Operadores lingüísticos OWA-RIM para la diagnosis de fallos en plantas complejas

A. Sánchez-Fernández, M.J. Fuente, G.I. Sainz-Palmero
Dpto. de Ingeniería de sistemas y Automática
Universidad de Valladolid
Valladolid, España
alvsan, mjfuente, gresai@eii.uva.es

Jose Manuel Benitez
Dpto de Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada
Granada, España
j.m.benitez@decsai.ugr.es

***Abstract*—En este trabajo se afronta el problema de la detección e identificación de fallos en plantas industriales complejas. Dicho problema se aborda como una toma de decisiones basada en operadores lingüísticos OWA, lo cual permite fusionar diversos métodos de identificación de fallos (FI) alternativos. De esta manera la diagnosis de fallos resulta más robusta, y por otro lado el aspecto lingüístico de los operadores manejados encaja fácilmente en el contexto de la detección e identificación de fallos. La identificación se lleva a cabo usando varios métodos de FI muy utilizados, la solución de cada método se agrega usando operadores del tipo Ordered Weighed Average (OWA), basados en cuantificadores Regular Increasing Monotone (RIM). En este artículo se ha hecho una comparativa de los términos lingüísticos más conocidos para implementar estos operadores OWA-RIM en el contexto de la identificación de fallos. Esto se ha aplicado a un benchmak de plantas depuradoras de aguas residuales.**

***Palabras clave*—Identificación de fallos, Fusión de información, OWA, Cuantificadores.**

I. INTRODUCIÓN

En cualquier tipo de instalaciones, plantas industriales, maquinaria, etc. existe el problema de verificar si tanto el propio sistema como sus dispositivos de control están trabajando correctamente y sin fallos. Además, en caso de detectar algún fallo, es necesario localizar los puntos de la planta en los que ha aparecido el fallo, esto es, analizar las causas del comportamiento defectuoso. Por otro lado, en muchos casos es difícil y costoso detener la instalación, y por tanto, se debe conocer cómo evoluciona el sistema después del fallo para tomar la decisión de parar la planta para reparar el fallo o continuar trabajando con ella si el fallo no afecta a la integridad de la misma, ni a la calidad del proceso, ni resulta peligroso para el personal.

En sistemas de cierta envergadura se cuenta con gran cantidad de variables y todas ellas deben ser monitorizadas para detectar y localizar cualquier problema. Por todo esto se hace imprescindible tener una herramienta que facilite la diagnosis de fallos en la planta.

En el momento en el que un fallo es detectado se requiere saber dónde está el problema. Tradicionalmente los diagramas de contribución [1] han sido la herramienta más utilizada para localizar fallos basadas en datos, y en concreto en el contexto de PCA. Algunos autores han propuesto otras soluciones: en [2] se combinan los estadísticos tradicionales del PCA (T^2 y Q) en un nuevo estadístico φ , en [3] se desarrolló el método de contribución basada en la reconstrucción, y en [4] se introdujeron los diagramas de contribución modificados. A pesar de toda esta variedad de métodos existentes no hay una técnica universal, cada una tiene ventajas e inconvenientes. En este contexto se plantea el uso de múltiples técnicas de identificación de fallos y la "fusión" de sus resultados para así conseguir una diagnosis más robusta. Este trabajo está basado en la metodología propuesta en [5] de identificación de fallos, donde la diagnosis final es una toma de decisiones multicriterio usando operadores lingüísticos OWA (Ordered Weighted Average)-Regular Increasing Monotone (RIM), [6].

La diagnosis [5] se basa en dos procesos de identificación del fallo, uno al detectarse éste, y otro una vez que la planta ha evolucionado en condiciones de fallo y llega a un nuevo estado estacionario. En el primer caso, sólo se tienen en cuenta las variables afectadas por el fallo en los primeros instantes. En el segundo caso, se tienen en cuenta otras variables que serán las que se verán afectadas por la propagación el fallo a lo largo de la instalación. El objetivo es conocer el origen del fallo y la posterior evolución del mismo a través del sistema. Algunos autores realizan únicamente la identificación en el instante de detección del fallo [1], mientras que otros la realizan a lo largo de un período de tiempo extenso después de dicha detección: Yue et al. introdujeron el índice de reconstrucción [2], Alcala et al. ampliaron el trabajo acerca de dicho índice [3], Liu et al. aplicaron diagramas de contribución modificados [4], Dunia et al. trabajaron con los subespacios del fallo [7].

