

Diseño de un algoritmo Fuzzy C-Means para predicción de contaminación de aire basado en VHDL

Juan M. De la Cruz-Aguirre¹, Marco A. Aceves-Fernández²✉, Efren Gorrostieta-Hurtado², José Emilio Vargas-Soto²

1 Facultad de Informática, Universidad Autónoma de Querétaro, Querétaro, México
juanmca@gmail.com

2 Facultad de Ingeniería, Universidad Autónoma de Querétaro, Querétaro, México
marco.aceves@gmail.com

Resumen

Las partículas de suspensión contaminantes, como por ejemplo las PM10 y las PM25, son algunas de las más dañinas para la humanidad dado el diámetro mínimo aerodinámico que ellas tienen. Esto, aunado a la creciente demanda de factores que las producen en las ciudades cada vez más pobladas, las hacen un riesgo serio para sus habitantes. Los factores de detección deben ser cada vez más efectivos, precisos y accesibles para que puedan llegar a donde más se necesitan. A lo largo de este trabajo se explora y deduce un algoritmo de predicción basado en el método de Clustering Difuso C-Means suficientemente capaz de pronosticar estos índices contaminantes, además se propone y compara su implementación en VHDL y en Matlab para determinar la obtención de los mismos resultados con ambas arquitecturas, complementando de esta forma la factibilidad de que pueda ser implementada de forma más rápida y con menos recursos.

Palabras clave: VHDL, Fuzzy C-Means, PM10, Contaminantes.

1. Introducción

La acción de respirar obliga a un contacto permanente entre el aparato respiratorio y el medio ambiente. Esta relación es imprescindible para la vida, aunque nos hace vulnerables a la acción de los contaminantes suspendidos en el aire que respiramos [1]. Los pulmones se convierten en la puerta de entrada, con frecuencia de una manera invisible, para un gran número de sustancias con capacidad de causar enfermedad respiratoria, cardíaca o de otros órganos. La preocupación por conseguir un aire limpio es compartida por científicos e instituciones [2].

La acción de respirar obliga a un contacto permanente entre el aparato respiratorio y el medio ambiente. Esta relación es imprescindible para la vida, aunque nos hace vulnerables a la acción de los contaminantes suspendidos en el aire que respiramos. Los pulmones se convierten en la puerta de entrada, con frecuencia de una manera invisible, para un gran número de sustancias con capacidad de causar enfermedad respiratoria, cardíaca o de otros órganos. La preocupación por conseguir un aire limpio es compartida por científicos e instituciones [1].

Esto ha servido de fundamento para determinar que la calidad del aire que respiramos es de vital importancia para nuestra salud y bienestar; esta calidad depende de la presencia en la atmósfera de algunos contaminantes, que en concentraciones superiores a los niveles límite establecidos pueden resultar peligrosos para la salud y/o el bienestar de las personas, o para la calidad de los ecosistemas naturales. El diseño de las ciudades es de gran importancia en cuanto a la calidad del aire local se refiere, ya que los elementos que lo configuran (principalmente aspectos relacionados con la movilidad y los procesos industriales) determinan, junto con las condiciones meteorológicas, la emisión, la distribución y la difusión de los contaminantes atmosféricos [3]. Los principales

contaminantes atmosféricos son el dióxido de azufre, el dióxido de nitrógeno, el monóxido de carbono, los compuestos orgánicos volátiles, las partículas en suspensión y el plomo [4].

Existe una asociación directa entre el tamaño y la masa de las partículas suspendidas, con los efectos en la salud humana. La principal vía de ingreso al organismo es durante la respiración, pero no todas las partículas en suspensión ingresan al sistema respiratorio, entre más pequeñas son, más profundamente ingresan. Las partículas más grandes se retienen en la región nasal, mientras que las menores a 10 μm pueden penetrar más allá de la laringe. Las más pequeñas, menores a 1 μm , son capaces de ingresar hasta la región alveolar, que es la zona en donde se realiza el intercambio de oxígeno. En el monitoreo de la calidad del aire se utilizan dos indicadores para el monitoreo de las partículas suspendidas: las partículas menores a 10 μm (PM10) y las partículas menores a 2.5 μm (PM2.5) [5].

