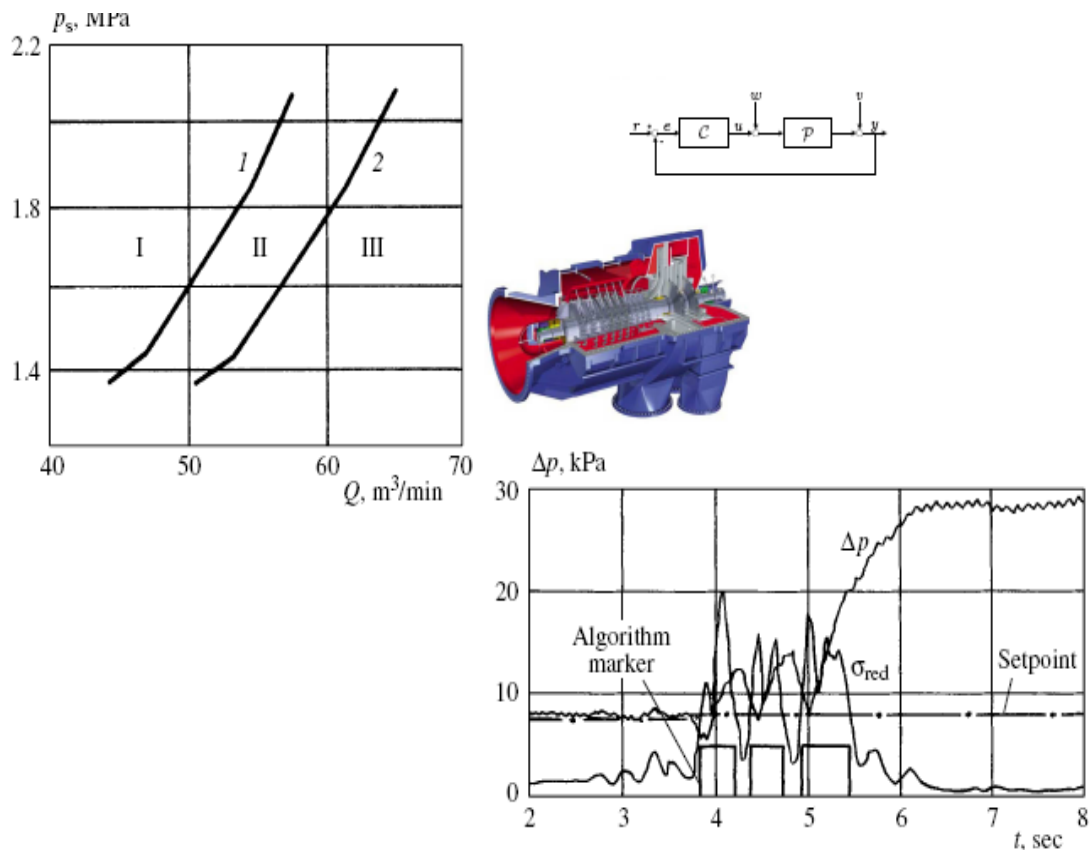


Figura 12. Métodos Anti efecto bombeo más utilizados. Fuente: autor

### 3.1. Sistemas preventivos

Utilizan la información de la instrumentación ubicada en el compresor (medición de temperatura, presión y flujo), para determinar el punto de operación. Realizan una comparación entre el punto de operación actual y el límite de bombeo, generando una señal de error. Un control tipo PID procesa dicha señal y desarrolla en respuesta las ordenes para la válvula de recirculación. Cuando ésta se abre, una parte del gas de descarga es regresado a la succión para garantizar el flujo mínimo, esperando que el compresor regrese a la operación normal, momento en el cual se puede cerrar la válvula nuevamente.

Los requerimientos más importantes para garantizar la efectividad de estos métodos según [9] son:



- Un modelo del límite de bombeo preciso. Debe predecir el límite, bajo las posibles condiciones de operación y características del gas.
- Un algoritmo de control apropiado. Debe garantizar que el compresor no ingrese en operación inestable sin que las acciones de control afecten el proceso del cual forma parte el compresor.
- La instrumentación adecuada. Los sensores deben ser seleccionados para cumplir los requerimientos de velocidad, rango de operación y precisión.
- Válvula de recirculación seleccionada de acuerdo al compresor y el proceso. Debe ajustarse al compresor, debe ser capaces de soportar cambios en la carga largos y rápidos, así como pequeños y lentos. Además la válvula debe tener la velocidad y volumen suficientes para garantizar que no se supere el límite de bombeo durante el apagado. El sistema de ductos es el factor dominante en los tiempos de respuesta del sistema, razón por la cual deben ser analizados y entendidos antes de la selección de este componente.

En la figura 13 se muestra un esquema de control típico, donde el filtro (*scrubber*) y el enfriador (*cooler*) son opcionales.

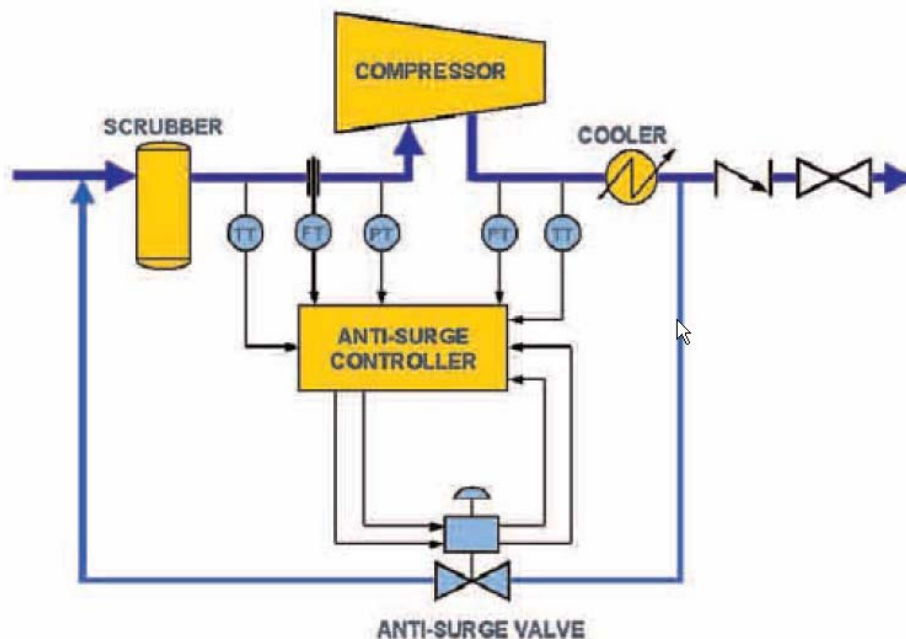


Figura 13. Esquema típico anti efecto bombeo. Fuente: Tomado de [9]



La operación de un compresor se encuentra determinada por tres parámetros: presión, velocidad y flujo. La relación entre el punto de operación y el efecto de bombeo puede ser determinada por cualquier dos de estos parámetros, tal y como se muestra en la figura 14. En las dos graficas a la izquierda se observa la relación con respecto a la velocidad, la cual se ve afectada fuertemente por variaciones de la composición del gas, debido al cambio en el número mach. En la tercera se observa una curva donde las características del gas son menos significativas, la cual se obtiene a partir de mediciones de presión diferencial a través del compresor y presión en el medidor. Esta curva es proporcionada por el fabricante del compresor o bien puede ser obtenida a través de mediciones teórico-experimentales.

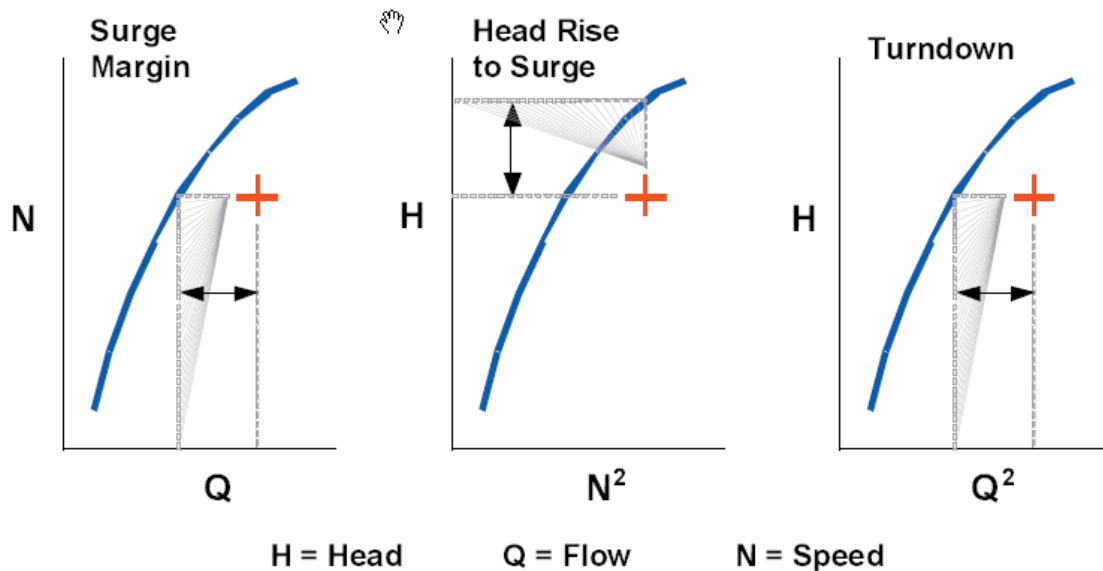


Figura 14. Limite de bombeo bajo diversos sistemas de referencia.

Fuente: Tomado de [9]

Como se había mencionado antes, la caracterización precisa de esta curva es necesaria, sin embargo esto no garantiza completamente el éxito del esquema de control. Si se analizan las parámetros que influyen en el comportamiento de la figura 14, (ecu. 6), se llega a la conclusión de que son variables para las diferentes circunstancias que se puedan presentar durante el funcionamiento del compresor, razón por la cual se observan cambios en la posición de la curva más o menos pronunciados, de acuerdo a la magnitud del cambio de la variable y de la variable en si. Teniendo en cuenta este fenómeno se opta por establecer una línea de



control ubicada a la derecha de la curva de anti bombeo, que cree un margen de confianza lo suficientemente amplio para evitar la operación inestable del compresor. Dicho margen debe hacerse lo más estrecho posible a fin de obtener una máxima amplitud del rango de operación de la máquina.

La comparación entre el punto de operación de la máquina y la curva de anti bombeo, puede ser llevada a cabo por un micro controlador o en caso de que se requiera un algoritmo de control más complejo, por un computador.

### **3.2. Sistemas que responden a las características incipientes**

Los estudios realizados acerca de las características del efecto bombeo en su fase inicial han arrojado un patrón. Sin embargo, muchas de las características son opacadas por el ruido inherente a los sensores, convertidores y perturbaciones externas. La solución a este tipo de inconvenientes se puede encontrar en el campo de la teoría de la probabilidad, donde se utilizan métodos propios de dicha rama para la detección de señales en ambientes ruidosos cuyos comportamientos sean pseudoaleatorios. Es importante comprender algunos conceptos básicos de la teoría de la probabilidad a fin de analizar los alcances de los sistemas tratados en este numeral.

En primer lugar, un proceso aleatorio es aquel cuyas muestras pueden ser caracterizadas por variables aleatorias, donde cada una de ellas tiene un orden determinado en el proceso. Más formalmente y para señales discretas, Allan Oppenheim en [2] afirma: "Un proceso aleatorio es una familia indexada de variables aleatorias  $\{X_n\}$  caracterizada por un conjunto de funciones de distribución de probabilidad que en general, pueden ser función del índice  $n$ ...el índice  $n$  esta relacionado con la variable temporal". La importancia de los procesos aleatorios radica en el modelado de señales discretas como una secuencia muestra de dichos procesos.

La esperanza matemática de una variable aleatoria representa el valor esperado de dicha variable y esta definida por (ecu. 7), donde  $x_i$  es el valor de la variable  $x$  en el instante  $i$ -ésimo y  $p(x_i)$  es la probabilidad de que la variable  $x_i$  ocurra. Si la función  $p(x_i)$  es constante, la esperanza matemática se conoce como la media cuya definición se encuentra determinada por (ecu. 8).



$$E(x) = \sum_{t=0}^{N-1} x_t \cdot p(x_t) \quad (\text{ecu. 7})$$

$$\bar{x} = \frac{1}{N} \sum_{t=0}^{N-1} x_t \quad (\text{ecu. 8})$$

La dispersión hace referencia al distanciamiento que tiene una variable aleatoria respecto a su valor central. Un estimador de la dispersión es la varianza, la cual se define como el promedio de los cuadrados de las desviaciones respecto a la media (ecu.9).

$$\sigma_x^2 = \frac{1}{N} \sum_{t=0}^{N-1} (x_t - \bar{x})^2 \quad (\text{ecu. 9})$$

La función de auto correlación, es la correlación cruzada de una variable aleatoria con una versión desplazada de si misma. Mide la dependencia entre valores de un proceso aleatorio en instantes de tiempo diferentes y se encuentra definida en (ecu. 10). Una propiedad importante de esta función establece que su transformada de Fourier, es la densidad espectral de potencia de la señal.

$$R_{xx}(m) = \frac{1}{N-m} \sum_{t=0}^{N-m-1} x_t \cdot x_{t+m} \quad (\text{ecu. 10})$$

Es en este marco conceptual, donde surgen varios métodos basados en el procesamiento estadístico de señales, provenientes de los sensores que monitorean los parámetros de funcionamiento del compresor. Se van a mencionar los dos métodos más pertinentes para el desarrollo de la solución propuesta en este trabajo de grado.

### 3.2.1. Método del cálculo de la función de auto correlación.

Se basa en el estudio de las características del efecto de bombeo incipiente a través del cálculo de la función de correlación de la señal de los sensores. Los valores descritos a continuación son propios de la aplicación descrita en [4].



El procedimiento es el siguiente:

- Se realizan agrupan los datos históricos del sensor en intervalos de tiempo de un minuto.
- Los datos acumulados en cada intervalo se dividen en 648 secciones.
- A cada sección se le calcula la función de autocorrelación (ecu. 10).
- La condición de detección se basa en el establecimiento de un valor máximo de la función de autocorrelación por intervalo, cuya gráfica en tiempo arroje una función parecida al coseno.

### 3.2.2. Método del cálculo de la dispersión de la señal.

Este método ha arrojado los mejores resultados para la detección del fenómeno de interés y se basa en el cálculo de la esperanza matemática y la varianza de la señal. De igual manera los valores enunciados a continuación son propios de la aplicación descrita en [4].

El procedimiento es el siguiente:

- Los datos recopilados en los históricos de la maquina se dividen en  $j$  secciones.
- Cada sección debe poseer una cantidad de valores de la cantidad medida, tal que el tiempo transcurrido entre el primero y el último no exceda los .2 segundos, garantizando de esta manera la precisión requerida.
- Se calcula la media y la varianza de cada sección.
- El valor calculado se asigna al instante de tiempo correspondiente al final de la sección.
- Se introduce un cierto valor de umbral ( $\sigma_{\text{thresh}}^2$ ) el cual es utilizado como condición de detección. Cuando el nivel de dispersión, mostrado en (ecu. 11), supere el valor de umbral se determina el inicio del efecto bombeo.

$$\sigma_{red}^2 = k \cdot \frac{\sigma_x^2}{\bar{x}^2} \quad (\text{ecu. 11})$$

Es necesario resaltar la importancia de escoger adecuadamente la frecuencia de muestreo de los sensores así como de la cantidad de puntos para el cálculo de la dispersión a fin de obtener resultados precisos y rápidos, pues existe un compromiso entre estos dos parámetros: A medida que se reducen el número de puntos a fin de realizar un cálculo de la dispersión más veloz, se afecta la efectividad del método ya que aumenta la probabilidad de falsas detecciones a



causa del ruido y las perturbaciones, si por el contrario se aumenta el número de puntos para mejorar la efectividad del método se reduce la rapidez de detección. En este orden de ideas, ubicar el punto de operación óptimo se traduce en una tarea de prueba y error donde el objetivo final es balancear los parámetros de acuerdo a las condiciones de la máquina.

Entre las ventajas del algoritmo se pueden enunciar:

- Versatilidad. Presenta resultados igualmente satisfactorios en diversos tipos de pruebas y compresores.
- Alta confiabilidad y eficiencia en la detección del fenómeno.
- Tolerancia a perturbaciones y ruido.
- Ausencia de falsas señales de detección.
- Potencial para la implementación práctica en tiempo real utilizando PLC modernos.

En la figura 15 se muestran algunos resultados obtenidos por el método en la aplicación enunciada en [4], donde la escogencia de la señal de detección y los valores de los sensores son propios de la aplicación.

