



# “Propuesta de procedimiento para configurar una red neuronal artificial de Base Radial con aplicaciones en el diagnóstico de fallos”

*Ing. Egly Barrero Viciado, Ing. Beatriz Fernández Díaz, Dr .Orestes Llanes Santiago*

## **RESUMEN / ABSTRACT**

En este artículo se presenta un procedimiento que permite la configuración de los parámetros de una arquitectura de red neuronal artificial de Base Radial para tareas de diagnóstico de fallos, luego de establecer un orden lógico para la selección de los mismos. Dicho procedimiento garantiza la obtención del número necesario de neuronas ocultas a partir de fijar el error de diagnóstico deseado por los expertos para cada proceso, permitiendo también seleccionar el método para estimar los anchos de las neuronas ocultas y la ecuación de distancia para la propagación del espacio de entradas. El procedimiento se aplica al proceso de prueba “Tanque Reactor Continamente Agitado” para demostrar su efectividad. Como resultado de los experimentos realizados, se concluye que la elección de la función de distancia no influye sobre la selección del método para estimar los anchos. Se comprueba que esta arquitectura, con el entrenamiento adecuado, exhibe buenas propiedades de sensibilidad y robustez para el diagnóstico de fallos.

Palabras claves: red neuronal artificial de Base Radial, diagnóstico de fallos, robustez.

*In this article a procedure is shown up for allowing the configuration of parameters of artificial neuronal network architecture of Radial Basis for failure diagnosis tasks after establishing a logical order for their selection. Such procedure guarantees the procurance of a necessary number of hidden neurons starting from fixing the diagnosis error desired by the experts for each process, also allowing the selection of the method for esteeming the widths of the hidden neurons and the equation of distance for the propagation of the inlets vector. The procedure applies to the testing process of “Reactor Tank Constantly Agitated” searching to demonstrate its effectiveness. As a result of the realized experiments, it is confirmed that within the topology of the Radial Basis net, the calculation of the centers and widths of the hidden neurons intervene decisively in its execution, as well as the fixed-distance equation, but the election of the function of distance does not act on the selection of the method for esteeming the widths. It is verified that this architecture, with the adequate training, shows good properties of sensibility and robustness.*

**Keywords:** *Radial Basic artificial neural networks, fault diagnosis, robustness.*

**Proposal of a procedure for setting an artificial neural network of Radial Basis with applications in fault diagnosis.**

## INTRODUCCIÓN

Con el paso del tiempo se van desarrollando cada vez más las industrias, volviéndose más complejos los procesos que tienen lugar en las mismas [24], por lo cual se hace necesario mejorar las técnicas para determinar los fallos que afectan dichos sistemas. Se considera un fallo a cualquier desviación no permitida, de al menos un parámetro del sistema, de su condición usual, aceptable o estándar [25]. Un temprano diagnóstico de las afectaciones de un proceso, mientras la planta continúa operando en una región controlable, puede ayudar a evitar la progresión de eventos anormales y reducir las pérdidas en la producción, reportando beneficios en la economía, en la seguridad de los procesos y en la mejora de las labores de mantenimiento [1, 22, 23, 25]. Por otra parte, también es fundamental evitar la ocurrencia de falsas alarmas debido a la presencia de ruidos o alteraciones temporales en el proceso, ya que esto puede provocar paros innecesarios y desconfianza de los operadores en el sistema de diagnóstico, trayendo consigo que pasado un tiempo prefieran ignorar las indicaciones que resulten del mismo.

Desde el punto de vista del control automático, el término robustez es usado para describir el grado de independencia entre el rendimiento de un sistema de control y la influencia de las perturbaciones o de la variación de sus parámetros. En el campo del diagnóstico de fallos se sigue el mismo principio, sin que esto implique una pérdida de sensibilidad ante los fallos para los que se diseña el diagnosticador. Se hace entonces necesario establecer un compromiso pues al beneficiar una de estas propiedades puede verse afectada la otra. También es fundamental no supeditar ni robustez ni sensibilidad a una situación anómala en particular, sino que deben tratar de fijarse de igual manera para todos los fallos que se incluyan en el diseño del sistema de diagnóstico de fallos [21].

Se tiene entonces que las características deseadas de un sistema de diagnóstico de fallos (SDF) son: el rápido diagnóstico de los fallos del proceso, teniendo en cuenta que si el umbral de detección se fija muy bajo puede ser sensible al ruido y por tanto provocar falsas alarmas durante la operación normal; el aislamiento del fallo para hacer al sistema capaz de distinguir entre distintos fallos; la robustez ante el ruido; y la adaptabilidad ante cambios en las condiciones de operación del proceso [25].

Desde hace varios años y hasta la actualidad se han realizado investigaciones en el campo del diagnóstico de fallos, abogándose por las redes neuronales artificiales como una posible técnica a emplear, arrojando resultados satisfactorios [2, 8, 20, 22, 24, 26, 28, 30]. Se han estudiado un número de posibles arquitecturas y se conoce que el diseño de la topología de la red [3, 5, 11, 18, 29] y la selección del algoritmo de aprendizaje [11, 27] influyen sobre la capacidad de diagnóstico de dicha herramienta, y por ende, en la calidad del mismo. La forma usual de aplicar las redes neuronales artificiales al diagnóstico de fallos es para clasificar los datos del proceso de acuerdo con la operación del mismo. Este método de clasificación no toma en cuenta las propiedades dinámicas del proceso porque solo utiliza patrones de mediciones individuales y no tiene información sobre la dirección o magnitud del cambio en las mediciones [23].

Dentro de las arquitecturas de redes neuronales que más auge han tenido en aplicaciones de diagnóstico de fallos se encuentran las redes neuronales artificiales de Base Radial, ya que su estructura básicamente está concebida para que de forma natural agrupe los patrones en regiones del espacio, acorde a las clases a la cual pertenecen. Su empleo generalmente se realiza con su arquitectura clásica: una capa de entrada, una capa oculta con una función de activación gaussiana y una capa de salida con una función de activación lineal [13, 22]. Esta arquitectura presenta mayormente un buen desempeño en problemas de clasificación de patrones, sin embargo, para esto se necesitan muchas neuronas en la capa oculta, lo que hace que la red sea muy extensa y el tiempo de cómputo en fase de ejecución elevado [18]. Se considera que el número necesario de neuronas ocultas crece exponencialmente con el aumento del número de variables de entrada de la red (cantidad de componentes del espacio de entradas), es decir, con el aumento de las dimensiones del espacio de entradas [6, 21, 24]. Se conoce que ha sido utilizada como diagnosticador en problemas que presentan un intercambiador de calor y un tanque reactor continuamente agitado [13, 22], observándose la necesidad de un buen ajuste de sus parámetros para que los resultados puedan ser satisfactorios.

Debido a lo anterior, se ha mostrado interés en la comunidad científica por reformular la red de Base Radial entrenándola mediante algoritmos supervisados como el "gradiente descendiente", con el objetivo de mejorar su aprendizaje y por tanto su desempeño. Las comparaciones entre una red neuronal artificial de Base Radial clásica y una reformulada, aplicadas al mismo caso de estudio, demuestran que la red reformulada tiene igual o menor por ciento de error en la clasificación y necesita un menor número de neuronas en la capa oculta [11]. Se han propuesto nuevos métodos de aprendizaje basados principalmente en el algoritmo de Retropropagación del Error (*Backpropagation* (BP)), que proveen un número mínimo de neuronas ocultas en problemas de diagnóstico de fallos, pero para lograrlo necesitan tiempos de entrenamiento elevados [3]. Se han implementado nuevos métodos que permiten optimizar la arquitectura de Base Radial con el objetivo de disminuir el número de neuronas necesarias en la capa oculta, logrando reducir el tiempo de respuesta de la misma.