En clustering tradicional u otros modelos como los mostrados en [6] [7] [8], una entidad pertenece únicamente a un solo cluster. En esta contribución, se presenta la técnica llamada Fuzzy Clustering Means (FCM), en la cual una entidad puede pertenecer a uno o más clusters basado en diversos parámetros como su localización en el histograma y que grado de pertenencia ocupa en cada cluster. Esta versatilidad provee una mayor certeza en la predicción, en este caso, de contaminantes.

2. Marco Teórico

El nombre de éste segundo título dependerá de cómo se haya estructurado el artículo. En caso de que la estructura del artículo requiera de subsecciones, estas se enumerarán de acuerdo a su orden de aparición. Todos los títulos (Resumen, Introducción, etc.) se escribirán usando el tipo Arial normal de 12 puntos.

2.1 Fuzzy C-Means (FCM)

(FCM) es uno de los métodos de clustering más utilizados. Desarrollado por Dunn [8], y más adelante mejorado por Bezdek [9], permite a un sector de datos pertenecer a uno o más clusters con grados variantes de membresía. En aplicaciones reales, en ocasiones no hay una frontera puntual visible entre clusters colindantes. Esto conduce a un clustering difuso mejor situado en los datos. Los grados de Membresía entre cero y uno son usados en clustering difuso en lugar de asignaciones llanas de los datos a los clusters. Una descripción detallada de los fundamentos de clustering difuso, algoritmos básicos, implementaciones, así como evaluación de validez de cluster y resultados de visualización fueron propuestas por Kruse [10].

En el presente trabajo, y como se espera para el algoritmo, inicialmente los centros están ubicados en lugares inexactos y para cada punto de datos se obtiene su correspondiente membresía para cada cluster. Por iteraciones, los centros de los clusters y los grados de membresía son actualizados para cada punto de datos minimizando una función objetivo que representa la distancia de cada punto dado a un centro de cluster. El resultado final de un sistema de clustering difuso es una lista de centros de clusters y los respectivos grados de membresía para cada punto de datos, los cuales servirán de antecedente para pronosticar las partículas de agentes contaminantes en el algoritmo a ser implementado. Este Algoritmo está basado en la minimización de la función objetivo [11] mostrada en (1).

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2, 1 \leq m < \infty \quad (1)$$

En donde para desarrollar un algoritmo que pueda ser implementado en código es necesario separarlo en componentes específicos. Primero se definen los datos específicos que soportaran la definición del algoritmo, dentro de los cuales se encuentran el elemento a evaluar, el centro actual evaluado, la secuencia de centros que formarán parte de la operación y el Fusificador (Fuzzifier) utilizado como valor constante igual a 2 para las iteraciones de entrenamiento de la red de lógica difusa.

Tomando este valor de 2 se puede definir la potencia a la que será evaluada el divisor, la cual toma un resultado de dos al realizar la operación respectiva de $2/(m-1)$ que se define en la ecuación de Membrecía [12] de (2) y (3).

$$u_{ij} = \frac{1}{\sum_{k=1}^C \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \quad (2)$$

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m \cdot x_i}{\sum_{i=1}^N u_{ij}^m} \quad (3)$$

El proceso iterativo de optimización termina cuando su criterio de termino ∂ se cumple, por ejemplo, en (4), donde ∂ es el criterio de término entre 0 y 1 y k es el número de iteración. Después de la formación del cluster, la red se separa en agrupaciones (clusters) y el proceso comienza localmente con cada uno de los clusters.

$$\max_{ij} \left\{ \left| u_{ij}^{(k+1)} - u_{ij}^{(k)} \right| \right\} < \partial \quad (4)$$

2.2 Modelo del algoritmo

Dentro del clustering o agrupamiento de formas de datos existen metodologías diferentes y cada una de ellas puede aplicarse como mejor corresponda a un objetivo específico. En el caso del presente desarrollo se optó por el esquema difuso que nos permitirá definir un modelo que pronostique de forma efectiva agentes contaminantes en un entorno común.

El modelo, inicia con un grupo de valores reales de agentes contaminantes que se irán agrupando en grupos (clusters). Cada uno de estos clusters tiene un valor centro, y cada uno de los elementos reales tiene cierto grado de membrecía con respecto al centro de cada grupo.