La novedad presentada en este trabajo consiste en la aplicación de una amplia variedad de operadores lingüísticos OWA-RIM: *Existe, Todos, Al menos* α*,...* para comprobar las prestaciones de dichos cuantificadores y ver si obtienen una buena diagnosis: en términos de una correcta identificación de las variables responsables del fallo y de separabilidad entre las variables candidatas al fallo y el resto de variables de la planta.

El resto de este artículo está organizado de la siguiente manera: la Sección II explica los aspectos teóricos en los que se basa este trabajo. La metodología de diagnosis se

Este trabajo ha sido financiado parcialmente por el Ministerio de Economía y Competitividad del Gobierno español y por el Fondo Europeo de Desarrollo Regional (FEDER) a través de los proyectos: DPI2015-67341-C2-2-R, TIN2013-47210-P, TIN2016-81113-R.

explica brevemente en la Sección III. En la Sección IV se explica la planta donde se ha probado el método propuesto y los resultados obtenidos. Finalmente, las conclusiones de este trabajo están contenidas en la Sección V.

II. TEORÍA

A. Análisis de Componentes Principales

El Análisis de Componentes Principales (PCA) [8] es una técnica muy utilizada para reducción de la dimensionalidad, construcción de modelos, análisis de datos, así como para detectar y diagnosticar fallos [1], [3], [4], [9].

El modelo PCA parte de una matriz $\mathbf{X}$ $(n \times m)$ con datos de la planta en condiciones normales, que es normalizada a media 0 y varianza 1: $\mathbf{X^{norm}}$. Su matriz de covarianza se descompone en valores singulares, dando como resultado:

$$\mathbf{S} = \frac{1}{(n-1)}(\mathbf{X}^{norm})^T\mathbf{X}^{norm} = \mathbf{V\Lambda V}^T \tag{1}$$

donde $\mathbf{\Lambda}$ (m x m) es una matriz diagonal con los valores singulares, y $\mathbf{V}$ (m x m) contiene los vectores propios. Si se elige un número a de elementos de $\mathbf{\Lambda}$ que representen un porcentaje suficientemente alto de la varianza total de los datos, se obtiene una matriz $\mathbf{\Lambda_a}$. Las a primeras columnas de $\mathbf{V}$, forman la *matriz de cargas* $\mathbf{P}$.

1) Detección de fallos: La opción más usada para realizar la detección de fallos con PCA se basa en el cálculo de las estadísticas T^2 y Q como sigue [9], [1]:

$$T^2 = z^T\mathbf{P\Lambda}_a^{-1}\mathbf{P}^T z \tag{2}$$

$$Q = r^T r, \; r = (I - \mathbf{PP}^T)z^T \tag{3}$$

Siendo z una nueva observación e I la matriz identidad. Si los valores de T^2 y Q están por debajo de unos umbrales T^2_α y Q_α, el sistema trabaja en condiciones normales.

Hay un tercer estadístico, que es combinación de los dos anteriores: φ [2].

$$\varphi = \frac{Q}{Q_a} + \frac{T^2}{T^2_\alpha} \tag{4}$$

Los respectivos umbrales que no se deben superar se pueden calcular según lo expuesto en [8], [2].

La alarma saltará para T^2, Q o φ cuando uno de ellos sobrepase su umbral un número determinado de observaciones consecutivas. Ese número de observaciones consecutivas debe escogerse para evitar falsas alarmas pero no debe retrasar mucho la detección.

Los valores de los umbrales calculados antes son teóricos, y es necesario reajustarlos a valores más realistas usando los datos disponibles. Para ello, se toman datos del sistema funcionando sin fallo y se ajusta un umbral de modo que sólo un α % de las observaciones supere dicho umbral.

2) Identificación de fallos: Después de detectar un fallo, es necesario identificar las variables responsables del mismo. El análisis de contribuciones [1] es una de las técnicas más utilizadas para diagnosticar fallos. Se basa en la influencia de cada variable del sistema en el fallo. Las variables con mayor contribución al estadístico fuera de control (que ha detectado el fallo) son las responsables. Además de dicho método existen varios otros métodos que permiten identificar las variables responsables del fallo, los utilizados en este artículo se describen a continuación.

a) Contribuciones a T^2: En la observación *x*, para cada una de los componentes principales normalizados de más alto valor se calculan las contribuciones de cada variable x_j en el componente principal normalizado *a* como [1] :

$$cont_{a,j} = \frac{t_a}{(\sigma_a)^2}(x_j - \mu_j) \tag{5}$$

si es negativa, $cont_{a,j}$ se deja a cero. Se calcula la contribución total de cada variable *j*:

$$CONT_j = \sum_{a=1}^{n} cont_{a,j} \tag{6}$$

Las variables con la mayor contribución serán las que hayan influido más en la aparición del fallo.