Para ello, se ha usado un aprendizaje secuencial de la función de Base Radial, en el cual se combina el criterio de crecimiento de la Asignación de Recursos de la Red Neuronal (RAN) de Platt, con una estrategia reducida basada en la contribución relativa de cada neurona oculta a la salida general de la red (MRAN). Esta aplicación demostró que la red de Base Radial entrenada con este algoritmo es más compacta y obtiene un error de clasificación similar al de una red Perceptrón Multicapa (*Multilayer Perceptron* (MLP)), entrenada con una variante del algoritmo de aprendizaje BP, en un tiempo computacional mucho menor [8, 29].

Como se aprecia en [19] se han utilizado algoritmos evolutivos para determinar de forma automática, en un problema específico, parámetros de las redes de Base Radial como la cantidad de neuronas ocultas y sus respectivos centros y anchos, demostrándose que esta técnica es capaz de encontrar valores óptimos de dichos parámetros manteniendo un pequeño error de generalización.

Sin embargo a pesar de las investigaciones realizadas en el campo de las redes de Base Radial, específicamente en la configuración de sus parámetros, no existe una solución generalizable que permita configurar los parámetros de la red de Base Radial, por lo que este sigue siendo un problema sin solución definitiva en el campo de aplicación de esta arquitectura al diagnóstico de fallos. A pesar de los muchos algoritmos y métodos que se siguen investigando, con el objetivo de perfeccionar esta arquitectura, no se encontró en la literatura un procedimiento, o forma similar, que brinde una vía para configurar completamente una red de Base Radial con el objetivo de emplearla en la clasificación de los estados de un proceso (diagnóstico de fallos). Otra dificultad es que muchos de los métodos propuestos en la literatura, y las modificaciones que se realizan a los mismos, se centran generalmente en la capa oculta de esta red (anchos, centros y números de neuronas) y no incluyen en el entrenamiento la variación de otros parámetros de esta arquitectura.

Por tanto el objetivo del presente trabajo es desarrollar un procedimiento que permita, luego de establecer los parámetros que más influyen en el desempeño de esta arquitectura en el diagnóstico de fallos, proponer una vía lógica basada en pruebas estadísticas para determinarlos, de manera que se garanticen adecuadas propiedades de sensibilidad y robustez. El artículo está estructurado en cinco secciones además de introducción y conclusiones. La sección "Redes neuronales artificiales de base radial" aborda los aspectos teóricos básicos relacionados con las redes de Base Radial. En la sección "Procedimiento para la obtención de una arquitectura de red neuronal artificial de base radial aplicada al diagnóstico de fallos" se presentan los pasos del procedimiento para obtener una arquitectura de Base Radial con aplicación en el diagnóstico de fallos con adecuadas propiedades de sensibilidad y robustez. En las secciones "Caso de Estudio" y "Experimentos Realizados" se explica brevemente el caso de estudio empleado y los experimentos realizados con los principales resultados obtenidos; y la sección "Análisis de los Resultados" está dedicada al análisis de los resultados alcanzados en los experimentos efectuados.

## REDES NEURONALES ARTIFICIALES DE BASE RADIAL

Es un modelo de red unidireccional que puede considerarse de tipo híbrido por incorporar aprendizaje supervisado y no supervisado. La arquitectura de una red neuronal artificial de Base Radial (*Radial Basis Function*, (RBF)) (figura 1) cuenta con tres capas de neuronas: una de entrada, una oculta y otra de salida. En las neuronas de la capa oculta es donde radica la mayor diferencia de esta arquitectura con respecto a las demás, y es que su función de activación es de simetría radial y opera en base a la distancia que separa al espacio de entradas con respecto al vector de pesos sinápticos que almacena cada neurona, evaluando la función radial en dicha diferencia. Las neuronas de la capa de salida son lineales y esencialmente calculan la suma ponderada de las salidas que proporciona la capa oculta [6, 15, 27].

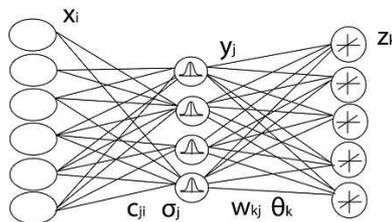
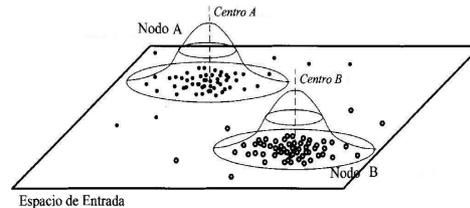


Figura 1. Arquitectura de la Red Neuronal Artificial de Base Radial.

En las neuronas ocultas de la RBF la respuesta es localizada, es decir, solo responden con una intensidad apreciable cuando el espacio de entradas presentado y los pesos sinápticos de la neurona (también conocidos como centros) pertenecen a una zona próxima, como se muestra en la figura 2. Cada neurona oculta se ocupa de una zona del espacio de entradas y el conjunto de neuronas debe cubrir totalmente la zona de interés, pero debe hacerse suavemente, controlando el número de neuronas de la capa oculta y el ancho de cada una, para evitar aumentos innecesarios en la cantidad de neuronas ocultas.



**Figura 2. Respuesta localizada de las neuronas ocultas en la RBF.**

En las neuronas ocultas figuran dos parámetros que son esenciales para un buen desempeño de la red neuronal en general, el centro  $c$  y el ancho  $\sigma$ , los cuales tomarán su valor definitivo una vez realizado el aprendizaje de la red neuronal artificial. Así, si el espacio de entrada coincide con el centro de una neurona esta responde con la unidad (máxima salida). Cuando el espacio de entrada se encuentra en una región cercana al centro de una neurona la respuesta de la misma tiende a uno, lo que significa que está activa, indicando que reconoce el patrón de entrada; por el contrario, si el patrón de entrada es muy diferente del centro la respuesta tiende a cero, indicando que la neurona está inactiva. Por otra parte las neuronas ocultas pueden tomar diferentes formas, aunque en la mayoría de los casos se asumen formas circulares.

### **Aprendizaje de las Redes Neuronales Artificiales de Base Radial**

El aprendizaje de las redes neuronales artificiales de Base Radial generalmente es de tipo híbrido, realizándose en dos etapas. Primeramente, se lleva a cabo un entrenamiento no supervisado en la capa oculta y posteriormente un entrenamiento supervisado en la capa de salida. Con esta división se consigue acelerar notablemente el proceso de aprendizaje con respecto a otros algoritmos [6, 24]. Este proceso de aprendizaje se realiza de forma iterativa, pues el ajuste final de los parámetros se logra a partir de aproximaciones sucesivas de los mismos.

En esta arquitectura es imprescindible elegir la cantidad de neuronas radiales, ya que cada una cubre una parte del espacio de entrada, siendo preciso buscar el número adecuado para que se cubra completamente dicho espacio. En la práctica se emplean diferentes técnicas para determinar el número adecuado de neuronas ocultas en dependencia de la aplicación, por ejemplo la mostrada en [29], y las propuestas más recientes de algoritmos de entrenamiento se preocupan mucho por este parámetro. El valor de los centros se puede obtener empleando algún algoritmo supervisado para agrupamiento (*clustering*) como el algoritmo de Kohonen simple [6], o el conocido algoritmo de las  $k$ -medias [6, 17]. Una vez determinado el valor de los centros se procede al cálculo de los anchos de cada neurona. Para obtener los valores de los anchos usualmente se utilizan criterios heurísticos.