El primer paso del algoritmo a implementar consiste en definir un umbral inicial con el cual mediremos la aproximación de nuestro modelo, este límite marcará la diferencia máxima que puede existir entre los valores pronosticados que se definen con nuestros centros calculados con respecto a valores proporcionados con un sistema real. Este valor es fundamental ya que de él dependerá la certeza con el que el algoritmo podrá predecir los valores objetivo esperados.

Una vez definido un umbral es necesario establecer un primer modelo con el cual trabajar, para esto se asignan centros aleatorios para cada grupo; estos valores de centros tendrán que ser representativos del grupo en total.

Con estos primeros centros se procede a ir obteniendo los datos reales y junto con los valores de los centros se continúa con los cálculos respecto a la membrecía.

Para proceder a calcular los valores de membrecía se requiere tener todos los datos dentro de un arreglo para poder proceder a la fórmula de membrecía, a ser descrita en la siguiente parte de este capítulo. De esta forma se va preguntando si es el último dato y en caso afirmativo se pasa a calcular los valores de membrecía de los elementos del grupo con respecto a los centros predeterminados. Si no es el último dato se va adquiriendo nuevamente hasta completar el arreglo.

3. Arquitectura

3.1 Definición

El sistema propuesto de predicción consta de 3 pasos principales. Dichos pasos se ejecutan de la siguiente manera. Primeramente, se obtienen las lecturas de las membresías de los grupos actuales. Posteriormente, se calculan los valores de los centros de cada cluster a ser utilizados en la predicción del día posterior. Por último, se calculan los nuevos centros y se cambia el umbral para realizar la predicción. Las funciones principales que integran el código en VHDL se mencionan a continuación.

3.2 Arquitectura del Sistema

El algoritmo se implementa por medio de la descripción de mayor nivel. La arquitectura se muestra en la figura 1.

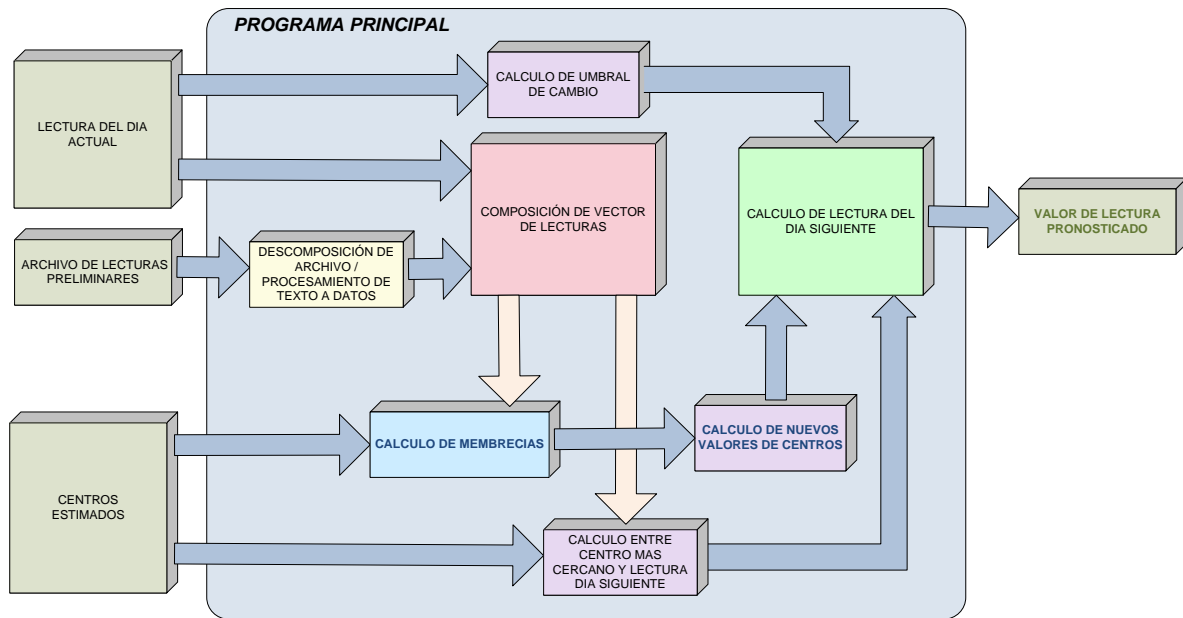


Figura 1. Arquitectura del Algoritmo VHDL para pronosticar partículas contaminantes basado en Fuzzy-C Means.

