

Reconocimiento de clases de aeronaves con base en patrones del ruido en el despegue

Luis Alejandro Sánchez Pérez, Luis Pastor Sánchez-Fernández, Sergio Suarez-Guerra

Instituto Politécnico Nacional,
Centro de Investigación en Computación, Ciudad de México,
México

l.alejandro.2011@gmail.com, lsanchez@cic.ipn.mx, ssuarez@cic.ipn.mx

Resumen. En este trabajo se examina el reconocimiento de la clase de aeronaves a partir de patrones del ruido en el despegue. Se analiza la segmentación de la señal en tiempo y el uso de una red neuronal MLP por cada segmento. Asimismo, se examinan varios algoritmos de decisión por comité para la agregación de las múltiples salidas de los clasificadores paralelos, así como la extracción y selección de características con base en el análisis del espectro del ruido de aeronaves. Por otro lado, se explora un método para estimar la trayectoria georeferenciada durante el despegue únicamente a partir de la señal. La metodología y los resultados están sustentados en la literatura actual.

Palabras clave. Ruido en aeropuertos, clasificación de aeronaves, segmentación de la señal, redes neuronales.

Aircraft Class Recognition based on Take-off Noise Patterns

Abstract. In this work the aircraft class recognition of based on take-off noise patterns is examined. Signal segmentation in time is analyzed as well as using a MLP neural network as the classifier for each segment. Also, several algorithms for decision by committee in order to aggregate the multiple parallel outputs of the classifiers are examined along with feature extraction and selection based on spectrum analysis of the aircraft noise. Also, a method for georeferenced estimation of the take-off flight path based only on the noise signal is explored. The methodology and results are sustained in the current literature.

Keywords. Airport noise, aircraft class recognition, signal segmentation; neural network.

1. Introducción

El ruido en aeropuertos es una preocupación importante en las sociedades modernas. La extensión así como la cantidad de aeropuertos que están siendo construidos es cada vez mayor, con el fin de poder manejar el tráfico aéreo en crecimiento [1-3]. En este sentido, mucho esfuerzo se ha realizado para abordar el problema del ruido. Las principales áreas donde se viene trabajando para lograr mejoras en el entorno acústico de los aeropuertos son la reducción en la fuente, la planificación y gestión del uso del suelo, los procedimientos operativos de reducción y las restricciones de operación [4].

La clave en relación con los esfuerzos anteriores es la identificación de un problema de ruido en el aeropuerto [4]. En general existen dos técnicas para evaluar la situación sonora, que son medir o estimar los niveles de ruido en ubicaciones alrededor del aeropuerto. Por un lado, se han definido varios métodos para calcular las curvas de ruido alrededor de los aeropuertos [5-9]. Una curva de ruido es una línea de valor constante que representa el promedio durante largos períodos de tiempo calculado con base en un día común en el año. Por otra parte, el monitoreo permanente del ruido y las operaciones de aeronaves en las proximidades de los aeropuertos se ha convertido en una práctica común [10-12]. En estos casos, las señales registradas en cada evento se utilizan solamente para el cálculo de algunos indicadores estadísticos, tales como el nivel sonoro continuo equivalente.

La condición sonora de los puntos en tierra debido a las aeronaves que vuelan hacia y desde

un aeropuerto depende de una serie de factores. Entre ellos, la clase de aeronave que usa el aeropuerto es el más importante, cuando se lleva a cabo una evaluación o acción en particular [4]. Por ejemplo, todas las aeronaves que aterrizan o despegan del *Ronald Regan National Airport* después de las 9:59 pm y antes de las 7:00 am están sujetas a la regulación *DCA Night-time Noise Rule* impuesta por la *Metropolitan Washington Airports Authority* [13].

En este sentido y a fin de extraer mayor información de las señales registradas, se propone el reconocimiento de la clase de aeronave a partir de patrones del ruido en el despegue con base en la segmentación de las señales, así como la estimación de la trayectoria georreferenciada, añadiendo así potencialidad y robustez en el monitoreo y evaluación sin excluir la posibilidad de validación a partir de múltiples fuentes de información.

2. Reconocimiento de la clase de aeronave usando segmentación en tiempo

El espectro del ruido de las aeronaves cambia en amplitud y frecuencia durante un despegue debido a múltiples factores. La señal muestreada $x(t)$ en un punto receptor puede ser definida de acuerdo a (1), donde A_{ij} es la amplitud, w_j es la frecuencia y φ_j es la fase. La señal $x(t)$ puede ser representada con matrices como se muestra en la Eq. (2), donde cada fila denota un lapso de tiempo disjuncto mientras que cada columna define una componente de frecuencia como se muestra en la Eq. 1 y 2:

$$x(t) = \begin{bmatrix} A_{11} \cos(w_1 t + \varphi_1) \\ A_{21} \cos(w_1 t + \varphi_1) \\ \vdots \\ A_{m1} \cos(w_1 t + \varphi_1) \end{bmatrix} + \dots + \begin{bmatrix} A_{1n} \cos(w_n t + \varphi_n) \\ A_{2n} \cos(w_n t + \varphi_n) \\ \vdots \\ A_{mn} \cos(w_n t + \varphi_n) \end{bmatrix}, \quad \begin{matrix} 0 \leq t \leq t_1 \\ t_1 < t \leq t_2 \\ \vdots \\ t_{m-1} < t \leq t_m \end{matrix} \quad (1)$$

cualquier frecuencia puede o no estar presente si la amplitud relativa A_{ij} es despreciable o no:

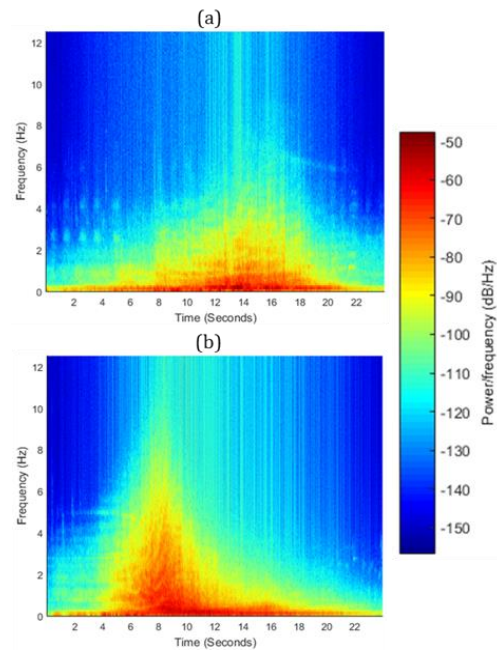


Fig. 1. Espectrograma de dos señales de ruido de aeronaves muestreadas a 25 kHz: (a) ATR 42-500, (b) Boeing 737-200

$$X = \begin{bmatrix} A_{11} & \dots & A_{1n} \\ A_{21} & \dots & A_{2n} \\ \vdots & & \vdots \\ A_{m1} & \dots & A_{mn} \end{bmatrix} \times \cos \left(\begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \times t + \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \vdots \\ \varphi_n \end{bmatrix} \right) \quad (2)$$

Cada operación de despegue produce una combinación única de parámetros A_{ij} , w_j y φ_j , debido a muchos factores como la atenuación atmosférica [14,15], el efecto Doppler y otras cuestiones inherentes al ruido de las aeronaves como el mecanismo de generación de ruido que tiene direccionalidad variable [16] y la atenuación lateral [17, 18]. Una revisión completa está disponible en [19]. Este fenómeno se ilustra en la Fig. usando espectrogramas. Las áreas azules denotan frecuencias despreciables mientras que las áreas rojas representan las significativas.

Más aún, la distribución en tiempo de las componentes de frecuencia es diferente en cada espectrograma.

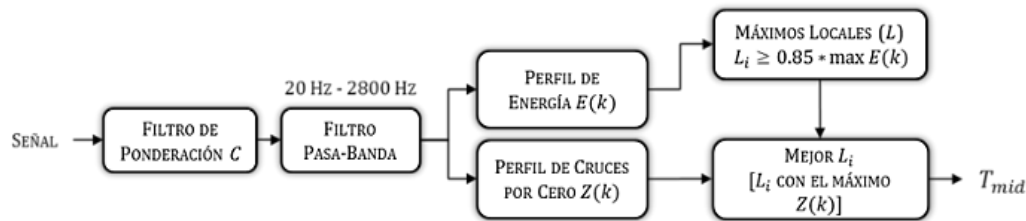
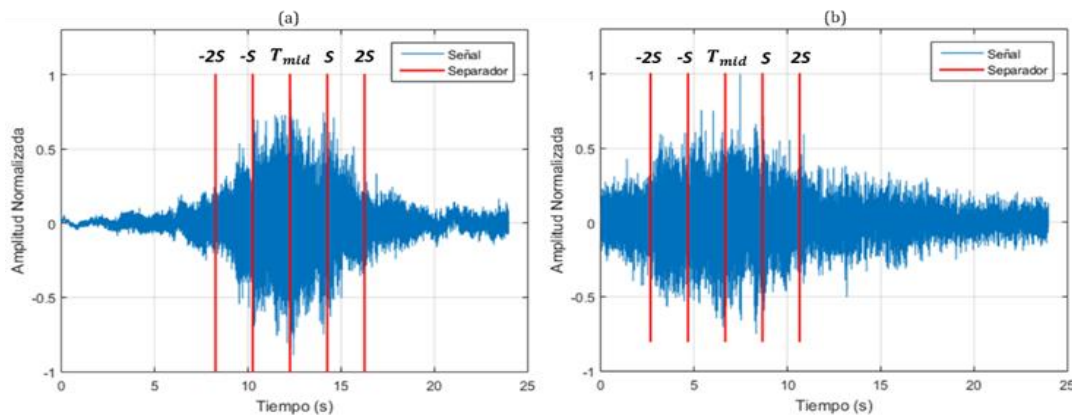
Fig. 2. Algoritmo para calcular T_{mid} 

Fig. 3. Ubicación de los segmentos: (a) Fokker F100, (b) Airbus A320

Varios métodos han sido propuestos para identificar la clase de aeronave que despegue usando rasgos del espectro de la señal [20-23].

Todos ellos usan la señal completa como entrada principal para la extracción de rasgos. Sin embargo, la señal del ruido de las aeronaves es un proceso no estacionario que conlleva a la variación del espectro durante el despegue. Por lo tanto, al usar la señal completa se pudiera enmascarar ciertos rasgos temporales que solo están presentes durante un corto período de tiempo durante el despegue. En esta sección se propone un modelo computacional para identificar las clases de aeronaves con base en la extracción selectiva de rasgos a partir de diferentes segmentos de la señal.

2.1. Segmentación de la señal en tiempo

La segmentación en el tiempo implica definir un punto común para todas las señales, a partir del cual se pueda extraer los segmentos. En esta

sección, dicho punto es nombrado T_{mid} e intenta representar la posición de la aeronave respecto al punto de medición más cercana durante el despegue. Encontrar este punto no es un problema trivial ya que no se cuenta con información espacial. Varios métodos conocidos son evaluados con el objetivo de hallar T_{mid} , tales como el máximo de amplitud, el máximo de energía y el centro de masa [24]. Todos generan resultados erróneos en un porcentaje significativo de las mediciones, lo que se discute a detalles en [19].

En esta sección se propone un algoritmo para estimar T_{mid} utilizando la energía y el cruce por cero, como se muestra en la Fig. 2.

El punto T_{mid} es definido como el instante de tiempo correspondiente al máximo local, superior o igual al $0.85[\max E(q)]$ con el mayor valor $Z(q)$, donde $E(q)$ es el perfil de la energía calculado usando (3) y $Z(q)$ es el perfil de cruce de ceros obtenido mediante la Eq. (4 y 5):

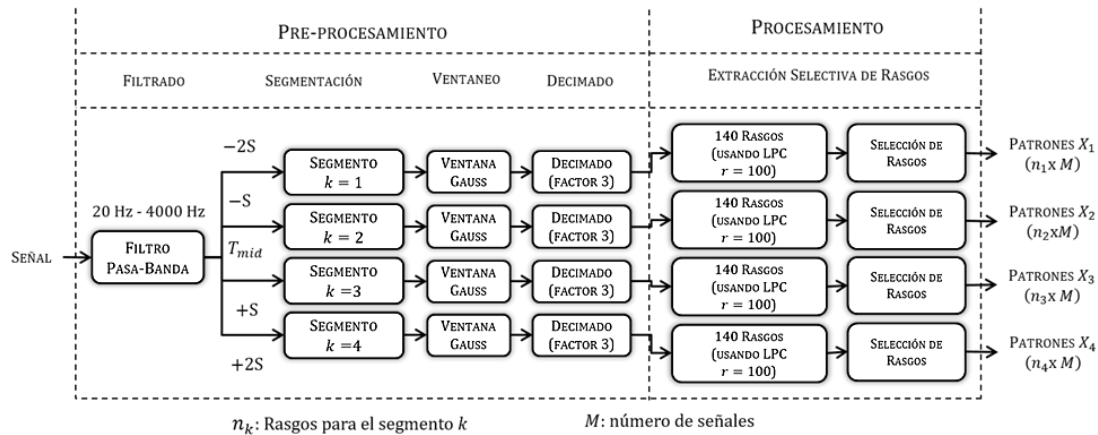


Fig. 4 Procedimiento de extracción de rasgos selectivo

$$E(q) = \sum_{x=n}^{n+S_E} |y(x)|^2, \quad (3)$$

$$F(x) = \begin{cases} 1, & \text{sign}[y(x)] \neq \text{sign}[y(x+1)] \\ 0, & \text{otherwise} \end{cases}, \quad (4)$$

$$Z(q) = \sum_{x=n}^{n+S_E-1} F(x),$$

$$\forall q \mid q = 1, 2, \dots, \left\lfloor \frac{(N - S_E)}{S_E(1 - O_{S_E}/100)} \right\rfloor, \quad (5)$$

$$n = (q - 1) \left(S_E \left(1 - \frac{O_{S_E}}{100} \right) \right),$$

donde q denota un fragmento de la señal $y(x)$, S_E representa que la longitud del fragmento q , N denota la longitud de la señal, $\text{sign}[\]$ simboliza el operador signo, O_{S_E} indica el porcentaje de solapamiento, y $\lfloor \]$ denota el entero menor más cercano.

En esta sección se usan los siguientes valores: $S_E = 200$ ms, $O_{S_E} = 20\%$ y $0.85[\max E(k)]$, como umbral.

La ubicación de los segmentos se determina de la siguiente manera: si el número de segmentos (N_f) es par, se seleccionan $N_f/2$ segmentos del mismo tamaño a cada lado de T_{mid} , si N_f es impar, entonces se seleccionan $(N_f - 1)/2 + 1$ segmentos delante de T_{mid} y $(N_f - 1)/2$ detrás de

T_{mid} . La Fig. 3 muestra las ubicaciones de cuatro segmentos extraídos para dos señales diferentes.