b) Errores normalizados de las variables: Si una observación *x* tiene un valor del estadístico Q que supera el umbral, la contribución de cada variable se calcula como el cuadrado del error normalizado de cada variable x_j, [1] que es:

$$cont_{x_j} = (\hat{x}_j - x_j)^2 = e_j^2 \tag{7}$$

donde $\hat{x}_j$ es el valor de la variable predicha por el modelo PCA y e_j es el error de predicción,

c) Contribución a φ : La contribución a φ de la variable *i* se calcula como [3]:

$$c_i^\varphi = \left(\xi_i^T \Phi^{\frac{1}{2}} x\right)^2 \tag{8}$$

donde ξ_i es la *i-ésima* columna de la matriz identidad.

d) Indices basados en reconstrucción: El índice basado en la reconstrucción (RBC) [3] para una variable *i* se obtiene de la siguiente manera:

$$RBC_i^{index} = x^T M \xi_i \left(\xi_i^T M \xi_i\right)^{-1} \xi_i^T M x \tag{9}$$

donde *x* es la observación actual y la matriz M se calcula de la forma siguiente:

$$M = D = P\Lambda^{-1}P^T para\ T^2$$
$$M = \tilde{C} = \tilde{P}\tilde{P}^T\ para\ Q$$
$$M = \Phi = \frac{\tilde{C}}{T^2_\alpha} + \frac{D}{Q_\alpha}\Omega\ para\ \varphi$$

$\tilde{P}$ es la matriz formada por las últimas *m-a* columnas de V (matriz de vectores singulares).

e) Diagramas de contribución modificados: Este método se basa en encontrar las variables que, después de ser reconstruidas, consiguen las mayores reducciones del estadístico combinado φ hasta que este caiga por debajo de su umbral [4]. Para la variable reconstruida *i* la reducción del índice (o estadístico) combinado (RCI) es:

$$c_i^{RCI} = \left[(x_{nf} - x_{nf}^*)^T (\xi^T \Phi \xi)^{0.5} \xi_i\right]^2 \tag{10}$$

donde x_{nf} es el conjunto de variables anómalas y x_{nf}^* es la reconstrucción de esas variables anómalas:

$$x_{nf}^* = -\left(\xi^T \Phi \xi\right)^{-1} \xi^T \Phi \left(1 - \Gamma\right) x \tag{11}$$

donde Γ es una matriz diagonal con unos en las posiciones de la diagonal que corresponden con las variables anómalas y ceros en en las posiciones de las variables no anómalas.

La reducción del estadístico combinado después de la reconstrucción se puede calcular así:

$$\varphi - \varphi_{nf}^* = (x_{nf} - x_{nf}^*)^T (\xi^T \Phi \xi)(x_{nf} - x_{nf}^*) \tag{12}$$

Cuando el estadístico combinado cae por debajo de su umbral, las variables contenidas en el conjunto de variables anómalas, con mayores índices de contribución son las consideradas responsables del fallo.

B. Operadores lingüísticos

Para obtener una diagnosis global y más robusta, se requiere una toma de decisiones considerando las siete técnicas de diagnosis anteriores. Esto se implementa mediante operadores lingüísticos basados en OWA-RIM.

Operadores OWA: Los operadores OWA [6] son ampliamente conocidos y empleados, se definen como una función $F : \mathbb{R}^n \longrightarrow \mathbb{R}$ asociada al vector ω:

$$\omega = (\omega_1, \omega_2, \omega_3, ..., \omega_n) \tag{13}$$

donde $\omega_j \in [0, 1]$ y $1 \leq j \leq n$, cumpliéndose, además, que

$$\sum_{j=1}^{n} \omega_j = 1 \tag{14}$$

Si se dispone de un conjunto de resultados $< \alpha_1, \alpha_2, \alpha_3, ..., \alpha_n >$ como consecuencia de aplicar metodologías alternativas, y se ordenan de mayor a menor, resultando en $< \beta_1, \beta_2, \beta_3, ..., \beta_n >$, siendo β_1 el mayor y β_n el menor, el resultado de aplicar el operador OWA es:

$$F(\alpha_1, \alpha_2, \alpha_3, ..., \alpha_n) = \sum_{j=1}^{n} \omega_j \beta_j \tag{15}$$

En [10] se presentó una forma de determinar los elementos del vector de pesos usando cuantificadores lingüísticos *fuzzy*. El vector ω de pesos asociado a un cuantificador Q se obtiene de la siguiente forma:

$$\omega_j = Q\left(\frac{j}{n}\right) - Q\left(\frac{j-1}{n}\right) \tag{16}$$

con $j = 1, ..., n$.