En las siguientes secciones, se habla del resto de los módulos requeridos para realizar las predicciones.

4. Definición de módulos

4.1 Ciclo de partícula y apertura de archivo de datos

El primer paso después de iniciar el programa de ejecución principal es darle valores preliminares a los centros que se utilizarán para las operaciones, los valores que se incluyen son dados de acuerdo con los centros calculados del grupo de datos guardado y que se basa en predicciones anteriores.

4.2 Valores de centroide inicial y leer los datos de

El primer paso en el programa principal es asignar valores preliminares a los centros, ellos pueden ser definidos de acuerdo con los análisis estadísticos o crear una tabla de dispersión de la lista de lectura de datos. Los valores de los centros de los grupos donde la mayor concentración de partículas es percibida son seleccionados como los valores iniciales.

4.3 Calcular umbral de cambio y reemplazo de lecturas

Una vez que tenemos los datos de las partículas para realizar el pronóstico procedemos a hacer el cálculo del umbral de cambio que nos servirá para el pronóstico de las partículas; el UMBRAL DE CAMBIO como lo hemos definido es la distancia de la lectura que tenemos guardada en nuestros datos con respecto a la lectura que se está leyendo y de ahí definir cuanto están cambiando los agentes contaminantes. Para esto, primeramente, debemos definir si la lectura del día es mayor o menor que la lectura guardada en los datos para ese día, esta comparación se hace para sabiendo cuál de las dos es mayor proceder a restar una de la otra y obtener la diferencia.

De esta forma ya tenemos cuanto cambia la lectura guardada con respecto a la leída (UMBRAL DE CAMBIO) y si este valor tiene que ser sumado o restado a la nueva partícula que definiremos con nuestros centros y membrecías. Una vez terminado el cálculo del umbral de cambio procedemos a reemplazar los valores leídos en nuestro grupo de valores almacenados, el grupo se encuentra en la variable lecturas y está apuntando al día de lectura actual; los valores guardados del día están en la variable a la que se iguala y que es lecturaDiaParticula. También en los dos arreglos se indica la partícula actual que está siendo evaluada.

De esta forma, con el nuevo grupo de lecturas, incluyendo las lecturas recientes, podemos dar paso a la obtención de los valores de membrecías. Mostrando esta función en un diagrama Top-Down queda como se muestra en la figura 3.

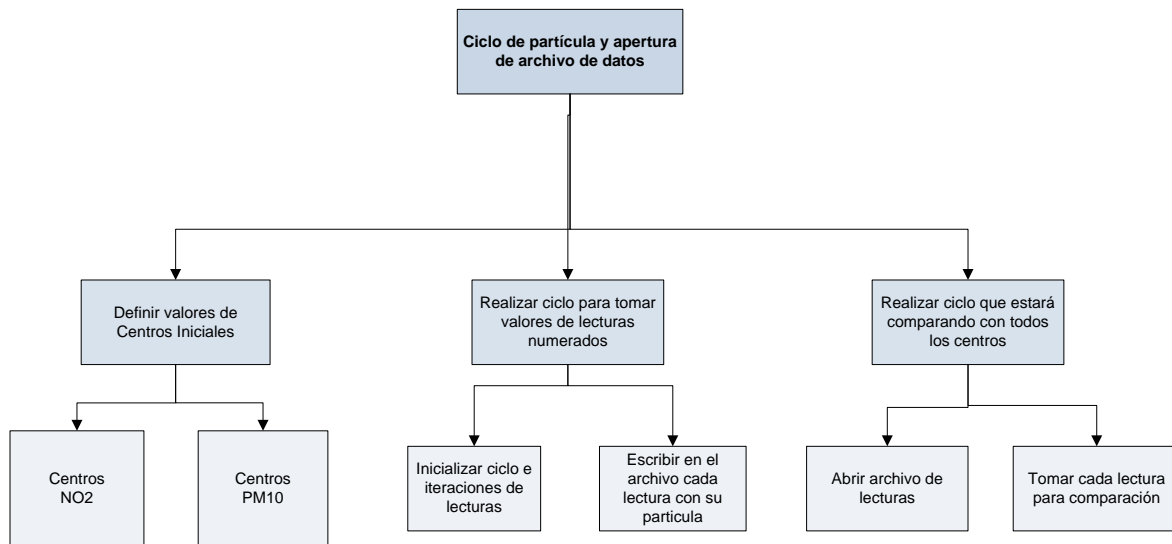


Figura 2. Diagrama Top-Down de Ciclo de partícula y apertura de archivo de datos.