2.2. Extracción de rasgos

Dado que la frecuencia de muestreo es de 25 kHz, el rango espectral es 12 kHz ($F_s/2$). Además, como la longitud de los segmentos es de 2s, se tienen 50000 muestras en cada segmento. Consecuentemente, la precisión espectral es de 0.5 (F_s/N) y la longitud del espectro es 25000 puntos. A fin de hacer la clasificación computacionalmente aceptable, se debe realizar una reducción dimensional.

Los coeficientes LPC se obtienen mediante el análisis de predicción lineal [25]. En general, estos coeficientes se utilizan para describir el tracto vocal como un filtro IIR (Respuesta Infinita al Impulso). Este filtro puede ser descrito usando la Eq. (6):

$$H_s(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{i=1}^r a_i z^{-i}}. \quad (6)$$

Las técnicas que usan coeficientes LPC están fuertemente relacionadas con la idea de un codificador de voz mediante la simulación del tracto vocal humano [26-28]. Sin embargo, estos también han sido usados para representar la envolvente del espectro de frecuencia de una señal y lograr una reducción dimensional del mismo [21, 22]. El procedimiento de selección de

Tabla 1 Clases y Distribución de Patrones entre Conjuntos de Entrenamiento, Validación y Prueba

Clases de aeronave (Etiqueta en el modelo)	Conjuntos de patrones Entrenamiento/ Validación/ Prueba
A320 1 (A320 1)	08/03/2005
A320 2 (A320 2)	08/03/2006
A320 3 (A320 3)	08/03/2006
A320, B737-800 (A320 B737-800)	08/03/2007
ATR-42 (ATR-42)	08/03/2003
B737-100, B737-200 (B737-1/200)	08/03/2011
B737-600, B737-700 (B737-6/700)	08/03/2009
B747-400 (B747-400)	08/03/2004
F100 (F100)	08/03/2006
F100 2 (F100 2)	08/03/2006
F100, B737-200 (F100 B737-200)	08/03/2003
F100, B737-200 2 (F100 B737-200 2)	08/03/2004
MD87, MD88 (MD)	08/03/2009
TOTAL	104/39/79

rasgos hasta la obtención de los conjuntos de patrones X_1, X_2, X_3 y X_4 se muestra en la Fig. 4, (Refiérase a [19] para más detalles).

Con el objetivo de mejorar la clasificación es necesario qué rasgos son redundantes o irrelevantes. En esta sección se utiliza el método de selección de rasgos para redes neuronales MLP propuesto por [29]. Este utiliza un criterio $\hat{s}^P(j)$ para medir la importancia del rasgo j^{th} .

El criterio $\hat{s}^P(j)$ puede ser usado con eliminación recursiva (RFE, por sus siglas en inglés). En este enfoque se elimina el rasgo menos importante (menor $\hat{s}^P(j)$), de manera recursiva por medio de entrenamientos sucesivos de la red neuronal MLP. Este método en conjunto, denominado MLP-FSPP-RFE, es menos costoso computacionalmente y generalmente provee mejores resultados que otros métodos [29].

En esta sección, se aplica el método de selección de rasgos MLP-FSPP-RFE para cada segmento. En una iteración, la red neuronal correspondiente al segmento analizado se entrena 10 veces y se pone a prueba con 39 patrones (3

para cada clase), obteniéndose así el error de prueba. La red con el error de prueba más bajo se selecciona para calcular $\hat{s}^P(j)$. Luego, se eliminan los 5 rasgos menos importantes. El algoritmo se detiene cuando no se observa un claro avance con más eliminaciones. Cada red neuronal tiene una capa oculta de 30 neuronas (ninguna mejora significativa se consigue variando este número en el intervalo 25-45). La función de activación de la capa oculta y de salida es la función tangente hiperbólica. Las clases y la distribución de los patrones entre los conjuntos de entrenamiento, validación y prueba, se muestran en la Tabla 1. Para un análisis detallado de los resultados refiérase a [19].

2.3. Modelo computacional

Dado que cada red neuronal genera una salida, se necesita un modelo que tome ventaja de estas múltiples respuestas. En esta sección se propone un nuevo modelo computacional que mejora la clasificación por medio de cuatro redes neuronales

con diferentes rangos de acuerdo a su desempeño, las cuales interactúan como en un sistema de votación. La Fig. 5 muestra la topología del modelo computacional.

Cada red neuronal tiene un rango $I_r \in \{1,2,3,4\}$, relacionado con su rendimiento durante el entrenamiento y calculado con (7). El subíndice r no está asociada con el número del segmento y cada rango I_r debe cumplir con (8), de modo que NN_{I_1} es la red con el rango más bajo (I_1), mientras que NN_{I_4} es la red con el rango más alto (I_4):

$$I_r = 1 - \text{error de validación de la red } NN_{I_r}, \quad (7)$$

$$I_1 \leq I_2 \leq I_3 \leq I_4. \quad (8)$$

Cada salida $O_{NN_{I_r}}$ de la red NN_{I_r} se pondera en relación con su rango (I_r) por un factor a_r asociado con las dos salidas ponderadas anteriores F_{r-2} y F_{r-1} . Los valores de a_3 y a_4 se calculan mediante (9), mientras que $\alpha_1 = \alpha_2 = 0$ porque no existen dos salidas previas en ningún caso:

$$\alpha_3 = \alpha_4 = (I_4 - I_1) * 2. \quad (9)$$

Además, los valores a_3 y a_4 se actualizan de acuerdo a las siguientes reglas:

$$\text{Si } \text{argmax}(F_1) = \text{argmax}(F_2),$$

$$\text{entonces: } \alpha_3 = -\alpha_3;$$

$$\text{Si } \text{argmax}(F_2) = \text{argmax}(F_3),$$

$$\text{entonces: } \alpha_4 = -\alpha_4,$$

cuando $a_r \neq 0$, lo que se busca es aumentar la importancia de la salidas de la red NN_{I_r} cuando las clases identificadas por las redes neuronales $NN_{I_{r-2}}$ y $NN_{I_{r-1}}$ difieren y se disminuye la importancia cuando las clases identificadas son iguales.

2.4. Resultados y discusión

La base de datos introducida en [22], es usada para entrenar las redes neuronales y probar el modelo computacional. Por lo tanto, se utilizaron 143 señales para entrenamiento y 79 para prueba. En resumen, el modelo propuesto en esta sección se desempeña mejor que los modelos existentes,

reduciendo el número de errores del 9/79 a 4/79. Esta mejora está relacionada fundamentalmente con dos factores. En primer lugar, el proceso de selección de rasgos mejora el desempeño individual de la red neuronal de cada segmento. En segundo lugar, el modelo se beneficia de múltiples respuestas. Ambos factores son posibles debido a la segmentación de la señal que se propone en esta sección, la cual permite extraer información relevante de la frecuencia con respecto al tiempo, lo cual no se había realizado, incluso para el seguimiento continuo de la similitud del ruido de entrada y los sonidos de las aeronaves [30].

Además, se realizaron mediciones en cuatro días distintos con el fin de probar el modelo computacional en tiempo real. En total, 83 mediciones de despegue se clasificaron en tiempo real bajo diferentes condiciones climáticas (con excepción de lluvia) y diversos ruidos ambientales, como aves, aeronaves rodando, ladridos, entre otros. La efectividad mínima obtenida es del 85%, al menos un 5% superior a la reportada en [22].

3. Agregación jerárquica dinámica de salidas paralelas

El modelo propuesto en la sección anterior demuestra que la extracción de rasgos de segmentos adyacentes de la señal de ruido en el despegue aumenta el porcentaje de reconocimiento que al usar toda la señal. La motivación para la segmentación surge de considerar el ruido de las aeronaves en el despegue como un proceso no estacionario con espectro dinámico. El objetivo es dividir la señal en K segmentos con características espectrales diferentes. En este sentido, el modelo propuesto en la Sección 2.3 utiliza una red NN_k por cada segmento k y rasgos basados en los LPC.

Por otra parte, en [23], se evalúan concurrentemente rasgos extraídos a partir de los MFCC y bandas de 1/24 de octava evaluados por dos redes paralelas. En ambos modelos se define un algoritmo de agregación para ponderar las múltiples salidas producidas por las redes. En la Sección 2 la agregación se basa en una ponderación dinámica de la salida O_p^k con respecto

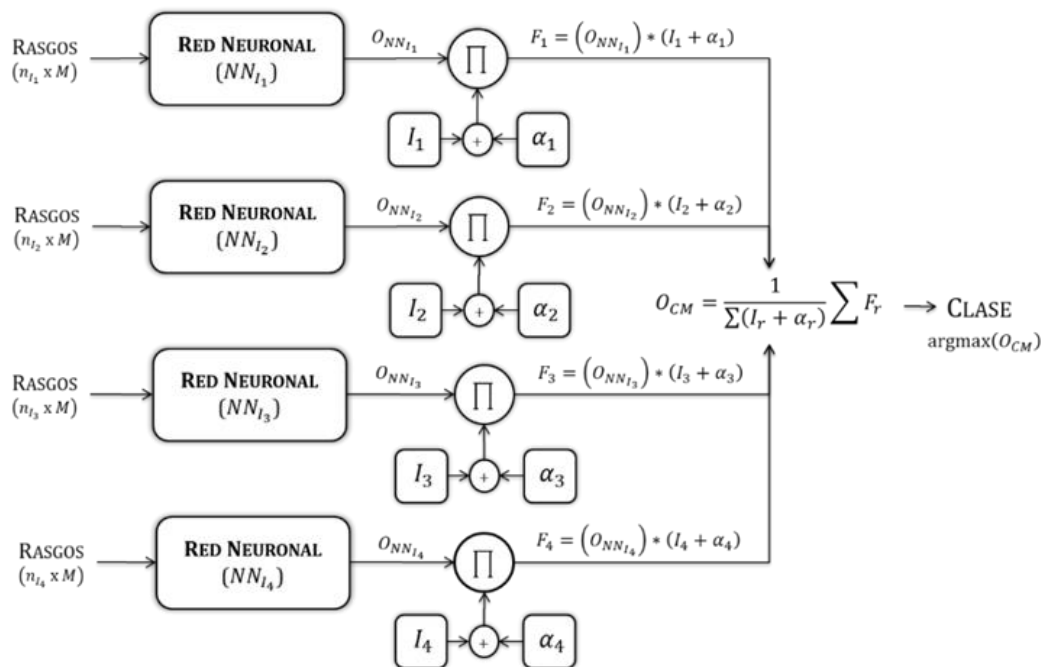


Fig. 5 Topología del modelo computacional

a la clase p de la red neuronal NN_k , la cual es entrenada para reconocer rasgos extraídos del segmento k . Sin embargo, el peso w_p^k para la salida O_p^k es el mismo para todas las clases de aeronaves $p = 1, 2, \dots, P$.

Por otra parte, la agregación propuesta en [23] se realiza con base en la suma ponderada $\sum_k w_p^k O_p^k$. Sin embargo, el peso w_p^k es estático y no cambia según el contexto (en este caso, k representa el tipo de rasgos utilizado: MFCC u octavas). Más aún, los métodos de agregación anteriores no están diseñados para contrarrestar la salida del mejor clasificador individual con respecto a la clase p , cuando este regresa un falso positivo para esa misma clase p .

Además, w_p^k se establece teniendo en cuenta únicamente los verdaderos positivos con respecto a la clase p .

En esta sección se define un modelo para la identificación de la clase p con base en la agregación jerárquica dinámica de las salidas O_p^k de K redes neuronales paralelas [31]. El modelo pondera dinámicamente cada salida O_p^k de la red neuronal NN_k con respecto a la clase p usando

una jerarquía de las redes neuronales. La jerarquía se determina con base a un criterio de ordenamiento definido específicamente para cada clase p de acuerdo al desempeño durante el entrenamiento. El criterio de ordenamiento utiliza la métrica F_β que combina la precisión y exhaustividad ("precision and recall") de la red neuronal NN_k con respecto a cada clase p .

3.1. Definición analítica del modelo de la sección 2

El modelo de la Sección 2 propone una red NN_k para cada segmento de señal k .

Dicho modelo propone la ponderación jerárquica dinámica de las salidas O_p^k de acuerdo con (10) y (11). La etiqueta prevista o nombre de la clase reconocida dada una entrada $\{x_1, \dots, x_K\}$, se determina una vez que la red NN_k ha evaluado el patrón x_k , resultando en la salida O_p^k , la cual se pondera de forma dinámica usando $w_k + f(k)$. El valor de w_k se calcula usando (12), con base en el error de validación e_k de la red NN_k , durante la fase de entrenamiento, mientras que la función

$f(k)$ devuelve un factor de ajuste α_k , de acuerdo a las salidas de las dos redes neuronales de menor rango anteriores $O_p^{i|\vec{r}(i)=\vec{r}(k)-2}$ y $O_p^{j|\vec{r}(j)=\vec{r}(k)-1}$. El valor de α_k se calcula usando (13). Además, el vector $\vec{r}(k)$ representa los rangos de las redes neuronales $NN_k | k = 1, 2, \dots, K$ de acuerdo a los pesos $w_k | k = 1, 2, \dots, K$ de forma tal que $\vec{r}(k)$ contiene la posición ordenada de manera ascendente de w_k como se muestra en (14) y (15). Por lo tanto, la componente $\vec{r}(k)$ denota el rango de la red neuronal NN_k , de tal forma que un valor más alto indica una mayor importancia y en consecuencia un mayor rango:

$$\operatorname{argmax}_{p=1,2,\dots,P} \left\{ \left(\begin{bmatrix} O_1^1 & O_1^2 & \dots & O_1^K \\ O_2^1 & O_2^2 & \dots & O_2^K \\ \vdots & \vdots & \ddots & \vdots \\ O_P^1 & O_P^2 & \dots & O_P^K \end{bmatrix} \begin{bmatrix} w_1 + f(1) \\ w_2 + f(2) \\ \vdots \\ w_{k=K} + f(K) \end{bmatrix} \right) \times \right. \\ \left. (1/\sum_{k=1}^K (w_k + f(k))) \right\}, \quad (10)$$

$$f(k) = \begin{cases} 0, & \vec{r}(k) \leq 2 \\ \alpha_k, & \vec{r}(k) > 2, \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{i|\vec{r}(i)=\vec{r}(k)-2}\} \neq \\ & \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{j|\vec{r}(j)=\vec{r}(k)-1}\}, \\ -\alpha_k, & \text{otherwise} \end{cases} \quad (11)$$

$$w_k = 1 - e_k, \quad (12)$$

$$\alpha_k = 2 \left(w_{i|\vec{r}(i)=K} - w_{j|\vec{r}(j)=1} \right), \quad (13)$$

$$\forall (w_1, w_2, \dots, w_K) | (w_1 < w_2 < \dots < w_K) \rightarrow \vec{r} = [1, 2, \dots, K], \quad (14)$$

$$\forall (w_1, w_2, \dots, w_K) | (w_K < w_{K-1} < \dots < w_1) \rightarrow \vec{r} = [K, K-1, \dots, 1]. \quad (15)$$

3.2. Modelo propuesto

En el algoritmo de agregación anterior las salidas O_p^k y $O_{p+n}^k | n \in \mathbb{Z} - \{0\} \wedge 1 \leq p+n \leq P$ son ponderadas de igual forma puesto que el peso w_k es el mismo, independientemente del desempeño de red neuronal NN_k con respecto a la clase p o

$p+n$, es decir, el error de validación e_p^k de la red NN_k con respecto a la clase p no se examina individualmente.