El funcionamiento de las redes neuronales artificiales de Base Radial se basa en gran medida en la distancia que existe entre los patrones de entrada y las neuronas de la capa oculta, de ahí la connotación del aprendizaje no supervisado de su capa intermedia. La función para el cálculo de dicha distancia no fija ningún método para la obtención de los restantes parámetros de la arquitectura y viceversa.

Como se mencionó anteriormente, para garantizar un buen desempeño de esta arquitectura es necesario determinar la cantidad de neuronas ocultas, con el valor de sus centros y anchos, aplicada al problema específico donde se desea utilizar. Hasta el momento no se conoce ningún procedimiento determinístico que facilite la obtención de dicha cantidad. Sin embargo, fijar el tamaño de la capa oculta constituye el punto de partida para determinar los centros y anchos de la misma, no teniendo sentido realizar el proceso a la inversa. Luego de determinar el tamaño de la capa oculta es necesario calcular los valores de sus restantes parámetros (centros y anchos) y para esto se parte de la obtención de uno de ellos y se halla el restante, pudiendo realizarse esta operación en cualquier orden. Otro parámetro de la capa oculta que puede ser ajustado es el umbral, haciendo uso generalmente del método de prueba y error.

Los métodos recomendados en la literatura para el cálculo de los centros, han demostrado ser efectivos en esta tarea. Por otro lado, los métodos utilizados para determinar los anchos brindan estimaciones que pueden ser valores más o menos acertados, por lo que el parámetro que generalmente se busca mejorar es este último.

El último paso en el aprendizaje de la red de Base Radial es el entrenamiento de la capa de salida, el cual es de tipo supervisado. En este momento las salidas de la capa oculta ya son cantidades conocidas, puesto que son halladas en función de los valores de entrada, centros, anchos y umbrales de las neuronas ocultas. A partir de este punto solo queda calcular los pesos sinápticos y umbrales de la capa de salida.

## **PROCEDIMIENTO PARA LA OBTENCIÓN DE UNA ARQUITECTURA DE RED NEURONAL ARTIFICIAL DE BASE RADIAL APLICADA AL DIAGNÓSTICO DE FALLOS**

A partir de contar con los datos históricos de un proceso se plantea un procedimiento, al cual se denominó “Configuración de una RBF para el diagnóstico de fallos”, para obtener la disposición de los parámetros de una red de Base Radial con el fin de diagnosticar los fallos presentes en el sistema bajo estudio, satisfaciendo el error de clasificación deseado y las propiedades de sensibilidad y robustez. Para la concepción del procedimiento se tuvieron en cuenta los siguientes aspectos:

- Los estados a diagnosticar usualmente se agrupan por clases, de forma tal que todos los datos identificativos de un mismo estado del proceso, por ejemplo un fallo, pertenecerán a la misma clase. Visto de esta forma y teniendo en cuenta el funcionamiento de la capa oculta de las redes RBF lo lógico sería tratar de agrupar el espacio de entradas de forma tal que cada neurona de la capa oculta cubra totalmente una clase y de esto no ser posible, ir aumentando la cantidad de neuronas de la capa oculta por clases (todas a la vez en cada iteración), hasta que grupos iguales de neuronas cubran cada clase.
- Usualmente en el diagnóstico de fallos se asocia cada neurona de la capa de salida con cada clase que se desea reconocer, así solamente es necesario verificar qué neurona de la capa de salida está activa para saber a qué clase pertenece el dato presente en la entrada de la red. De igual forma el tamaño de la capa de entrada se fija acorde a la cantidad de variables necesarias para realizar el diagnóstico de fallos.
- Otro aspecto importante es la influencia del método de estimación de los anchos de las neuronas ocultas. Si las neuronas de la capa oculta no tienen correctamente hallados sus anchos, el número necesario de neuronas ocultas que se obtenga mediante cualquier procedimiento no será el más adecuado. Por tanto es fundamental el método que se seleccione para calcular dichos anchos, y el mismo debe ser escogido antes de preocuparse por buscar el número de neuronas de la capa oculta. Ya luego de seleccionado el método de estimación se procede a calcular los valores de los anchos en dependencia de la cantidad de neuronas de la capa oculta ya que estos son dos parámetros que guardan una estrecha relación entre sí.
- El empleo de una función de distancia u otra puede influir en la cantidad de neuronas necesarias en la capa oculta para agrupar los patrones pertenecientes a una clase, ya que es precisamente esta función la encargada de “medir” la distancia existente entre el patrón de entrada y el centro de cada clase. Por lo tanto este parámetro también está vinculado con el valor de los anchos de las neuronas de la capa oculta, pero no influye en la elección del método que se emplee para obtener dichos anchos.
- Buscar el número mínimo de neuronas en la capa oculta es el proceso más lento del ajuste, ya que conlleva una búsqueda exhaustiva a partir de realizar pequeños cambios en esta capa. Sin embargo, puede hallarse un número adecuado de neuronas en la capa oculta, que permita a la red cumplir con los requerimientos para los cuales se diseña, y garantizar que si bien no es el mínimo al menos no aumenta excesivamente el tamaño de esta capa.
- Se conoce de la literatura que la búsqueda de la cantidad de neuronas ocultas se realiza usualmente de forma iterativa a partir de realizar aumentos (o disminuciones) de la cantidad de neuronas hasta obtener el desempeño adecuado. Partiendo entonces de que la función de distancia influye en el tamaño de la capa oculta, es lógico pensar que no tiene por qué coincidir la cantidad de neuronas ocultas, por clase, que provocan un cambio significativo en el error de clasificación con el empleo de diferentes funciones de distancia. Si primeramente se determina esta cantidad mínima necesaria, se ahorra tiempo de entrenamiento pues se reduce el proceso de buscar la cantidad de neuronas necesarias en la capa oculta ya que los aumentos (o disminuciones) que se realicen siempre provocarán un cambio en el error de clasificación de la red.

El procedimiento de forma general contempla tres etapas que se describen a continuación:

**Etapa 1:** Conformación de la arquitectura inicial.

**Paso 1.1:** Establecer la cantidad de neuronas en la capa de entrada ( $N_e$ ) igual al número de variables del proceso que son empleadas en el diagnóstico ( $X_i$ ).

**Paso 1.2:** Establecer la cantidad de neuronas en la capa oculta y la cantidad de neuronas en la capa de salida igual al número de fallos que se deseen diagnosticar más un estado de funcionamiento normal.

**Paso 1.3:** Seleccionar una función de distancia para la propagación del espacio de entrada hacia la capa oculta.

**Paso 1.4:** Seleccionar una función de activación con dependencia radial para las neuronas de la capa oculta.

**Paso 1.5:** Seleccionar un algoritmo de entrenamiento para la capa de salida.

La secuencia para realizar la segunda etapa se describe de forma general a continuación y se muestra en forma algorítmica en la figura 3:

**Etapa 2:** Selección del método de estimación para los anchos de las neuronas ocultas.

**Paso 2.1:** Seleccionar un algoritmo supervisado de agrupamiento para el cálculo de los centros de las neuronas ocultas.

**Paso 2.2:** Determinar el método de estimación adecuado para calcular el ancho de las neuronas ocultas.

**Paso 2.2.1:** Establecer métodos a competir.

**Paso 2.2.2:** Estimar los anchos de las neuronas ocultas y calcular el error de clasificación de la red para cada método.

**Paso 2.2.3:** Mediante el test de Friedman establecer si existen diferencias significativas en el empleo de los métodos.

**Paso 2.2.4:** Si existen diferencias utilizar el test de Wilcoxon para fijar el mejor método mediante el voto mayoritario.