1.1 Distancia entre centro más cercano y lectura día siguiente

Se procede entonces a calcular la distancia desde el centro más cercano a la lectura del día siguiente para tener un parámetro de concordancia el cual tener de referencia para un pronóstico más acertado. La realización de este cálculo inicia un nuevo ciclo en el que se irá tomando centro por centro para verificar si es el que se encuentra más cerca.

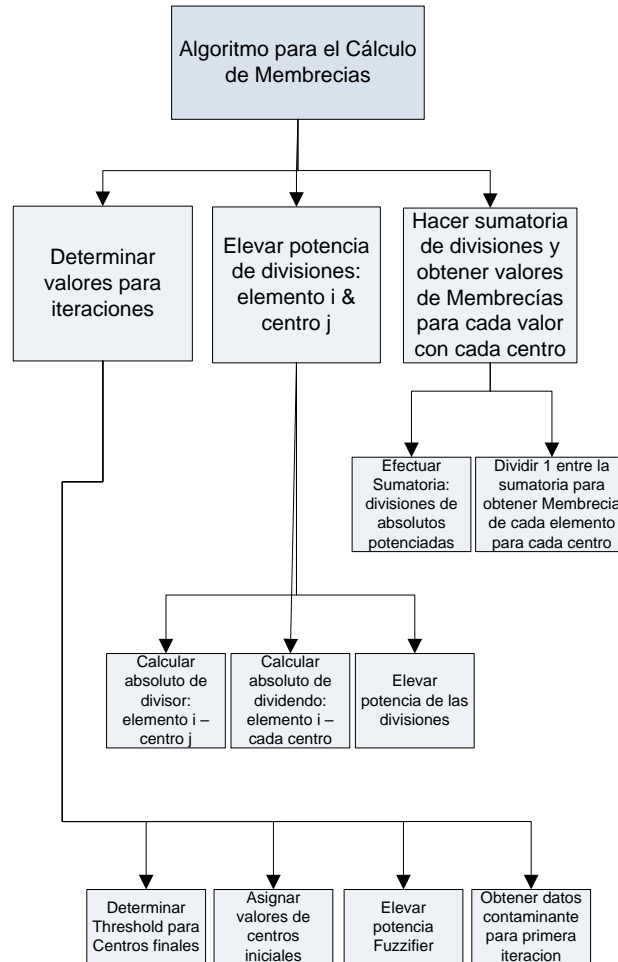


Figura 3. Diagrama Top-Down para el Cálculo de Membrecías.

Una vez que se tiene el centro de la iteración se comienzan a hacer operaciones con él, lo primero es comparar si este valor o el de la lectura son más grandes para que se proceda a evaluarse respecto al próximo centro en la iteración. Para obtener la lectura del día siguiente se le suma un valor de uno al índice del vector de Lecturas.

En caso de que esta distancia nueva calculada sea mayor, entonces el nuevo valor de la distancia del centro a esta lectura se guardará como el nuevo valor que estamos buscando, en caso contrario se descarta. También se guarda la iteración que corresponde al centro que debe usarse para modificar la lectura correspondiente con este valor de centro que se obtuvo de la iteración.

1.2 Calcular nuevos valores de centros con membreías obtenidas.

Ya que se han efectuado todas las operaciones necesarias con los valores de centros actuales se procede a calcular los nuevos valores de centros que servirán de base para el pronóstico de la lectura del día siguiente, para esto primeramente se inician dos ciclos uno para ir evaluando cada centro y dentro de este ciclo ir evaluando la lectura.