Más aún, el rendimiento de las redes neuronales $NN_k | k = 1, 2, \dots, K$ no se adhiere necesariamente a (16):

$$\forall (p+n) | n \in \mathbb{Z} - \{0\} \wedge 1 \leq p+n \leq P, e_p^k = e_{p+n}^k, \quad (16)$$

en este sentido, cuando la red neuronal NN_k tiene un buen rendimiento con respecto a la clase p pero pobre rendimiento en relación a la clase $p+n$, es decir, $e_p^k < e_{p+n}^k$, las salidas O_{p+n}^k y O_p^k siguen siendo igualmente ponderadas, aun cuando $e_p^k \neq e_{p+n}^k$. Por otra parte, dadas las redes neuronales NN_k y $NN_{k+m} | m \in \mathbb{Z} - \{0\} \wedge 1 \leq k+m \leq K$, si $w_k > w_{k+m}$, pero $e_p^k > e_p^{k+m}$ entonces la salida O_p^k sigue siendo más ponderada que O_p^{k+m} a pesar de que $e_p^k > e_p^{k+m}$.

3.2.1. Método de agregación

En esta sección, se propone una agregación con base en una jerarquía específica de las redes $NN_k | k = 1, 2, \dots, K$ por cada clase p . La etiqueta prevista para una entrada $\{x_1, \dots, x_K\}$, se calcula por medio de (17) y (18). La matriz de criterios de ordenamiento $R = [P \times K]$, establece el rango de las redes neuronales $NN_k | k = 1, 2, \dots, K$, con respecto a todas las clases p . El vector $R_{(p)}$ representa la fila p de la matriz R y contiene todos los rangos de las redes neuronales $NN_k | k = 1, 2, \dots, K$ con respecto a los pesos $\{w_p^1, w_p^2, \dots, w_p^K\}$, de manera que $R_{(p,k)}$, es la posición ascendente ordenada del peso w_p^k como se muestra en la Eq. (19 y 20).

Por lo tanto, la componente $R_{(p,k)}$, indica el rango de la red neuronal NN_k con respecto a la clase p de modo que un valor más alto indica una mayor importancia. La expresión $\operatorname{diag}(\vec{v})$, denota la matriz cuadrada con \vec{v} como diagonal y cero en cualquier otra componente. En este caso, el rango de la red neuronal NN_k , con respecto a la clase p depende del peso w_p^k calculado con (21), teniendo en cuenta el error de validación e_p^k de NN_k con respecto a la clase p . El factor α_p^k devuelto por $f(k, p)$ para ajustar w_p^k se calcula mediante (22):

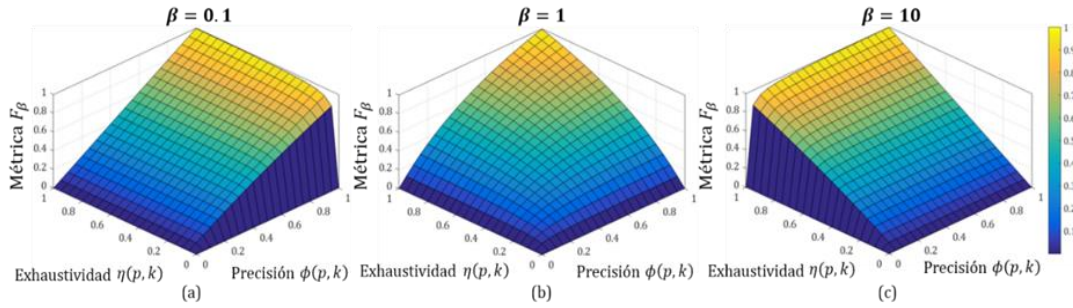


Fig. 6 Relación de la exhaustividad $\eta(p, k)$, precisión $\phi(p, k)$ y la métrica F_β con respecto a tres valores diferentes de β

$$y(x_1, \dots, x_K) = \operatorname{argmax}_{p=1,2,\dots,P} \left\{ \left(\sum_{k=1}^K \begin{bmatrix} O_1^k \\ O_2^k \\ \vdots \\ O_p^k \end{bmatrix} \times \operatorname{diag} \begin{bmatrix} w_1^k \\ w_2^k \\ \vdots \\ w_p^k \end{bmatrix} + \operatorname{diag} \begin{pmatrix} f(k, 1) \\ f(k, 2) \\ \vdots \\ f(k, P) \end{pmatrix} \right) \times \operatorname{diag} \left(\frac{1}{\sum_{k=1}^K \begin{bmatrix} w_1^k + f(k, 1) \\ w_1^k + f(k, 2) \\ \vdots \\ w_p^k + f(k, P) \end{bmatrix}} \right) \right\}, \quad (17)$$

$$f(k, p) = \begin{cases} 0, & R_{(p,k)} \leq 2 \\ \alpha_p^k, & R_{(p,k)} > 2, \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{i|R_{(p,i)}=R_{(p,k)}-2}\} \neq \\ & \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{j|R_{(p,j)}=R_{(p,k)}-1}\} \\ -\alpha_p^k, & \text{otherwise} \end{cases}, \quad (18)$$

$$\forall (w_p^1, w_p^2, \dots, w_p^K) \mid (w_p^1 < w_p^2 < \dots < w_p^K) \rightarrow R_{(p)} = [1, 2, \dots, K], \quad (19)$$

$$\forall (w_p^1, w_p^2, \dots, w_p^K) \mid (w_p^K < w_p^{K-1} < \dots < w_p^1) \rightarrow R_{(p)} = [K, K-1, \dots, 1], \quad (20)$$

$$w_p^k = 1 - e_p^k, \quad (21)$$

$$\alpha_p^k = 2 \left(w_p^{i|R_{(p,i)}=K} - w_p^{j|R_{(p,j)}=1} \right), \quad (22)$$

considerando $R_{(p)} = [1, 2, \dots, K]$ y $R_{(p+n)} = [K, K-1, \dots, 1]$, esto implica que la red neuronal $NN_{k=1}$ es la de menor rango con respecto a la clase p dado

que $R_{(p,1)} = 1$ (peso w_p^k más bajo) y la de más alto rango en cuanto a la clase $p+n$, dado que $R_{(p+n,1)} = K$ (peso w_p^k más alto).

Sin embargo, dado w_p^k está directamente relacionado con el error de validación e_p^k , si el número de verdaderos positivos con respecto a la clase p es alto (es decir, e_p^k es bajo y w_p^k es alto), pero el número de falsos positivos es alto también (examinando (21) los falsos positivos no están relacionados con e_p^k o w_p^k), un valor incorrecto O_p^k de red neuronal NN_k podrían ser también altamente ponderado por w_p^k . El problema anterior en relación con el peso w_p^k tiene un mayor impacto puesto que w_p^k no está relacionado con los falsos positivos obtenidos por la red neuronal NN_k .

En este sentido, la matriz de confusión $C_k = [P \times P]$ dada en (23) representa el desempeño de la red neuronal NN_k durante la etapa de validación e incluye verdaderos positivos $\{c_{p,p}^k\}$, falsos positivos $\{c_{i,p}^k \mid i \in \{1, 2, \dots, P\} - \{p\}\}$ y falsos negativos $\{c_{p,j}^k \mid j \in \{1, 2, \dots, P\} - \{p\}\}$, con respecto a la clase p . Usando la matriz C_k se pueden obtener dos medidas (exhaustividad y precisión). La exhaustividad $\eta(p, k)$ representa el número de elementos pertenecientes a la clase p que son clasificados correctamente por la red neuronal NN_k y se calcula utilizando (24). Vale la pena mencionar que la exhaustividad $\eta(p, k)$ es igual a $1 - e_p^k$. Por otro lado, la precisión $\phi(p, k)$ simboliza la fracción de elementos identificados por la red neuronal NN_k como de la clase p que realmente pertenecen a la clase p y se determina con base en la Eq. (25):

$$C_k = \begin{bmatrix} c_{1,1}^k & c_{1,2}^k & \dots & c_{1,p}^k \\ c_{2,1}^k & c_{2,2}^k & \dots & c_{2,p}^k \\ \vdots & \vdots & \ddots & \vdots \\ c_{p,1}^k & c_{p,2}^k & \dots & c_{p,p}^k \end{bmatrix}, \quad (23)$$

$$\eta(p, k) = \frac{c_{p,p}^k}{c_{p,p}^k + \sum_{j \in \{1,2,\dots,p\} - \{p\}} (c_{p,j}^k)} = 1 - e_p^k, \quad (24)$$

$$\phi(p, k) = \frac{c_{p,p}^k}{c_{p,p}^k + \sum_{i \in \{1,2,\dots,p\} - \{p\}} (c_{i,p}^k)}. \quad (25)$$

Dado que $\eta(p, k) = 1 - e_p^k$ entonces de acuerdo con la Eq. (21) w_p^k es igual a $\eta(p, k)$, lo que significa que el peso w_p^k de la red neuronal NN_k no está relacionado con la precisión $\phi(p, k)$, es decir, el número de elementos identificados por NN_k como clase p que no pertenecen a la clase p no son considerados en el cálculo de w_p^k .

En este sentido, en esta sección se propone calcular w_p^k con base en la métrica F_β de acuerdo con la Eq. (26-28). Usar F_β permite que el peso w_p^k refleje la precisión de la red neuronal NN_k con respecto a la clase p . Por otra parte, el factor β permite poner más énfasis en la precisión cuando $\beta < 1$ o en la exhaustividad cuando $\beta > 1$ como se muestra en la Fig. 6.

Cuando $\beta = 1$, la exhaustividad $\eta(p, k)$ y la precisión $\phi(p, k)$ tienen influencia similar sobre la métrica F_β tal como se muestra en la Figura 6b. Sin embargo, cuando $\beta = 0.1$, la métrica F_β depende principalmente de la precisión $\phi(p, k)$, ya que cualquier variación de $\eta(p, k)$ solo produce un ligero cambio en la métrica F_β como se ilustra en la Fig. 6a. Lo opuesto sucede cuando se usa $\beta = 10$ como se demuestra en la Fig. 6c.

En consecuencia, ahora la jerarquía de las redes neuronales $NN_k \mid k = 1, 2, \dots, K$ con respecto a la clase p se define no solo con base en $\eta(p, k)$ sino también con base en $\phi(p, k)$. De acuerdo con (26 y 28), los pesos w_p^k varían entre ε y uno. El término ε es un valor muy pequeño que se utiliza para evitar que la agregación sea cero, por ejemplo, cuando para una clase p , la expresión $\forall k, \phi(p, k) = 0 \vee \eta(p, k) = 0$ se cumple:

$$w_p^k = \begin{cases} \varepsilon, & \phi(p, k) \text{ is undefined} \\ \varepsilon, & \phi(p, k) = 0 \vee \eta(p, k) = 0, \\ F_\beta, & \text{otherwise} \end{cases}, \quad (26)$$

$$\varepsilon = 0.0001, \quad (27)$$

$$F_\beta = (1 + \beta^2) \frac{\phi(p, k) \times \eta(p, k)}{(\beta^2 \times \phi(p, k)) + \eta(p, k)}. \quad (28)$$

Por otra parte, la agregación dinámica propuesta en (17) Eq. (17) está relacionada con la función $f(k, p)$ que depende del factor α_p^k calculado mediante la Eq. (22). Dicho factor α_p^k se basa en la definición dada en la Eq. (13) y representa la diferencia entre el peso w_p^k más bajo y más alto. Sin embargo, cuando $w_p^{i|R(p,i)=K} \gg w_p^{j|R(p,j)=1}$, el factor α_p^k tiende a cancelar o magnificar de manera excesiva la ponderación $w_p^k + f(k, p)$ en relación con la salida O_p^k . Más aún, el factor α_p^k sigue siendo el mismo para todas las redes neuronales $NN_k \mid k = 1, 2, \dots, K$. En este sentido, se propone calcular el factor dinámico α_p^k con base en el peso w_p^k de red neuronal NN_k y los pesos $w_p^{i|R(p,i)=R(p,k)-2}$ y $w_p^{j|R(p,j)=R(p,k)-1}$ de las dos redes anteriores de menor rango como se muestra en la Eq. (29):

$$\alpha_p^k = \frac{2 \times w_p^k \times w_p^{i|R(p,i)=R(p,k)-2} \times w_p^{j|R(p,j)=R(p,k)-1}}{w_p^k + w_p^{i|R(p,i)=R(p,k)-2} + w_p^{j|R(p,j)=R(p,k)-1}}, \quad (29)$$

Dado que w_p^k toma valores en el rango $[0,1]$ entonces $\alpha_p^k \leq 2/3$. Usar (29) permite ajustar el peso w_p^k no sólo con base en los rangos de las redes neuronales $NN_{i|R(p,i)=R(p,k)-2}$ y $NN_{j|R(p,j)=R(p,k)-1}$, sino también con respecto al rango de la red NN_k , es decir, el factor α_p^k depende de los pesos respecto a la clase p de un conjunto de tres redes neuronales con rangos consecutivos. Por ejemplo, cuando los pesos $w_p^k = 0.5$ y $w_p^{i|R(p,i)=R(p,k)-2} = w_p^{j|R(p,j)=R(p,k)-1} = 0.5$ entonces $\alpha_p^k = 0.167$, pero cuando $w_p^k = 1$ entonces $\alpha_p^k = 0.25$, es decir, α_p^k es mayor cuando los pesos $w_p^{i|R(p,i)=R(p,k)-2}$ y $w_p^{j|R(p,j)=R(p,k)-1}$ permanecen

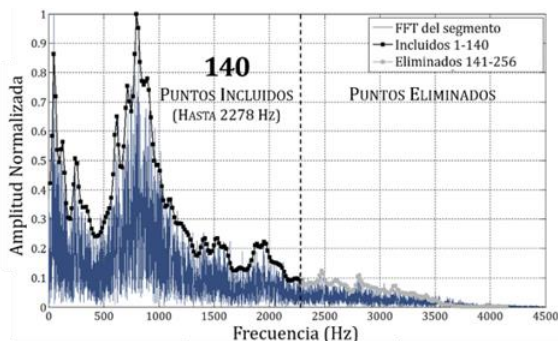


Fig. 7 Respuesta en frecuencia del filtro IIR obtenida a partir de los coeficientes \vec{a} para el primer segmento de la señal del ruido en el despegue de un Boeing 737-200

iguales, pero w_p^k aumenta. De la misma manera, cuando $w_p^{i|R(p,i)=R(p,k)-2} = w_p^{j|R(p,j)=R(p,k)-1} = 0.3$ y $w_p^k = 1$, entonces $\alpha_p^k = 0.112$, es decir, el factor α_p^k disminuye cuando los pesos de las redes $NN_{R(p,i)=R(p,k)-2}$ y $NN_{j|R(p,j)=R(p,k)-1}$ con respecto a la clase p disminuyen. Por otra parte, la relación α_p^k/w_p^k aumenta a medida que la diferencia entre pesos $\left|w_p^k - w_p^{i|R(p,i)=R(p,k)-2}\right|$, o $\left|w_p^k - w_p^{j|R(p,j)=R(p,k)-1}\right|$, disminuye, es decir, existe una fuerte cohesión respecto a α_p^k entre todas las redes neuronales implicadas en la función $f(k, p)$.