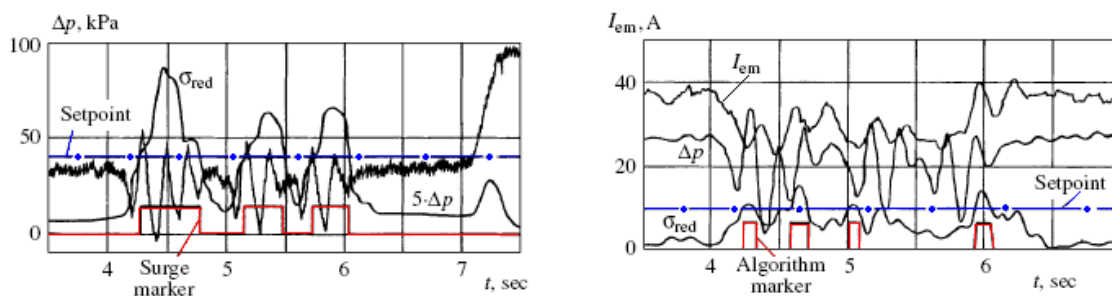


Figura 15. Resultados del algoritmo para dos variables diferentes.

Fuente: Tomado de [4] modificado.

En la parte izquierda de la figura 15 se observa una gráfica del diferencial de presión en la máquina contra el tiempo. De igual manera, se muestran la señal  $\sigma_{red}^2$ , el setpoint y los marcadores donde el algoritmo detectó la presencia del fenómeno. De igual manera, en la parte derecha se observan los resultados para la corriente en el manejador eléctrico del compresor. Lo importante es observar como ante la aleatoriedad de la señal, el algoritmo reconoce posibles instantes de inicio del fenómeno del bombeo. Además, cabe resaltar la versatilidad del algoritmo para detectar las características del efecto de bombeo en diversos tipos



de sensores, diferentes a los de presión, como por ejemplo, en la corriente del manejador eléctrico.

## **CAPITULO 4.**

### **RESULTADOS**

En capítulos anteriores, se establecieron las bases para la implementación de una solución para la detección incipiente del fenómeno del bombeo. Dadas las condiciones del problema y las herramientas con las que se cuenta para solucionarlo, se estructuró una estrategia, basada en los históricos de la máquina y los métodos probabilísticos que se han estudiado.

En primer lugar se analizó la información disponible sobre la máquina y el fenómeno, posteriormente se desarrolló el código para la programación de los algoritmos de detección y automatización de las pruebas, seguidamente se dieron ajustes al algoritmo y finalmente se analizaron los resultados.

#### **4.1. Desarrollo del software.**

En cuanto al código, se utilizó la plataforma Matlab © en la versión 7, de la compañía Mathworks, dada la rapidez, confiabilidad y versatilidad a la hora de procesar datos. Sin embargo, se trató en lo posible de llevar a cabo una programación muy básica, carente de toda la sofisticación de las funciones preconcebidas en el programa, con miras a ser implementada, en etapas futuras de este proyecto, en un dispositivo embebido cuyo lenguaje de programación sea compatible o muy similar al que se desarrolló en este trabajo de grado. Ahora bien, no basta con tener compilados los algoritmos, es necesario visualizar en un entorno amable, sencillo y confiable, los resultados que paso a paso se van suscitando, siendo además de utilidad, una herramienta software para el procesamiento offline de los datos del compresor. Es esta la razón para la creación de una interfaz gráfica, la cual fue desarrollada con ayuda de la herramienta GUIDE (Graphical User Interface Development Environment) incorporada en Matlab.

##### **4.1.1. Algoritmo de cálculo de la función de Autocorrelación.**

De acuerdo con la definición de la función (ecu. 10), se requiere conocer las muestras de la señal y una versión desplazada de ella, para una longitud de sección establecida. En otras palabras, la función solo requiere como parámetro de entrada la señal que se va a procesar, cuyas muestras se encuentran almacenadas en forma de vector y donde la posición de la muestra esta relacionada con la variable temporal. La información sobre la longitud de la señal, se puede calcular a



partir de la longitud del vector, valor que se asigna a la variable N, según la nomenclatura definida en (ecu. 10) y que a su vez debe ser la longitud de la función de autocorrelación. Ahora bien, se necesita realizar operaciones con muestras posteriores de la señal, por ende, se recurre al indexado propio del vector de muestras. Finalmente, se debe establecer el rango de variación de m, tal que el algoritmo no solicite muestras que no se encuentran disponibles de la señal:  $0 \leq m \leq N-1$ , siendo m entero.

En este orden de ideas, se plantea y programa en la función llamada "aucorr" (ver anexo A), el diagrama de flujo mostrado en la figura 16, para dar un orden de ejecución de los procesos y obtener la función como la función de autocorrelación como variable de salida.

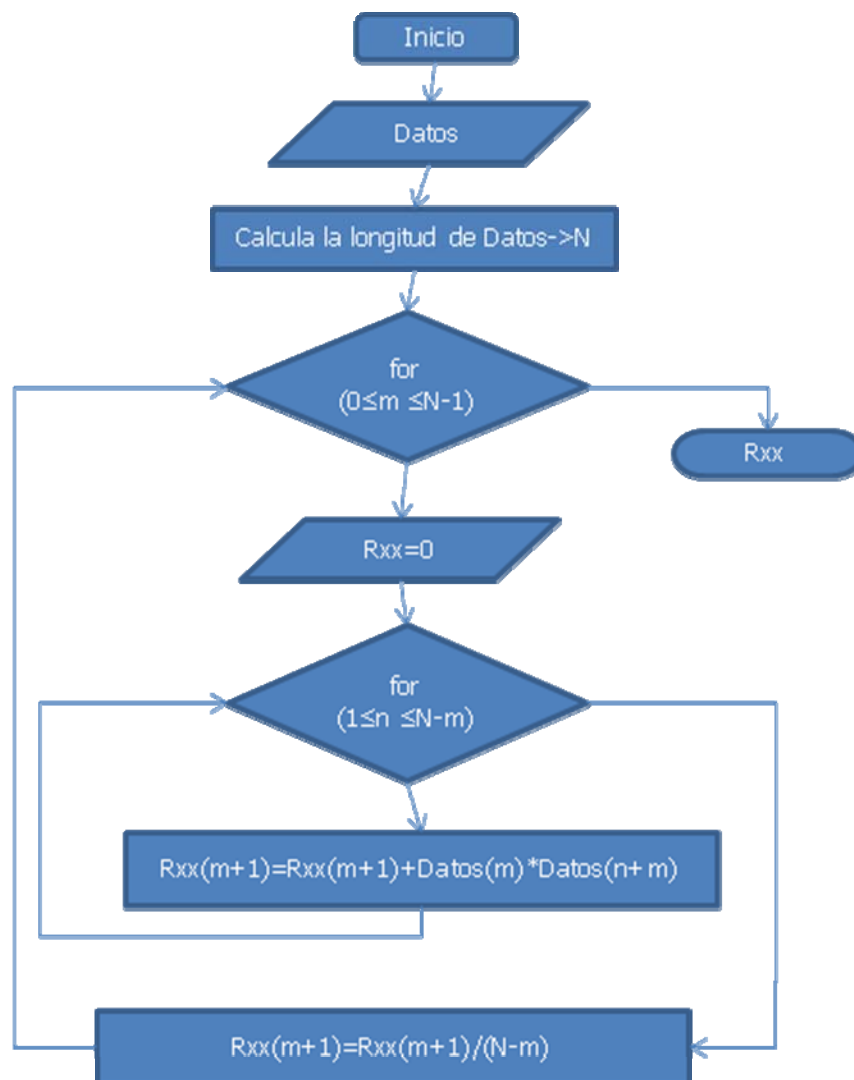




Figura 16. Diagrama de flujo de la función de autocorrelación. Fuente:autor

Con la función de autocorrelación programada, el paso a seguir era definir una función que permitirá escoger el número de secciones en que se iban a dividir los datos. Es así como surge "secciones" cuyos parámetros de entrada son la totalidad de los datos y el número de puntos por sección. Se aclara, que no se ingresan el número de secciones para evitar problemas relacionados con número de puntos por sección no enteros. La función selecciona la cantidad de muestras consecutivas, determinada por la variable de entrada, garantizando que todas las secciones tengan la misma longitud y en caso de que la última sección resulte más corta, no la procesa. Luego, llama a la función de autocorrelación, se la aplica a los datos seleccionados y guarda el resultado en la variable de salida. Este proceso se repite consecutivamente hasta abarcar la totalidad de los datos. El diagrama de flujo se presenta en la figura 17 y su codificación en el anexo B.

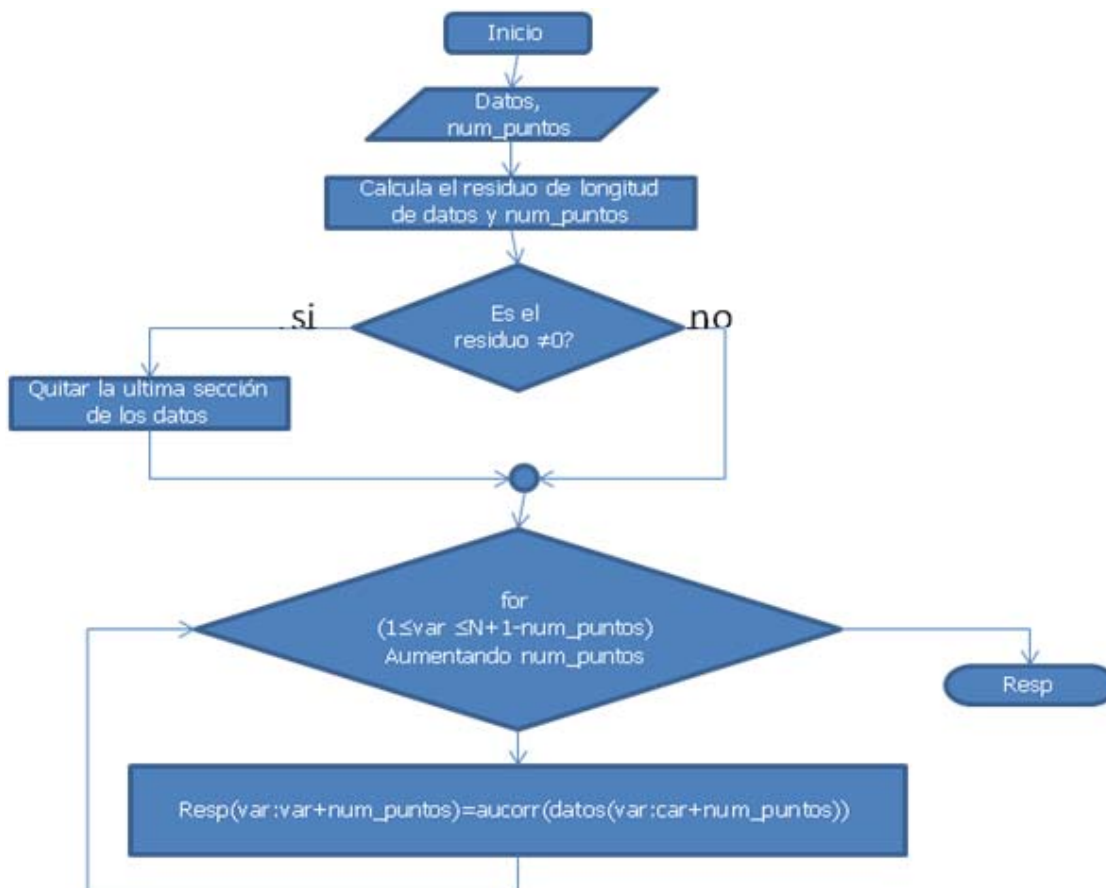




Figura 17. Diagrama de flujo del método de la función de autocorrelación  
Fuente: autor

Nota: Las demás características del método, tales como la comparación con los valores de umbral se codificaron dentro las GUI desarrolladas para la interfaz gráfica.

#### 4.1.3. Algoritmo de cálculo del nivel de dispersión.

De acuerdo con la definición presentada en (ecu. 11), el nivel de dispersión requiere el cálculo de la varianza y de la media para cada intervalo de datos. Es por esto que se plantean dos funciones que calculan estos dos parámetros para los datos que se le estipulen como entrada: "media" y "dispersión".

La función media, tal y como su nombre lo indica, calcula el promedio de los datos definidos como entrada aplicando (ecu. 8). Con ayuda de un bucle *for*, obtiene los valores de cada una de las posiciones del vector de datos, las suma y finalmente, divide por el número total de muestras. El diagrama de flujo se muestra en la figura 18, mientras que el código de la función se puede hallar en el anexo C.

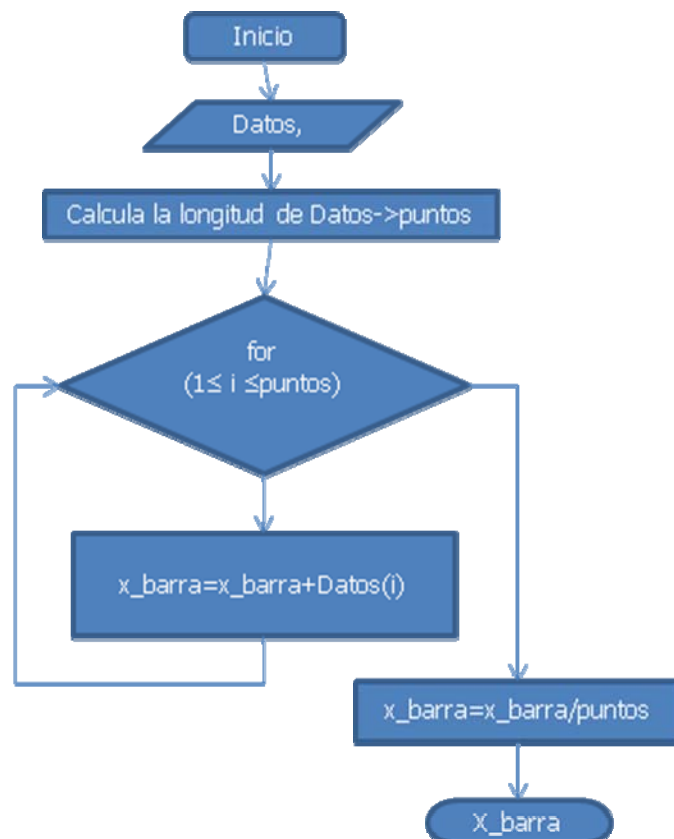




Figura 18. Diagrama de flujo de la función media. Fuente: autor.

De igual manera que la función media y según (ecu. 9), se diseña el algoritmo para el cálculo de la varianza. Las entradas serían en este caso, los datos y la media de los mismos, y el procesamiento consiste en calcular dentro de una estructura *for*, la diferencia de cada dato respecto a la media y elevar dicha diferencia al cuadrado, dentro de una variable acumulativa. Este procedimiento se encuentra ilustrado en el diagrama de flujo de la figura 19 cuyo código se presenta en el anexo D.

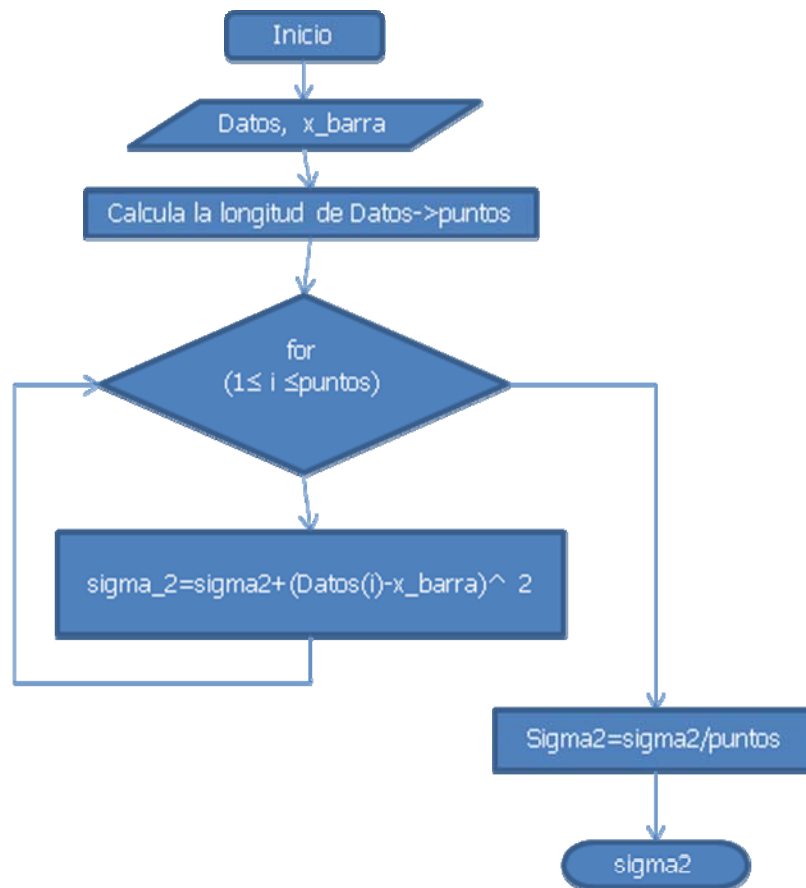


Figura 19. Diagrama de flujo de la función varianza. Fuente: autor.