En [11] se recopilan una serie de cuantificadores lingüísticos del tipo Regular Increasing Monotone (RIM), como las que aparecen en la Tabla I y que son empleados en este trabajo.

Table I
CUANTIFICADORES LINGÜÍSTICOS RIM

Término lingüístico	Función de pertenencia	Orness
There exists	$Q(x) = \begin{cases} 0 & \text{si } x = 0 \\ 1 & \text{si } x \neq 0 \end{cases}$	1
All	$Q(x) = \begin{cases} 0 & \text{si } x = 1 \\ 1 & \text{si } x \neq 1 \end{cases}$	0
Most (Wang) [12]	$Q(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq 0.3 \\ 2(x-0.3) & \text{si } 0.3 < x \leq 0.8 \\ 1 & \text{si } 0.8 < x \leq 1 \end{cases}$	0.45
At least half	$Q(x) = \begin{cases} 2x & \text{si } 0 \leq x \leq 0.5 \\ 1 & \text{si } 0.5 < x \leq 1 \end{cases}$	0.75
As many as possible	$Q(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq 0.5 \\ 2x & \text{si } 0.5 < x \leq 1 \end{cases}$	0.25
Average	$Q(x) = x \text{ si } 0 \leq x \leq 1$	0.5
More than α	$Q(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq \alpha \\ \frac{x}{1-\alpha} & \text{si } \alpha < x \leq 1 \end{cases}$	$\frac{1-\alpha}{2}$
At least α	$Q(x) = \begin{cases} \frac{x}{\alpha} & \text{si } 0 \leq x \leq \alpha \\ 1 & \text{si } \alpha < x \leq 1 \end{cases}$	$1 - \frac{\alpha}{2}$
Most (Pasi) [13]	$Q(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq 0.4 \\ 2(x-0.4) & \text{si } 0.4 < x \leq 0.9 \\ 1 & \text{si } 098 < x \leq 1 \end{cases}$	0.35
Most (Feng) [14]	$Q(x) = \begin{cases} 0 & \text{si } 0 \leq x \leq 0.5 \\ (2x-1)^{0.5} & \text{si } 0.5 < x \leq 1 \end{cases}$	0.33

El comportamiento del operador OWA va de un nivel más alto a un nivel menor de compensación: el operador es más tipo OR si no penaliza en exceso la presencia de elementos de bajo valor en la bolsa, y es más tipo AND si penaliza más la presencia de esos elementos. [6] propone una forma de clasificarlos según su comportamiento:

$$orness(\omega) = \frac{1}{n-1} \sum_{j=1}^{n} (n-j)\omega_j \tag{17}$$

El valor del *orness* de un operador puede tomar valores entre 0 y 1. Si es superior a 0.5, el operador es tipo OR, pero si es inferior a 0.5 es del tipo AND.

III. Diagnosis de Fallos

La metodología de diagnosis, como ya se ha comentado anteriormente, se basa en la descrita en [5]: la identificación de fallos se realiza en dos instantes de tiempo: t_0, justo después de la detección, y t_1 una vez que el sistema, en condiciones de funcionamiento defectuoso, ha llegado a un nuevo punto de funcionamiento estacionario. Se buscaba conocer el origen del fallo y, también, ver cómo evoluciona el sistema después de dicho fallo, para saber si es necesario realizar reparaciones, detener el proceso, continuar con él, etc. Por otro lado, también se busca conseguir una diagnosis más robusta aplicando varios métodos de identificación de fallos y alcanzando una diagnosis global. Combinando así las prestaciones de casa método mediante OWA-RIM basado en el cuantificador $Q(r)$ [15]:

$$Q_\alpha(r) = r^\alpha \tag{18}$$

que proporciona los pesos ω_i:

$$\omega_j = \left(\frac{j}{n}\right)^\alpha - \left(\frac{j-1}{n}\right)^\alpha \tag{19}$$

Y su nivel de *orness* es:

$$orness(Q_\alpha) = \frac{1}{1+\alpha} \tag{20}$$

En este artículo se introducen en la diagnosis propuesta, los cuantificadores descritos en la Tabla I, cuantificadores lingüísticos de fácil introducción en los entornos industriales de mantenimiento. Y los resultados se analizan mediante la conocida distancia de Damerau-Levenshtein [16], [17], entre la secuencia de diagnosis estimada y la conocida.