El cálculo de los nuevos valores de centros implica para cada lectura la división de, la sumatoria de la membreía actual correspondiente elevada al valor del fuzzifier multiplicada por la lectura que se está evaluando, entre la sumatoria de todas las membreías de esta lectura elevadas al valor del fuzzifier con respecto a todos los centros. Para esto, primero se encuentra el valor del divisor para el cual el primer paso es iniciar un ciclo para ir calculando todas las membreías. De esta forma tendremos los nuevos valores de centros para obtener la lectura del día siguiente por partícula, como se muestra en la figura 4.

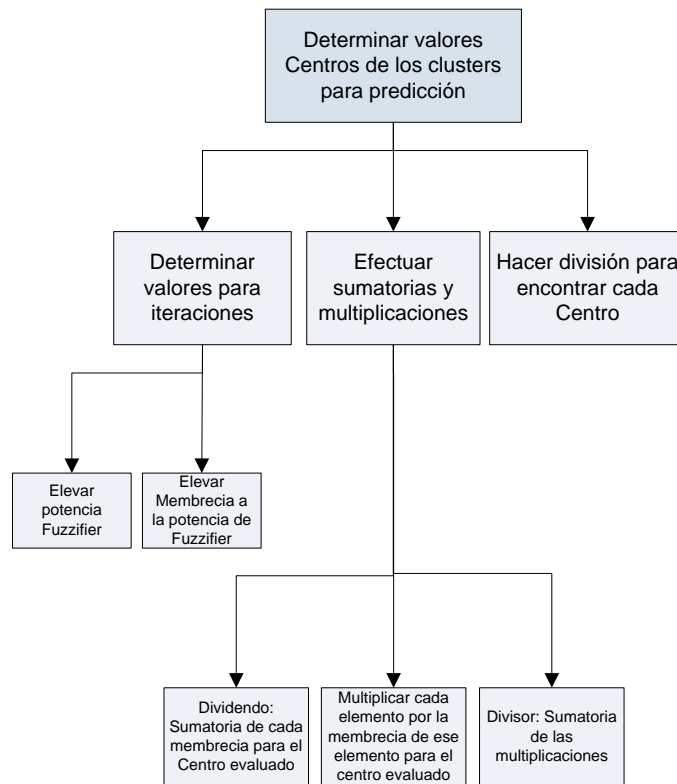


Figura 4. Diagrama Top-Down para el Cálculo de Nuevos Centros.

1.3 Pronosticar lectura del día siguiente.

Una vez que se tienen los valores de membreías y los nuevos valores de centros calculados se inician las operaciones para encontrar la lectura del día siguiente. Para esto primeramente tenemos que determinar cómo será aplicada la variable calculoUmbralDeCambio que pueden ser sumada o restada, de acuerdo a como se haya determinado.

Después se procede a evaluar también la variable `calculoDiaSiguieteCentro` y en caso de que las dos se sumen se procede a realizar la adición de ambas al parámetro que guardará nuestro valor final de la lectura del día siguiente el cual se denominó `lecturaDiaSiguiete`, este parámetro tiene un sub-parámetro `particula` para indicar a cuál tipo de esta pertenece y que está siendo evaluada.

En caso de que la condición resulte en que la distancia del centro sea restada entonces se procede a restar la variable `sumarDistanciaCentro` del parámetro que contiene el valor de la lectura anterior y que finalmente contendrá el nuevo valor de partícula; habiendo evaluado si ambas condiciones se suman se procede a cerrar el condicional que evalúa si se suma la distancia hacia el centro. Por último, se cierran el condicional respectivo que se estaba operando, el ciclo correspondiente y el proceso general del programa VHDL respectivo para finalizar con las operaciones. La figura 5 muestra el Diagrama Top-Down correspondiente.