A partir de esta etapa se introduce el empleo de los test estadísticos con el objetivo de tener pruebas válidas que permitan interpretar los resultados [10, 14]. Primeramente es necesario decidir si existen diferencias significativas entre los métodos propuestos, porque de no existir se ahorra tiempo en la comparación uno a uno, para lo cual se seleccionó el test de Friedman. Para el caso de que existan diferencias significativas entre los métodos es necesario contar con un test que permita demostrar si existen o no diferencias significativas en la comparación método a método, para lo cual se propuso el test de Wilcoxon, que permite demostrar si existen o no diferencias significativas entre las medias de dos muestras relacionadas.

La secuencia para realizar la tercera etapa se describe de forma general a continuación y se muestra en forma algorítmica en la figura 4:

**Etapa 3:** Configuración de la capa oculta.

**Paso 3.1:** Aplicar el método para estimar los anchos de las neuronas ocultas seleccionado en la etapa 2.

**Paso 3.2:** Determinar la función para la propagación del espacio de entradas que mejor se adecua al proceso.

**Paso 3.2.1:** Establecer las funciones de distancia a competir y un criterio para el error de clasificación deseado.

**Paso 3.2.2:** Determinar los aumentos significativos en la cantidad de neuronas ocultas para cada función de distancia.

**Paso 3.2.3:** Aumentar en cantidades significativas las neuronas en la capa oculta hasta cumplir con el error fijado.

Para evitar caer en lazos infinitos en esta última etapa se proponen los siguientes criterios:

- En el paso 3.2.2 fijar el número máximo de neuronas en la capa oculta que provocan un cambio significativo en el error como 10 veces el tamaño inicial ( $10 * N^{\circ} \text{Clases}$ ). Si en el proceso de buscar los aumentos de neuronas que sean significativos para el error de clasificación de la red se llega a esta cantidad máxima se detiene el paso y se toma dicho número para comenzar a realizar los aumentos en la capa oculta buscando obtener el error de clasificación máximo deseado para la red.
- En el paso 3.2.3 fijar un límite mínimo de iteraciones consecutivas (se sugiere 10) durante las cuales no se aprecie un cambio en el error de clasificación y una vez alcanzado dicho límite detener el algoritmo y tomar ese menor error alcanzado ya que no es posible obtener el deseado.

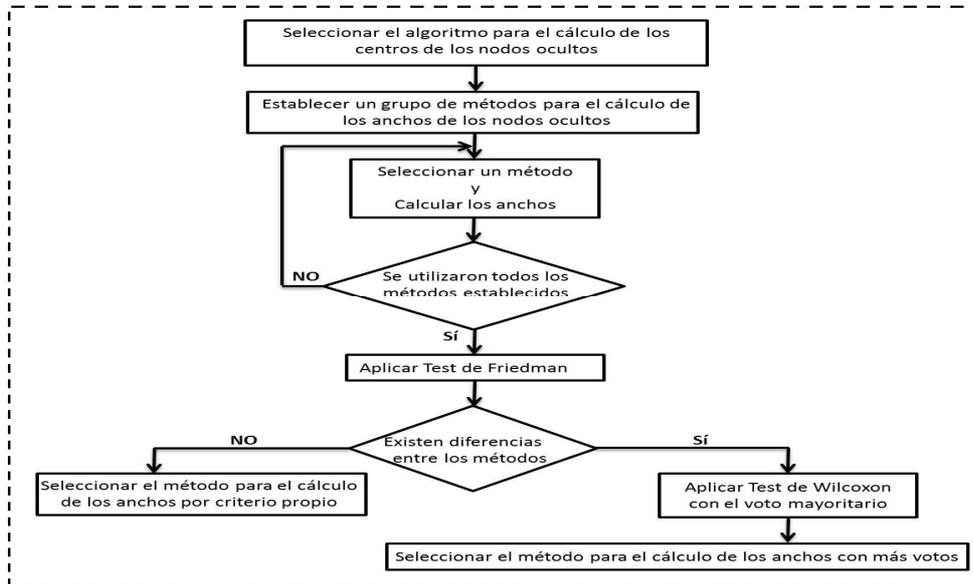


Figura 3. Selección del método para el cálculo de las neuronas ocultas.

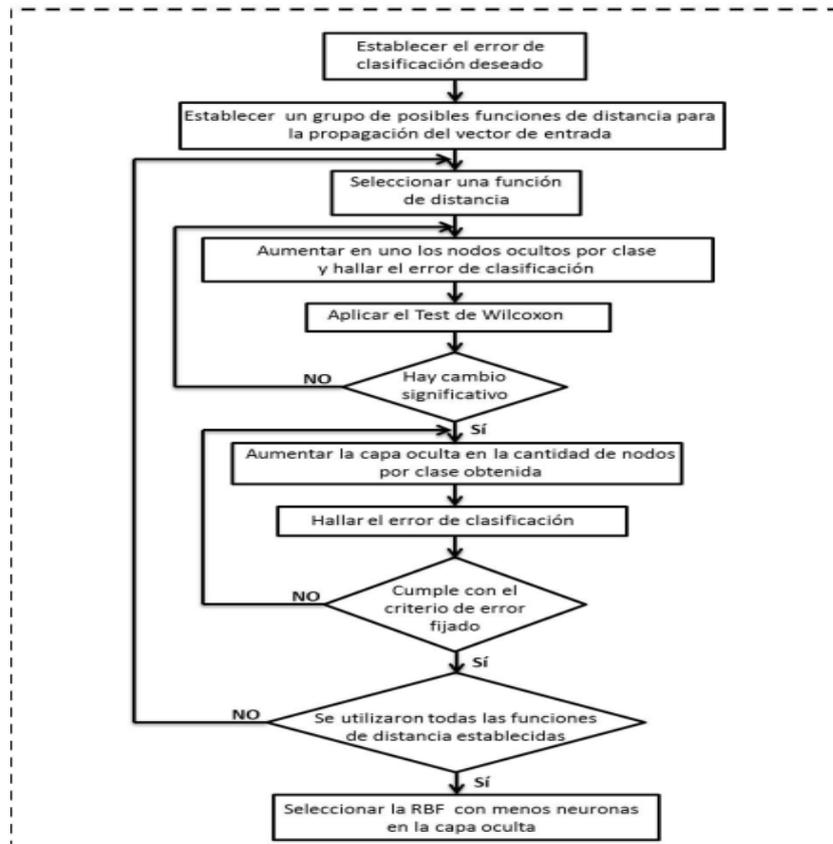


Figura 4. Configuración de la capa oculta.

## CASO DE ESTUDIO

Para realizar los experimentos se seleccionó el proceso de prueba Tanque Reactor Continuamente Agitado (TRCA) por ser un proceso aplicado en la literatura para la evaluación de diagnosticadores [13, 22, 23], siendo el mismo un sistema donde ocurre una reacción exotérmica irreversible, convirtiéndose el reactivo A en el producto B y está compuesto por un tanque, una chaqueta enfriadora, un agitador y una bomba. La instrumentación del mismo cuenta con dos válvulas de control, un transmisor de nivel, dos transmisores de flujo, un transmisor de temperatura y cuatro controladores de tipo PI. En este proceso las variables controladas son temperatura de la reacción ( $T$ ) y el nivel de solución en el tanque ( $h$ ), y las variables manipuladas son flujo de alimentación del refrigerante de la chaqueta ( $Q_c$ ) y el flujo de salida del reactor ( $Q$ ). Toda la descripción del proceso y sus características, así como los fallos empleados, su generación y otros aspectos importantes aparecen en [4, 9]. El esquema general se muestra en la figura 5.