Con todas las variables del método a disposición, se procede a estructurar los pasos del algoritmo dentro del diagrama de flujo. En primer lugar, se garantiza que el número de datos en los cuales se dividen los datos genere divisiones iguales, quitando los datos que resulten insuficientes para conformar la división final. Luego se selecciona dentro del total de datos la primera sección a la cual se le halla la media, la varianza y el nivel de dispersión, guardando el resultado en la



variable de salida. Se aclara que el factor de normalización mencionado en (ecu. 11) se hace igual a 1 por practicidad. Implementando el ciclo *for*, se realiza la misma operación para las demás secciones, procesando el total de datos. La longitud de la variable de salida se comprimirá por el factor del número de puntos por división utilizado en el algoritmo. Por ejemplo, si la longitud del vector de datos es de 2700 y el número de puntos por sección es de 10, la longitud del vector de salida será 270. Esto implica que se debe realizar una expansión de las muestras para contrastar los datos con los resultados. Este proceso se ilustra en la figura 20, y su respectivo código se plasma en el anexo E.

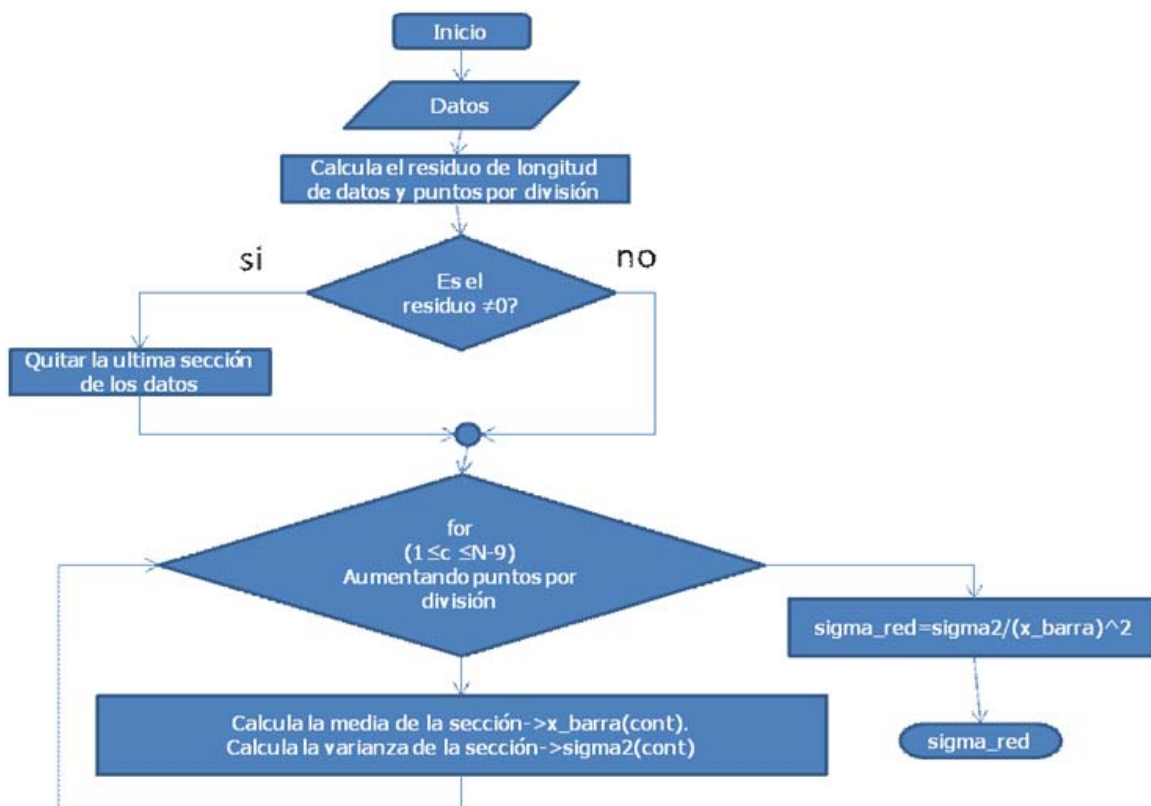


Figura 20. Diagrama de flujo del algoritmo del cálculo del nivel de dispersión.

Fuente: autor.

Nota: Las demás características del método, tales como la comparación con los valores de umbral y la expansión temporal del vector de salida para a fin de realizar su gráfica, se codificaron dentro las GUI desarrolladas para la interfaz.

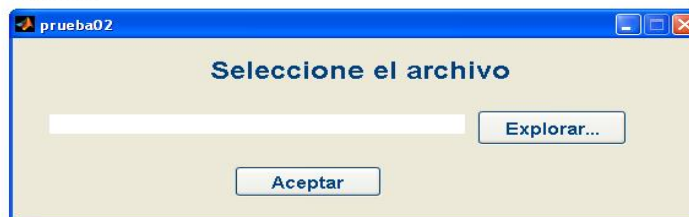
#### 4.1.3. Interfaz Gráfica

En primera instancia, el entorno grafico brinda la opción de cargar el archivo donde se encuentran los históricos, mediante la exploración de carpetas hasta la



ubicación del archivo de interés. Es necesario que el archivo a cargar posea una extensión .xls, es decir, que se haya creado con una hoja de cálculo compatible con este formato. Una vez se ha seleccionado el paquete de datos históricos a cargar, se pulsa aceptar y el programa solicita al usuario que seleccione los datos, dentro del paquete, que va a utilizar. Cabe resaltar que, si bien, carga los datos de todos los sensores, en una instancia posterior, se presenta la opción de escoger dentro de todos los sensores, aquel sobre el cual se va a trabajar. Cuando el usuario completa la selección, estos son cargados por Matlab y puestos a disposición para los algoritmos. Esta serie de pasos se observa en la figura 21.

**Paso 1.**

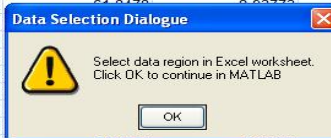


**Paso 2.**



**Paso 3.**

	A	B	C	D	E	F	G
1	TIME	FC-PDI-503,VALUE	FC-FI-515,VALUE	FC-ZI-500R,VALUE	FC-ZI-5003,VALUE	FC-ZI-5002,VALUE	FC-ZI-5001,VALUE
2	01-02-2007 00:00:00	31,7069	72051.6	61,8429	-2,93773	0,416606	0,121337
3	01-02-2007 00:00:03	31,7077	72050.4	61,8417	-2,93773	0,41685	0,123352
4	01-02-2007 00:00:06	31,7091	72052.7	61,8427	-2,93773	0,415385	0,125855
5	01-02-2007 00:00:09	31,7105	72055.1	61,8437	-2,93773	0,41392	0,128358
6	01-02-2007 00:00:12	31,712	72057.4	61,8447	-2,93773	0,412454	0,130861
7	01-02-2007 00:00:15	31,7134	72059.7	61,8457	-2,93773	0,410989	0,133364
8	01-02-2007 00:00:18	31,7148	72062	61,8468	-2,93773	0,409524	0,135867
9	01-02-2007 00:00:21	31,7162	72064.4	61,8478	-2,93773	0,408059	0,13837
10	01-02-2007 00:00:24	31,7176	72066.7	61,8488	-2,93773	0,406594	0,140873
11	01-02-2007 00:00:27	31,719	72069	61,8498	-2,93773	0,405128	0,143376
12	01-02-2007 00:00:30	31,7204	72071.3	61,8508	-2,93773	0,403663	0,145879
13	01-02-2007 00:00:33	31,7218	72073.6	61,8518	-2,93773	0,402198	0,148382
14	01-02-2007 00:00:36	31,7232	72076	61,8528	-2,93773	0,400733	0,150885
15	01-02-2007 00:00:39	31,7246	72078.3	61,8538	-2,93773	0,399267	0,153388
16	01-02-2007 00:00:42	31,726	72080.6	61,8548	-2,93773	0,397802	0,155891
17	01-02-2007 00:00:45	31,7274	72082.9	61,8558	-2,93773	0,396337	0,158394
18	01-02-2007 00:00:48	31,7288	72085.3	61,857	-2,93773	0,394872	0,160898
19	01-02-2007 00:00:51	31,7302	72087.6	61,858	-2,93773	0,393407	0,163401
20	01-02-2007 00:00:54	31,7316	72089.9	61,859	-2,93773	0,391941	0,165904
21	01-02-2007 00:00:57	31,733	72092.2	61,86	-2,93773	0,390476	0,168407



**Paso 4.**





Figura 21. Pasos para cargar los datos históricos al programa. Fuente: autor.

Cuando se recibe la confirmación de la carga exitosa de los datos, el programa cierra la ventana de adquisición y le ofrece al usuario el entorno de procesamiento de los datos, tal y como se observa en la figura 22.

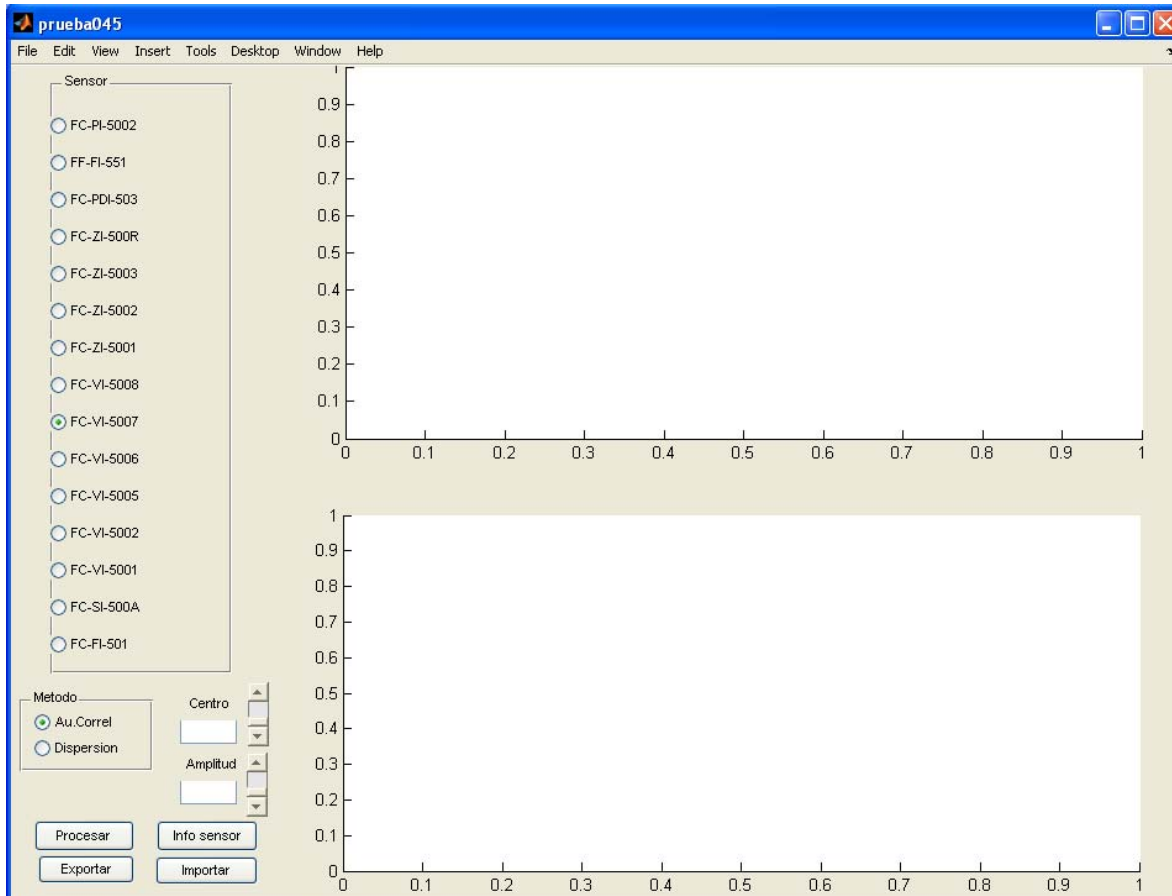


Figura 22. Entorno de procesamiento de datos. Fuente: autor.

Esta ventana presenta varias opciones para el usuario. En la parte superior izquierda se encuentra el panel de sensores, en el cual se muestran activos aquellos sensores cuyos datos fueron cargados en el programa. La selección de uno u otro sensor es excluyente, es decir, solo se procesan los datos de un sensor por vez. Como en principio cuesta trabajo familiarizarse con las referencias de los sensores, en la parte inferior izquierda existe un botón denominado "Info Sensor" cuya función es precisamente, indicarle al usuario en donde se encuentra ubicado y que variable mide el instrumento de medida seleccionado en el panel. Debajo de éste, se encuentra el panel de selección del algoritmo a implementar para el procesamiento de los datos. Igual que sucede con los sensores, para los algoritmos, la selección es excluyente y solo se puede aplicar un algoritmo por vez.



A la derecha del panel de selección del método se encuentra el área de "set point", lugar donde el usuario puede ingresar un valor de referencia para la detección del fenómeno, tal y como lo indican los algoritmos, bien sea escribiendo el valor en el recuadro o deslizando la barra hasta observar en el recuadro el valor deseado. En el extremo derecho de la pantalla se encuentran ubicados dos pares de ejes coordenados, los cuales tienen por objetivo presentar de forma gráfica los resultados del procesamiento: En el superior, la gráfica de los valores del sensor para diversas muestras cuyo orden está relacionado con la variable temporal; en la parte inferior, la gráfica del valor de la función de autocorrelación o del nivel de dispersión según se escoja, del setpoint y los marcadores de detección del fenómeno del bombeo. En la parte inferior, junto al botón "info sensor", se encuentra ubicado el botón "procesar", el cual después de haber seleccionado los parámetros antes mencionados, da inicio al procesamiento de los datos y presentación gráfica de resultados. Además, la ventana cuenta con herramientas de Matlab para la visualización, tales como zoom, data cursor, grid, rotación, entre otras. Un ejemplo de lo que el usuario puede obtener por resultado se observa en la figura 23.

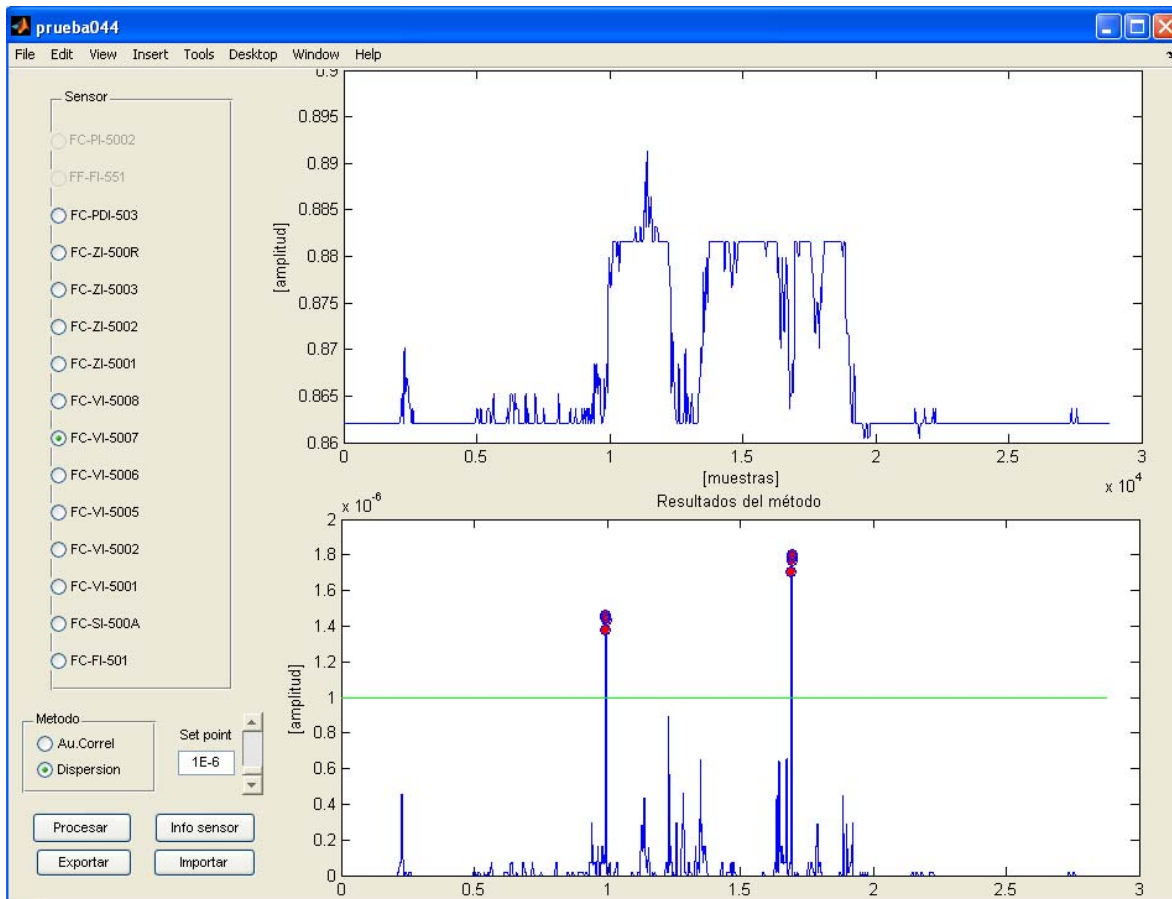




Figura 23. Resultado típico del entorno de procesamiento. Fuente: autor.