IV. Caso de estudio: Planta depuradora de aguas residuales

A. Depuradora de agaus residuales

La planta Benchmark Simulation Model 2 (BSM2) es un benchmark de amplio uso para estrategias de control en plantas depuradoras de aguas residuales [18]. Implementado en Matlab-Simulink, los datos que proporciona representan el estado del agua residual a la salida de distintos elementos de la red. Se dispone de 20 puntos de medida en los que se miden 7 variables, y, además, se dispone de 13 fallos implementados (más detalles en [5]).

1) Metodología experimental: Los parámetros de la diagnosis de fallos son: ventana temporal *Early* de 5 instantes de muestreo, y ventana temporal *Steady* de 5 instantes, $Th_{Steady} = 0.5$ y $Var_{Steady} = 0.001$.

Los cuantificadores lingüísticos a aplicar son los mostrados en la Tabla I. En el caso de *'More than α'* y *'At least α'*, el parámetro α tomará los siguientes valores: $\alpha = \{0.4, 0.5, 0.6\}$.

La evaluación del desempeño de cada cuantificador lingüístico OWA-RIM, que considera una agregación del resultado de la diagnosis de cada método FI considerado, se ha basado en el porcentaje de acierto en la identificación de las variables responsables del fallo. Para ello se ha analizado el grado de coincidencia entre las variables identificadas como responsables del fallo por un determinado método con las variables que se saben afectadas por el fallo. Dicho grado de coincidencia se ha medido con la distancia de Damerau-Levenshtein. Los pesos utilizados para calcular esta medida fueron: borrado de elementos: 0.1, introducción de elementos: 0.4, sustitución de elementos: 0.4 y transposición de elementos: 0.1. Es decir, está más penalizado introducir elementos nuevos o sustituirlos que eliminarlos o cambiarlos de posición. Por tanto, los métodos que no hayan encontrado alguna variable responsable estarán muy penalizados, al contrario de los que sí las hayan encontrado. Si la distancia de Damerau-Levenshtein obtenida es 0 implica una identificación perfecta (el método ha identificado todas las variable responsables del fallo, y no ha incluido ninguna más); y cuanto peor sea el resultado de identificación, es decir, no identifica alguna o ninguna de las variables responsables del fallo, mayor será esta distancia.

Cada fallo tiene una serie de variables en las que se origina dicho fallo, en la tabla II se detallan esas variables para cada fallo, así como se da una idea del tipo de fallo, se han considerado cuatro fallos distintos con diferentes magnitudes, dando un total de 13 conjuntos diferentes de datos de fallo. Cada método correspondiente a cada cuantificador, entregará una lista de variables ordenadas según la puntuación obtenida al aplicar OWA, ordenadas de mayor a menor. En algunos fallos, hay una variable responsable del fallo y en otros hay dos (ver tabla II), por tanto, cuando se haga la evaluación del rendimiento de cada cuantificador, se tomarán la primera o las dos primeras variables respectivamente, de la lista de variables entregada por cada método y se medirá la distancia de dichas variables con las realmente responsables del fallo.

Table II
Variables responsables de los fallos y tipos de fallo

	Fallos 1-5	**Fallos 6-8**	**Fallos 9-11**	**Fallos 12-13**
Variables	42 125	8 15	22	80 86
Tipo de fallo	Sensor de O_2	Alcalinidad en el influente	Alcalinidad en el reactor 1	Cambio en los flujos Q_r y Q_w

2) Resultados: El objetivo es analizar las prestaciones de los cuantificadores para una diagnosis más robusta, por tanto que coincida en la diagnosis conocida en el mayor grado posible. Los resultados de la identificación para la ventana *Early* se muestran en la tabla III.

Los fallos 1 a 5 son los más difíciles de identificar, y ahí es dónde los métodos dan, en general, peor resultado, mientras que en los fallos 6 a 13 la mayoría de métodos obtienen mejores resultados. Las mejores diagnosis (distancias más pequeñas) se han obtenido usando los cuantificadores del tipo *'Most'* (Wang, Pasi y Feng), *'As many as possible'*, *'More than* 0.4*'* y *'More than* 0.5*'* que tienen valores de *orness* moderadamente bajos (entre 0.25 y 0.45), es decir, requieren que hayan una mayoría de métodos que elijan a las mismas variables como candidatas (aunque no tienen por qué coincidir todos los métodos).