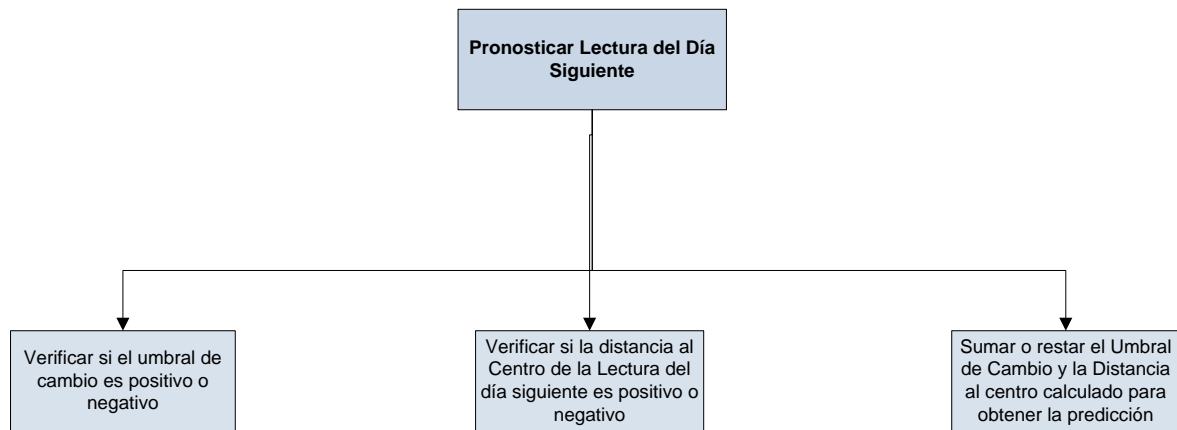


Figura 5. Diagrama Top-Down para Pronosticar la Lectura del Día Siguiete.

En la siguiente sección se revisarán la simulación y los resultados que arroja el algoritmo.

2. RESULTADOS

2.1 Simulación en VHDL.

Un sistema de predicción de partículas contaminantes debe ser capaz de interactuar con ciudades donde la contaminación es grave y poder así ser una herramienta útil para poder realizar predicciones. De esta forma, para las etapas de simulación y verificación, datos reales han sido usados. Datos previamente guardados son evaluados y divididos en archivos de 8760 muestras cada uno. Cada archivo representa un año de información guardada de NO₂ y PM₁₀, respectivamente. Sin embargo, la dispersión del año completo es considerablemente grande y la búsqueda de centros a través de las iteraciones, es compleja, como se muestra en la figura 6.

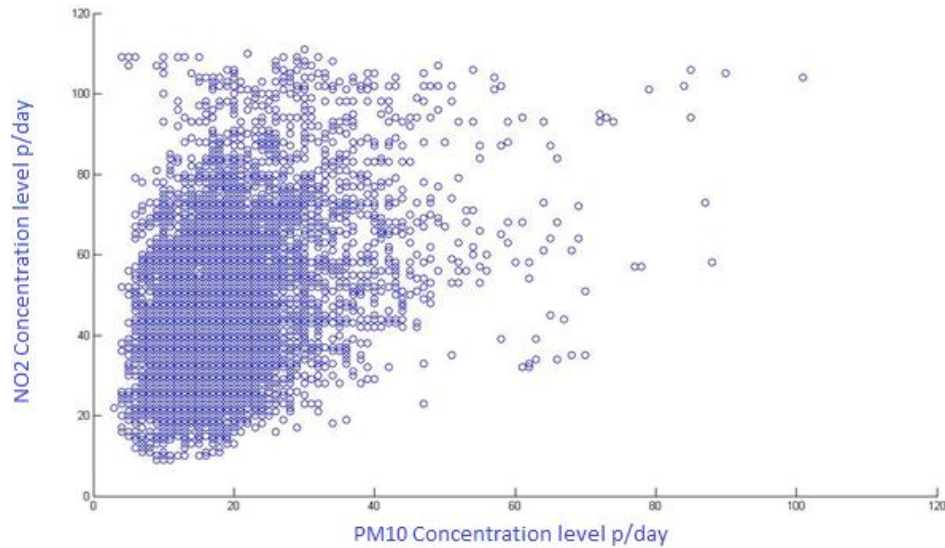


Figura 6. Mapa de dispersión de las partículas medidas por año.