Finalmente, en la parte inferior izquierda se encuentran ubicados los botones "Exportar" e "importar". Con el primero se puede crear una figura con los datos arrojados por el proceso a fin de permitir su almacenamiento o edición. Con el segundo, se cierra la ventana actual y se muestra la interfaz de adquisición de datos históricos. De esta manera, el usuario puede procesar los datos históricos de la instrumentación del compresor de forma semiautomática con control de algunos parámetros del algoritmo.

Todo esto es posible gracias al diseño, configuración y programación de dos GUI (Graphical user Interface), la primera para cargar los datos y la segunda para procesarlos. No es la intención de este trabajo realizar una experimentación profunda de GUIDE, razón por la cual, se dará una explicación somera sobre el funcionamiento de las interfaces, omitiendo muchos detalles sobre la creación de las mismas.

Para la interfaz de carga de datos se utilizaron dos botones (explorar y aceptar), dos textos estáticos ("seleccione el archivo" y "pulse aceptar y espere por favor...") y un texto editable (La barra blanca donde aparece la ruta del archivo). Sobre este cuadro se hacía importante deshabilitar el texto "pulse aceptar y espere por favor..." al cargar la pantalla y permitir su visualización después de haber ingresado una ruta. Es por esto que se modificaron las propiedades del texto haciéndolo invisible y cambiando este estado después de que se presiona el botón de explorar.

Una vez es presentada la pantalla inicial, el programa queda en espera de acciones por parte del usuario, sin embargo esta diseñado de forma tal que la acción posible es la búsqueda del archivo a cargar, pues en caso de que decida pulsar el botón aceptar, este activará un mensaje indicándole la acción adecuada que debe realizar. Cuando se pulsa explorar, se activa el *callback* de dicho botón, en el cual se recurre a la función *uigetfile* para abrir una venta de exploración del sistema operativo, que permite la ubicación del archivo; esta ubicación es almacenada en el programa para su uso posterior. Si el programa identifica que la ruta es no vacía procede a mostrar un mensaje que le propone al usuario utilizar el botón de aceptar para continuar con el proceso. Oprimir dicho botón activa su *callback*, el cual comprueba que no haya sido oprimido antes de presionar "explorar". A continuación, utiliza el comando *xlsread* junto con la ruta almacenada, para abrir los datos en la hoja de cálculo. Allí el usuario es cuestionado acerca de los datos que desea que sean cargados, a lo cual el debe responder seleccionando aquellos



de su interés. Es importante que allí escoja los que desee, sin omitir las columnas relacionadas con la variable temporal y la fila de los títulos del o los sensores cuyos datos desea analizar. Se aconseja escoger todos los datos disponibles, pues el tiempo invertido en adquirirlos, es inferior al tiempo empleado en volver a cargar toda la interfaz, el número de veces que corresponden al número de sensores. Cuando se sienta seguro de que su selección es correcta, debe retornar a la ventana de carga de datos para presionar el botón ok, tal y como se lo indica la ventana de mensaje. Estos pasos se encuentran ilustrados en la figura 21, y constituyen los procesos realizables en esta primera GUI. Para mayor información se consigna el código de la interfaz en el anexo F.

La segunda interfaz se carga automáticamente tras la obtención exitosa de los datos. En ella se procesa la información de tal forma que se distinguen los datos correspondientes a cada sensor y el instante de tiempo que corresponde a cada muestra. Con estos datos y antes de que el usuario pueda notarlo, se habilitan en el panel de botones las opciones correspondientes a los sensores de los cuales se posee información, dejando deshabilitados aquellos que no. Este procedimiento se lleva a cabo mediante un *case* que compara, los encabezados previamente introducidos en el programa, con aquellos que vienen incluidos en los datos cargados. Tras este proceso, la interfaz queda en espera de las órdenes del usuario, las cuales deben ser:

- Seleccionar el sensor del cual desea procesar la información.
- Escoger el método para el procesamiento.
- Ingresar el valor del set point para la detección.
- Pulsar procesar para visualizar en el par de ejes coordenados los resultados.

En términos generales, al pulsar "procesar", la interfaz guarda la información de las opciones seleccionadas por el usuario, selecciona los datos correspondientes al sensor dentro del global, sigue las opciones correspondientes al método seleccionado y finalmente grafica, los datos, las funciones, el set point y los marcadores de detección. La selección de datos se lleva a cabo con una variable índice que permite identificar la columna que se debe seleccionar para trabajar, copiándola en la variable *datos\_utiles*. El selector del método guarda en una variable el valor que distingue un método u otro, cuya utilización se escoge a través de la estructura *if*. En caso de escoger la función de autocorrelación, el programa aplica el método a la variable *datos\_utiles* obteniendo la función como resultado. Se calcula su longitud y se genera la variable temporal necesaria para realizar su gráfica. Posteriormente se genera un vector con el valor del set point y de la misma longitud de la función de autocorrelación, que será comparado



muestra a muestra a fin de establecer los valores que se encuentran por fuera del rango. Para terminar, se realiza la grafica del setpoint y de las muestras detectadas fuera del intervalo del mismo. Si por el contrario, el método seleccionado es el de la dispersión, el programa aplicará a los datos, la función de dispersión y con su resultado, generará los valores de variable temporal adecuados para realizar su gráfica. De nuevo se llena el vector con el set point, se compara muestra a muestra con los valores de la función y se establecen los valores por encima del rango, para finalmente ser graficados junto con el valor del setpoint. En la fase final del proceso se realiza la gráfica de los datos.

Las otras tres funcionalidades, "info sensor", "exportar" e "Importar", buscan ayudar al usuario. La primera compara mediante un *case*, el sensor seleccionado, con la información de los sensores previamente guardada en el programa, dando por resultado una ventana que indica la información del sensor solicitada. La segunda funcionalidad, hace una copia de los resultados del método en una nueva figura, la cual se haya externa a la interfaz, desde donde se puede modificar y guardar si es el interés del usuario. Esto es posible gracias a la función *copyobj* que permite tomar todos los valores del par de ejes coordenados y pegarlos en una nueva figura. La tercera ejecuta la sentencia *close all*, lo que cierra todas las ventanas abiertas, y a seguir llama a la interfaz de carga de datos.

Esta descripción del procedimiento busca simplificar el funcionamiento, y detalles, como la lectura de la barra deslizante del set point, o la lectura del valor en el cuadro de texto del mismo, se omitieron por simplicidad. Para mayor claridad, se adjunta el código utilizado para la programación en el anexo G.

#### **4.2. Pruebas Realizadas y análisis de resultados**

En primer lugar se comprobó el correcto funcionamiento de la programación, tanto de los algoritmos como de las interfaces. Para esto se ingresaron datos aleatorios a las funciones y se contrastó con la función correspondiente implementada en Matlab. Allí se presentó un inconveniente, relacionado con los estimadores utilizados en Matlab, pues diferían en algunos términos, razón por la cual, los resultados también eran diferentes. Por citar un ejemplo, el algoritmo de la varianza se basa en (ecu. 9), donde la sumatoria se divide por N, sin embargo, la función del software realiza la estimación de la misma sumatoria pero la divide por el factor N-1. Algo similar, sucede con la función de autocorrelación cuando se compara con su correspondiente *autocorr*, donde las sumatorias, en un caso y en otro, poseen diferentes factores de escala. La alternativa que se propuso, fue el seguimiento paso a paso del programa y compararlo con el cálculo manual de los



valores de una señal aleatoria. Estos resultados corroboraron la efectividad del código programado, en cuanto al cálculo correcto de las variables internas y por ende, las de salida. Además, se realizaron ajustes que garantizaban cierta inmunidad a errores del usuario. Por ejemplo, se garantizó que la longitud de todas las secciones en las cuales se dividen la totalidad de los datos, tengan la misma cantidad de muestras. De igual manera, se protegió la integridad de las funciones haciéndolas transparentes para el usuario, presentándole solo los resultados de las mismas.

En cuanto a la interfaz gráfica se corroboró el correcto funcionamiento de las operaciones básicas, dedicando especial atención, a aquellas posibles secuencias de órdenes por parte del usuario, que generaran un funcionamiento inapropiado del programa. Todo paso inadecuado dentro del esquema general de funcionamiento, se trataba de enmendar mostrando, mediante un mensaje, el paso correcto a fin de obtener el resultado. A pesar de que se corrigieron y ajustaron varias situaciones inadecuadas, el programa no se encuentra exento de algunas otras, que puedan por en riesgo la integridad del mismo o de los resultados, las cuales sólo se podrán detectar y corregir a medida que su uso se haga más frecuente.

Una vez se ha establecido la confiabilidad de los resultados y de los procedimientos, se llevan a cabo, pruebas que buscan experimentar el desempeño de los métodos de detección.

#### **4.2.1. Método basado en el cálculo de la función de autocorrelación**

Con respecto a éste método se experimento en primer lugar con el bloque de datos correspondientes a la falla, buscando esclarecer un poco más la información sobre la condición de detección. Es por esto que se calculan los máximos de cada sección, ubicándolos al final de la misma, para a través de un análisis de su gráfica, encontrar algún patrón de comportamiento como el estipulado en la teoría del método. La función es calculada para los siguientes parámetros:

Número de datos total=2700

Numero de secciones=270

Sensores utilizados= FC-VI-5001, FC-VI-5002, FC-ZI-5001, FC-ZI-5002, FC-SI-500A, FC-FI-501, FC-PDI-503, FC-ZI-500R.

Bloque de datos procesados=ICP-Datas3



Los resultados gráficos de esta experiencia se muestran en las figuras 24 para el caso de los cuatro primeros sensores y 25 para los cuatro últimos.

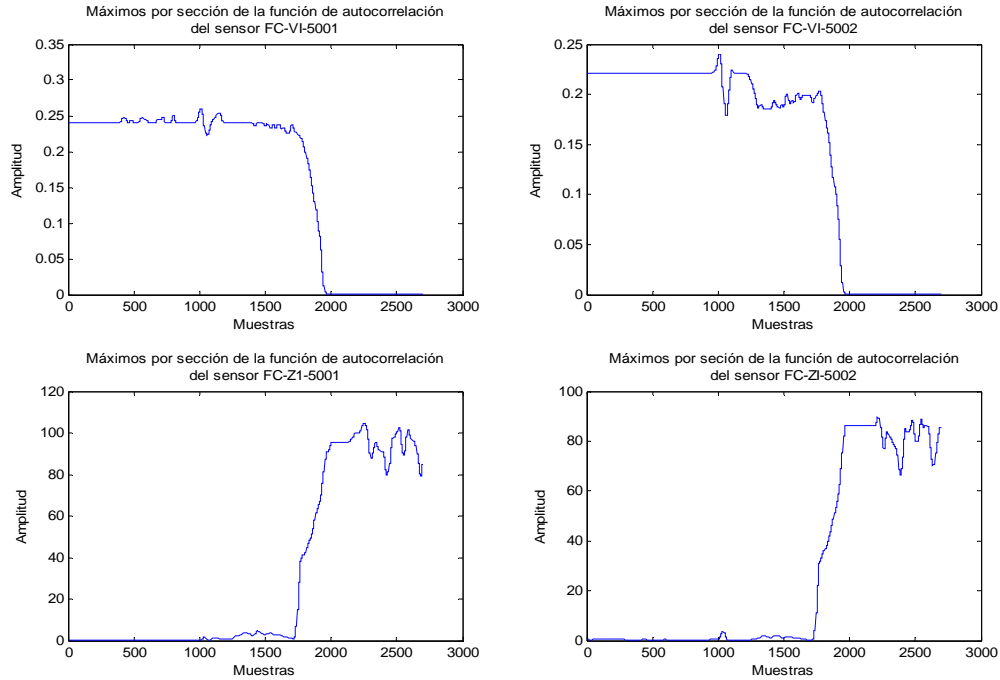


Figura 24. Resultados gráficos de aplicar el algoritmo basado en el cálculo de la autocorrelación, a los datos históricos del día del fallo. Fuente: autor.

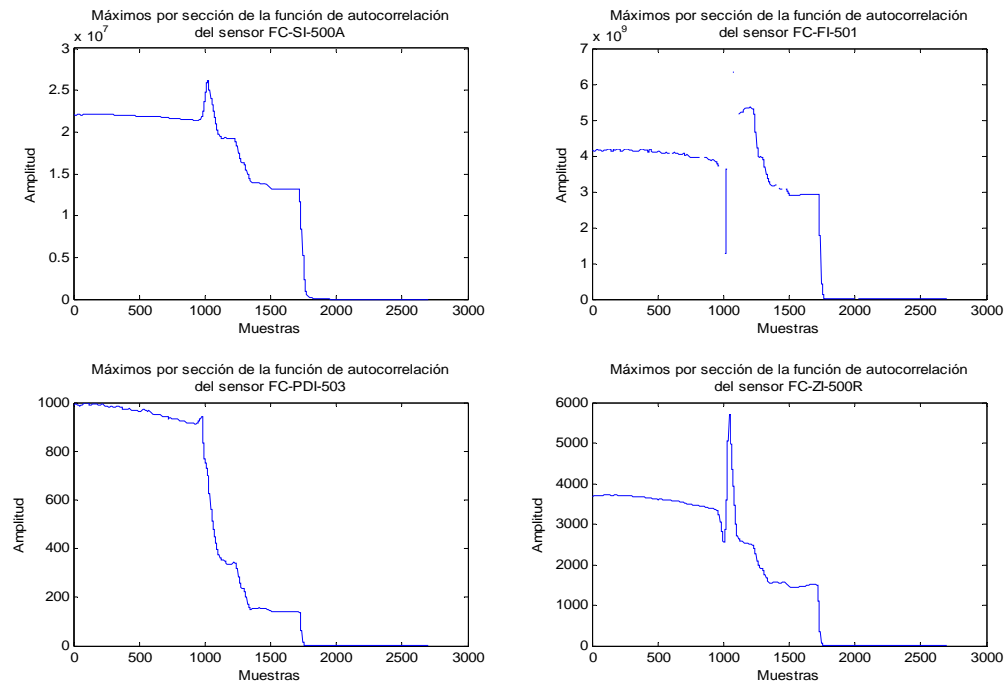




Figura 25. Resultados gráficos de aplicar el algoritmo basado en el cálculo de la autocorrelación, a los datos históricos del día del fallo. Fuente: autor.

Analizando estas gráficas cerca de la muestra 1000, que es el punto alrededor del cual surge el fenómeno, se pueden observar cambios pronunciados, tanto por encima como por debajo del valor promedio que venían presentando, pero no se alcanza a visualizar, por lo que mucho menos se puede presumir, la presencia de una forma cosenoidal de los datos alrededor de dicho punto. En otras palabras, la condición de detección del algoritmo no se hace evidente durante el procedimiento, y por ende la detección no se puede llevar a cabo.

La intención era detectar el patrón cosenoidal, aplicar algún algoritmo de coincidencia parcial, y establecer el origen del fenómeno. Sin embargo, la señal cosenoidal esperada, tendría una frecuencia de oscilación cercana la frecuencia del *surge*, razón por la cual, se debería contar con muestras que permitieran un espectro de análisis hasta de 30 Hz, lo que implica, poseer datos de los sensores cada 33.3 ms, un valor bastante lejano del real, que escasamente llega a los 0.5 Hz, o sea, 2 s. No obstante, se puede proponer una modificación al algoritmo, de tal forma que la condición de detección, se base en la detección de las variaciones fuera del rango de operación estable. Esta alternativa, busca aprovechar los picos de amplitud observados durante el análisis de las figuras 24 y 25 respectivamente. En este orden de ideas, se llevan a cabo una serie de pruebas a fin de determinar el mejor valor para el número de secciones en que se dividen los datos y el rango de variación de la función en estado estable a partir de los datos disponibles.

A fin de garantizar que el algoritmo se ajuste a lo descrito en [4], y teniendo en cuenta la cantidad y frecuencia de datos, se establece que durante un minuto se pueden observar 30 datos, por lo tanto el número de secciones para el bloque de datos debe ser 90. A partir de este valor se realiza una variación por encima y por debajo del número de secciones, tal y como se muestra en la figura 26, donde se presentan los resultados de aplicar el planteamiento anterior a uno de los sensores, para ilustrar el procedimiento. La escogencia del sensor se hizo de manera aleatoria y se comprobó con los demás, tal que las tendencias allí mostradas se pudieran generalizar.

Estos resultados indican una señal de detección más fuerte a medida que se aumenta el número de secciones, es decir, con un número más pequeño de muestras por sección. La gráfica correspondiente a la línea roja muestra picos más altos cerca del punto donde se presume el origen del fenómeno, comportamiento bastante deseable para aplicar la modificación planteada al método. Si embargo,



270 secciones implican un total de 10 muestras por sección, y tratar de aumentar este valor resulta poco práctico según la definición de la función, y las características de los datos. A partir de este resultado se evidencia con más claridad la necesidad de una frecuencia de muestreo de los datos mayor. No obstante, la intención es generar resultados con los elementos proporcionados, y en pro de este objetivo, se selecciona el valor de 10 muestras por sección para los procedimientos posteriores.