Table III
DIAGNOSIS VENTANA *Early* BASADA EN LA DISTANCIA DE DAMERAU-LEVENSHTEIN: $0 \rightarrow$ DIAGNOSIS PERFECTA.

Fallo	**There**	**All**	**Most** (Wang)	**At least half**	**As many**	**Average**	**More than** 0.4	**More than** 0.5	**More than** 0.6	**At least** 0.4	**At least** 0.5	**At least** 0.6	**Most** (Pasi)	**Most** (Feng)
1	0.8	0.8	0	0.8	0	0	0	0	0.3	0.8	0.8	0	0	0
2	0.8	0.8	0	0	0	0	0	0	0.3	0.8	0	0	0	0
3	0.8	0.3	0	0.8	0	0.8	0	0	0	0.8	0.8	0.8	0	0
4	0.8	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
5	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0.4	0.4	0	0	0	0	0	0	0	0	0	0	0	0
10	0.4	0.4	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Media	**0.392**	**0.3**	**0.092**	**0.215**	**0.092**	**0.154**	**0.092**	**0.092**	**0.138**	**0.277**	**0.215**	**0.1538**	**0.092**	**0.092**

Por contra, los métodos *'At least* 0.4*'*, *'At least* 0.5*'* y *'At least half'* con un *orness* entre 0.8 y 0.75, tienen unas de las puntuaciones peores de la comparativa, siendo los métodos con *orness* extremos (0 y 1), como *'There exists'* y *'All'*, los que dan un peor resultado.

Los métodos con *orness* entre 0.5 y 0.7 como son *'Average'* y *'At least* 0.6*'* consiguen resultados buenos, pero sin llegar a los de los mejores. A su vez, el método *'More than* 0.6*'*, con un orness de 0.2 también consigue un resultado aceptable.

Viendo estos resultados se puede conluir que la mayoría de los métodos de FI que se han usado dan un buen resultado en la diagnosis temprana, es decir diagnosticar el fallo justo en el momento en que se detecta el fallo. La agregación de todos los métodos FI da una diagnosis más correcta y robusta, sobre todo usando métodos OWA con cuantificadores que tienen valores *orness* bajos, en cuyo caso la distancia es cero, es decir, diagnostican siempre las variables responsables del fallo.

Los resultados de la identificación para la ventana *Steady* se muestran en la tabla IV. En este caso, los resultados se han igualado bastante entre los métodos. En los fallos 1 al 5 y 9 al 11, el fallo se propaga por varias zonas de la planta y es más complejo encontrar las variables causantes del fallo. El cuantificador *'There'* sigue siendo el peor método, y el resto han dado resultados muy similares, siendo *'All'* el mejor, seguido a poca distancia por *'More than* 0.4*'*, *'More than* 0.5*'* y *'More than* 0.6*'*. En cualquier caso, en esta ventana de identificación no se aprecian grandes diferencias entre usar un cuantificador u otro.

Estos resultados demuestran, que cada método de FI usado identifica variables diferentes como responsables del fallo, ya que ahora hay muchas variables afectadas por el fallo además de las que lo originaron. Por lo que diagnosticar fallos sólo en el estacionario después del fallo como hacen varios autores no es adecuado en todos los tipos de fallos, ya que la diagnosis calculada no es la correcta.

3) Ejemplo: fallo 2: Se muestran en la tabla V, como ejemplo, los resultados del fallo 2 (fallo en un sensor de O_2) obtenidos por cada uno de los métodos, en la que se muestran las puntuaciones para la ventana *Early*, obtenidas con cada método para las 6 primeras variables del ranking.

Los métodos *'There'* y *'At least 0.4'* no consiguen encontrar las variables responsables del fallo. Por contra, los métodos *'Most (Wang)'*, *'At least half'*, *'As many as possible'*, *'Average'*, *'More than* 0.4*'*, *'More than* 0.5*'*, *'At least* 0.5*'*, *'At least* 0.6*'*, *'Most (Pasi)'* y *'Most (Feng)'* han identificado las dos variables responsables del fallo lo que les permite obtener una distancia de 0 (identificación perfecta), como se ve en la tabla III. *'More than* 0.6*'* tiene una variable responsable en la primera posición, pero la segunda no es una de las responsables, por tanto, su distancia es de 0.3 (tabla III). El método *'All'* no identifica ninguna variable como responsable del fallo ya que sus contribuciones son todas cero.