3.2.2. Arquitectura del modelo

Cada patrón de entrada x_k es un vector correspondiente al segmento k con $J - r_k$ rasgos, donde J denota el número de rasgos inicialmente extraídos y r_k el número de rasgos eliminados después de un proceso de selección de rasgos. La red neuronal NN_k devuelve un vector O_p^k que representa qué tan relacionada está la entrada x_k con cada clase $p \in \{1, 2, \dots, P\}$ con base en los patrones de entrada utilizados para entrenar la red neuronal NN_k . Después, la agregación O_p^+ de las salidas $O_p^k | k = 1, 2, \dots, K$ se realiza de acuerdo con el algoritmo propuesto anteriormente.

3.2.3. Extracción de rasgos

El proceso de extracción de rasgos aplicado se divide en tareas de pre-procesamiento y procesamiento con base en aquel propuesto en la

Sección 2.2. Luego, se definen las entradas x_k para ser evaluadas por las redes neuronales $NN_k | k = 1, 2, \dots, K$.

La Fig. 7 representa la respuesta en frecuencia del filtro IIR respecto al segmento $k = 1$ de una señal de un B737-200. Además en dicha figura se muestra la FFT del segmento $k = 1$. Similar a lo propuesto en el Sección 2.2, se seleccionan los primeros 140 puntos de los 256 disponibles, lo cual cubre frecuencias hasta 2278 Hz. La Fig. 7 confirma la estrecha relación entre los rasgos basados en los LPC y el espectro de la señal de ruido del segmento $k = 1$.

El siguiente paso es la selección de rasgos lo cual implica la eliminación de aquellos redundantes o irrelevantes. Para ello se utiliza el método para redes neuronales MLP propuesto por [29], el cual se describe en la Sección 2.2. El método calcula la importancia del rasgo j^{th} usando un criterio de ordenamiento $\hat{s}(j)$. Dicho criterio $\hat{s}(j)$ utiliza la diferencia absoluta entre la función $\hat{p}(w_c|x_i)$ de densidad de probabilidad de x_i con el rasgo j^{th} sin cambiar y $\hat{p}(w_c|x_{(j),i})$, la cual representa la función de densidad de probabilidad de x_i evaluadas por el mismo clasificador MLP después de una permutación aleatoria de los elementos en el conjunto $x^j := \{x_i^j\}_1^N$, donde N es el número de muestras y P el número de clases.

El criterio orden $\hat{s}(j)$ se usa en el enfoque de eliminación recursiva (RFE) para remover los r rasgos menos importantes después de entrenamientos sucesivos de la red neuronal NN_k . Al igual que en la Sección 2, se realizan 100 entrenamientos por iteración usando NN_k . El error de prueba se determina después de cada entrenamiento de manera tal que la red NN_k con menor error de prueba, de acuerdo con los pesos W , se selecciona para el cálculo del criterio de ordenamiento $\hat{s}(j)$. Luego, se eliminan los $r = 5$ rasgos menos importantes.

3.3. Experimentos

En esta sección se utiliza la misma base de datos usada en la Sección 2 para evaluar el rendimiento del nuevo modelo propuesto en esta sección contra el propuesto en la Sección 2.3. En total se usan $P = 13$ clases.

Tabla 2. Topología de la red neuronal NN_k

k	Topología de NN_k (Neuronas por capa)	n_k
1	95-30-13	45
2	140-30-13	0
3	95-30-13	45
4	85-30-13	55

Tabla 3. Pesos w_p^k calculados para $k = 1, 2, \dots, K$ Y $p = 1, 2, \dots, P$

p / k	1	2	3	4
1	0.1976	0.0535	0.0682	0.9802
2	0.3699	0.7370	0.4925	0.4225
3	0.4925	0.7370	0.2693	0.6535
4	0.7370	0.7370	0.9612	0.3936
5	0.4925	0.5905	0.6535	0.7370
6	0.4225	0.5905	0.3289	0.4925
7	0.5905	0.9612	0.4225	0.1060
8	0.0341	0.1647	0.0848	0.0309
9	0.7370	0.9612	0.9802	0.9612
10	0.0267	0.0449	0.0848	0.9802
11	0.5905	0.4913	0.4225	0.7370
12	0.3267	0.6535	0.3284	0.1795
13	0.7370	0.7370	0.7370	0.7370

Ya que el mismo número de segmentos es usado ($K = 4$) para ambos casos, las mismas redes neuronales (topología y pesos) se utilizan para los experimentos en esta sección con el fin de comparar ambos modelos con respecto al algoritmo de agregación. Sin embargo, el modelo propuesto en esta sección también es aplicable a cualquier otro conjunto de K redes neuronales paralelas.

La Tabla 2 muestra las topologías de las redes neuronales resultantes de la Sección 2. Estas topologías se determinaron después de la aplicación del algoritmo de selección de rasgos descrito en la misma. Las salidas $O_p^k \mid p = 1, 2, \dots, P$ de la red NN_k se modifican de acuerdo a la función softmax definida en (30):

$$O_p^k = \frac{e^{(o_p^k)}}{\sum_{n=1,2,\dots,P} e^{(o_n^k)}}, \quad (30)$$

En esta sección varios valores de β fueron evaluados a fin de analizar experimentalmente la contribución de la precisión $\phi(p, k)$ y la exhaustividad $\eta(p, k)$ respecto a la salida final. Para este caso en particular, un mejor rendimiento se obtiene generalmente al usar valores que cumplen con $\beta < 1$. Esto refuerza la motivación para el cálculo de los pesos w_p^k con base en la métrica F_β porque reduce el efecto negativo sobre la agregación de una red neuronal NN_k con muchos verdaderos positivos (alta exhaustividad $\eta(p, k)$) y muchos falsos positivos (baja precisión $\phi(p, k)$) en relación a la clase p . Es decir, la red anterior tendrá un rango inferior utilizando $\beta < 1$ que al usar $\beta \geq 1$ y en consecuencia, el peso w_p^k que corresponde a la salida O_p^k será menor también. En general, se logra el mejor rendimiento con la métrica $F_{0.1}$ de manera que $\beta = 0.1$.

La Tabla 3 muestra los pesos w_p^k calculados para las redes neuronales $NN_k \mid k = 1, 2, \dots, K$ que se describen en la Tabla 2 usando (26 a 28), como parte del nuevo algoritmo de agregación propuesto en la Sección 3.2. De acuerdo con los rangos de las redes $NN_k \mid k = 1, 2, \dots, K$ utilizadas en la Sección 2.3, los pesos w_p^k se definen de la siguiente manera $\forall p \mid p = 1, 2, \dots, P; w_p^1 = 0.897, w_p^2 = 0.950, w_p^3 = 0.949, w_p^4 = 0.846$.

Haciendo referencia a la Tabla 3, el peso de la red NN_2 con respecto a la clase $p = 2$ es $w_2^2 = 0.7370$, que es el más alto con respecto a esa clase. Sin embargo, en relación a la clase $p = 1$, el peso de la red neuronal NN_2 es el más bajo dado que $w_1^2 = 0.0535$. La Tabla 3 confirma que el mejor clasificador con respecto a la clase p no es necesariamente el mejor clasificador respecto a la clase $p + n \mid n \in \mathbb{Z} - \{0\} \wedge 1 \leq p + n \leq P$.

Por otro lado, la matriz R que se da en (31) se construye a partir de los pesos que se presentan en la Tabla 3 y refleja los rangos de las redes $NN_k \mid k = 1, 2, \dots, K$ con respecto a la clase $p = 1, 2, \dots, P$. En este sentido, la matriz R muestra que la red de mayor rango $NN_k \mid k = \underset{k}{\operatorname{argmax}}(R_{(p)})$ respecto a la clase $p = 1, 2, \dots, P$, no es la misma para todas las clases, por ejemplo, la red NN_2 es la de menor rango respecto la clase $p = 1$ dado que $R_{(1,2)} = 1$ y la de mayor rango con respecto a la clase $p = 2$, dado que $R_{(2,2)} = 4$.

Tabla 4. Salidas del modelo con respecto a una medición de la clase B737-600/700

p	Etiqueta de Clases	O_p^1	O_p^2	O_p^3	O_p^4	Modelo de la Sección 2.3 ^a (O_p^+)	Modelo introducido en esta sección ^b (O_p^+)
1	MD	0.0733	0.0678	0.0665	0.0684	0.0687	0.0690
2	F100_B737-200 2	0.0740	0.0676	0.0665	0.0685	0.0689	0.0685
3	F100_B737-200	0.0733	0.0676	0.1670	0.1280	0.1100	0.0980
4	F100 2	0.0733	0.0734	0.0924	0.1170	0.0878	0.0858
5	F100	0.1160	0.0676	0.0665	0.0712	0.0787	0.0765
6	ATR-42	0.0739	0.0676	0.0665	0.0684	0.0688	0.0689
7	A320_B737-800	0.0733	0.0676	0.0665	0.0684	0.0687	0.0690
8	A320 3	0.0768	0.0676	0.0665	0.0684	0.0695	0.0684
9	A320 2	0.0733	0.0676	0.0665	0.0684	0.0687	0.0683
10	A320 1	0.0733	0.0677	0.0665	0.0684	0.0687	0.0683
11	B747-400	0.0733	0.0676	0.0718	0.0684	0.0702	0.0702
12	B737-6/700	0.0733	0.1830	0.0706	0.0684	0.1030	0.1230
13	B737-1/200	0.0733	0.0676	0.0665	0.0684	0.0687	0.0686

^a El modelo será referido de aquí en adelante como *Modelo ESWA*

^b El modelo será referido de aquí en adelante como *Nuevo Modelo*

$$R_{[P \times K]} = \begin{bmatrix} 3 & 1 & 2 & 4 \\ 1 & 4 & 3 & 2 \\ 2 & 4 & 1 & 3 \\ 2 & 3 & 4 & 1 \\ 1 & 2 & 3 & 4 \\ 2 & 4 & 1 & 3 \\ 3 & 4 & 2 & 1 \\ 2 & 4 & 3 & 1 \\ 1 & 2 & 4 & 3 \\ 1 & 2 & 3 & 4 \\ 3 & 2 & 1 & 4 \\ 2 & 4 & 3 & 1 \\ 1 & 2 & 3 & 4 \end{bmatrix} \quad (31)$$

Por otra parte, la red NN_4 es la de mayor rango con respecto a la clase $p = 1$ ($R_{(1,4)} = 4$) pero la segunda de menor rango con respecto a la clase $p = 2$ ($R_{(2,4)} = 2$).

En relación con la dinámica introducida en el nuevo modelo por medio de (17) y (18), el término $w_2^4 + f(4,2)$ se reduce a w_2^4 dado que $R_{(2,4)} \leq 2$ y por lo tanto $f(4,2) = 0$, es decir, la salida O_2^4 de la red neuronal NN_4 siempre es ponderada por la misma cantidad $w_2^4 + 0$. Por otro lado, el peso w_1^4 de la red neuronal NN_4 con respecto a la clase $p = 1$ cambia dinámicamente de acuerdo con $f(4,1)$ dado que $R_{(1,4)} > 2$. En este sentido, con base en

(18) el aumento o disminución de w_1^4 depende de si las redes $NN_{i=3}$ y $NN_{j=1}$ concuerdan en sus salidas respecto a $\operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{i|R_{(p,i)}=R_{(p,k)}-2}\}$ y $\operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{j|R_{(p,j)}=R_{(p,k)}-1}\}$. Dado que $R_{(1,4)} = 4$, $i = 3$ cumple con $i|R_{(p,i)} = R_{(p,k)} - 2$ puesto que $R_{(1,3)} = R_{(1,4)} - 2 = 2$.

Similarmente, $j = 1$ cumple con $j|R_{(p,j)} = R_{(p,k)} - 1$ dado que $R_{(1,1)} = 3$ y $R_{(1,4)} - 1 = 3$. En consecuencia, si las redes neuronales NN_3 y NN_1 coinciden, es decir, $\operatorname{argmax}_{p=1,2,\dots,P} \{O_p^3\} = \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^1\}$ entonces la salida O_1^4 se pondera por $w_1^4 - \alpha_1^4$ donde α_1^4 se calcula con (29). En este caso, el factor α_1^4 resta importancia a la salida O_1^4 dado que las redes NN_3 y NN_1 coinciden respecto a la clase identificada.

La Tabla 4 presenta las salidas O_p^+ después de aplicar el nuevo modelo propuesto en esta sección (*Nuevo Modelo*) y aquel presentado en la Sección 2.3 (*Modelo ESWA*) para una medición perteneciente a la clase 737-600/700 ($p = 12$). Las salidas agregadas O_p^+ se muestran para ambos modelos, así como las salidas O_p^k de las redes NN_k | $k = 1, 2, \dots, K$ descritas en la Tabla 6.

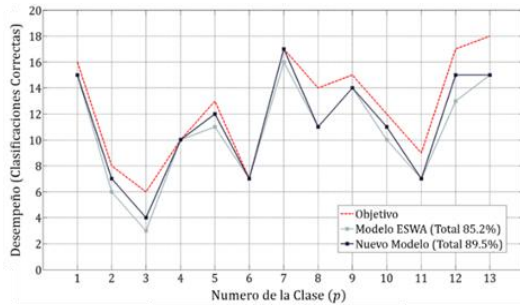


Fig. 8. Comparación gráfica de ambos modelos

Respecto al *Modelo ESWA*, las salidas O_p^k son agregadas usando (10) a (15) y los pesos antes mencionados: $w_1 = 0.897, w_2 = 0.950, w_3 = 0.949, w_4 = 0.846$. Respecto al *Modelo Nuevo*, los resultados agregados se calculan como se define en la Sección 3.2.