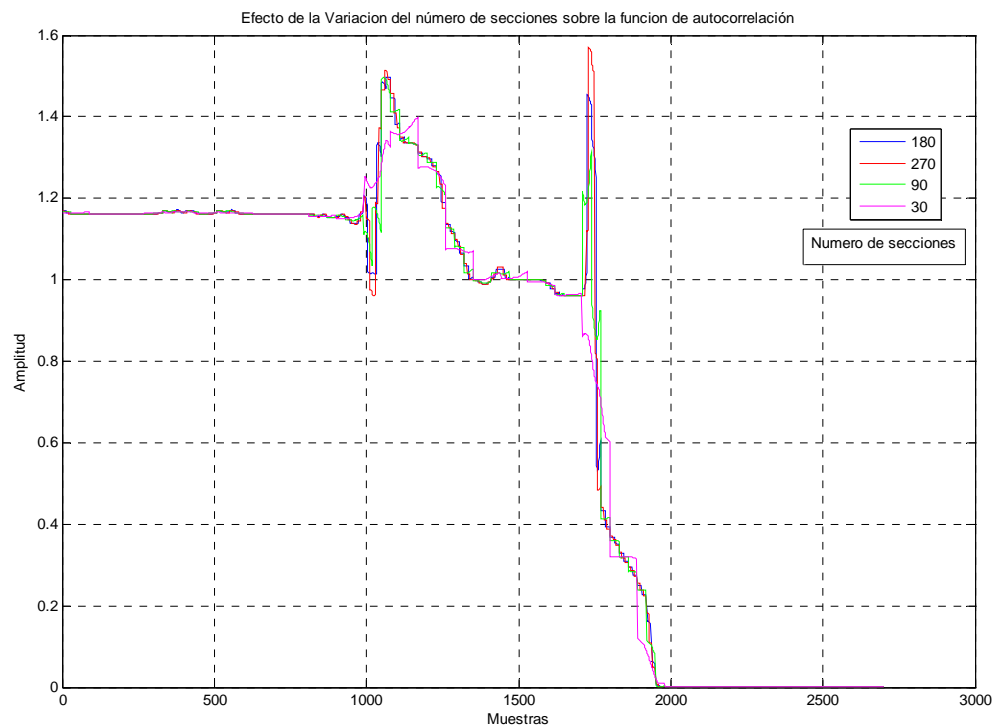


Figura 26. Efecto de la variación del número de secciones sobre la función de autocorrelación. Fuente: autor.

En la figura 27 se muestran los resultados de procesar las amplitudes máximas y mínimas de los datos históricos de un sensor, así como de la función de autocorrelación, en estado cuasi estable. La información contenida en cada bloque de datos, ejemplifica diversos modos de operación, razón por la cual, se debe tener presente el concepto de estado estable cuando se escoge el rango de análisis de los datos. Es decir, se deben considerar los máximos y mínimos de la función, así como de los datos, en intervalos donde las muestras presenten un comportamiento cuasi estable, para evitar introducir información que no aplica



para la situación. Los resultados de esta experiencia, con cada uno de los sensores disponibles, se encuentran consignados en la tabla I.

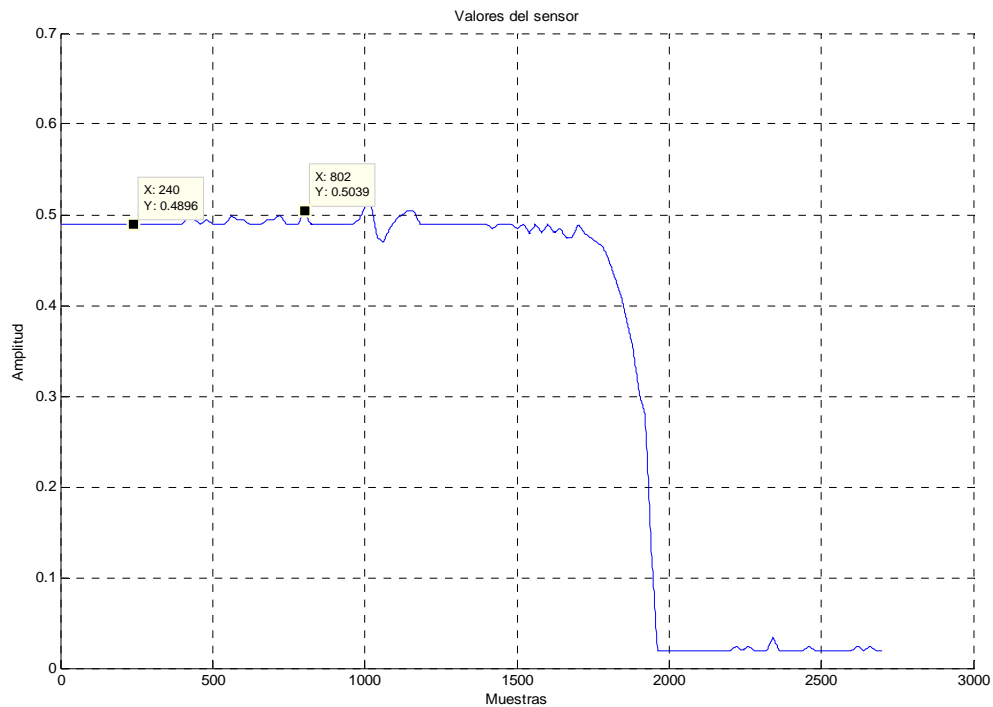
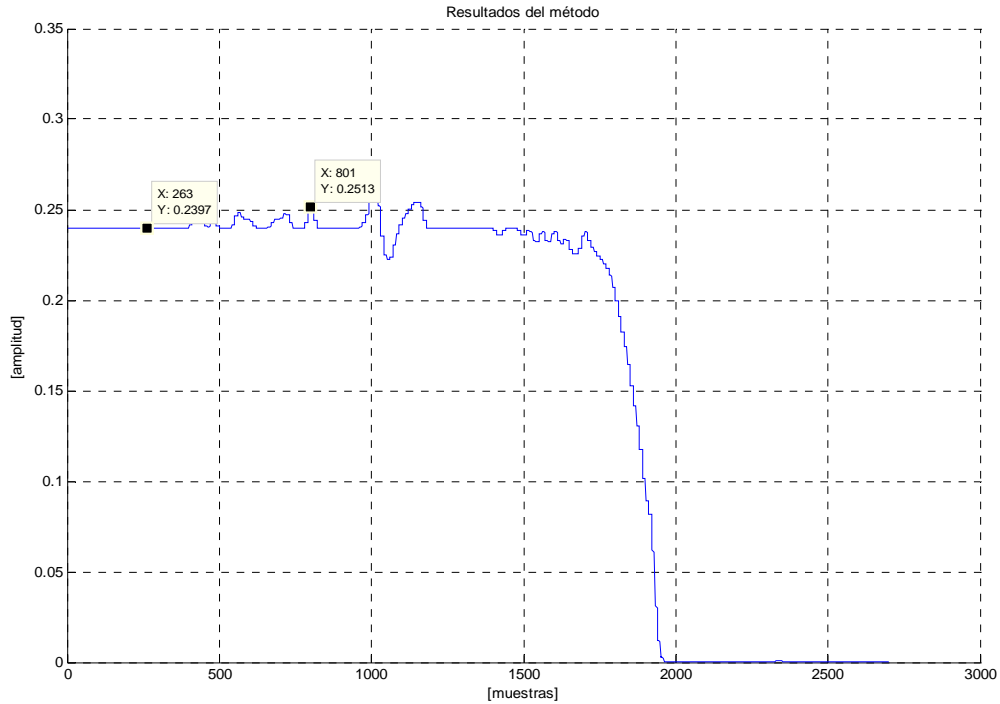




Figura 27. Ejemplo del proceso de cálculo del rango de variación de la función de autocorrelación y de los datos, en estado cuasi-estable. Fuente: autor.

Tabla I. Valores en estado estable de los datos y la F.A.C.

		Sensor								
Bloque		FC-VI-5001			FC-VI-5002			FC-VI-5005		
		Max	Min	$\Delta$	Max	Min	$\Delta$	Max	Min	$\Delta$
Datas2	Valor	0,5091	0,4896	0,0195	0,4896	0,4636	0,026	0,6862	0,6309	0,0553
	Función	0,2589	0,2397	0,0192	0,2397	0,2154	0,0243	0,4709	0,398	0,0729
Datas3	Valor	0,5039	0,4896	0,0143	0,4701	0,4701	0	0,652	0,6471	0,0049
	Función	0,2513	0,2397	0,0116	0,221	0,221	0	0,4444	0,4188	0,0256
Datas4	Valor	0,4896	0,4701	0,0195	0,4358	0,4116	0,0242	0,6861	0,6571	0,029
	Función	0,2397	0,221	0,0187	0,1977	0,1693	0,0284	0,4444	0,4344	0,01
		Sensor								
Bloque		FC-VI-5006			FC-VI-5007			FC-VI-5008		
		Max	Min	$\Delta$	Max	Min	$\Delta$	Max	Min	$\Delta$
Datas2	Valor	0,6471	0,5678	0,0793	0,8913	0,8604	0,0309	1,1563	1,0782	0,0781
	Función	0,4188	0,3225	0,0963	0,7936	0,7403	0,0533	1,337	1,163	0,174
Datas3	Valor	0,5921	0,5873	0,0048	0,862	0,8476	0,0144	1,083	1,078	0,005
	Función	0,3449	0,3449	0	0,7431	0,7231	0,02	1,171	1,152	0,019
Datas4	Valor	0,5678	0,5482	0,0196	0,9206	0,8816	0,039	1,098	1,078	0,02
	Función	0,3224	0,3006	0,0218	0,811	0,7772	0,0338	1,19	1,162	0,028
		Sensor								
Bloque		FC-ZI-5001			FC-ZI-5002					
		Max	Min	$\Delta$	Max	Min	$\Delta$			
Datas2	Valor	3,96E-01	-1,06E+00	1,46E+00	5,93E-01	-9,24E-01	1,52E+00			
	Función	1,12E+00	-1,92E-04	1,12E+00	8,24E-01	-4,85E-04	8,25E-01			
Datas3	Valor	1,16E-01	-1,80E-01	2,96E-01	8,23E-01	4,21E-01	4,03E-01			
	Función	2,37E-02	-2,92E-03	2,67E-02	6,28E-01	1,95E-01	4,33E-01			
Datas4	Valor	5,81E-01	-2,73E-01	8,54E-01	1,18E+00	5,95E-02	1,12E+00			
	Función	2,92E-01	-3,86E-03	2,96E-01	1,39E+00	3,39E-04	1,39E+00			
		Sensor								
Bloque		FC-ZI-5003			FC-SI-500A					
		Max	Min	$\Delta$	Max	Min	$\Delta$			
Datas2	Valor	- 1,53E+00	-3,64E+00	2,11E+00	4,79E+03	4,70E+03	9,66E+01			
	Función	1,32E+01	2,34E+00	1,09E+01	2,93E+07	2,21E+07	7,21E+06			
Datas3	Valor	-	-2,70E+00	2,34E-01	4,70E+03	4,63E+03	7,00E+01			



		2,47E+00					
	<b>Función</b>	7,31E+00	6,10E+00	1,21E+00	2,21E+07	2,14E+07	6,80E+05
Datas4	<b>Valor</b>	-3,89	-4,36E+00	4,69E-01	4,52E+03	4,26E+03	2,57E+02
	<b>Función</b>	1,90E+01	1,62E+01	2,80E+00	2,03E+07	1,81E+07	2,20E+06
<b>Sensor</b>							
<b>Bloque</b>		<b>FC-FI-501</b>			<b>FC-ZI-500R</b>		
		<b>Max</b>	<b>Min</b>	<b>Δ</b>	<b>Max</b>	<b>Min</b>	<b>Δ</b>
Datas2	<b>Valor</b>	6,51E+04	6,42E+04	8,16E+02	6,39E+01	6,14E+01	2,55E+00
	<b>Función</b>	4,23E+09	4,13E+09	9,30E+07	4,09E+03	3,77E+03	3,11E+02
Datas3	<b>Valor</b>	6,49E+04	6,35E+04	1,34E+03	6,10E+01	5,90E+01	2,01E+00
	<b>Función</b>	4,19E+09	3,96E+09	2,38E+08	3,72E+03	3,48E+03	2,44E+02
Datas4	<b>Valor</b>	6,29E+04	6,09E+04	1,97E+03	5,20E+01	4,91E+01	2,95E+00
	<b>Función</b>	3,95E+09	3,73E+09	2,19E+08	2,71E+03	2,41E+03	2,98E+02
<b>Sensor</b>							
<b>Bloque</b>		<b>FC-FI-501</b>			<b>FC-ZI-500R</b>		
		<b>Max</b>	<b>Min</b>	<b>Δ</b>	<b>Max</b>	<b>Min</b>	<b>Δ</b>
Datas2	<b>Valor</b>	6,51E+04	6,42E+04	8,16E+02	6,39E+01	6,14E+01	2,55E+00
	<b>Función</b>	4,23E+09	4,13E+09	9,30E+07	4,09E+03	3,77E+03	3,11E+02
Datas3	<b>Valor</b>	6,49E+04	6,35E+04	1,34E+03	6,10E+01	5,90E+01	2,01E+00
	<b>Función</b>	4,19E+09	3,96E+09	2,38E+08	3,72E+03	3,48E+03	2,44E+02
Datas4	<b>Valor</b>	6,29E+04	6,09E+04	1,97E+03	5,20E+01	4,91E+01	2,95E+00
	<b>Función</b>	3,95E+09	3,73E+09	2,19E+08	2,71E+03	2,41E+03	2,98E+02
<b>Sensor</b>							
<b>Bloque</b>		<b>FC-PDI-503</b>					
		<b>Max</b>	<b>Min</b>	<b>Δ</b>			
Datas2	<b>Valor</b>	3,19E+01	3,13E+01	6,30E-01			
	<b>Función</b>	1,02E+03	9,81E+02	3,94E+01			
Datas3	<b>Valor</b>	3,16E+01	3,10E+01	6,20E-01			
	<b>Función</b>	9,99E+02	9,12E+02	8,68E+01			
Datas4	<b>Valor</b>	2,84E+01	2,69E+01	1,43E+00			
	<b>Función</b>	5,77E+02	8,03E+02	-2,26E+02			

El análisis de estos resultados debe hacerse con cuidado, a fin de establecer el valor correcto para el rango de detección. Por tal motivo, se estudia cada sensor de forma independiente buscando la validez de los resultados y el establecimiento del valor de rango correcto, haciendo especial énfasis en los resultados obtenidos en Datas 4. La metodología consiste en encontrar el máximo rango de variación



de la función de autocorrelación, excluyendo aquellos valores disímiles que indiquen un estado de inestabilidad. El resultado serán los límites del rango de variación establecido, junto con, el centro y la amplitud del setpoint. Para garantizar una detección más holgada, es decir, no tan cercana a los valores normales de la función, se aconseja aumentar la amplitud del setpoint en un porcentaje. Haciendo un análisis minucioso de las variables presentadas, en la tabla I, se puede llegar a la conclusión de que los datos etiquetados como ICP-Datas4, muestran en su totalidad, una condición de operación inestable en gran parte del tiempo y que los valores finales no han alcanzado el funcionamiento nominal de la máquina, razón por la cual, no deben ser tenidos en cuenta a la hora de establecer los rangos de variación de la función de autocorrelación y por ende en los resultados de detección. En este orden de ideas, se obtienen los resultados mostrados en la tabla II.

Tabla II. Resultados estimados para el parámetro de detección.

<b>Sensor</b>	<b>Máximo</b>	<b>Mínimo</b>	<b>Centro</b>	<b>Amplitud</b>
<b>FC-VI-5001</b>	0,2589	0,2397	0,2493	0,0192
<b>FC-VI-5002</b>	0,2397	0,2154	0,22755	0,0243
<b>FC-VI-5005</b>	0,4709	0,398	0,43445	0,0729
<b>FC-VI-5006</b>	0,4188	0,3225	0,37065	0,0963
<b>FC-VI-5007</b>	0,7936	0,7231	0,75835	0,0705
<b>FC-VI-5008</b>	1,337	1,152	1,2445	0,185
<b>FC-ZI-5001</b>	1,12E+00	-2,92E-03	5,57E-01	1,12E+00
<b>FC-ZI-5002</b>	8,24E-01	-4,85E-04	4,12E-01	8,25E-01
<b>FC-ZI-5003</b>	1,32E+01	2,34E+00	7,79E+00	1,09E+01
<b>FC-SI-500A</b>	2,93E+07	2,14E+07	2,53E+07	7,85E+06
<b>FC-FI-501</b>	4,23E+09	3,96E+09	4,09E+09	2,70E+08
<b>FC-ZI-500R</b>	4,09E+03	3,48E+03	3,78E+03	6,09E+02
<b>FC-PDI-503</b>	1020	911,8	965,9	108,2

Estos resultados se ponen a prueba en la interfaz gráfica cargando los datos catalogados como ICP-Datas3, que corresponden al momento del fallo, aplicando el centro sugerido por la tabla II y aumentando en un 50% el valor de la amplitud del rango propuesto en dicha tabla. Como resultado se obtienen los diferentes puntos donde se realiza detección, tal y como se aprecia en la tabla III, de los



cuales, se extrae el primero de ellos como indicador del surgimiento del fenómeno. Un resultado escogido al azar es presentado en la figura 28 para ilustrar el procedimiento.