### Fallos del proceso

Del conjunto de fallos que permite estudiar el proceso fueron seleccionados ocho, de los cuales cuatro son positivos y el resto negativos. Los fallos se encuentran numerados según aparece en la literatura donde se aborda este caso de estudio [4, 9], correspondiendo cada número con una descripción particular de la situación de fallo, y adicionalmente a continuación del número se identifica el signo del fallo, positivo o negativo, con las letras “p” y “n” respectivamente. El origen de los fallos tomados en consideración está en perturbaciones externas al proceso y ninguno es debido a fallo en los sensores o en los componentes propios del proceso [4, 9]. Las características de cada uno de los fallos tomados en cuenta se muestran en la tabla 1.

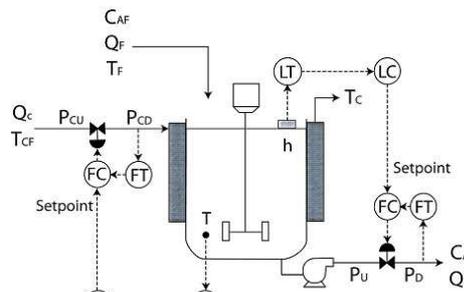


Figura 5. Diagrama Tecnológico del TRCA.

Fallo	Descripción	Valor	Variación
6p y 6n. Cambio escalón en $Q_F$	$Q_F$ se mantiene en su nuevo valor por 100 min	+/- 10 L/min	66.6 %
7p y 7n. Cambio tipo rampa en $C_{AF}$	$C_{AF}$ aumenta en rampa por 100 min	+/- $6 \cdot 10^{-4}$ (mol/L)/min	6 %
8p y 8n. Cambio tipo rampa en $T_F$	$T_F$ aumenta en rampa por 100 min	+/- 0.1 K/min	3 %
9p y 9n. Cambio tipo rampa en $T_{CF}$	$T_{CF}$ aumenta en rampa por 100 min	+/- 0.1 K/min	3 %

Tabla 1. Fallos del TRCA.

De la tabla 1:

$Q_F$ : Flujo de alimentación del reactor

$C_{AF}$ : Concentración de alimentación.

$T_F$ : Temperatura de alimentación

$T_{CF}$ : Temperatura de alimentación del refrigerante

### Generación de los datos históricos

Los datos históricos se generaron por medio de simulación, utilizando el software Simulink del asistente matemático Matlab® 2011. La condición normal de operación y cada uno de los fallos fueron simulados para un tiempo de 100 minutos. El número de observaciones generadas para cada fallo fue de 1190 por cada una de las variables. También se obtuvo un conjunto de datos adicionales donde el tamaño de los fallos fue cambiado respecto al valor inicial visto en la tabla 1, con el fin de probar la capacidad de generalización del diagnosticador propuesto. Se realizaron simulaciones donde se contaminaron las señales con ruido, con el objetivo de emular la variabilidad presente en un proceso real.

## EXPERIMENTOS REALIZADOS.

Durante la aplicación del procedimiento propuesto se hace uso de la validación cruzada para calcular el error de clasificación, por lo que se dividieron los datos de los fallos y de la operación normal en 10 particiones. La aplicación del voto mayoritario en la etapa dos consistió en la adición de puntos al método que se seleccione como el mejor en cada prueba, siguiendo las siguientes pautas:

- Las comparaciones entre los métodos se realizan por pares, asignándole un punto al método con mejores resultados, según el test aplicado, y cero puntos al restante.
- Si los resultados del test no permiten diferenciar entre los dos métodos se le otorgan 0.5 puntos a cada método.

Al concluir la aplicación del test para todas las combinaciones posibles, se dice que el mejor método es el que más puntos obtiene.

### 1: Aplicación del procedimiento al proceso de prueba para los datos de los fallos y la operación normal sin ruido

Etapa 1: Inicialmente la RBF está compuesta por diez neuronas en la capa de entrada, acorde a las diez variables que se miden en el proceso, nueve neuronas en la capa oculta, correspondiendo a un criterio inicial donde se le asigna una neurona a cada clase (fallo) y otra para el funcionamiento normal, y nueve neuronas en la capa de salida siguiendo el criterio anterior. Esta arquitectura se expresa de forma sintetizada como 10-9-9. Se escogió la función de la distancia Euclídea para la propagación del espacio de entrada y la función gaussiana como función de activación de la capa oculta [13, 15, 16, 17, 19, 22, 24], por ser las más comúnmente utilizadas. El algoritmo de entrenamiento seleccionado para la capa de salida fue el de Mínimos Cuadrados (*Least Mean Square* (LMS)), por ser comúnmente utilizado en las capas lineales [6, 7, 12, 16, 18, 27].

Etapa 2: Para el cálculo de los centros de las neuronas ocultas se seleccionó el algoritmo de las k-medias. Partiendo de los centros obtenidos se emplearon las siguientes alternativas para calcular los anchos:

A) Teniendo en cuenta los centros de las neuronas que están más próximas a la neurona que se calcula (ecuación 1).

$$\sigma_j^2 = \frac{1}{N} * \sum_{l=1}^N \|c_l - c_j\|^2 = \frac{1}{N} * \sum_{l=1}^N \sum_k (c_{lk} - c_{jk})^2 \quad (1)$$

donde:

- $c_l$  son los  $N$  centros cercanos a la neurona  $j$ .
- $c_j$  es el centro de la neurona  $j$ .
- $N$  es la cantidad de centros que se tienen en cuenta para calcular el ancho de la neurona  $j$ .
- $\sigma_j$  es el ancho de la neurona  $j$ .

B) Teniendo en cuenta los patrones que se desean clasificar en la neurona que se calcula (ecuación 2).

$$\sigma_j^2 = \frac{1}{N_j} * \sum_{x \in \text{nodoj}} \|x - c_j\|^2 \quad (2)$$

donde:

- $N_j$  es la cantidad de patrones correspondientes a cada centro  $c_j$ .

C) Mediante la implementación de un algoritmo genético que estime este parámetro en base al error de clasificación que comete la RBF aplicando la validación cruzada. Para la selección de los parámetros del algoritmo se tomaron varios trabajos consultados en la literatura donde se aplicaba este algoritmo luego de una selección previa de sus parámetros. Partiendo entonces de estas configuraciones se probaron en el problema abordado realizando 25 experimentos con cada una. Posteriormente se promediaron los errores obtenidos para cada configuración de los parámetros y finalmente se seleccionó la que logró como promedio minimizar más la función objetivo (error de clasificación de la red). Dicha configuración de los parámetros del algoritmo se muestra en la tabla 2.

<b>Parámetros del algoritmo genético</b>	
Tamaño de la población	60
Límite inferior	1x10 <sup>-9</sup>
Límite superior	500
Factor de cruzamiento	0.95
Función de cruzamiento	heurística
Función de mutación	uniforme
Dirección de migración	adelante
Generaciones	500 000
Límite de generaciones sin cambio en el error	50

**Tabla 2. Configuración de los parámetros del algoritmo genético.**

Una vez obtenidos los anchos de las neuronas gaussianas, empleando cada uno de los métodos propuestos, se calcula el error de clasificación de la red para cada variante. Los resultados se muestran en la tabla 3.

<b>Método</b>	<b>Error de clasificación</b>
Ecuación 1	0.07086
Ecuación 2	0.12875
Algoritmo Genético	0.02810

**Tabla 3. Resultados de los métodos aplicados para calcular los anchos.**

La aplicación del test de Friedman arrojó que sí existen diferencias significativas entre las diferentes vías de cálculo de los anchos, por tanto, se procede a la aplicación del test de Wilcoxon para determinar cuál es la que mejores resultados brinda. Ambos test fueron aplicados con una probabilidad del cinco por ciento. Los resultados de la aplicación del voto mayoritario (tabla 4) evidencian que, para este caso, la mejor forma de calcular los anchos de las neuronas ocultas es mediante el uso del algoritmo genético.