Las clases $p = 3, p = 4, p = 5$ y $p = 12$ son las más relevantes en el ejemplo de la Tabla 4; la última de ellas es la clase objetivo a identificar. Respecto a la clase $p = 5$, las salidas $O_5^k | k = 1, 2, \dots, K$ no son relevantes, salvo la salida O_5^1 que introduce cierta incertidumbre. Sin embargo, la salida errónea de la red NN_1 es anulada en ambos modelos con base en una ponderación baja respecto a la clase $p = 5$. El *Modelo ESWA* utiliza w_1 , que es el segundo peso más bajo mientras que el *Nuevo Modelo* usa w_5^1 , que es el más bajo con respecto a la clase $p = 5$. En ambos casos, la agregación de las salidas $O_5^k | k = 1, 2, \dots, K$ conduce a un bajo valor de O_5^+ .

Respecto a la clase $p = 4$, las salidas $O_4^k | k = 1, 2, \dots, K$ también producen cierta incertidumbre dado que O_4^3 y O_4^4 son altas en comparación con el resto. No obstante, en ambos modelos se logra un valor O_4^+ bajo. En este sentido, el *Modelo ESWA* pondera las salidas de la red NN_4 con un peso w_4 bajo, de modo que aun cuando O_4^4 es el valor más alto con respecto a la clase $p = 4$, su peso es el más bajo también. Algo similar se consigue con el *Nuevo Modelo* dado que w_4^4 es el peso más bajo respecto a la clase $p = 4$. Sin embargo, el *Modelo ESWA* devuelve incorrectamente el valor O_p^+ más alto, para la clase $p = 3$ debido, principalmente, a que O_3^3 es la segunda salida más alta $\forall k, p$ y las salidas $O_p^3 | p = 1, 2, \dots, P$ de la red NN_3 son

ponderadas considerablemente alto. Al contrario, el *Nuevo Modelo* contrarresta con éxito las salidas $O_p^3 | p = 1, 2, \dots, P$ con base en una ponderación más precisa, por ejemplo, el peso w_3^3 es el más bajo con respecto a la clase $p = 3$.

Respecto a la clase objetivo $p = 12$, solamente la salida O_{12}^2 de la red NN_2 es significativa. En este sentido, el *Nuevo Modelo* agrega dinámicamente O_{12}^2 de acuerdo con $(w_{12}^2 + \alpha_{12}^2)O_{12}^2$ donde $w_{12}^2 + \alpha_{12}^2 = 0.950 + 0.107$ mientras que el *Modelo ESWA* pondera O_{12}^2 con base en $(w_2 + f(2))O_{12}^2$ donde $w_2 + f(2) = 0.950 + 0.208$. No obstante, aun cuando $(w_2 + f(2))O_{12}^2 > (w_{12}^2 + \alpha_{12}^2)O_{12}^2$, el *Nuevo Modelo* obtiene el mayor valor para la salida O_{12}^+ , lo cual conlleva a una clasificación exitosa ya que la contribución en la agregación de la salida O_{12}^2 es mayor en el *Nuevo Modelo* dado que $w_{12}^2 / \sum_k (w_{12}^k + f(k, 12)) > w_2 / \sum_k (w_k + f(k))$. Al final, el *Modelo ESWA* no puede clasificar correctamente la medición de la Tabla 4 debido a una ponderación inadecuada de las salidas O_p^k . Para más ejemplos y análisis refiérase a [31].

3.4. Resultados y discusión

Con base en los extensos experimentos con la base de datos de mediciones del ruido en el despegue, el modelo propuesto en esta sección (*Nuevo Modelo*) mejora el reconocimiento de clases de aeronaves con los modelos existentes en la literatura y aquel introducido en la Sección 2.3. La comparación global entre el *Nuevo Modelo* y el *Modelo ESWA* se presenta en la Fig. 8. Ambos modelos dependen de la agregación de las salidas O_p^k de K redes neuronales paralelas, donde la red $NN_k | k = 1, 2, \dots, K$ es entrenada para clasificar patrones del segmento de señal k . El primero utiliza una agregación dinámica basada en una jerarquía selectiva con respecto a las clases $p = 1, 2, \dots, P$ mientras que el segundo usa una jerarquía general.

El mejor rendimiento del *Nuevo Modelo* está estrechamente relacionado con la capacidad de cambiar la dinámica de agregación con respecto a cada clase p , es decir, la agregación de las salidas $O_p^k | k = 1, 2, \dots, K$ se determina con base en la jerarquía respecto a la clase p y no afecta la manera en que las salidas $O_{p+n}^k | n \in \mathbb{Z} - \{0\} \wedge 1 \leq$

$p + n \leq P$ son agregadas. Además, la agregación propuesta en la Sección 3.2.1 utiliza un peso específico w_p^k para ponderar la salida O_p^k de la red neuronal NN_k de manera que w_p^k no es necesariamente igual a w_{p+n}^k . En este sentido, la jerarquía propuesta en esta sección se basa en el ordenamiento ascendente de pesos w_p^k . De hecho, el rendimiento de la red $NN_k \mid k = 1, 2, \dots, K$, que se pondera con base en w_p^k , varía en dependencia de la clase p .

Además, dada las redes neuronales NN_k y $NN_{k+m} \mid m \in \mathbb{Z} - \{0\} \wedge 1 \leq k + m \leq K$ entonces se cumple que $w_p^k > w_p^{k+m} \nRightarrow w_{p+n}^k > w_{p+n}^{k+m}$, es decir, el hecho de que la red NN_k sea el mejor clasificador individual respecto a la clase p no implica que NN_k sea también el mejor clasificador individual con respecto la clase $p + n$. Además, la definición de los pesos w_p^k propuestos en la Sección 3.2.1 permite la cuantificación del desempeño de la red NN_k con base en la exhaustividad $\eta(p, k)$ y la precisión $\phi(p, k)$. La métrica F_β mejora la agregación reduciendo el efecto negativo de falsos positivos usando valores de $\beta < 1$, ya que w_p^k estaría más sesgado hacia $\phi(p, k)$ que hacia $\eta(p, k)$.

Otro factor que contribuye al mejor desempeño del *Nuevo Modelo* es la función $f(k, p)$ introducida en la Sección 3.2.1. La función $f(k, p)$ devuelve un factor único α_p^k para cambiar dinámicamente el peso w_p^k . El factor α_p^k se calcula con base en los pesos w_p^k , $w_p^{i|R(p,i)=R(p,k)-2}$ y $w_p^{j|R(p,j)=R(p,k)-1}$, es decir, depende de un conjunto de tres redes neuronales con rangos consecutivos. Teniendo en cuenta que los rangos de las redes neuronales $NN_k \mid k = 1, 2, \dots, K$ varían según la clase p , la dinámica de agregación también cambia según la clase p .

Más aún, el factor α_p^k representa la relación entre los pesos de las redes neuronales con rangos consecutivos de manera que, cuando $NN_{i|R(p,i)=R(p,k)-2}$ y $NN_{j|R(p,j)=R(p,k)-1}$ clasifican las entradas x_i y x_j como de la clase p , el factor α_p^k disminuye a medida que los pesos $w_p^{i|R(p,i)=R(p,k)-2}$ o $w_p^{j|R(p,j)=R(p,k)-1}$ aumentan y por tanto el término $w_p^k + f(k, p)$ disminuye, es decir, si

$NN_{i|R(p,i)=R(p,k)-2}$ y $NN_{j|R(p,j)=R(p,k)-1}$ tienen un mejor rango, entonces el cambio dinámico del peso w_p^k es mayor.

Además, dado que la arquitectura propuesta para el *Nuevo Modelo* se basa en decisión por comité, el resultado final es alcanzado por un conjunto de redes neuronales $NN_k \mid k = 1, 2, \dots, K$ en lugar de un clasificador único. En consecuencia, si un clasificador falla pero las salidas de los otros coinciden, aún es posible obtener una salida final correcta. Esto depende del desempeño de las redes $NN_k \mid k = 1, 2, \dots, K$ y de la agregación de las salidas $O_p^k \mid p = 1, 2, \dots, P$ del conjunto de redes antes mencionado. Por ese motivo, el algoritmo de la Sección 3.2.1 se define con el fin de disminuir el efecto no deseado en la decisión final de los falsos positivos respecto a la clase p .

Por otra parte, dado que cada segmento de señal k sólo es analizado por la red NN_k , las alteraciones encontradas durante el tiempo cubierto por el segmento de señal k , sólo puede afectar al rendimiento de NN_k y no aquel de la red $NN_{k+m} \mid m \in \mathbb{Z} - \{0\} \wedge 1 \leq k + m \leq K$, es decir, cualquier otra red. Algunas perturbaciones tales como perros ladrando, personas hablando o gritando, canto de pájaros y claxon de autos, están presentes en las mediciones obtenidas del ruido en el despegue.

Sin embargo, en la mayoría de los casos sólo algunas de las características de los segmentos de señal en los extremos ($k = 1$ y $k = 4$) se ven afectados debido a que dichas perturbaciones no son efectivas, en general, a lo largo del espectro completo de la señal del ruido en el despegue, lo cual no deteriora por completo el desempeño de la red neuronal correspondiente.

Además, de acuerdo con la segmentación de la señal descrita en la Sección 2.1, los segmentos interiores ($k = 2$ y $k = 3$) están siempre más cerca del punto de mayor amplitud como se muestra en la Fig. 3, donde es poco probable que se enmascare la señal del ruido de los aviones por su gran intensidad durante dicha etapa en el despegue. En consecuencia, dichos segmentos y las redes asociadas (NN_2 y NN_3), se ven menos afectadas por perturbaciones tales como las mencionadas anteriormente.

4. Modelo multicapa neuro-difuso

En la Secciones 2 y 3, se utiliza una red neuronal NN_k para cada segmento k (véase [19, 31]) mientras que en [23] se usan dos redes paralelas para evaluar dos tipos diferentes de rasgos extraídos de la misma señal. En todos los casos se establece un algoritmo de agregación para construir la respuesta final a partir de las múltiples salidas producidas por todas las redes neuronales. El algoritmo de agregación es muy importante ya que involucra la ponderación de múltiples respuestas sobre el mismo espacio de clases $p = 1, 2, \dots, P$ incluyendo las salidas erróneas. Hasta ahora los algoritmos propuestos anteriormente sólo permiten ponderar cada clase p individualmente, por lo que las relaciones entre las salidas O_p^k y O_{p+n}^k de una misma red neuronal NN_k no se consideran. En esta sección se propone un modelo multicapa neuro-difuso para el reconocimiento de la clase de aeronave con base en la segmentación de la señal. El modelo usa redes neuronales y lógica difusa secuencialmente, aprovechando la generalización e inferencia difusa bajo la incertidumbre cognitiva respecto al rendimiento de las redes neuronales. El algoritmo usa la agregación difusa de múltiples salidas O_p^k para la generación de una salida final con respecto a la clase p .

4.1. Arquitectura multicapa neuro-difusa

Los sistemas híbridos que combinan redes neuronales y lógica difusa han demostrado su eficacia en una amplia variedad de problemas reales [32-34]. Los $140 - n_k$ rasgos iniciales extraídos como se presenta en la Sección 2.2 son básicamente una reducción dimensional del espectro de la señal del ruido en el despegue, donde n_k se determina a partir del método de selección de rasgos descrito en la misma sección. Sin embargo, todavía es complicado derivar de manera sencilla, un mapeo directo entre los rasgos del segmento k y las clases. Por tanto, una red NN_k para cada $k \in \{1, 2, \dots, K\}$ es utilizada para pre-procesar los $140 - n_k$ rasgos y transformarlos al espacio R^P , donde P es el número de clases. La transformación anterior representa cuán relacionada está la entrada x_k con cada clase $p \in$

$\{1, 2, \dots, P\}$, con base en los patrones de entrada que se usan durante el entrenamiento supervisado de las redes.

La salida $O_p^k(x, W)$ de la red neuronal NN_k del segmento k , dada una entrada $x_k \in R^{140-n_k}$ y pesos W , se determina usando (32):

$$O_p^k(x_k) = \gamma_p \left(W_{p0}^2 \cdot b_k^1 + \sum_{u=1}^m W_{pu}^2 \cdot \varphi_u \left(W_{u0}^1 \cdot b_k^0 + \sum_{j=1}^{140-n_k} W_{uj}^1 \cdot x_k^j \right) \right), \quad (32)$$

donde W_{ij}^l denota los valores de los pesos de la j^{th} neurona de la capa $l - 1$ a la i^{th} neurona de la capa l , W es la colección de W_{ij}^l , $\forall i, j, l$ de la red, $\varphi_u = \tanh(\cdot)$ es la función de activación de la u^{th} neurona en la capa uno, $\gamma_C = \tanh(\cdot)$ es la función de activación del p^{th} salida y b^l representa el sesgo para la capa l .

En general, las salidas $\{O_p^k \mid p = 1, 2, \dots, P\}$ de la red NN_k dada una entrada x_k se espera que cumplan con (33 y 34):

$$\forall y \mid y = \operatorname{argmax}_{p=1,2,\dots,C} \{O_p^k\} : O_y^k = 1, \quad (33)$$

$$\forall y' \in \{1, 2, \dots, P\} - \{y\} : O_{y'}^k = 0. \quad (34)$$

Las condiciones anteriores indican que la red neuronal NN_k clasifica totalmente la entrada x_k como clase y , es decir, NN_k no detecta similitud alguna con respecto a las clases y' . Bajo una buena generalización NN_k normalmente devuelve salidas que no cumplen con una o ambas condiciones. Más aún, las salidas $\{O_p^k \mid p = 1, 2, \dots, P\}$ pueden incluir más de una posible clase, lo cual se puede representar con (35). En este sentido, tener en cuenta la relación entre clases y e $y' \in \{1, 2, \dots, P\} - \{y\}$ es particularmente significativo si x_k pertenece a la clase y , pero $y' = \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^k\}$.

$$\begin{aligned} \exists y \in \{1, 2, \dots, P\} - \left\{ \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^k\} \right\} \mid \beta_y \\ = \frac{\left| O_y^k - \max_{p=1,2,\dots,P} \{O_p^k\} \right|}{\left| \min_{p=1,2,\dots,P} \{O_p^k\} - \max_{p=1,2,\dots,P} \{O_p^k\} \right|} \lll 1 \end{aligned} \quad (35)$$

La agregación descrita en la Sección 4 se lleva a cabo con base en la ponderación independiente de K salidas para la clase y , es decir, $\forall k \in \{1, 2, \dots, K\}$, O_y^k no tiene ningún efecto sobre el valor final de la clase $y' \in \{1, 2, \dots, P\} - \{y\}$. Por otra parte, se muestra también que $\forall y \in \{1, 2, \dots, P\}$, $\forall k \mid 2 < \bar{r}(k) \leq K$, el valor final de la clase y' estará sesgado hacia $O_y^{i|\bar{r}(i)=\bar{r}(k)-2}$, si y sólo si $y = \operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{i|\bar{r}(i)=\bar{r}(k)-2}\}$, por lo tanto el modelado de las relaciones entre las clases y y y' es limitado.