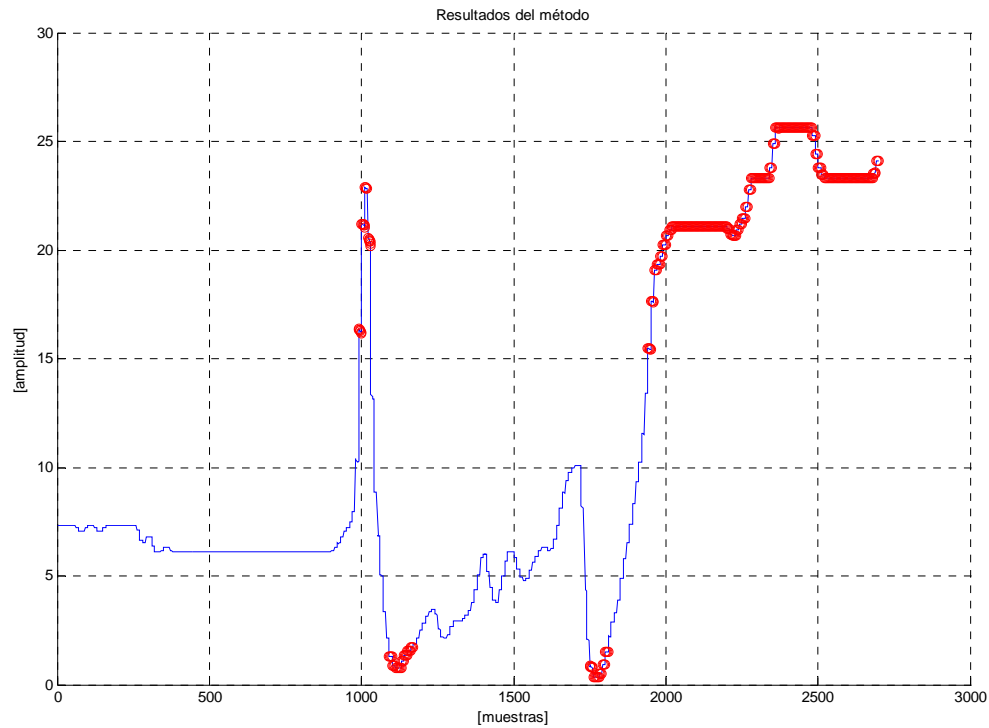


Figura 28. Resultado obtenido al procesar el sensor FC-ZI-5003 con el algoritmo de la función de autocorrelación.

Tabla III. Resultados de detección del fenómeno del bombeo mediante el método de la función de autocorrelación.

Sensor	Muestra	Centro	Amplitud	Sensor	Muestra	Centro	Amplitud
<b>FC-VI-5001</b>	1031	0,2493	0,02112	<b>FC-ZI-5002</b>	1011	0,412	0,9075
<b>FC-VI-5002</b>	1031	0,22755	0,02673	<b>FC-ZI-5003</b>	991	7,79	11,99
<b>FC-VI-5005</b>	1001	0,43445	0,08019	<b>FC-SI-500<sup>a</sup></b>	1091	2,53E+07	8,64E+06
<b>FC-VI-5006</b>	1001	0,37065	0,10593	<b>FC-FI-501</b>	881	4,09E+09	2,97E+08
<b>FC-VI-5007</b>	1001	0,75835	0,07755	<b>FC-ZI-500R</b>	831	3780	669,9



<b>FC-VI-5008</b>	951	1,2445	0,2035	<b>FC-PDI-503</b>	981	965,9	119,02
<b>FC-ZI-5001</b>	1031	0,557	1,232				

En estos resultados se observa la detección por parte de todos los sensores, sin embargo, difieren en el momento exacto del surgimiento del fenómeno. Esto se entiende desde el punto de la generalización las variables, pues la propuesta hecha como ajuste al método se basa en resultados globales de los sensores y no en los particulares. En otras palabras, se esta tomando como referencia para la detección valores de muestras en instantes de tiempo, que dada la aleatoriedad de la señal, no tienen ninguna relación. Esta observación apunta hacia la obtención de la condición de detección a partir de los valores que presenta el sensor en momentos previos a la inestabilidad, siendo esta, una justificación y buen preámbulo para el método de la dispersión.

Otra premisa, que se puede plantear a partir de estos resultados, consiste en comparar la detección tomando como referencia los demás sensores. Esto busca aprovechar la información arrojada por el algoritmo al aplicarlo a todos los instrumentos de medida, donde la detección temprana se pueda complementar con la detección posterior de otro sensor, y así sucesivamente, a fin de generar una señal de alarma confiable.

Como no hay una certeza respecto al momento en el que surge el fenómeno, en el siguiente numeral, se van a presentar las pruebas realizadas al método del cálculo del nivel de dispersión, para posteriormente contrastar los resultados de ambos métodos.

#### **4.2.2. Método basado en el cálculo del nivel de dispersión.**

Los lineamientos para el desarrollo del método impartidos por [4], mencionaban un total de muestras por sección, tal que la duración temporal no exceda los 0.2 segundos. Aplicando este criterio a los datos de la máquina con los que se cuenta, ni siquiera, una sección de longitud 1 (muestras) cumpliría con los requerimientos. Sin embargo, el criterio evidencia una necesidad de secciones no muy extensas, temporalmente hablando, lo que se traduce, en el caso del compresor estudiado, en un número de muestras por sección pequeño. Apoyado en los resultados obtenidos con el otro método aplicado, y entendiendo que la sección no se puede hacer tan pequeña como se quisiera, se utiliza una longitud de 10 muestras por sección, como parámetro del algoritmo.

Se aplica el método a cada uno de los sensores, tal y como se comentó en el numeral anterior, a fin de determinar los valores máximos de la función de



dispersión calculada, para instantes de tiempo donde el compresor se encontraba en estado cuasi-estable. En la figura 29 se muestra un ejemplo de dicho procedimiento, cuyos resultados se encuentran tabulados en la tabla IV.

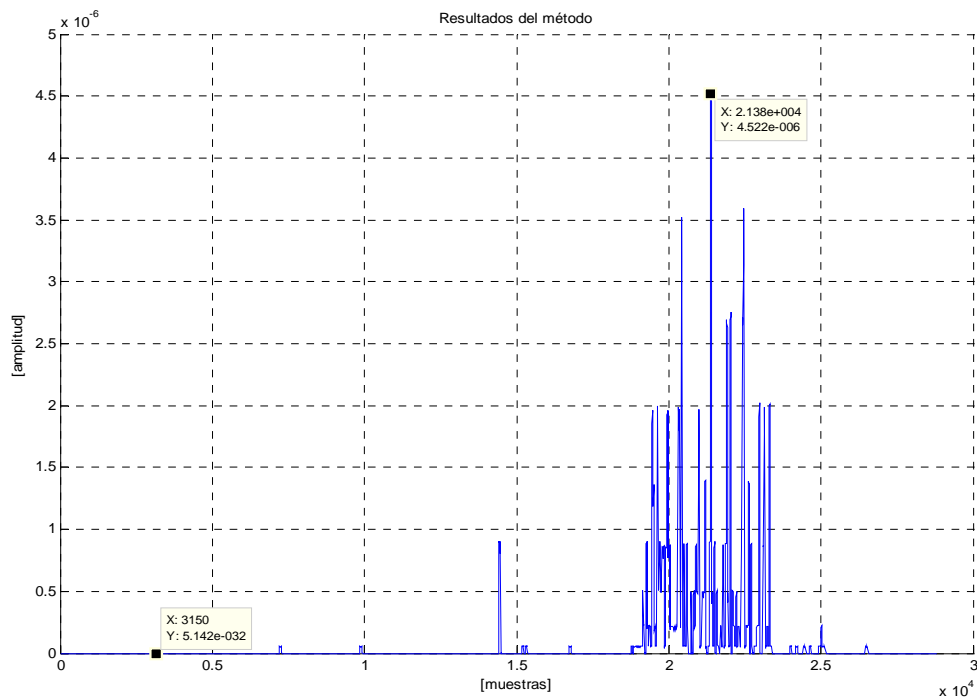
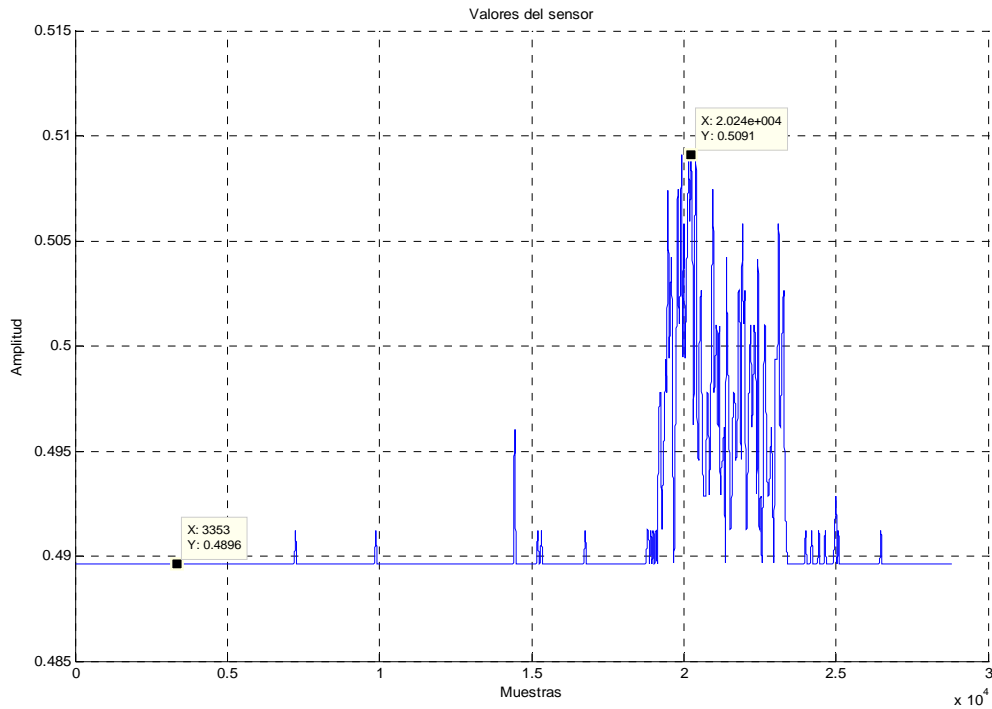




Figura 29. Cálculo de los valores extremos de la función de dispersión en estado cuasi-estable. Fuente: autor.

Tabla IV. Valores máximos de la Función de Dispersión en estado estable, para los sensores ubicados en el compresor.

Bloque	Sensor					
	FC-VI-5001	FC-VI-5002	FC-VI-5005	FC-VI-5006	FC-VI-5007	FC-VI-5008
Datas2	4,52E-06	2,98E-05	1,83E-06	2,77E-06	1,77E-06	5,68E-07
Datas3	1,81E-05	0,00E+00	4,66E-06	1,42E-06	6,02E-06	4,23E-07
Datas4	1,93E-05	9,95E-06	4,52E-06	1,43E-05	5,04E-06	1,67E-06
Bloque	Sensor					
	FC-ZI-5001	FC-ZI-5002	FC-ZI-5003	FC-SI-500A	FC-FI-501	FC-ZI-500R
Datas2	8,88E+02	2,41E+01	5,85E-05	6,26E-09	2,81E-06	3,45E-07
Datas3	1,29E+01	2,88E-03	1,00E-04	2,68E-08	1,62E-05	5,20E-07
Datas4	3,09E+04	3,81E-03	3,46E-05	2,80E-07	2,48E-05	9,86E-06
Bloque	Sensor					
	FC-PDI-503					
Datas2	3,04E-07					
Datas3	1,57E-06					
Datas4	3,60E-06					

Los valores presentados en la tabla están relacionados con tres situaciones de operación de la máquina, que son, operación normal, parada de emergencia y arranque. Se hace necesario, establecer de ellos, el valor de umbral adecuado para la detección, razón por la cual se estima el valor máximo de las tres situaciones entendido como, el peor valor que podría presentar el nivel de dispersión en estado cuasi-estable. Dichos valores se consignan en la tabla V.

Tal y como sucedió con el método anterior, los valores arrojados por el paquete de datos correspondientes a la arrancada, no brinda información adecuada sobre el estado estable de los sensores, puesto que, dicho estado no es alcanzado durante el periodo de obtención de los datos.

A fin de contrastar los valores sugeridos por la tabla V con la realidad, se aplica, con ayuda de la interfaz gráfica, el algoritmo a los datos que presumen la presencia del fenómeno del bombeo. Con el objetivo de brindar holgura a la detección y por ende a la operación del compresor, se toma un valor



porcentualmente mayor del valor de umbral (10%), tal y como se muestra en la tabla VI, donde se plasman los resultados de esta experiencia. Además, en la figura 30, se presenta un resultado gráfico del procedimiento descrito anteriormente.

Tabla V. Valores de umbral obtenidos durante el análisis en estado estable.

		Sensor					
		FC-VI-5001	FC-VI-5002	FC-VI-5005	FC-VI-5006	FC-VI-5007	FC-VI-5008
Valor de umbral		1,81E-05	2,98E-05	4,66E-06	2,77E-06	6,02E-06	5,68E-07
		Sensor					
		FC-ZI-5001	FC-ZI-5002	FC-ZI-5003	FC-SI-500A	FC-FI-501	FC-ZI-500R
Valor de umbral		8,88E+02	2,41E+01	1,00E-04	2,68E-08	1,62E-05	5,20E-07
		Sensor					
		FC-PDI-503					
Valor de umbral		1,57E-06					

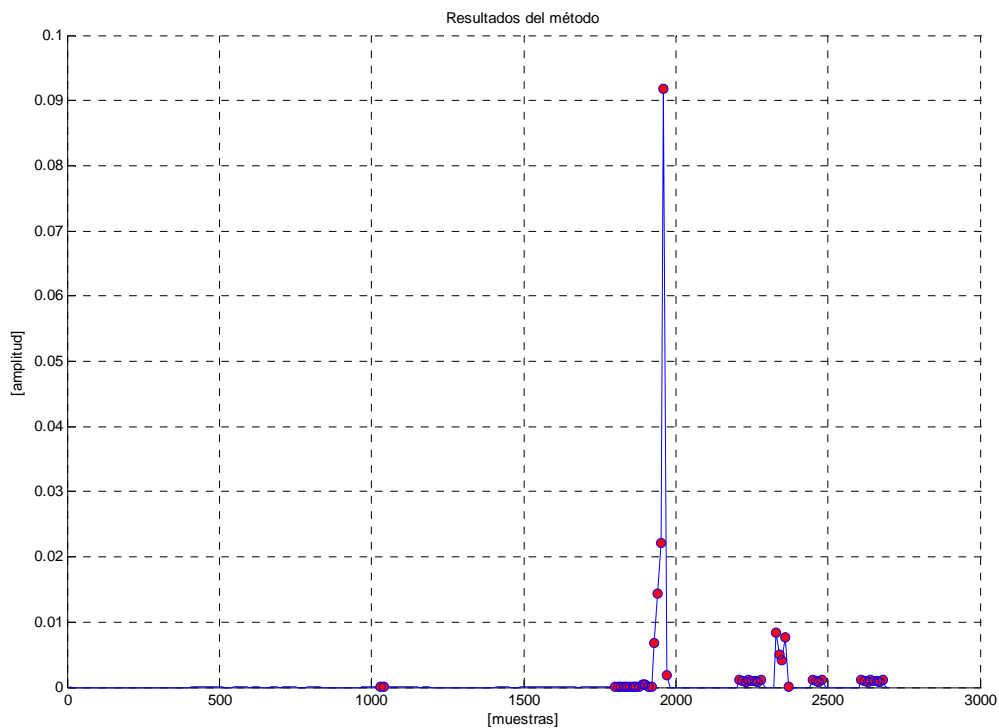




Figura 30. Resultado de procesar el sensor FC-VI 5001 con el algoritmo de la dispersión. Fuente: autor.

Tabla VI. Resultados de la detección del fenómeno del bombeo mediante el método del cálculo de la dispersión.

Sensor	Valor de umbral	Muestra	Sensor	Valor de umbral	Muestra
<b>FC-VI-5001</b>	1,99E-05	1030	<b>FC-ZI-5002</b>	26,48371	no detectó
<b>FC-VI-5002</b>	3,28E-05	1030	<b>FC-ZI-5003</b>	1,10E-04	990
<b>FC-VI-5005</b>	5,126E-06	990	<b>FC-SI-500A</b>	2,95E-08	970
<b>FC-VI-5006</b>	3,047E-06	970	<b>FC-FI-501</b>	1,78E-05	1020
<b>FC-VI-5007</b>	6,622E-06	1010	<b>FC-ZI-500R</b>	5,72E-07	950
<b>FC-VI-5008</b>	6,248E-07	950	<b>FC-PDI-503</b>	1,73E-06	970
<b>FC-ZI-5001</b>	976,767	no detectó			

En la tabla VI, se observan varias particularidades, una de ellas consiste en la no detección del efecto, por parte de dos, de los tres sensores de desplazamiento. Analizando las posibles causas de este hecho, se observó el alto valor de umbral determinado para dichos sensores, el cual no obedece a algún error de cálculo o programación, sino a la definición del índice de dispersión presentada en (ecu. 11). El parámetro en mención, corresponde a un cociente de dos variables, la varianza y la media, de donde se deduce que, a medida que la varianza se hace más grande el índice aumenta, que es el principio de detección del método, pero si la media se hace muy pequeña, el índice también aumenta. La media en el caso particular de los sensores FC-ZI-5001 Y FC-ZI-5002, se hace pequeña a causa de la bipolaridad de la medida del sensor, logrando de esta manera valores de índice de dispersión grandes por una causa diferente al efecto de bombeo. Este hecho no se ve reflejado en el sensor FC-ZI-5003, puesto que la medida no alcanza a hacerse negativa, algo que obedece a una casualidad y no a un buen funcionamiento del método para ese sensor. Estos hechos reflejan una desventaja del método, cuya posible solución puede consistir, en desplazar verticalmente una



cantidad adecuada, todos los valores del sensor durante el procesamiento, a fin de evitar los cruces por cero. Dicho desplazamiento tendría sentido únicamente para la detección, más no como análisis del comportamiento de la variable.