<b>Ecuación 1</b>	<b>Ecuación 2</b>	<b>Algoritmo Genético</b>
1	0.5	1.5

**Tabla 4. Resultados del Voto Mayoritario.**

Etapa 3: Configuración de la capa oculta.

Para la propagación del espacio de entradas se proponen las siguientes funciones:

- Función de la distancia Euclídea
- Función de la distancia de Mahalanobis

Se fija el error de clasificación de la red en un 95% de aciertos. Se procede a determinar cuál de estas funciones de distancia permite que la arquitectura cumpla con el error de clasificación establecido, utilizando un menor número de neuronas ocultas. Aplicando el algoritmo genético para la obtención de los anchos de las neuronas gaussianas se alcanzaron las arquitecturas que se muestran en la tabla 5. No fue necesario determinar la cantidad de neuronas ocultas por clase que representan un cambio significativo en el error de clasificación debido a que con la arquitectura inicial fijada se logra cumplir con el criterio de error establecido.

Como las dos arquitecturas presentan la misma cantidad de neuronas ocultas el uso de una función de distancia u otra dependerá de otros criterios, tales como, el mínimo error obtenido, la sencillez de la arquitectura, el tiempo de procesamiento, etc. En este caso se seleccionó la RBF que utiliza la función de la distancia Euclídea porque esta función es más sencilla de implementar y por tanto la red presenta una menor complejidad.

Función de distancia	Arquitectura	Error de clasificación
Euclídea	10-9-9	4.53 %
Mahalanobis	10-9-9	4.78 %

**Tabla 5. Arquitecturas RBF obtenidas.**

**Comparación de la arquitectura obtenida mediante el procedimiento con una RBF creada por Matlab®**

La herramienta matemática Matlab® 2011 posee una función llamada *newrb* que crea y entrena una red neuronal artificial de Base Radial. Se conoce que este comando, por defecto, fija la dispersión de las neuronas ocultas en 1 y el número máximo posible lo limita con la cantidad de datos que contenga el vector de entrada, aunque ambos parámetros pueden variarse. No proporciona ninguna forma de estimación para el ancho de las neuronas ocultas, recomendando que se prueben diferentes valores hasta lograr el resultado deseado. Tampoco brinda ninguna forma de ajuste para el número de neuronas en la capa oculta que no sea “prueba y error”.

Se empleó este comando con el objetivo de comparar, en cuanto al número de neuronas en la capa oculta, la RBF obtenida mediante la aplicación del procedimiento propuesto con la obtenida mediante la herramienta matemática. El empleo de la función *newrb* se realizó con los valores que trae por defecto, excepto el error que se fijó en un 5 %. Como se puede observar en la tabla 6, la RBF creada por Matlab® necesitó un mayor número de neuronas en su capa oculta que la obtenida aplicando el procedimiento propuesto, para cumplir con el criterio del 5 % de error de clasificación.

RBF obtenida mediante:	Arquitectura	Error de clasificación
Procedimiento	10-9-9	4.53 %
Matlab®	10-12-9	4.91 %

**Tabla 6. RBF creada por Matlab® y RBF obtenida aplicando el procedimiento propuesto.**

**2: Aplicación del procedimiento al proceso de prueba para los datos de los fallos y la operación normal con un 2 % de ruido**

Etapa 1: La conformación de la arquitectura inicial de la RBF, para este caso, arrojó los mismos resultados que para los datos del proceso sin ruido, por lo que se obtuvo una red con igual configuración de partida. Fue seleccionado el mismo algoritmo de entrenamiento para la capa de salida que el empleado para los datos del proceso sin ruido.

Etapa 2: En la segunda etapa se propusieron los mismos métodos que los aplicados a los datos del proceso sin ruido, tanto para hallar los centros de las neuronas ocultas como para el cálculo de los anchos de las mismas, manteniendo la configuración de los parámetros del algoritmo genético empleada anteriormente. Los resultados se muestran en la tabla 7.

Método	Error de clasificación
Ecuación 1	0.32539
Ecuación 2	0.46890
Algoritmo Genético	0.26778

**Tabla 7. Resultados de los métodos aplicados para calcular los anchos.**

Nuevamente se evidenció (tabla 8) que el algoritmo genético es el método que mejores resultados brinda en el cálculo de los anchos de las neuronas gaussianas para esta aplicación.

Ecuación 1	Ecuación 2	Algoritmo Genético
0.5	0.5	2

**Tabla 8. Resultados del Voto Mayoritario.**

Etapa 3: En la tercera etapa se propusieron las mismas funciones de distancia para la propagación del espacio de entradas que en el experimento anterior. Utilizando el algoritmo genético para la obtención de los anchos de las neuronas gaussianas y las funciones de distancia planteadas, se evidenció que para la distancia Euclídea el aumento de nueve neuronas en la capa oculta representa un cambio significativo en el error de clasificación de la RBF, mientras que con el uso de la distancia de Mahalanobis se necesitan aumentos de 18 neuronas en la capa oculta para lograr un cambio significativo en el error de clasificación de la red.

Al continuar aumentando las neuronas de la capa oculta en cantidades significativas, para obtener una arquitectura de red que cumpla con el criterio de error fijado, se obtuvieron los resultados que se muestran en la tabla 9.

Función de distancia	Arquitectura	Error de clasificación
Euclídea	10-90-9	4.3 %
Mahalanobis	10-45-9	3.95 %

**Tabla 9. Arquitecturas RBF obtenidas.**

En este caso la RBF que mejores resultados brinda es la que utiliza la distancia de Mahalanobis como función de propagación del espacio de entrada, ya que solo necesitó 45 neuronas en la capa oculta para lograr el error de clasificación deseado.

#### **Comparación de la arquitectura obtenida mediante el procedimiento con una RBF creada por Matlab®**

En la tabla 10 se muestra la comparación entre la RBF obtenida mediante la aplicación del procedimiento propuesto y la desarrollada por la herramienta matemática Matlab® haciendo uso de su función *newrb*, cuya aplicación se realizó igual que en el experimento anterior.

RBF obtenida mediante:	Arquitectura	Error de clasificación
Procedimiento	10-45-9	3.95 %
Matlab®	10-199-9	4.99 %

**Tabla 10. RBF creada por Matlab® y RBF obtenida aplicando el procedimiento propuesto.**

Nuevamente, la RBF creada por Matlab® necesitó un mayor número de neuronas en su capa oculta que la obtenida aplicando el procedimiento propuesto, para cumplir con el criterio del 5 % de error de clasificación.

## **ANÁLISIS DE LOS RESULTADOS.**

Se conoce que el proceso de aprendizaje de las redes neuronales artificiales se basa en la información contenida en los patrones de entrenamiento, por lo que resulta de vital importancia verificar la correcta asimilación de la información presentada, y también la capacidad de la red de generalizar el conocimiento adquirido. Estos procesos se llevan a cabo cuando se simula la red neuronal, luego de entrenada, ante patrones no vistos en el proceso de aprendizaje.

Por otra parte dentro del campo del diagnóstico de fallos es importante que la herramienta seleccionada, en este caso la RBF, sea robusta ante el ruido presente en las mediciones y la variabilidad en los datos que el mismo provoca, pero que también presente sensibilidad ante los fallos para los que fue diseñada cuando estos comienzan a presentarse y su representación en las mediciones no es tan evidente. Para comprobar estas dos características se realizan pruebas tomando datos contaminados con diferentes niveles de ruido así como también datos con cambios en la magnitud de los fallos que representan.