El algoritmo de agregación propuesto en la Sección III-B-1 permite cambiar dinámicamente el peso w_p^k perteneciente a O_p^k con respecto a las clases $\operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{i|\bar{r}(i)=\bar{r}(k)-2}\}$, y $\operatorname{argmax}_{p=1,2,\dots,P} \{O_p^{j|\bar{r}(j)=\bar{r}(k)-1}\}$, lo cual representa una ponderación selectiva dinámica con respecto a la clase p .

En este caso, $f(k, p)$ devuelve un factor α_p^k diferente para cada clase p . Sin embargo, si la clase y cumple con (35), entonces O_y^k no tendrá relevancia en relación con la clase y' incluso cuando $O_y^k \approx O_{y'}^k$ y $O_{y'}^k = \max_{p=1,2,\dots,P} \{O_p^k\}$.

Por lo tanto, las asociaciones entre y y y' no están modeladas bajo coincidencia parcial, lo que podría llevarse a cabo con base en la cuantificación del valor de verdad de $O_y^k \approx \max_{p=1,2,\dots,P} \{O_p^k\}$ o $\beta_y \lll 1$. Además, cuantificar el valor de verdad de $O_y^k \approx \max_{p=1,2,\dots,P} \{O_p^k\}$ o $\beta_y \lll 1$ con base en la lógica clásica binaria (verdadero o falso), conlleva las mismas limitaciones dado que también se deben establecer límites estrictos.

De acuerdo a lo anterior, se propone una arquitectura neuro-difusa. El modelo calcula el

Tabla 5. Descripción de las bases de datos

Nombre ^a	Mediciones	P	Introducida por
BD Anterior	305	13	Sánchez-Fernández et al. [22], Sección 2
BD Nueva	426	13	Sección 4.2.2

^a Los nombres serán usados a partir de aquí para referirse a las BD

valor de membresía c_p a la clase $p \in \{1, 2, \dots, P\}$ con base en un sistema de inferencia difuso del tipo Mamdani (FIS). Las entradas \hat{O}_p^k se definen con base en la fuzzificación de $O_p^k \mid p = 1, 2, \dots, P$ de cada red neuronal $NN_k \mid 1, 2, \dots, K$ mediante la cuantificación del valor de verdad de $O_p^k \approx 1$ con universo $U = [0, 1]$. Tres valores lingüísticos se definen para cada entrada difusa \hat{O}_p^k (*Baja*, *Media* y *Alta*).

La salida difusa $Z_p \mid p = 1, 2, \dots, P$ representa el grado de membresía a la clase p . Cinco valores lingüísticos son definidos para cada salida difusa (*Muy Baja*, *Baja*, *Media*, *Alta* y *Muy Alta*). Después, las relaciones entre entradas difusas \hat{O}_p^k y cada clase p se modelan utilizando reglas SI-ENTONCES, que se construyen a partir de razonamiento deductivo.

Las reglas de agregación conllevan a un conjunto difuso Reglas rendimientos de agregación a un conjunto C_p con respecto a la clase p , el cual es defuzificado con base el método del centroide, obteniendo el valor real de membresía c_p . A continuación, la etiqueta y predicha para una entrada $\{x_1, \dots, x_K\}$ está dada por (36):

$$y(x_1, \dots, x_K) = \operatorname{argmax}_{p=1,2,\dots,P} (c_p), \quad (36)$$

La nueva arquitectura permite la combinación de cualquier número $i = 1, 2, \dots, K \times P$ de entradas difusas con el fin de representar el conocimiento con base en la inferencia difusa mediante reglas. La integración de múltiples salidas O_p^k con base en inferencia difusa permite lo siguiente:

Tabla 6. Salidas del Modelo con respecto a una Medición de la Clase A320 1

<i>p</i>	Etiqueta	Modelo propuesto en este sección ^a					Modelo propuesto en el [19] ^b				
		O_p^1	O_p^2	O_p^3	O_p^4	Final	O_p^1	O_p^2	O_p^3	O_p^4	Final
		-	-	-	-	(c_p)	0.897	0.950	0.949	0.846	(c_p)
1	Clase MD	3.77E-09	5.87E-08	6.38E-05	3.9E-14	0.0774	0.0689	0.068	0.0676	0.0551	0.0654
2	Clase F100_B737-200 2	0.000404	0.000676	2.94E-07	2.07E-06	0.0774	0.0689	0.0681	0.0676	0.0551	0.0654
3	Clase F100_B737-200	0.000166	0.000527	9.86E-05	1.02E-10	0.0774	0.0689	0.068	0.0676	0.0551	0.0654
4	Clase F100 2	0.00232	2.38E-07	5.18E-06	0.00866	0.0774	0.0691	0.068	0.0676	0.0556	0.0655
5	Clase F100	4.68E-12	1.11E-15	1.36E-06	0.00014	0.0774	0.0689	0.068	0.0676	0.0551	0.0654
6	Clase ATR-42	0.0727	0.00135	2.02E-12	0	0.0774	0.0741	0.0681	0.0676	0.0551	0.0666
7	Clase A320_B737-800	0.00587	0.753	0.0275	0.996	0.5	0.0693	0.144	0.0695	0.149	0.107
8	Clase A320 3	2.85E-05	0.124	1	0.998	0.564	0.0689	0.077	0.184	0.149	0.121
9	Clase A320 2	1.89E-05	0.000873	0.038	2.87E-13	0.0774	0.0689	0.0681	0.0702	0.0551	0.0662
10	Clase A320 1	0.00814	0.368	2.42E-07	0.999	0.866	0.0695	0.0983	0.0676	0.15	0.0939
11	Clase B747-400	0.874	0.000266	0.00438	2.21E-12	0.0774	0.165	0.068	0.0679	0.0551	0.0868
12	Clase B737-6/700	0.0217	0.000113	0.00204	1.66E-05	0.0774	0.0704	0.068	0.0677	0.0551	0.0658
13	Clase B737-1/200	2.91E-10	5.26E-07	8.77E-09	0	0.0774	0.0689	0.068	0.0676	0.0551	0.0654

^a El modelo será referido de aquí en adelante como *Nuevo Modelo*^b El modelo será referido de aquí en adelante como *Modelo ESWA***Tabla 7.** Comparación general contra modelos existentes

<i>p</i>	Total	Clasificaciones correctas			
		Modelo en [22]	Modelo en [23]	Modelo ESWA [19]	Nuevo Modelo [35]
1	12	10	10	10	12
2	15	13	14	14	14
3	14	10	11	11	12
4	17	15	14	16	16
5	7	6	6	7	7
6	18	15	15	15	16
7	17	13	13	13	16
8	9	7	7	7	7
9	13	11	11	11	11
10	10	9	10	10	10
11	6	3	3	3	3
12	8	6	6	6	8
13	16	12	15	15	15
TOTAL (%)	162 (100%)	130 (80%)	135 (83.3%)	138 (85.2%)	147 (90.7%)

1. Jerarquía y ponderación individual de clases:

SI \hat{O}_p^k es *Alta* **ENTONCES** Z_p es *Alta*.

La membresía difusa a una clase Z_p será alta si la entrada difusa \hat{O}_p^k es alta sin importar qué valor tenga \hat{O}_p^{k+n} , es decir, el resultado respecto a la clase p está altamente relacionado con las salidas de la red neuronal NN_k .

Por otra parte, sólo las reglas que tengan Z_p como consecuente afectarán a la salida respecto a la clase p . Si una regla tiene un consecuente compuesto que consiste de $R > 1$ salidas difusas, entonces dicha regla pudiera ser descompuesta en R reglas con el mismo antecedente y sólo una salida difusa Z_p .

2. Ponderación de la "concordancia de respuestas":

SI \hat{O}_p^k es *Alta* **Y** \hat{O}_p^{k+1} es *Alta* **Y** \hat{O}_p^{k+2} es *Alta* **ENTONCES** Z_p es *Muy Alta*.

La membresía difusa a una clase Z_p va a ser muy alta si las salidas de las redes neuronales NN_k , NN_{k+1} y NN_{k+2} concuerdan regresando valores altos respecto a la clase p .

3. Anular clasificaciones erróneas individuales:

SI \hat{O}_p^k es *Alta* **Y** \hat{O}_p^{k+1} no es *Baja* **Y** \hat{O}_{p+n}^{k+1} no es *Baja* **ENTONCES** Z_p es *Alta* **Y** Z_{p+n} no es *Alta*.

Suponiendo que la red neuronal NN_{k+1} encuentra cierta correlación entre la clase p y $p+n$ con respecto a la entrada x_{k+1} , que en realidad pertenece a la clase p la regla anterior explota el hecho anterior combinando \hat{O}_p^k con \hat{O}_p^{k+1} y \hat{O}_{p+n}^{k+1} para compensar la incertidumbre producida por la red NN_{k+1} .

Además, el nuevo modelo introduce un enfoque flexible para cambiar la salida final respecto a la clase p con base en la generación de conocimiento a través de la adición o supresión de reglas o el ajuste de las funciones de membresía, sin volver a calcular los pesos o reentrenar las redes neuronales.

4.2. Experimentos

Extensos experimentos se llevan a cabo con dos bases de datos con mediciones reales del ruido en el despegue. Una descripción general de ambas bases de datos se da en la Tabla XI. Los experimentos incluyen lo siguiente: 1) dividir la señal del despegue en segmentos; 2) extraer los rasgos; 3) aplicar el algoritmo de selección de rasgos; 4) entrenar y evaluar las redes neuronales; 5) definir las funciones de membresía para las entradas y salidas difusas; 6) crear las reglas de inferencia; y 7) evaluar el desempeño del modelo neuro-difuso. Más información de estos experimentos se puede encontrar en [35].

Los experimentos descritos en la Sección 4.2.2 se utilizan para comparar el desempeño del modelo propuesto y aquellos definidos en [19, 22, 23] usando ambas bases de datos de la Tabla 5. Los experimentos de la Sección 4.2.2 utilizan una nueva base de datos más compleja y actualizada, estrictamente construida a partir de mediciones en tiempo real y usando varias fuentes para verificar el horario del despegue, el tipo de aeronave y motor de avión.

4.2.1. Experimentos con la BD anterior

Las mismas redes neuronales (topología y pesos) obtenidas en la Sección 2 se utilizan para los experimentos descritos en esta sección a fin de comparar ambos métodos solo con respecto al algoritmo de agregación. La Tabla 2 muestra la topología de las redes anteriormente referidas.

Cada salida O_p^k de la red NN_k se fuzzifica en \hat{O}_p^k , la cual es etiquetada de acuerdo a *Etiqueta Corta de Clase de Aeronave + NN_k* de modo que la salida de la red neuronal NN_1 con respecto a la clase *ATR-42* se etiqueta como *ATR-42 NN1*. Un conjunto difuso $A_{k,p}^i$ se define para cada valor lingüístico i de \hat{O}_p^k con base en la función de Gauss definida en (51). Dado el conocimiento contextual y semántico sobre las salidas de la red NN_k (O_p^k representa la similitud de x_k con los patrones que se enseñan a la red NN_k con respecto a la clase p), las funciones de Gauss se ajustan manualmente usando la intuición [36, 37]. Además, los mismos conjuntos difusos $A_{k,p}^i$ son utilizados para todas las entradas \hat{O}_p^k ya que la evaluación de técnicas para el ajuste automático

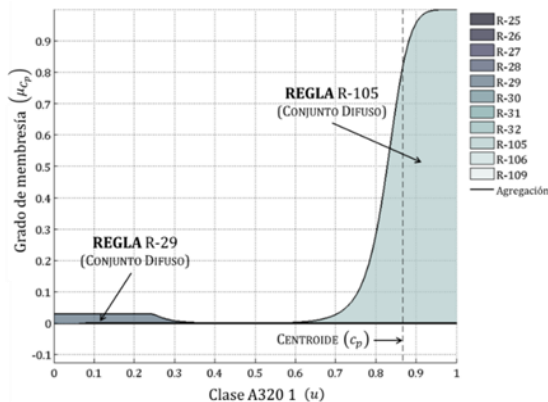


Fig. 9. Salida del FIS para la Clase A320 1

de las funciones de memberships está fuera del alcance de este trabajo:

$$f(x, \alpha, \beta) = e^{-\frac{(x-\beta)^2}{2\alpha^2}}, \quad (37)$$

En esta sección $A_{k,p}^1$ (Baja) se construye con $\alpha = 0.1403, \beta = 0$; $A_{k,p}^2$ (Media) con $\alpha = 0.1021, \beta = 0.5$; y $A_{k,p}^3$ (Alta) con $\alpha = 0.187, \beta = 1$.

Del mismo modo, las salidas difusas Z_p están etiquetadas de acuerdo a Clase + Etiqueta Corta de Clase de Aeronave de modo que la salida correspondiente a la clase ATR-42 se etiqueta como Clase ATR-42. Del mismo modo, la definición de los conjuntos difusos B_p^j para Z_p se realiza a partir de intuición. La función de curva de campana definida en (52) se usa como función de membresía de los valores lingüísticos Muy Baja, Baja, Alta y Muy Alta; mientras que la función de membresía combinación combinada de Gauss presentada en (53) se utiliza para Media.

$$h(x, \alpha, \beta, \gamma) = \frac{1}{1 + \left| \frac{x-\gamma}{\alpha} \right|^{2\beta}}, \quad (38)$$

$$g(x, \alpha_1, \beta_1, \alpha_2, \beta_2) = \begin{cases} f(x, \alpha_1, \beta_1), & 0 \leq x \leq \beta_1 \\ 1, & \beta_1 < x < \beta_2, \\ f(x, \alpha_2, \beta_2), & \beta_2 \leq x \leq 1 \end{cases} \quad (39)$$

En esta sección B_p^1 (Muy Baja) se construye con $\alpha = 0.15, \beta = 3.132, \gamma = -0.0119$; B_p^2 (Baja) con $\alpha = 0.25, \beta = 4.42, \gamma = -0.05525$; B_p^4 (Alta)

con $\alpha = 0.201, \beta = 3.54, \gamma = 1.028$; y B_p^5 (Muy Alta) con $\alpha = 0.113, \beta = 2.91, \gamma = 0.9913$; mientras que B_p^3 (Media) se construye con $\alpha_1 = 0.064, \beta_1 = 0.321, \alpha_2 = 0.14, \beta_2 = 0.4344$.