Otro hecho importante es la detección en diferentes instantes de tiempo. Esta situación se va a contrastar con los resultados obtenidos con el método anterior. Un primer dato para comparar es, cuanta variación hay en la detección de un método y otro, es decir, cual es tiempo entre la detección más temprana y la más tardía. Otro sería la variación de los tiempos dentro del conjunto, o sea, que tan dispersos están los tiempos en que ocurrió la detección y por último, la relación de similitud entre la ubicación del sensor y el tiempo en que se detecta el efecto bombeo. Estas y otras variables son comparadas en la tabla VII, donde se hace un intento por esbozar las diferencias encontradas, con base en la experiencia de procesamiento de los datos.

Tabla VII. Cuadro comparativo de los dos métodos.

<b>Parámetro</b>	<b>Método de la autocorrelación</b>	<b>Método de la dispersión</b>
Variación	1091-831=260 muestras =520 segundos.	1030-950= 80 muestras =160 segundos.
Dispersión	Se presenta una mayor concentración de los datos hacia la muestra 1000.	La concentración es mucho más fuerte hacia la muestra 1000.
Relación ubicación – detección	Los sensores ubicados en lugares similares, presentan detección en tiempos similares.	Los sensores ubicados en lugares similares, presentan detección en tiempos similares.
Amplitud del set point	La amplitud del set point es grande.	La amplitud del set point es pequeña.
Facilidad de detección	Es más sensible a variaciones del set point, pues la variación de la función es pequeña respecto al valor nominal.	Es más marcado el cambio de valores, cerca de la región de detección
Set point	La detección debe realizarse tanto por encima, como por debajo de un valor de referencia.	La detección solo se realiza por encima del valor de referencia. .
Complejidad	Complejidad baja, tiempo de proceso relativamente corto (sin contar el tiempo que se demora la interfaz en cargar los datos)	



## Capítulo 5.

### CONCLUSIONES

#### 5.1. Conclusiones.

Analizando la información compilada en este trabajo de grado, las experiencias, dificultades y soluciones planteadas a las mismas, se establecen las siguientes conclusiones:

- ☑ El método de cálculo del nivel de dispersión, como estrategia para la detección del fenómeno del bombeo, en el compresor de la unidad de cracking catalítico de la Refinería de Cartagena, brinda mejores resultados que su homólogo basado en el cálculo de la función de autocorrelación, bajo las condiciones de trabajo a las que fueron sometidos. Al decir mejores resultados, se hace referencia a la facilidad con que se puede llevar a cabo la detección, la menor dispersión del instante en que surge el fenómeno (Comparando las tablas VI y III donde se observa que las muestras detectadas como origen del fenómeno, se encuentran más cerca unas de otras en los resultados del método de la dispersión) y menor probabilidad de falsa detección.
- ☑ El fenómeno del bombeo se refleja en todas las variables medidas por los sensores, tal y como lo indican las tabla III y VI, o sea, en el diferencial de presión succión/descarga, en la velocidad de la turbina, en el desplazamiento axial y en la vibración, tanto en el eje de referencia x como y, de los acoples, la cola del compresor y el lado de la turbina. En consecuencia, las muestras de estos sensores, adquiridas con una frecuencia de muestreo baja, se hacen útiles en la implementación del mecanismo de detección e información de la presencia del fenómeno del bombeo en el compresor centrífugo estudiado.
- ☑ Recurriendo a estructuras computacionales básicas, tales como *for*, *case* e *if*, y al indexado propio de los vectores, se puede programar sin inconvenientes la lógica de procesamiento y detección de los algoritmos estudiados (ver anexos), característica de utilidad, para su implementación en un dispositivo embebido, que soporte condiciones de operación ruidosas como las que se presentan en el entorno de trabajo del compresor. Esto permite, el procesamiento y detección online del fenómeno del bombeo, siempre y cuando, se cumplan los requerimientos de frecuencia de muestreo y velocidad de procesamiento, establecidos por cada uno de los métodos, siendo una ventaja, el bajo consumo de memoria para la codificación de los algoritmos, dejando disponible, la mayor parte de dicho recurso para el procesamiento de los datos.



- ☑ Las exigencias del método de detección basado en el cálculo de la función de autocorrelación, superan las limitantes impuestas por los datos disponibles para el desarrollo de este trabajo, razón por la cual, no se observa el comportamiento descrito en la teoría para la detección (ver figuras 24 y 25). No obstante, la propuesta de utilizar una variación de dicha condición, a fin de utilizar las características observadas en la forma de onda de la función de autocorrelación, brinda resultados satisfactorios en cuanto a la detección del fenómeno, de acuerdo a los datos mostrados en la tabla III.
- ☑ Determinar el valor de umbral implementado en cada método (ver tabla I), en función de la variación de los valores de cada sensor durante operación nominal, resultó una estrategia válida, pues se detectó efectivamente el fenómeno (ver tablas III y VI), en los instantes esperados, antes de que el comportamiento del fenómeno alcanzará amplitudes desastrosas para la máquina.
- ☑ El método de cálculo de la función de autocorrelación, con la condición de detección modificada, exige rangos de comparación muy cercanos al valor nominal de la función, lo que dificulta el establecimiento oportuno del punto de origen del *surge*, obteniendo tiempos dentro de un rango más variable, que aquel calculado con el método de cálculo del nivel de dispersión.
- ☑ Existe una relación entre el tipo y ubicación de cada sensor, con el momento en el que detecta el fenómeno del bombeo (ver tabla III y VI), siendo más rápidos aquellos ubicados en la cola del compresor, cuando se comparan los diversos sensores de vibración analizados. Sin embargo, no se puede establecer una preferencia de tipo, es decir, afirmar que son mejores para la detección los sensores que miden una variable u otra, pues los estudios realizados, no aportan la información suficiente para determinar una tendencia.
- ☑ Es importante garantizar la operación del compresor en sus valores nominales, antes de activar la detección del efecto de bombeo, pues las condiciones de arranque de la planta implican una constante de tiempo del orden de horas antes de poder llevar al compresor a su operación normal, tiempo durante el cual, las muestras de los sensores presentan alta inestabilidad, afectando el correcto funcionamiento de los algoritmos, tal y como lo demuestran los valores en estado estable para el bloque Datas4 correspondiente a el arranque, con relación a los valores de Datas2 y Datas3, en la tabla I.
- ☑ En caso de variación de las condiciones nominales de operación del compresor centrífugo, se debe hacer una evaluación de las mismas antes de aplicar el algoritmo, pues dicha variación debe inducir un cambio en los valores de umbral para la detección.



## 5.2. Sugerencias

De acuerdo con la experiencia adquirida al procesar la información de los sensores, y gracias a la interacción detallada con los algoritmos de detección estudiados en este trabajo de grado, se plantean las siguientes sugerencias a fin de mejorar, complementar y desarrollar estrategias que brinden un mejor resultado:

- En lo posible, se debe aplicar el método de detección basado en el cálculo de la función de dispersión, a sensores cuyo rango de operación sea mayor que cero, o cuya variación entre valores positivos y negativos no sea muy frecuente, a fin de garantizar que el promedio de los datos arrojados por el sensor sea mayor que cero evitando una falsa detección.
- Se recomienda aumentar la frecuencia de muestreo de los datos adquiridos para el procesamiento, a fin de comprobar la tendencia establecida como criterio de selección del número de muestras por sección. Es decir, se requiere comprobar que los resultados para ambos métodos son mejores cuando se procesan muestras con una frecuencia de muestreo mayor.
- Es aconsejable llevar a cabo pruebas con máquinas cuyo estado es defectuoso, con el objetivo de comparar el efecto que causa la falla, en las variables medidas por los sensores y el ocasionado por el efecto de bombeo, para de este modo establecer, una situación real que puede afectar la confiabilidad del método de detección. Entiéndase falla como, desbalance del eje, fallas en los rodamientos, problemas relacionados con el manejador eléctrico.
- Para la implementación online del sistema de detección del fenómeno, son más adecuados los resultados obtenidos con el método de la dispersión.
- De acuerdo con el estudio del comportamiento del fenómeno en el compresor, y los resultados obtenidos con las variables medidas, se puede extender la detección, a otro tipo de sensores, como la corriente en el manejador eléctrico, la temperatura y la presión acústica.
- El orden tan pequeño de la señal de setpoint, así como de la función nivel de dispersión, puede tornarse en un inconveniente a la hora de implementar físicamente el esquema de detección, pues el ambiente ruidoso puede ocasionar variaciones de la señal de umbral, introduciendo errores al método. Es por esto que se recomienda modificar el factor de escala "k" descrito en (ecu. 11) para magnificar los resultados hasta un punto tal, que las pequeñas variaciones no afecten en gran medida, la magnitud de la señal.
- Es aconsejable brindar holgura al parámetro de detección a fin de permitir un mayor rango de operación del compresor. En el caso particular de este estudio se dio un 10% de holgura a los valores de umbral, debido a las limitantes



aportadas por los datos, pero en condiciones más favorables se puede extender dicha holgura evitando el detenimiento de la máquina ante falsas señales de alarma.

### 5.3. Evaluación de objetivos.

Ante el planteamiento del problema de identificar las características previas al efecto de bombeo mediante el análisis de datos offline, utilizando la autocorrelación y la dispersión, como técnicas de procesamiento digital de señales, se desarrollaron una serie de objetivos específicos que definían los lineamientos suficientes para satisfacer las expectativas del estudio realizado. A continuación se plantean dichos objetivos y se desarrolla una autoevaluación a fin de establecer el grado de cumplimiento de lo mismos y por ende el índice de satisfacción del trabajo desarrollado.

- *Desarrollar un algoritmo para el procesamiento offline de los datos, basado en el cálculo de autocorrelación y dispersión estadística.*

El desarrollo de los algoritmos se realizó teniendo en cuenta la referencia [4] y las condiciones establecidas por la frecuencia de muestreo, cantidad y tipo de datos históricos disponibles. Ambos inician dando un tratamiento a los datos con una serie de parámetros cuya justificación no se establecía claramente. No obstante, se realizaron pruebas con el objetivo de analizar dichos parámetros, dando por resultado, el ajuste de los valores hasta tal punto, que se obtuviera la situación más apropiada a pesar de las limitaciones impuestas por las muestras. Se hace referencia a las limitaciones, pues no se contaba con una frecuencia de muestreo ideal, aún así, se programan satisfactoriamente todos los pasos, previendo posibles situaciones de error introducidas por el usuario y esperando que dicha situación no atenuara el desempeño del método total. En la segunda parte, el algoritmo toma los resultados de las funciones y realiza la detección. De nuevo, existen inconvenientes con relación a la frecuencia de los datos, especialmente en el algoritmo de la función de autocorrelación, el cual requiere indispensablemente un espectro de análisis mayor al que se tenía para poder detectar las características incipientes del bombeo. A pesar de dicha situación, se plantea una modificación de la condición de detección del algoritmo, con miras a utilizar características interesantes, percibidas en la forma de onda de la función de autocorrelación. Cabe resaltar, el hecho de que se logró la detección con todos los sensores a los que se les aplicó la metodología, con una precisión bastante aceptable. En cuanto al algoritmo del



cálculo de la dispersión de la señal, los resultados fueron aún más prometedores. Finalmente, se mejoró la interacción de los algoritmos con el usuario, mediante la creación de una interfaz gráfica, que automatiza la mayor parte del proceso, presenta los resultados y facilita la comparación de los mismos con la señal del sensor. En este orden de ideas, se puede considerar cumplido el objetivo, pues se sobrepasaron las dificultades, y por si fuera poco, se creó una herramienta que puede ser utilizada en el futuro para complementar los análisis y conclusiones obtenidas.

- *Aplicar el algoritmo a los datos proporcionados por los sensores ubicados en la máquina, seleccionando en los históricos, momentos críticos donde la máquina presentó dicho fallo.*

Se procesó la información correspondiente a 13 sensores de diversos tipos (6 de vibración, 3 de desplazamiento, uno de velocidad, uno de presión diferencial, uno de posición y uno de carga), donde se caracterizaban tres modos de operación diferente: Operación nominal (estable), operación bajo el efecto bombeo, y arranque. Esta selección permitió vislumbrar las características de la máquina, desde la óptica que cada variable medida le imprimía, logrando así establecer una relación de detección entre los instrumentos de medida. Es decir, la escogencia de los datos facilitó el reconocimiento de los estados de operación, normales y anormales del compresor, logrando establecer relaciones de tipo y ubicación del sensor con la detección del inicio del fenómeno. Los resultados que dan fe de las pruebas realizadas, se encuentran plasmados en este documento, razón por la cual se puede sustentar el cumplimiento de este objetivo.

- *Contrastar los resultados del algoritmo con los datos históricos de los momentos donde se tiene conocimiento de la presencia del fenómeno como medida de la confiabilidad del algoritmo.*

De antemano se conocía la existencia del fenómeno en uno de los bloques de información disponible. De igual manera se sabía la ausencia del fenómeno en los otros dos paquetes de muestras de los sensores. Esta información debía ser verificada y corroborada con la aplicación de los algoritmos, los cuales en lo posible, debían detectar el fenómeno antes de que la variable presentara un comportamiento incontrolable. La comparación realizada permitió validar el método dada la precisión con que indicaba el origen del fenómeno, pero además, permitió obtener información que en la literatura consultada no se estipulaba, respecto a características y falencias. Tal y como se mencionó anteriormente, la posición, y el tipo de sensor influía en la detección del



momento de surgimiento del fenómeno, además se pudo establecer una condición necesaria para la aplicabilidad del algoritmo: Garantizar la operación en estado estacionario del compresor. Aunque esta situación es evidente en el contexto de la máquina, puesto que los sistemas de alarma relacionados al compresor, se activan o desactivan de acuerdo al modo de operación en el que el operario ubica a la máquina, no se hacía relevante en la teoría. Esta condición tan importante no sólo le da validez al método, sino que plantea limitantes del mismo, así como el establecimiento de estrategias que puedan llegar a solventar dicha situación. En este orden de ideas, se considera alcanzado éste objetivo, y más aún, se superan las expectativas concebidas para los resultados.

➤ *Plantear mejoras y sugerencias al método.*

En el capítulo 5 se realiza una realimentación de los resultados y experiencias adquiridas durante el estudio del fenómeno, siendo una de sus secciones el planteamiento de sugerencias. Las ideas allí plasmadas, dan a conocer realidades propias de los algoritmos, además, proponen posibles soluciones a los inconvenientes, y pruebas complementarias que enriquezcan y confirmen los resultados obtenidos. De esta manera se da por concluido este objetivo y se pone en manifiesto la satisfacción, no solo por los resultados, sino por las expectativas que estos plantean, ante la posibilidad de utilizar este estudio para ejecutar la detección online del fenómeno, en etapas posteriores.



## BIBLIOGRAFIA

- [1] Advant OCS product Guide, course A340.
- [2] Centrifugal & Axial compressor (pdf) consultado en [http://www.gepower.com/businesses/ge\\_oilandgas/en/literature/en/brochures\\_catalogues.htm](http://www.gepower.com/businesses/ge_oilandgas/en/literature/en/brochures_catalogues.htm) el día 2 de enero de 2008.
- [3] GREENE, Richard W., Guía para el uso de Compresores y Ventiladores, Mc Graw Hill
- [4] GUZEL'BAEV, Ya. Z., KHAVKIN, A. L., KHISAMEEV, I. G., Methods of Rotating Stall and Surge Detection in Centrifugal Compressors, Chemical and Petroleum Engineering, Vol. 42, Nos. 5–6, 2006.
- [5] HELVORIT, Jan van, Centrifugal Compressor surge modeling and identification for control, Tesis doctoral, Universidad Técnica de Eindhoven, 2007
- [6] MEULEMAN, Corina HJ, Measurement and Unsteady Flow Modelling of Centrifugal Compressor Surge, Tesis doctoral, Universidad Técnica de Eindhoven, 2002.
- [7] OPPENHEIM, Alan V., SCHAFER, Ronald W., BUCK, John R., Tratamiento de Señales en Tiempo Discreto, 2ª edición, Prentice Hall, 1999.
- [8] Rojas R. y Castaño J, Manual de cracking. , versión 01, 2005, coordinación de Cracking, Refinería de Cartagena.
- [9] WHITE, Robert C, KURZ, Rainer, Surge Avoidance for Centrifugal compressors, Solar Turbines Inc, Estados Unidos.