En la tabla VI se muestran los resultados para el fallo 2 con la ventana *Steady*. En este caso, las puntuaciones OWA son similares entre las variables, lo que indica la expansión del fallo por la planta. *'All'* es el único método capaz de identificar a una de las variables responsables del fallo, el resto no tienen dichas variables en las dos primeras posiciones, por tanto, obtienen una peor distancia (ver tabla IV).

V. CONCLUSIONES

En este artículo se ha realizado una comparativa entre diferentes variantes de un método de identificación de fallos multi-método cuando se introducen nuevos cuantificadores lingüísticos. Cada método de diagnosis realiza un identificación, y se obtiene una diagnosis global agregando la solución de cada método FI, utilizando operadores OWA-RIM con diferentes cuantificadores lingüísticos, comprobando el nivel de robustez de la identificación obtenida con cada cuantificador estudiado.

Los resultados muestran que los cuantificadores con valores de *orness* comprendidos entre 0.3 y 0.5 son los que mejor funcionan en la identificación inmediatamente posterior a la detección del fallo. Además estos métodos dan una diagnosis correcta, ya que la distancia calculada es cero. Mientras, en la identificación que se realiza cuando el sistema llega a un estado estacionario al evolucionar después del fallo, los resultados se igualan para todos los cuantificadores y dan peores resultados.

Como trabajo futuro, necesario para mejorar los resultados de la identificación en la ventana estacionaria, consistirá en

Table IV
Diagnosis Ventana *Steady* basada en la Distancia de Damerau-Levenshtein: 0 → diagnosis perfecta.

Fallo	**There**	**All**	**Most** (Wang)	**At least half**	**As many**	**Average**	**More than** 0.4	**More than** 0.5	**More than** 0.6	**At least** 0.4	**At least** 0.5	**At least** 0.6	**Most** (Pasi)	**Most** (Feng)
1	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
2	0.8	0.3	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
3	0.8	0.3	0.8	0.8	0.8	0.8	0	0	0	0.8	0.8	0.8	0.8	0.8
4	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
5	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8	0.8
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
10	0.4	0.4	0	0.4	0.4	0	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0
11	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
15	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0
Media	**0.446**	**0.323**	**0.369**	**0.4**	**0.4**	**0.369**	**0.338**	**0.338**	**0.338**	**0.4**	**0.4**	**0.4**	**0.4**	**0.369**

Table V
Resultados de cada cuantificador. Fallo 2 Ventana *Early*

Método	**Variables**	**Puntuaciones**
There	62 69 76 83 56 55	0.88 088 0.84 0.84 0.84 0.81
All	1 2 3 4 5 6	0 0 0 0 0 0
Most (Wang)	42 125 49 86 82 21	0.35 0.26 0.22 0.088 0.07 0.07
At least half	42 125 55 59 79 7	0.48 0.48 0.47 0.46 0.45 0.43
As many	42 125 49 86 21 82	0.6 0.52 0.38 0.13 0.11 0.1
Average	42 125 55 49 59 79	0.33 0.28 0.24 0.23 0.23 0.23
More than 0.4	42 125 49 86 82 20	0.51 0.43 0.33 0.16 0.13 0.12
More than 0.5	42 125 49 86 21 82	0.61 0.52 0.38 0.13 0.11 0.1
More than 0.6	42 49 125 86 21 82	0.69 0.36 0.16 0.11 0.1 0.09
At least 0.4	55 59 79 7 14 62	0.58 0.57 0.56 0.52 0.52 0.5
At least 0.5	42 125 55 59 79 7	0.48 0.48 0.47 0.46 0.45 0.43
At least 0.6	42 125 55 59 79 7	0.46 0.45 0.4 0.38 0.38 0.37
Most (Pasi)	42 125 49 86 82 21	0.27 0.18 0.16 0.06 0.05 0.05
Most (Feng)	42 125 49 86 21 82	0.27 0.19 0.17 0.05 0.05 0.04