Teniendo en cuenta lo anterior se toman las dos RBF ya entrenadas obtenidas mediante la aplicación del procedimiento propuesto y se simulan con nuevos patrones del proceso que presentan variaciones en su magnitud y por ciento de ruido, con el objetivo de observar cómo se afecta el desempeño de dichas redes en la clasificación del estado del sistema.

#### **Resultados para la RBF entrenada con los datos sin ruido.**

Seguidamente se muestran los resultados de las simulaciones realizadas para la RBF entrenada con los datos de los fallos y operación normal del proceso sin ruido y empleando como función de propagación la distancia Euclídea (ver tabla 5), ante variaciones en la magnitud (tabla 12), adición de ruido a los datos del proceso (tabla 11) y ambas afectaciones a la vez (tabla 13).

Por ciento de ruido	Error de Clasificación
Sin ruido (datos del entrenamiento)	4.53 %
2 %	39.78 %
3 %	42.78 %

**Tabla 11. Error de clasificación de la RBF ante variaciones en el por ciento de ruido de los datos.**

Observando la tabla 11 se puede afirmar que la red presenta un deterioro considerable en su capacidad de clasificación cuando procesa datos afectados por diferentes niveles de ruido, lo cual significa que si fue entrenada con datos que presentan poca variabilidad (sin ruido) no responderá adecuadamente en su funcionamiento cuando clasifique datos que si estén afectados por

perturbaciones. Esto quiere decir que la red aprendió los valores y no los patrones presentes en los datos empleados durante el entrenamiento.

Cambios en la magnitud	Error de Clasificación
Disminuyendo	6.12 %
Sin cambios	4.53 %

**Tabla 12. Error de clasificación de la RBF ante cambios en la magnitud de los datos.**

En cuanto a los resultados mostrados en la tabla 12 se puede observar que la sensibilidad de la red para reconocer los fallos para los cuales fue diseñada sigue siendo adecuada cuando las mediciones reflejan menos la presencia del fallo en el proceso (menor magnitud) pues el error de clasificación no se deterioró de forma considerable como sucedió con el ruido.

Cambios en la magnitud y por ciento de ruido	Error de Clasificación
Disminuyendo y 3 % de ruido	53.98 %

**Tabla 13. Error de clasificación de la RBF ante cambios en la magnitud y adición de ruido en los datos.**

Con respecto a los resultados mostrados en la tabla 13 se puede finalmente concluir que la red entrenada con datos del proceso sin ruido no es adecuada para el diagnóstico de los fallos del proceso si posteriormente las mediciones presentarán afectaciones por el ruido, aunque pueda ser sensible a los fallos de menor magnitud. La variabilidad de los datos afecta evidentemente su desempeño.

**Resultados para la RBF entrenada con un 2 % de ruido en los datos.**

Seguidamente se muestran los resultados de las simulaciones realizadas para la RBF entrenada con los datos de los fallos y operación normal del proceso con un 2 % de ruido y empleando como función de propagación la distancia de Mahalanobis (ver tabla 9), ante variaciones en la magnitud (tabla 15), cambios en el por ciento de ruido de los datos del proceso (tabla 14) y ambas afectaciones a la vez (tabla 16).

Por ciento de ruido	Error de Clasificación
Sin ruido	7.61 %
2 %	3.95 %
3 %	8.73 %

**Tabla 14. Error de clasificación de la RBF ante variaciones en el por ciento de ruido de los datos.**

Observando la tabla 14 se puede apreciar que la red responde adecuadamente cuando procesa datos que o bien no presentan ruido o bien están afectados por un mayor por ciento de ruido que el contenido en los datos de entrenamiento. Este comportamiento es el deseado, por lo que esta selección de los datos de entrenamiento (con ruido) es más adecuada para lograr un buen desempeño de la red en su posterior funcionamiento. En esencia estos resultados representan que la red fue capaz de asimilar los patrones presentes en los datos que empleó durante el entrenamiento.

Cambios en la magnitud	Error de Clasificación
Disminuyendo	9.96 %
Sin cambios	3.95 %

**Tabla 15. Error de clasificación de la RBF ante cambios en la magnitud de los datos.**

Con respecto a su sensibilidad se puede apreciar (tabla 15) que el desempeño de la red sigue siendo adecuado cuando procesa datos en los cuales el fallo afecta de forma menos evidente a las variables medidas. Se observa además que el error aumentó un poco con respecto a la red anterior (entrenada sin ruido) pero aun así sigue teniendo un desempeño satisfactorio. Es bueno considerar que esta red está clasificando datos que no solo tiene cierta variabilidad (ruido) sino que además ahora reflejan menos la presencia del fallo (disminución de la magnitud).

<b>Cambios en la magnitud y por ciento de ruido</b>	<b>Error de Clasificación</b>
Disminuyendo y 3 % de ruido	18.92 %

**Tabla 16. Error de clasificación de la RBF ante cambios en la magnitud y adición de ruido en los datos.**

En los resultados de la última prueba realizada a esta red (tabla 16) se puede ver que su desempeño sufrió bastante deterioro pues aumentó su error en la clasificación. Sin embargo, si tenemos en cuenta que esta simulación combina una variación en los datos de entrada (ruido) no vista en el entrenamiento con el cambio en la magnitud de los fallos (disminución) podemos decir que la red todavía responde de manera aceptable ya que en aproximadamente el 80% de los casos en que los datos presenten ambas afectaciones la clasificación será correcta.

#### **Evaluación de la influencia de los parámetros de la red neuronal artificial de Base Radial en el diagnóstico de fallos.**

Al simular la RBF entrenada con los datos que presentan ruido se evidenció una mejora considerable en la robustez del diagnóstico. El empleo de la distancia de Mahalanobis demostró superioridad en cuanto al número de neuronas necesarias en la capa oculta, para poder diagnosticar con el por ciento de error de clasificación deseado. Lo anterior es un resultado lógico debido a que esta distancia tiene en cuenta las relaciones entre las variables de entrada, logrando extraer con mayor facilidad el patrón dentro de la información con ruido. Es entonces evidente que la función de distancia influye sobre el rendimiento de esta arquitectura en el diagnóstico de fallos y es un parámetro importante a tener en cuenta.

La significancia del método de estimación de los anchos de las neuronas ocultas, así como la cantidad adecuada de neuronas en la capa oculta, son aspectos ya conocidos y probados en la literatura consultada, y se reafirman con los resultados obtenidos en este trabajo.

Por tanto, la selección de los parámetros de esta arquitectura tomados en cuenta en el procedimiento para buscar su adecuada elección (método de estimación de los anchos de las neuronas ocultas, función de distancia y cantidad de neuronas ocultas), resulta correcta ya que afectan su desempeño como clasificador del estado de un proceso.

## **CONCLUSIONES**

Como resultado del trabajo se obtuvo un procedimiento general que permite configurar los parámetros de interés establecidos en esta investigación para una red de Base Radial, con el propósito de aplicarla al diagnóstico de los fallos presentes en un proceso industrial. Dicho procedimiento garantiza la adecuada selección de parámetros tan importantes como la función de distancia y el método de estimación de los ancho de las neuronas ocultas, a partir de poner a competir las diferentes propuestas y basando la selección en pruebas estadísticas reconocidas.

La aplicación del procedimiento propuesto permite reducir la cantidad de experimentos a realizar en el proceso de configurar una red de Base Radial para tareas de diagnóstico de fallos ya que plantea una selección secuencial de los parámetros más importantes de la arquitectura basado en un orden de configuración lógico, lo cual se demostró mediante los experimentos realizados.