## ANEXOS

### Anexo A

#### Código de programación de la función de autocorrelación

```
function [Rxx]=aurcorr(data)
puntos=length(data);
for m=0:1:puntos-1
    Rxx(m+1)=0;    %crea el elemento del vector a llenar
    for n=1:1:puntos-m
        Rxx(m+1)=Rxx(m+1)+data(n).*data(n+m);    % Calcula la sumatoria como el valor ubicado
                                                    % en el elemento mas el nuevo producto.
    end
    Rxx(m+1)=Rxx(m+1)/(puntos-m);    %Divide el vector de acuerdo a la definición de la función.
end
```

### Anexo B

#### Código de programación del algoritmo de la función de autocorrelación

```
function [Resp]=secciones(datos, num_puntos) %Establece los datos de entrada y el número de
                                           puntos por sección
%calculo de la autocorrelación de la señal para un número de puntos
residuo=rem(length(datos),num_puntos);
if (residuo~=0)
    datos1=datos(1:length(datos)-residuo);
else
    datos1=datos;
end
for var00=1:num_puntos:length(datos1)+1-num_puntos;
    Resp(var00:var00+num_puntos-1)=aurcorr(datos1(var00:var00+num_puntos-1));
end
```

### Anexo C

#### Código de programación del algoritmo de la función media

```
function [x_barra]=media(datos)
puntos=length(datos);
x_barra=0;
for i=1:1:puntos
    x_barra=x_barra+datos(i);    %calcula la sumatoria de los datos
end
x_barra=x_barra./puntos;    %divide por el numero de datos
```



## Anexo D

### Código de programación del algoritmo de la función de autocorrelación

```
function [sigma2]=varianza(datos, x_barra)
puntos=length(datos);
sigma2=0;
for i=1:1:puntos
    sigma2=sigma2+(datos(i)-x_barra).^2; %calcula la sumatoria de las desviaciones respecto a la
    media
end
sigma2=sigma2/puntos; %divide por el numero de puntos
```

## Anexo E

### Código de programación del algoritmo del método del cálculo de la dispersión

```
function [sigma_red, x_barra, sigma_2]=metodo2(datos)
%calculo del promedio de la señal y su dispersión cada 10 puntos
residuo=rem(length(datos),10);
if (residuo~=0)
    datos1=datos(1:length(datos)-residuo)
else
    datos1=datos;
end
cont=1; %inicializa el contador
for c=1:10:length(datos1)-9
    x_barra(cont)=media(datos1(c:c+9));
    sigma_2(cont)=varianza(datos1(c:c+9),x_barra(cont));
    cont=cont+1;
end
sigma_red=sigma_2./(x_barra.^2);
```

## Anexo F

### Código de programación de la interfaz para la adquisición de los datos históricos

```
function varargout = prueba02(varargin)
% PRUEBA02 M-file for prueba02.fig
% PRUEBA02, by itself, creates a new PRUEBA02 or raises the existing
% singleton*.
%
% H = PRUEBA02 returns the handle to a new PRUEBA02 or the handle to
% the existing singleton*.
%
% PRUEBA02('CALLBACK',hObject,eventData,handles,...) calls the local
% function named CALLBACK in PRUEBA02.M with the given input arguments.
%
% PRUEBA02('Property','Value',...) creates a new PRUEBA02 or raises the
% existing singleton*. Starting from the left, property value pairs are
% applied to the GUI before prueba02_OpeningFunction gets called. An
% unrecognized property name or invalid value makes property application
% stop. All inputs are passed to prueba02_OpeningFcn via varargin.
```



```

%
% *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
% instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
% Copyright 2002-2003 The MathWorks, Inc.
% Edit the above text to modify the response to help prueba02
% Last Modified by GUIDE v2.5 18-Oct-2007 00:04:51
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @prueba02_OpeningFcn, ...
                  'gui_OutputFcn', @prueba02_OutputFcn, ...
                  'gui_LayoutFcn', [], ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
% --- Executes just before prueba02 is made visible.
function prueba02_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin    command line arguments to prueba02 (see VARARGIN)
% Choose default command line output for prueba02
handles.output = hObject;
global control
control=0;
% Update handles structure
guidata(hObject, handles);
% UIWAIT makes prueba02 wait for user response (see UIRESUME)
% uiwait(handles.figure1);
% --- Outputs from this function are returned to the command line.
function varargout = prueba02_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Get default command line output from handles structure
varargout{1} = handles.output;
% --- Executes on button press in bot_exp.
function bot_exp_Callback(hObject, eventdata, handles)
% hObject    handle to bot_exp (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```



```

global control
[nombre, directorio]=uigetfile('*.*xls','Abrir');
handles.ruta=[directorio nombre];
set(handles.text1,'String',handles.ruta);
if isempty(str2num(get(handles.text1,'String')))==1
    control=1;
end
set(handles.text4,'Visible','off')
set(handles.text3,'Visible','on')
guidata(hObject,handles);
% --- Executes on button press in acep_bot.
function acep_bot_Callback(hObject, eventdata, handles)
% hObject    handle to acep_bot (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
clear datos
global control datos nombres nom_sens
if control==0
    set(handles.text4,'Visible','on')
else
    nom_sens={'FC-PI-5002,VALUE' 'FF-FI-551,VALUE' 'FC-PDI-503,VALUE' 'FC-ZI-500R,VALUE' 'FC-
ZI-5003,VALUE' 'FC-ZI-5002,VALUE' 'FC-ZI-5001,VALUE' 'FC-VI-5008,VALUE' 'FC-VI-5007,VALUE'
'FC-VI-5006,VALUE' 'FC-VI-5005,VALUE' 'FC-VI-5002,VALUE' 'FC-VI-5001,VALUE' 'FC-SI-
500A,VALUE' 'FC-FI-501,VALUE'};
    [datos,nombres]=xlsread(handles.ruta,-1); %'03abr07'
    close (prueba02)
    h5=msgbox('i Carga Exitosa !','Mensaje','warn');
    prueba045
    close(h5);
end

```

## Anexo G

### Código de programación de la interfaz para el procesamiento de los datos históricos

```

function varargout = prueba045(varargin)
% PRUEBA045 M-file for prueba045.fig
%   PRUEBA045, by itself, creates a new PRUEBA045 or raises the existing
%   singleton*.
%
%   H = PRUEBA045 returns the handle to a new PRUEBA045 or the handle to
%   the existing singleton*.
%
%   PRUEBA045('CALLBACK',hObject,eventData,handles,...) calls the local
%   function named CALLBACK in PRUEBA045.M with the given input arguments.
%
%   PRUEBA045('Property','Value',...) creates a new PRUEBA045 or raises the
%   existing singleton*. Starting from the left, property value pairs are
%   applied to the GUI before prueba045_OpeningFunction gets called. An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to prueba045_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%   instance to run (singleton)".

```



```

%
% See also: GUIDE, GUIDATA, GUIHANDLES
% Copyright 2002-2003 The MathWorks, Inc.
% Edit the above text to modify the response to help prueba045
% Last Modified by GUIDE v2.5 23-Jan-2008 16:23:21
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
                  'gui_Singleton', gui_Singleton, ...
                  'gui_OpeningFcn', @prueba045_OpeningFcn, ...
                  'gui_OutputFcn', @prueba045_OutputFcn, ...
                  'gui_LayoutFcn', [] , ...
                  'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
% --- Executes just before prueba045 is made visible.
function prueba045_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to prueba045 (see VARARGIN)
% Choose default command line output for prueba045
handles.output = hObject;
global nombres nom_sens puntero
puntero=zeros(15,1);
[fila,col]=size(nombres);
for i=1:1:col
    for j=1:1:15
        if (strcmp(nombres(1,i),nom_sens(1,j)))
            switch(j)
                case 1
                    set(handles.s01,'Enable','on');
                    puntero(1)=i;
                    break;
                case 2
                    set(handles.s02,'Enable','on');
                    puntero(2)=i;
                    break;
                case 3
                    set(handles.s03,'Enable','on');
                    puntero(3)=i;
                    break;
                case 4
                    set(handles.s04,'Enable','on');
                    puntero(4)=i;
            end
        end
    end
end

```



```

        break;
    case 5
        set(handles.s05,'Enable','on');
        puntero(5)=i;
        break;
    case 6
        set(handles.s06,'Enable','on');
        puntero(6)=i;
        break;
    case 7
        set(handles.s07,'Enable','on');
        puntero(7)=i;
        break;
    case 8
        set(handles.s08,'Enable','on');
        puntero(8)=i;
        break;
    case 9
        set(handles.s09,'Enable','on');
        puntero(9)=i;
        break;
    case 10
        set(handles.s10,'Enable','on');
        puntero(10)=i;
        break;
    case 11
        set(handles.s11,'Enable','on');
        puntero(11)=i;
        break;
    case 12
        set(handles.s12,'Enable','on');
        puntero(12)=i;
        break;
    case 13
        set(handles.s13,'Enable','on');
        puntero(13)=i;
        break;
    case 14
        set(handles.s14,'Enable','on');
        puntero(14)=i;
        break;
    case 15
        set(handles.s15,'Enable','on');
        puntero(15)=i;
        break;
    end
end
end
end
set(handles.text2,'Visible','on');
set(handles.text3,'Visible','on');
set(handles.set_point2,'Visible','on');
set(handles.slider2,'Visible','on');
```



```

% Update handles structure
guidata(hObject, handles);
% UIWAIT makes prueba045 wait for user response (see UIRESUME)
% uiwait(handles.figure1);
% --- Outputs from this function are returned to the command line.
function varargout = prueba045_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Get default command line output from handles structure
varargout{1} = handles.output;
% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global puntero datos datos_utiles h h1 h2 h3 cuenta
%obtiene el sensor a procesar%
selec=get(handles.uipanel1,'SelectedObject');
%obtiene el valor del set point
setpoint=str2num(get(handles.set_point,'String'));
selec_tag=get(selec,'Tag');
indice=str2num(selec_tag(2))*10+str2num(selec_tag(3));
%obtiene el metodo a implementar
metodo=get(handles.uipanel2,'SelectedObject');
metodo_tag=get(metodo,'Tag');
%Selecciona en los datos los correspondientes al sensor
datos_utiles=datos(:,puntero(indice)-1);
%aplica el metodo a los datos útiles
if (metodo_tag=='metodo1')
    [aucorr_func]=secciones(datos_utiles,10);
    muestrasd=(1:1:length(aucorr_func));
    clear global h2
    axes(handles.axes2);
    h1=plot(muestrasd,aucorr_func);
    hold on;
    amplitud=str2num(get(handles.set_point2,'String'));
    setpointsup(1,1:1:length(aucorr_func))=setpoint+amplitud/2;
    setpointinf(1,1:1:length(aucorr_func))=setpoint-amplitud/2;
    plot(muestrasd,setpointsup,'g');
    plot(muestrasd,setpointinf,'g');
    cuenta=0;
    for i=1:1:length(aucorr_func)
        if(aucorr_func(1,i)>setpointsup(1,i)||aucorr_func(1,i)<setpointinf(1,i))
            cuenta=cuenta+1;
            exceso(cuenta)=i;
        end
    end
    if (cuenta~=0)
        h=stem(exceso,aucorr_func(exceso));
        set(h,'LineStyle','none','MarkerEdgeColor','red','Marker','o');
    end
end

```



```

    hold off;
else
    [s_red,prom,disp]=metodo2(datos_utiles);
    cuenta=0;
    for i=1:1:length(s_red)
        if(s_red(1,i)>setpoint)
            cuenta=cuenta+1;
            exceso(cuenta)=i;
        end
    end
    clear global h1
    axes(handles.axes2);
    muestras=10.*(1:1:length(s_red));
    muestrasd=(1:1:muestras(length(muestras)));
    h2=plot(muestras,s_red);
    hold on;
    setpoints(1,1:1:length(muestras))=setpoint;
    plot(muestras,setpoints,'g');
    if cuenta~=0
        h=plot(10.*exceso,s_red(exceso));
        set(h,'LineStyle','none','MarkerFaceColor','red','Marker','o');
    end
    hold off;
end
axes(handles.axes1);
h3=plot(muestrasd,datos_utiles(1:length(muestrasd)));
title('Valores de los sensores');
xlabel('[muestras]');
ylabel('[amplitud]');
axes(handles.axes2);
title('Resultados del método');
xlabel('[muestras]');
ylabel('[amplitud]');
% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
selec=get(handles.uipanel1,'SelectedObject');
selec_tag=get(selec,'Tag');
switch(selec_tag)
    case 's01'
        msgbox(' Presión Regenerador ','Mensaje','help');

    case 's02'
        msgbox('Carga al Reactor','Mensaje','help');

    case 's03'
        msgbox('Diferencial Succión - Descarga Soplador de Aire Principal','Mensaje','help');

    case 's04'
        msgbox('Posición Válvula Admisión de Vapor a Soplador','Mensaje','help');

```



```

case 's05'
    msgbox('Desplazamiento Axial lado Compresor del FC-C-501','Mensaje','help');

case 's06'
    msgbox('Desplazamiento Axial lado Turbina del FC-C-501','Mensaje','help');

case 's07'
    msgbox('Desplazamiento Axial lado Turbina del FC-C-501','Mensaje','help');

case 's08'
    msgbox('Vibración Eje X Cola Compresor del FC-C-501','Mensaje','help');

case 's09'
    msgbox('Vibración Eje Y Cola Compresor del FC-C-501','Mensaje','help');

case 's10'
    msgbox('Vibración Eje X Acople Compresor del FC-C-501','Mensaje','help');

case 's11'
    msgbox('Vibración Eje Y Acople Compresor del FC-C-501','Mensaje','help');

case 's12'
    msgbox('Vibración Eje X Lado Turbina del FC-C-501 ','Mensaje','help');

case 's13'
    msgbox('Vibración Eje Y Lado Turbina del FC-C-501','Mensaje','help');

case 's14'
    msgbox('Velocidad Turbina del FC-C-501','Mensaje','help');

case 's15'
    msgbox('Aire Total Succión FC-C-501','Mensaje','help');
end

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of slider
valor=get(handles.slider1,'Value')./10000;
set(handles.set_point,'String',num2str(valor));
guidata(hObject,handles);
% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
% Hint: slider controls usually have a light gray background, change
%       'usewhitebg' to 0 to use default.  See ISPC and COMPUTER.
usewhitebg = 1;
if usewhitebg

```



```

        set(hObject,'BackgroundColor',[.9 .9 .9]);
    else
        set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
    end
function set_point_Callback(hObject, eventdata, handles)
% hObject    handle to set_point (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'String') returns contents of set_point as text
%        str2double(get(hObject,'String')) returns contents of set_point as a double
% --- Executes during object creation, after setting all properties.
function set_point_CreateFcn(hObject, eventdata, handles)
% hObject    handle to set_point (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
% --- Executes on button press in exportar.
function exportar_Callback(hObject, eventdata, handles)
% hObject    handle to exportar (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global h h1 h2 h3 cuenta
figure
axes
if (isempty(h1))
    copyobj(h2,gca);
else
    copyobj(h1,gca);
end
hold on;
if cuenta~=0
    copyobj(h,gca);
end
title('Resultados del método');
xlabel('[muestras]');
ylabel('[amplitud]');
grid;
hold off;
figure
axes
copyobj(h3,gca);
title('Valores del sensor');
xlabel('Muestras');
ylabel('Amplitud');
grid;
% --- Executes when figure1 is resized.
function figure1_ResizeFcn(hObject, eventdata, handles)

```



```
% hObject handle to figure1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton4 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton5 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
close all;
prueba02;
function edit2_Callback(hObject, eventdata, handles)
% hObject handle to edit2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'String') returns contents of edit2 as text
% str2double(get(hObject,'String')) returns contents of edit2 as a double
% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
% hObject handle to edit2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns called
% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
% --- Executes on button press in metodo2.
function metodo2_Callback(hObject, eventdata, handles)
% hObject handle to metodo2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Hint: get(hObject,'Value') returns toggle state of metodo2
set(handles.text2,'Visible','off');
set(handles.text3,'Visible','off');
set(handles.set_point2,'Visible','off');
set(handles.slider2,'Visible','off');
guidata(hObject,handles);
% --- Executes on slider movement.
function slider2_Callback(hObject, eventdata, handles)
% hObject handle to slider2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'Value') returns position of slider
% get(hObject,'Min') and get(hObject,'Max') to determine range of slider
valor=get(handles.slider2,'Value')./10000;
set(handles.set_point2,'String',num2str(valor));
```



```
guidata(hObject,handles);
% --- Executes during object creation, after setting all properties.
function slider2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
% Hint: slider controls usually have a light gray background, change
%       'usewhitebg' to 0 to use default.  See ISPC and COMPUTER.
usewhitebg = 1;
if usewhitebg
    set(hObject,'BackgroundColor',[.9 .9 .9]);
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end
% --- Executes on button press in metodo1.
function metodo1_Callback(hObject, eventdata, handles)
% hObject    handle to metodo1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hint: get(hObject,'Value') returns toggle state of metodo1
set(handles.text2,'Visible','on');
set(handles.text3,'Visible','on');
set(handles.set_point2,'Visible','on');
set(handles.slider2,'Visible','on');
guidata(hObject,handles);
```