Se utiliza un sistema disyuntivo de reglas SI-ENTONCES. Las reglas se derivan aprendiendo por ejemplos; es decir, identificando relaciones entre las entradas difusas \hat{O}_p^k y cada salida difusa Z_p , con base en un conjunto de observaciones conocidas. En este caso, las observaciones se extraen aleatoriamente del conjunto de pruebas (desconocido para NN_k). Dos tipos de reglas intuitivas se definen para las pruebas con la base de datos actual (BD Nueva):

$$1. \forall (a, b, c) \mid a, b, c \in \{1, 2, \dots, K\}, a \neq b \neq c;$$

SI \hat{O}_p^a es Alta Y \hat{O}_p^b es Alta Y \hat{O}_p^c es Alta ENTONCES Z_p es Muy Alta.

$$2. \forall (a, b, c) \mid a, b, c \in \{1, 2, \dots, K\}, a \neq b \neq c;$$

SI \hat{O}_p^a es Baja Y \hat{O}_p^b es Baja Y \hat{O}_p^c es Baja ENTONCES Z_p es Muy Baja.

Dado que $K = 4, 2P[K!/(3!(K-3)!)] = 104$ reglas se crean con base en los criterios anteriores. Después, el sistema de inferencia difuso se mejora mediante la adición de siete reglas más. Tres de estas reglas están relacionadas con la Clase A320 1:

1. Regla 105: SI A320 1 NN4 es Alta Y A320_B737-800 NN4 no es Baja Y A320 3 NN4 no es Baja ENTONCES Clase A320 1 es Alta.
2. Regla 106: SI A320 1 NN2 es Alta ENTONCES Clase A320 1 no es Baja.
3. Regla 109: SI A320 1 NN1 no es Baja ENTONCES Clase A320 1 no es Baja.

La agregación de reglas resulta en un conjunto difuso C_p representado por $\mu_{C_p}(u)$, que se calcula de acuerdo con (54) como la unión difusa de cada consecuente de la regla r .

$$\begin{aligned} \mu_{C_p}(u \in U) &= \bigcup_{r=1, \dots, R} \left(\bigcap_{(k,p)} \left(\mu_{A_{k,p}}^r(O_p^k) \right) \cap \mu_{B_p}^r \right) \\ &= \max_{r=1, \dots, R} \left\{ \min_{(k,p)} \left\{ \mu_{A_{k,p}}^r(O_p^k) \right\} \cap \mu_{B_p}^r \right\} \end{aligned} \quad (40)$$

La contribución individual de la regla r con respecto a Z_p está determinada por la fuerza de

Tabla 8. Descripción de la nueva base de datos

p	Clase de Aeronave Modelo del Fuselaje (Tipo de Motor)	Total	Conjuntos Entrenamiento/ Validación/ Prueba
1	SU95 (SaM 146)	26	15/4/7
2	ERJ190 (CF34-10E)	36	15/4/17
3	ERJ170/175 (CF34-8E)	32	15/4/13
4	ERJ145 (AE3007)	37	15/4/18
5	B737-7xx (CF56-7B22-)	38	15/4/19
6	B737-8xx (CF56-7B22+)	39	15/4/20
7	B737-3xx (CFM56-3)	32	15/4/13
8	ATR72-600 (PW127M)	28	15/4/9
9	ATR42-500 (PW127E)	30	15/4/11
10	ATR42-300 (PW120)	25	15/4/6
11	A319-1xx (V25xx)	28	15/4/9
12	A320-2xx (V25xx)	34	15/4/15
13	A320-2xx (CFM56-5)	41	15/4/22
	TOTAL	426	195/52/179

Tabla 9. Topología de la Red Neurona NN_k

k	Topología de NN_k (Neuronas por capa)	n_k
1	85-30-13	55
2	90-30-13	50
3	110-30-13	30
4	130-30-13	10

disparo de la regla r y la función de membresía $\mu_{B_p}^r$, de acuerdo con el valor lingüístico j involucrado en la regla r . La fuerza de disparo de la regla r se calcula con base en las salidas O_p^k asociadas al antecedente de la regla r de acuerdo con la función de membresía $\mu_{A_{k,p}}^r$ de la entrada difusa \hat{O}_p^k , para el valor lingüístico i involucrado en la regla r .

El valor de membresía final c_p se calcula utilizando (55). La Fig. 9 presenta la agregación de reglas para una medición correspondiente a la Clase A320 1:

$$c_p = \frac{\sum_{u \in U} (\mu_{c_p}(u) \cdot u)}{\sum_{u \in U} (\mu_{c_p}(u))}, \quad (37)$$

La contribución resaltada de la *Regla 105* es mayor que la contribución de otras reglas y produce el valor real más alto de membresía como se muestra en la Tabla 6.

La Tabla 6 muestra los resultados después de aplicar ambos modelos, el propuesto en esta sección (*Nuevo Modelo*) y aquel presentado en [19] (*Modelo ESWA*) para una medición perteneciente a la Clase A320 1 ($p = 10$). La Fig. 9 también destaca la contribución individual de la siguiente regla:

Tabla 10. Salidas del Modelo con respecto a una Medición de la Clase ATR42-500 (PW127E)

p	Nuevo Modelo				Final (c_p)	Modelo ESWA				Final (c_p)
	O_p^1	O_p^2	O_p^3	O_p^4		O_p^1	O_p^2	O_p^3	O_p^4	
	-	-	-	-		0.500	0.654	0.558	0.519	
1	1.26E-07	1.14E-11	3.5E-13	1.66E-06	0.0774	0.0702	0.0695	0.0749	0.0664	0.0696
2	5.87E-05	0.112	6.83E-09	2.5E-06	0.271	0.0702	0.0777	0.0749	0.0664	0.0713
3	0.00311	4.53E-06	0.0324	3.04E-08	0.0774	0.0705	0.0695	0.0773	0.0664	0.07
4	4.03E-05	4.47E-07	9.36E-11	7.14E-10	0.0774	0.0702	0.0695	0.0749	0.0664	0.0696
5	0.284	0.0882	5.97E-05	1.36E-05	0.0824	0.0933	0.076	0.0749	0.0664	0.0781
6	0.336	6.09E-10	2.01E-05	0.547	0.439	0.0983	0.0695	0.0749	0.115	0.0938
7	1.2E-07	2.58E-12	2.26E-10	5.81E-10	0.0774	0.0702	0.0695	0.0749	0.0664	0.0696
8	0.343	0.011	0.212	0.845	0.129	0.099	0.0703	0.0925	0.155	0.11
9	1.41E-05	3.86E-06	4.82E-05	5.78E-05	0.528	0.0702	0.0695	0.0749	0.0664	0.0696
10	0.0919	0.767	0.0509	3.98E-05	0.0829	0.077	0.15	0.0788	0.0664	0.0895
11	2.53E-06	1.45E-09	1.99E-08	7.1E-05	0.0774	0.0702	0.0695	0.0749	0.0664	0.0696
12	0.000315	7.63E-07	2.29E-13	3.69E-05	0.0774	0.0703	0.0695	0.0749	0.0664	0.0696
13	5.64E-06	0.00514	0.0377	5.44E-08	0.0774	0.0702	0.0699	0.0777	0.0664	0.0701

Tabla 11. Comparación técnica entre $p = [8,9,10]$

p	Aeronave	Modelo del motor	Longitud	Extensión de las alas
8	ATR72-600	PW 127M	27.17 m	27.05 m
9	ATR42-500	PW 127E/M	22.67 m	24.57 m
10	ATR42-300	PW 120/1	22.67 m	24.57 m

1. Regla 29: **SI** A320 1 NN1 es Baja **Y** A320 1 NN2 es Baja **Y** A320 1 NN3 es Baja **ENTONCES** Clase A320 1 es Muy Baja.

Sin embargo, dados los conjuntos difusos $A_{1,10}^1$, $A_{2,10}^1$, $A_{3,10}^1$ (Baja) y salidas $O_{10}^1 = 0.00814$, $O_{10}^2 = 0.368$, $O_{10}^3 = 2.42 \times 10^{-7}$, la fuerza de disparo de la Regla 29 tiene poco impacto sobre el resultado final. La contribución individual de otras reglas no se aprecia en la Fig. 9, puesto que son insignificantes en comparación con la contribución de otras reglas como la Regla 105 o Regla 29. Más ejemplos y detalles de los resultados obtenidos puede consultarse en [35].

La Tabla 7 muestra una comparación general entre los modelos existentes y el nuevo modelo con respecto a la BD Anterior. Se evalúan un total de 162 mediciones. El Nuevo Modelo clasifica correctamente el 90.7% de estas.

4.2.2. Experimentos con la BD Nueva

La nueva base de datos utilizada en esta sección contiene mediciones del ruido en el despegue de aeronaves modernas actualmente en servicio. Un sistema de adquisición de datos similar al propuesto en [38] se utiliza para registrar el ruido de en el despegue durante 24 segundos a una frecuencia de muestreo de 51.2 kHz. Refiérase a [35] para más detalles.

Con el fin de identificar detalladamente el modelo de las aeronaves despegando, las autoridades aeroportuarias suministraron una lista completa de los despegues realizados durante el período de mediciones de cinco días diferentes. La lista incluye entre otras cosas: hora exacta del despegue, línea aérea, etiqueta corta del modelo (A320, AT42, B737, etc.), etiqueta única de

Tabla 12. Comparación General entre Ambos Modelos)

p	Mediciones	Clasificaciones Correctas	
		Modelo ESWA	Nuevo Modelo
1	7	5	7
2	17	13	14
3	13	9	11
4	18	15	16
5	19	11	16
6	20	13	18
7	13	10	11
8	9	6	8
9	11	7	9
10	6	4	6
11	9	6	7
12	15	11	13
13	22	19	20
TOTAL (%)	179 (100%)	129 (72.1%)	156 (87.2%)

registro (identifica plenamente la aeronave, por ejemplo, N833UA), y el número de vuelo.

Varias bases de datos, especificaciones de los fabricantes y páginas web de aerolíneas fueron revisadas para identificar con precisión el modelo de la aeronave [39-41]. Los resultados de la búsqueda en las bases de datos realizadas con base en la información provista por el aeropuerto se cotejaron con las observaciones en sitio durante las mediciones (hora del despegue, identificación visual del modelo de la aeronave y aerolínea así como el número de la medición). Las clases de aeronaves se definen con base en el modelo del fuselaje, el tipo de motor y las clases acústicas introducidas en [9, 42]. Por lo tanto, las clases que aparecen aquí son muy específicas. La Tabla 8 muestra la descripción de la *BD Nueva*.

Cuatro segmentos de señal ($K = 4$) se extraen de cada medición siguiendo el método descrito en la Sección 2.1, mientras que los 140 rasgos iniciales son definidos para cada segmento k de acuerdo con la Sección 2.2. Una red neuronal NN_k con topología 140- m -13 se define para el

segmento k . El número de neuronas ocultas m se determina con base en un proceso de validación cruzada por diez sobre el espacio $\{5,10,15, \dots, 60\}$. El número m con el promedio de error de prueba más bajo entre diez realizaciones es elegido. Posteriormente, el método de selección de rasgos descrito en la Sección 4.2.2 es aplicado a cada segmento k . El análisis y resultados del método anterior puede consultarse en [35]. La Tabla 9 presenta las topologías de las redes resultantes de la aplicación del método MLP-FSPP-RFE.

Cada salida O_p^k se fuzzifica en \hat{O}_p^k , la cual es etiquetada de acuerdo a *Clase de Aeronave + NN_k* . Las mismas funciones de membresía $A_{k,p}^i$ y conjuntos difusos $A_{k,p}^i$ usados para los experimentos con la *BD Anterior* en la Sección 4.2.1 se asignan a los valores lingüísticos *Baja*, *Media* y *Alta* de cada entrada difusa \hat{O}_p^k . Del mismo modo, las salidas difusas Z_p son etiquetadas de acuerdo con *Clase + Clase de Aeronave*. Las mismas funciones de membresía $\mu_{B_p}^r$ y conjuntos

difusos B_p^j usados para los experimentos con la *BD Anterior* se asignan a los valores lingüísticos *Muy Baja*, *Baja*, *Media*, *Alta* y *Muy Alta* con respecto a Z_p . Inicialmente tres tipos de reglas se utilizan para los experimentos con la *BD Nueva*:

1. $\forall(a, b, c) \mid a, b, c \in \{1, 2, \dots, K\}, a \neq b \neq c;$
SI \hat{O}_p^a es *Alta* **Y** \hat{O}_p^b es *Alta* **Y** \hat{O}_p^c es *Alta*
ENTONCES Z_p es *Muy Alta*.
2. $\forall(a, b, c) \mid a, b, c \in \{1, 2, \dots, K\}, a \neq b \neq c;$
SI \hat{O}_p^a es *Baja* **Y** \hat{O}_p^b es *Baja* **Y** \hat{O}_p^c es *Baja*
ENTONCES Z_p es *Muy Baja*.
3. $\forall(a, b) \mid a, b \in \{1, 2, \dots, K\}, a \neq b;$
SI \hat{O}_p^a es *Media* **Y** \hat{O}_p^b es *Media* **ENTONCES** Z_p es *Media*.

Con base en los tipos de reglas anteriores y dado que $K = 4$, $P\{2[K!/(3!(K-3)!)] + K!/(2!(K-2)!)\} = 182$ reglas son generadas. Posteriormente, se incluyen 23 reglas más para mejorar el sistema de inferencia difusa. La Tabla 10 presenta las salidas de ambos modelos con respecto a una medición de la *Clase ATR42-500 (PW127E)* (Respecto al *Modelo ESWA*, la importancia de la red NN_k se recalcula con base en los nuevos entrenamientos). Tres clases son relevantes en la medición anterior: *ATR72-600 (PW127M)* ($p = 8$), *ATR42-500 (PW127E)* ($p = 9$), y *ATR42-300 (PW120)* ($p = 10$).

Las redes neuronales regresan principalmente valores no despreciables para dos de estas clases $p = 8$ y $p = 10$. Como se muestra en la comparación técnica de la Tabla 11 [43-45], las clases $p = [8, 9, 10]$ están muy correlacionadas entre sí.

Dicha correlación produce incertidumbre entre las salidas O_p^k de las redes neuronales como se revela en la Tabla 10. Lo anterior sugiere que el conocimiento acerca de las clases de aeronave y su correlación (similitud) puede ser utilizado para identificar incertidumbres (múltiple clases identificadas como correctas) y traducirlos en una clasificación exitosa con base en un sistema de inferencia difusa. Por ejemplo, dado que el avión *ATR42-500* tiene ciertas características del *ATR42-300* y *ATR72-600*, la incertidumbre en la Tabla 10 con respecto a $p = [8, 9, 10]$ podría ser usada para inferir que es una medición de la *Clase*

ATR42-500 (PW127E). Más ejemplos y detalles de los resultados obtenidos puede consultarse en [35].