Con el principal aporte de este trabajo, el procedimiento de configuración, se resuelve parcialmente la carencia de métodos o procedimientos generales que guíen en la configuración de una red de Base Radial para aplicarla al diagnóstico de los fallos de un proceso industrial.

Esta arquitectura demuestra seguir siendo una buena herramienta para realizar tareas de diagnóstico de fallos ya que con un adecuado entrenamiento y una correcta selección de los datos de entrenamiento se garantizan buenas propiedades de sensibilidad y robustez. La aplicación del procedimiento para configurar la red no limita ni disminuye dichas propiedades, y si potencia seleccionar los parámetros más adecuados para cada proceso en particular, lo que trae buscar el mejor rendimiento de la red para cada caso.

Aunque los resultados mostrados en los experimentos son válidos solo para el proceso empleado como ejemplo, se considera que el procedimiento propuesto es, en general, aplicable a la configuración de una red de Base Radial para realizar tareas de diagnóstico de fallos en cualquier proceso que se desee, ya que el mismo no toma en cuenta particularidades de una aplicación específica y si se centra en la herramienta a emplear, en este caso, las redes de Base Radial.

## REFERENCIAS

1. C. Angeli and A. Chatzinikolaou, On-line fault detection techniques for technical systems: A survey, *International Journal of Computer Science & Applications* 1 (2004), no. 1, 12-30.
2. W. R. Becraft and P. L. Lee, An integrated neural network/expert system approach for fault diagnosis, *Computers and Chemical Engineering* 17 (1993), no. 10, 100-1014
3. A. G. Bors and M. Gabbouj, Minimal topology for a radial basis functions neural network for pattern classification, *Digital Signal Processing* 4 (1994), no. 3, 173-188.
4. O. Camacho, D. Padilla, and J. L. Gouveia, Diagnóstico de fallas utilizando técnicas estadísticas multivariantes, *Revista Técnica de Ingeniería. Luz* 30 (2007), no. 3, 253-262.
5. H. Cevikalp, D. Larlus, and F. Jurie, A supervised clustering algorithm for the initialization of rbf neural network classifiers, *IEEE 15th Signal Processing and Communications Applications*, (2007), 1-4.
6. B. Martín del Brío and A. Sanz Molina, *Redes neuronales y sistemas difusos*, 2 ed., RA-MA, Marzo 2001.
7. S. Elanayar and Y. C. Shin, Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems, *IEEE Transactions on Neural Networks* 5 (1994), no. 4, 594-603.
8. F. Gu, P.J. Jacob, and A.D. Ball, A rbf neural network model for cylinder pressure reconstruction in internal combustion engines, *IEE Colloquium on Modeling and Signal Processing for Fault Diagnosis* (1996), 1-11.
9. M. C. Johannesmeyer, A. Singhal, and D. E. Seborg, Pattern matching in historical data, *AICHE* 48 (2002), no. 9, 2022-2038.
10. D. J. Sheskin, *Parametric and nonparametric statistical procedures*, 2 ed., Chapman & Hall/CRC, 2000.
11. N. B. Karayiannis, Reformulated radial basis neural networks trained by gradient descent, *IEEE Transactions on Neural Networks* 10 (1999), no. 3, 657-671.
12. V. Kecman, *Learning and soft computing. Support vectors machines, neural networks and fuzzy logic models*, 2001.
13. H.N. Koivo, Artificial neural networks in fault diagnosis and control, *Control Engineering Practice* 2 (1994), no. 1, 89-101.
14. J. Luengo, S. García, and F. Herrera, A study on the use of statistical tests for experimentation with neural networks: Analysis of parametric test conditions and non-parametric tests, *Expert Systems with Applications* 36 (2009), no. 4, 7798-7808.
15. A. C. McCormick and A. K. Nandi, A comparison of artificial neural networks and other statistical methods for rotating machine condition classification, *IEE Colloquium on Modeling and Signal Processing for Fault Diagnosis* (1996), 2/1 - 2/6.
16. S.W. Park and B.T. Zhang, Learning constructive rbf networks by active data selection, *Proceedings of the International Conference on Neural Information Processing*, 2000, 1411-1415.
17. K. Patan., *Artificial neural networks for the modelling and fault diagnosis of technical processes*, vol. 377, Springer, 2008.
18. H. Peng, T. Ozaki, V. Haggan, and Y. Toyoda, A parameter optimization method for radial basis function type models, *IEEE Transactions on Neural Networks* 14 (2003), no. 2, 432-438.
19. V.M. Rivas, J.J. Merelo, P.A. Castillo, M.G. Arenas, and J.G. Castellano, Evolving rbf neural networks for time-series forecasting with evrbf, *Information Sciences* 165 (2004), no. 3-4, 107-220.
20. S. Sina Tayarani-Bathaie, Z.N. Sadough Vanini and K. Khorasani, Dynamic neural network-based fault diagnosis of gas turbine engines, *Neurocomputing*, 2014, 153-165
21. S. Simani, C. Fantuzzi, and R. J. Patton, *Model-based fault diagnosis in dynamic systems using identification techniques*, Springer, 2002.
22. T. Sorsa and H. N. Koivo, Application of artificial neural networks in process fault diagnosis, *Automatica*. 29 (1993), no. 4, 843-849.
23. T. Sorsa, H. N. Koivo, and H. Koivisto, Neural networks in process fault diagnosis, *IEEE Transactions on Systems and Cybernetics* 21 (1991), no. 4, 815-825.
24. S.G. Tzafestas and P. J. Dalianis, Fault diagnosis in complex systems using artificial neural networks, *Proceedings of the Third IEEE Conference on Control Applications*, vol. 2, IEE, 1994, 877-882.
25. V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, A review of process fault detection and diagnosis part 1: Quantitative model-based methods, *Computers and Chemical Engineering* 27 (2003), no. 3, 293-311.

26. S. Verron, T. Tiplica, and A. Kobi, Fault diagnosis of industrial systems by conditional Gaussian network including a distance rejection criterion, *Engineering Applications of Artificial Intelligence*, 2010, 1229-1235
27. M. Weerasinghe, J. B. Gonun, and D. Williams, Fault diagnosis of a nuclear processing plant at diferent operating points using neural networks, *IEE Colloquium on Fault Diagnosis in Process Systems*, vol. 1997, Abril 1997, 8/1-8/4.
28. Ch. Xinyi and Y. Xuefeng, Fault Diagnosis in Chemical Process Based on Self-organizing Map Integrated with Fisher Discriminant Analysis, *Process Systems Engineering and Process Safety*, 2013, vol. 21, no. 4, 382-387
29. L. Yingwei and N. Sundararajan, Performance evaluation of a sequential minimal radial basis function (rbf) neural network learning algorithm, *IEEE Transactions on Neural Networks* 9 (1998), no. 2, 308-318.
30. H. Zayandehroodi, A. Mohamed, M. Farhoodnea and M. Mohammadjafari, An optimal radial basis function neural network for fault location in a distribution network with high penetration of DG units, *Measurement*, 2013, 3319-3327

## AUTORES

Egly Barrero Viciado, Ingeniera Automática, CUJAE, La Habana, Cuba. [egly@electrica.cujae.edu.cu](mailto:egly@electrica.cujae.edu.cu)

Beatriz Fernández Díaz, Ingeniera Automática, Empresa Productora de Software para la Técnica Electrónica (Softel), La Habana, Cuba. [bettyf1989@yahoo.com](mailto:bettyf1989@yahoo.com)

Orestes Llanes Santiago, Ingeniero Electricista, Doctor en Ciencias Técnicas, CUJAE, La Habana, Cuba. [orestes@electrica.cujae.edu.cu](mailto:orestes@electrica.cujae.edu.cu)