Table VI
Resultados de cada cuantificador.Fallo 2. Ventana *Steady*

Método	**Variables**	**Puntuaciones**
There	19 12 5 123 94 100	0.91 0.91 0.89 0.89 0.89 0.89
All	125 1 2 3 4 5	0.04 0 0 0 0 0
Most (Wang)	62 69 56 125 49 42	0.79 0.56 0.49 0.49 0.49 0.37
At least half	61 68 50 42 125 43	0.64 0.58 0.47 0.47 0.47 0.42
As many	61 68 50 42 125 43	0.64 0.58 0.47 0.47 0.47 0.42
Average	62 69 42 56 125 49	0.69 0.56 0.49 0.49 0.49 0.4
More than 0.4	62 69 56 42 125 49	0.68 0.55 0.49 0.49 0.47 0.45
More than 0.5	62 69 56 42 125 49	0.69 0.56 0.49 0.49 0.49 0.4
More than 0.6	62 69 56 42 125 49	0.7 0.58 0.49 0.49 0.47 0.39
At least 0.4	121 97 103 109 115 114	0.61 0.61 0.61 0.61 0.61 0.61
At least 0.5	61 68 50 42 125 43	0.64 0.58 0.47 0.47 0.47 0.41
At least 0.6	61 68 42 125 50 43	0.58 0.53 0.45 0.45 0.4 0.34
Most (Pasi)	62 69 56 49 42 125	0.64 0.39 0.35 0.35 0.35 0.27
Most (Feng)	62 69 56 49 42 125	0.64 0.41 0.36 0.36 0.36 0.28

estudiar la distancia de Damerau-Levenshtein con cadenas de tamaño diferente, o introduciendo el valor de las contribuciones de cada variable en el cálculo de la distancia, no sólo su posición.

References

[1] T. Kourti and J. MacGregor, "Multivariate SPC methods for process and product monitoring," *Journal of Quality Technology*, vol. 28, pp. 409–428, 1996.

[2] H. Yue and S. J. Qin, "Recontruction based fault identification, using a combined index," *Industrial & Engineering Chemistry Research*, vol. 40(20), pp. 4403–4414, 2001.

[3] C. F. Alcala and S. J. Qin, "Reconstruction-based contribution for process monitoring," *Automatica*, vol. 45, pp. 1593–1600, 2009.

[4] J. Liu and D.-S. Chen, "Fault isolation using modified contribution plots." *Computers & Chemical Engineering*, vol. 61, pp. 9–19, 2014.

[5] A. Sánchez-Fernández, G. Sainz-Palmero, J. Benítez, and M. Fuente, "Linguistic OWA and two time-windows based fault identification in wide plants," *Computers & Chemical Engineering*, vol. 115, pp. 412–430, jul 2018.

[6] R.R. Yager, "On ordered weighted averaging aggregation operators in multi-criteria decision making," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 18, pp. 183–190, 1988.

[7] R. Dunia and S. J. Qin, "Subspace approach to multidimensional fault identification and reconstruction." *AIChE Journal*, vol. 44, pp. 1813–1831, 1998.

[8] J. E. Jackson, *A user's guide to principal components*. Wiley, 1991.

[9] P. Nomikos and J. F. MacGregor, "Multivariate SPC charts for monitoring batch processes," *Technometrics*, vol. 37, no. 1, pp. 41–59, 1995. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/00401706.1995.10485888

[10] R. Yager, "Quantifier guided aggregation using OWA operators," *International Journal of Intelligent Systems*, vol. 11, pp. 49–73, 1996.

[11] X. Liu and S. Han, "Orness and parameterized rim quantifier aggregation with owa operators: A summary," *International Journal of Approximate Reasoning*, vol. 48, no. 1, pp. 77–97, apr 2008, special Section: Perception Based Data Mining and Decision Support Systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0888613X07000709

[12] J. Wang and Y.-I. Lin, "A fuzzy multicriteria group decision making approach to select configuration items for software development," *Fuzzy Sets and Systems*, vol. 134, no. 3, pp. 343–363, 2003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016501140200283X

[13] G. Pasi and R. R. Yager, "Modeling the concept of majority opinion in group decision making," *Information Sciences*, vol. 176, no. 4, pp. 390–414, 2006, recent advancements of fuzzy sets: theory and practice. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025505002227

[14] L. Feng and T. Dillon, "Using fuzzy linguistic representations to provide explanatory semantics for data warehouses," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 1, pp. 86–102, jan 2003.

[15] R. Yager, "Families of OWA operators," *Fuzzy Sets and Systems*, vol. 59, pp. 125–148, 1993.

[16] F. J. Damerau, "A technique for computer detection and correction of spelling errors," *Commun. ACM*, vol. 7, no. 3, pp. 171–176, Mar. 1964. [Online]. Available: http://doi.acm.org/10.1145/363958.363994

[17] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Physics Doklady*, vol. 10, p. 707, 1966.

[18] J. Alex, L. Benedetti, J. Copp, K. Gernaey, U. Jeppsson, I. Nopens, M. Pons, C. Rosen, J. Steyer, and P. Vanrolleghem, "Benchmark simulation model no. 2," 2008.