La comparación global de ambos modelos con respecto a la *BD Nueva* se muestra en la Tabla 12. Se evaluaron un total de 179 mediciones, de las cuales el 87,2% son clasificadas correctamente por el *Nuevo Modelo*. El mejor desempeño del modelo propuesto está estrechamente relacionado con la capacidad de cambiar el resultado final respecto a la clase p con base en cualquier salida $O_{p+n}^k \mid n \in \mathbb{Z} - \{0\} \wedge 0 < p + n \leq P$ de la red neuronal NN_k . De hecho, cualquier combinación de las salidas $O_{p+n}^{k+m} \mid m \in \mathbb{Z} - \{0\} \wedge 0 < k + m \leq K$ puede usarse para inferir sobre con respecto a cualquier clase p . Además, la inferencia difusa con respecto a la entrada $\{x_1, \dots, x_K\}$ bajo la correspondencia parcial de las reglas relacionadas con la clase y reduce la incertidumbre incluso cuando $O_y^k \neq \max_{p=1,2,\dots,P} \{O_p^k\}$ y $\{x_1, \dots, x_K\}$ pertenezca a la clase y . Más aún, la jerarquía de la red neuronal NN_k respecto a la clase p está determinada por las reglas difusas que involucran \hat{O}_p^k como parte del antecedente y Z_p como consecuente, lo que reduce la incertidumbre introducida por un alto valor erróneo O_p^k cuando NN_k tiene una alta jerarquía en relación a la clase p , ya que podría ser apoyado por cualquier otra entrada difusa \hat{O}_{p+n}^{k+m} .

5. Conclusiones

Mediante la segmentación de la señal en el tiempo se logra identificar una disimilitud apreciable entre los espectros de cada segmento, confirmando que el ruido de las aeronaves en el despegue es un proceso no estacionario, que varía tanto en amplitud como en frecuencia. Además, se demuestra que la identificación de la clase de aeronave a partir de un modelo que combine los patrones extraídos por cada segmento es superior al uso de los patrones extraídos de la señal completa.

Dichos modelos con K clasificadores paralelos, uno por segmento $k = 1, 2, \dots, K$, generan múltiples salidas O_p^k sobre el mismo espacio de clases $p = 1, 2, \dots, P$. Las múltiples salidas O_p^k con respecto a

la clase p son combinadas usando un método de agregación que permite un mayor índice de reconocimiento. La agregación o combinación de las salidas se realiza a partir de la definición de una jerarquía por cada clase p de todos los clasificadores con base en la cuantificación del desempeño durante la etapa de validación usando la métrica F_{β} . Además se introduce la dinámica en el proceso de ponderado con base en la similitud de las respuestas de todos los clasificadores con respecto a la clase p .

Asimismo, se introduce un modelo multicapa neuro-difuso que permite decidir sobre la clase p no solamente con base en las salidas O_p^k sino también respecto a las salidas $O_{p+n}^{k+m} \mid m \in \mathbb{Z} - \{0\} \wedge 0 < k + m \leq K$. Ello permite modelar relaciones existentes entre clases que están correlacionadas y que algunos clasificadores del conjunto no logran diferenciar. Además, la arquitectura multicapa neuro-difusa propuesta permite una gran flexibilidad para cambiar el resultado final con respecto a la clase p a través de la adición/eliminación de reglas de inferencia difusas. Más aún, el nuevo modelo neuro-difuso es más escalable ya que permite adicionar una nueva serie de salidas O_p^k respecto a las clases $p = 1, 2, \dots, P$ al proceso de inferencia difusa.

Más detalles de la metodología y resultados obtenidos en este trabajo pueden ser consultados en [19, 22, 23, 31, 35, 38].

Referencias

1. **ICAO (2008)**. Environmental Protection. Aircraft Engine Emissions, International Civil Aviation Organization (ICAO). *Annex 16 to the Convention on International Civil Aviation*.
2. **ICAO (2008)**. Environmental Protection. Aircraft Noise, International Civil Aviation Organization (ICAO). *Annex 16 to the Convention on International Civil Aviation*.
3. **ICAO (2010)**. *Environmental Report, Aviation and Climate Change*. International Civil Aviation Organization (ICAO).
4. **ICAO (2007)**. *Guidance on the Balanced Approach to Aircraft Noise Management*. International Civil Aviation Organization (ICAO), Doc 9829 AN/451.
5. **ICAO (2008)**. *Recommended Method for Computing Noise Contours Around Airports*. International Civil Aviation Organization (ICAO), Canada, Doc. 9911.
6. **SAE (1986)**. *Procedure for the calculation of airplane noise in the vicinity of airports*. Society of Automotive Engineers (SAE), AIR 1845.
7. **ECAC (2005)**. *Methodology for Computing Noise Contours around Civil Airports. Application Guide*. European Civil Aviation Conference (ECAC), Doc. 29.
8. **ECAC (2005)**. *Methodology for Computing Noise Contours around Civil Airports. Technical Guide*. European Civil Aviation Conference (ECAC), Doc. 29.
9. **FAA (2013)**. *Integrated Noise Model (INM)*. Computer model to evaluate the aircraft noise impacts in the vicinity of airports, 7.0d ed: Federal Aviation Administration (FAA).
10. **MASSPORT (2015)**. *Logan Airport Noise Monitoring System*. <http://www.massport.com/environment/environmental-reporting/noise-abatement/noise-monitoring/>.
11. **SAE (2012)**. *Monitoring Aircraft Noise and Operations in the Vicinity of Airports: System Description, Acquisition, and Operation*, Society of Automotive Engineers (SAE). ARP 4721 Part 1.
12. **SAE (2012)**. *Monitoring Aircraft Noise and Operations in the Vicinity of Airports: System Validation*, Society of Automotive Engineers (SAE). ARP 4721 Part 2.
13. **MWAA (1981)**. *DCA Nighttime Noise Rule MWWA*. <http://www.metwashairports.com/reagan/7043.htm>.
14. **ISO (1996)**. *Acoustics - Attenuation of sound during propagation outdoors - Part 2: General method of calculation, in ISO 9613-2*. ed: International Organization for Standardization (ISO).
15. **ISO (1993)**. *Acoustics - Attenuation of sound during propagation outdoors - Part 1: Calculation of the absorption of sound by the atmosphere, in ISO 9613-1*. ed: International Organization for Standardization (ISO).
16. **NASA (1999)**. *Making future commercial aircraft quieter*. NASA Glenn Research Center, Cleveland, Ohio.
17. **SAE (2006)**. *Method for Predicting Lateral Attenuation of Airplane Noise*. Society of Automotive Engineers (SAE), AIR 5662.
18. **Plotkin, K. J., Hobbs, C. M., & Bradley K. A. (2000)**. *Examination of the Lateral Attenuation of Aircraft Noise*.
19. **Sánchez-Pérez, L. A., Sánchez-Fernández, L. P. Suárez-Guerra, S., & Carbajal-Hernández, J. J.**

- (2013). Aircraft class identification based on take-off noise signal segmentation in time. *Expert Systems with Applications*, Vol. 40, pp. 5148–5159.
20. **Sánchez-Fernández, L. P., Pogrebnyak, O., Oropeza-Rodríguez, J. L., & Suárez-Guerra, S. (2007).** *Noise Pattern Recognition of Airplanes Taking Off: Task for a Monitoring System*. Progress in Pattern Recognition, Image Analysis and Applications, pp. 831–840.
 21. **Rojo-Ruiz, A., Sánchez-Fernandez, Felipe-Riverón, L. E., & Suárez-Guerra, S. (2008).** *Computational Model for Aircraft's Takeoffs Pattern Recognition*. Progress in Pattern Recognition, Image Analysis and Applications, Vol. 5197, Springer Berlin Heidelberg, pp. 14–21.
 22. **Sánchez-Fernández, L. P., Sánchez-Pérez, L. A., Carbajal-Hernández, J. J., & Rojo-Ruiz, A. (2013).** Aircraft Classification and Acoustic Impact Estimation Based on Real-Time Take-off Noise Measurements. *Neural Processing Letters*, Vol. 38, pp. 239–259.
 23. **Márquez-Molina, M., Sánchez-Fernández, L. P., Suárez-Guerra, S., & Sánchez-Pérez, L. A. (2014).** Aircraft take-off noises classification based on human auditory's matched features extraction. *Applied Acoustics*, Vol. 84, pp. 83–90.
 24. **Serway, R. A. & Jewett, J. W. (2010).** *Physics for Scientists and Engineers*. 8 ed. Belmont, USA: Brooks/Cole.
 25. **Markel, J. & Gray, A. (1976).** *Linear Prediction of Speech*. Berlin, Germany: Springer-Verlag.
 26. **Li, D., Acero, A., & Bazzi, I. (2006).** Tracking vocal tract resonances using a quantized nonlinear function embedded in a temporal constraint. *Audio, Speech, and Language Processing, IEEE Transactions on*, Vol. 14, pp. 425–434.
 27. **Hagen, R., Paksoy, E. & Gersho, A. (1999).** Voicing-specific LPC quantization for variable-rate speech coding. *Speech and Audio Processing, IEEE Transactions on*, Vol. 7, pp. 485–494.
 28. **Mi Suk, L., Hong Kook, K., & Hwang Soo, L. (1999).** LPC analysis incorporating spectral interpolation for speech coding. *Electronics Letters*, Vol. 35, pp. 200–201.
 29. **Jian-Bo, Y., Kai-Quan, Chong-Jin, S. O., & Xiao-Ping, L. (2009).** Feature Selection for MLP Neural Network: The Use of Random Permutation of Probabilistic Outputs. *Neural Networks, IEEE Transactions on*, Vol. 20, pp. 1911–1922.
 30. **Asensio, C., Ruiz, M., & Recuero, M. (2010).** Real-time aircraft noise likeness detector. *Applied Acoustics*, Vol. 71, pp. 539–545.
 31. **Sanchez-Perez, L.A., Sanchez-Fernandez, L.P., Suarez-Guerra, S., & Lopez-Pacheco M.G. (2015).** Dynamic hierarchical aggregation of parallel outputs for aircraft take-off noise identification. *Engineering Applications of Artificial Intelligence*, Vol. 46, Part A, pp. 33–42.
 32. **Ramya, R., Anandanatarajan, R. Priya, R., & Selvan G. A. (2012).** Applications of fuzzy logic and artificial neural network for solving real world problem. *Advances in Engineering. Science and Management (ICAESM), International Conference on*, India.
 33. **Özbay, Y., Ceylan, R., & Karlik, B. (2011).** Integration of type-2 fuzzy clustering and wavelet transform in a neural network based ECG classifier. *Expert Systems with Applications*, Vol. 38, pp. 1004–1010, Jan.
 34. **Yörükoğlu, A. & Erdinç, A. (2013).** Estimation of Unbalanced Loads in Washing Machines Using Fuzzy Neural Networks. *Mechatronics, IEEE/ASME Transactions on*, Vol. 18, pp. 1182–1190.
 35. **Sanchez-Perez, L.A., Sanchez-Fernandez, L.P., Shaout, A., & Suarez-Guerra, S. (2016).** Airport take-off noise assessment aimed at identify responsible aircraft classes. *Science of The Total Environment*, Vol. 542, Part A, pp. 562–577.
 36. **Zadeh, L.A. (1972).** A Rationale for Fuzzy Control. *Journal of Dynamic Systems, Measurement, and Control*, Vol. 94, pp. 3–4.
 37. **Ross, T.J. (2010).** *Fuzzy Logic with Engineering Applications*. 3 ed. United Kingdom: John Wiley & Sons, Ltd.
 38. **Sánchez-Pérez, L.A., Sánchez-Fernández, L.P., Suárez-Guerra, S., & Márquez-Molina, M. (2014).** Geo-referenced flight path estimation based on spatio-temporal information extracted from aircraft take-off noise. *Digital Signal Processing*, Vol. 30, pp. 1–14.
 39. **FAA (2015).** *Releasable Aircraft Database*. <http://registry.faa.gov/database/AR022015.zip>
 40. **Aumont, A., Antonakis, A., Croy, B., Sloman, B., Duijnmayr, K., Blinkhorn, P., & et al. (2015).** *Aircraft Registration Database*. Available: <http://www.airframes.org/>
 41. **Flightradar24 (2015).** *Airline fleet database*. <http://www.flightradar24.com/data/airplanes>.
 42. **ANP (2012).** The Aircraft Noise and Performance (ANP) Database. *An international data resource for aircraft noise modellers*, <http://www.aircraftnoisemodel.org/>

43. **ATR (2011).** *ATR 42-300-320 Brochure.*
http://www.atraircraft.com/products_app/media/pdf/ATR_42-300-320.pdf.
44. **ATR (2014).** *ATR-500 Series Brochure.*
http://www.atraircraft.com/products_app/media/pdf/Fiche_ATR%2042-500_septembre2014.pdf.
45. **ATR (2014).** *ATR 72-600 Leaflet.*
http://www.atraircraft.com/products_app/media/pdf/Fiche_72-600_septembre%202014.pdf.

Luis Alejandro Sánchez-Pérez obtuvo su doctorado en el Centro de Investigación en Computación (CIC) del Instituto Politécnico Nacional (IPN), México 2016, recibiendo la Medalla Lázaro Cárdenas en la categoría de Doctorado en Ingeniería y Ciencias Físico Matemáticas, el mayor reconocimiento académico que otorga el instituto. Actualmente pertenece al Sistema Nacional de Investigadores del CONACYT con nombramiento de Investigador Nacional SIN-I. En este momento pertenece al Departamento de Ingeniería Eléctrica y Computación de la Universidad de Michigan – Dearborn como Académico Afiliado y Posdoctorante. Sus áreas de investigación son Machine Learning, Inteligencia Artificial, Procesamiento Digital de Señales, Arreglo de Sensores y Procesamiento Multicanal.

Luis Pastor Sánchez-Fernández obtuvo su doctorado en ciencias técnicas por el Instituto Superior Politécnico José Antonio Echeverría en 1998, Habana, Cuba. Actualmente se desempeña como Profesor/Investigador en el Centro de Investigación en Computación (CIC) del Instituto Politécnico Nacional (IPN). Pertenece al Sistema Nacional de Investigadores en la categoría de Investigador Nacional SNI-I. Sus áreas de investigación son Instrumentación Virtual, Procesamiento de Señales, Reconocimiento de Patrones y Control Automático.

Sergio Suarez-Guerra obtuvo su doctorado en ciencias técnicas en el Centro de Computación de la Academia de Ciencias de la Unión de Repúblicas Soviéticas (URSS) en 1979. Actualmente es Profesor/Investigador en el Centro de Investigación en Computación (CIC) del Instituto Politécnico Nacional. Pertenece al Sistema Nacional de Investigadores en la categoría de Investigador Nacional SNI-I. Sus áreas de investigación son Procesamiento de Señales, Reconocimiento de Patrones y Voz.

*Artículo recibido el 09/08/2016; aceptado 02/11/2016.
Autor de correspondencia. Luis Alejandro Sánchez Pérez.*