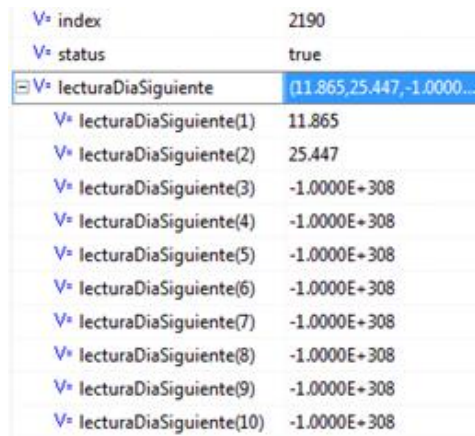
Estas 8760 muestras proporcionan un escenario mayúsculo de información el cual no puede ser analizado en conjunto debido a que los parámetros de contaminación se comportan de manera distinta dependiendo la época del año debido principalmente a los factores de humedad y temperatura. Con esto en mente se procede a dividir los datos en los archivos fuente en cuatro partes que representarían las estaciones del año, en donde se propondría un pronóstico más acertado para condiciones climatológicas similares.

Después de implementar el código VHDL, las iteraciones muestran los datos analizados conjuntados en vectores y el algoritmo va produciendo el pronóstico del más probable entorno de contaminación que se presentará al día siguiente. En la figura 9 se visualizan los vectores de entrada y procesamiento que se obtienen al simular el algoritmo. En la parte superior de la lista de variables se perciben los vectores de membrecías a evaluar que sirven para la ejecución del ciclo, después los valores de centros con los que se hizo la simulación, las lecturas de entrada, las membrecías resultantes en el vector de membrecías y las variables que conforman la operación del sistema para cada ciclo. Estas variables son la iteración del centro, la membrecía evaluada, los dividendos y divisores para el cálculo del nuevo centro, el umbral de cambio y el factor de separación con la distancia del Centro correspondiente a la posible lectura del día siguiente.

⊕ V= membreciasEvaluar	(0.41438,0.18653,0.202...
⊕ V= Centros	((9.9312,12.082,13.865)...
⊕ V= lecturas	(1.0000,20.000,18.000,...
⊕ V= membrecias	(0.61063,0.00000,0.041...
V= iteracionCentro	-2147483648
V= membreciaEvaluada	0.088519
V= dividendoCentro	49782.
V= divisorCentro	2574.4
V= umbralDeCambio	0.00000
V= distanciaCentroLecturaSig...	7.0000
V= centroLecturaDiaSiguiete	2
V= calculoUmbralDeCambio	1
V= calculoDiaSiguieteCentro	1

Figura 7. Vectores de entrada y procesamiento del algoritmo Fuzzy-Clustering

De la misma forma en un vector de resultados se obtiene la lectura más probable para el día siguiente y en este se van acumulando los datos conforme las iteraciones van corriendo. En la figura 8 se muestra este vector que incluye la lectura más próxima para los dos días siguientes:



V ^a index	2190
V ^a status	true
V ^a lecturaDiaSiguiente	(11.865,25.447,-1.0000...
V ^a lecturaDiaSiguiente(1)	11.865
V ^a lecturaDiaSiguiente(2)	25.447
V ^a lecturaDiaSiguiente(3)	-1.0000E+308
V ^a lecturaDiaSiguiente(4)	-1.0000E+308
V ^a lecturaDiaSiguiente(5)	-1.0000E+308
V ^a lecturaDiaSiguiente(6)	-1.0000E+308
V ^a lecturaDiaSiguiente(7)	-1.0000E+308
V ^a lecturaDiaSiguiente(8)	-1.0000E+308
V ^a lecturaDiaSiguiente(9)	-1.0000E+308
V ^a lecturaDiaSiguiente(10)	-1.0000E+308

Figura 8. Vector de salida que incluye los pronósticos del día siguiente.

3. VALIDACIÓN

Con el objetivo de tener la capacidad de valorar la eficiencia y desempeño del algoritmo que se implementa en VHDL, el algoritmo descrito en el capítulo III se implementó en Matlab mediante el cual se debían obtener resultados de MEMBRECIA similares a los que se obtuvieron en las secciones anteriores, ya que se hubieran logrado las similitudes se procederá a la comparación. Para este desarrollo la interface SIMULINK con la que cuenta el software tuvo un papel preponderante puesto que más que utilizar o analizar las ventajas de algún modelo existente era precisamente poder situar el modelo que ya se tiene bajo un esquema diferente, pero el mismo modelo que se tiene en VHDL, como se muestra en la figura 9.

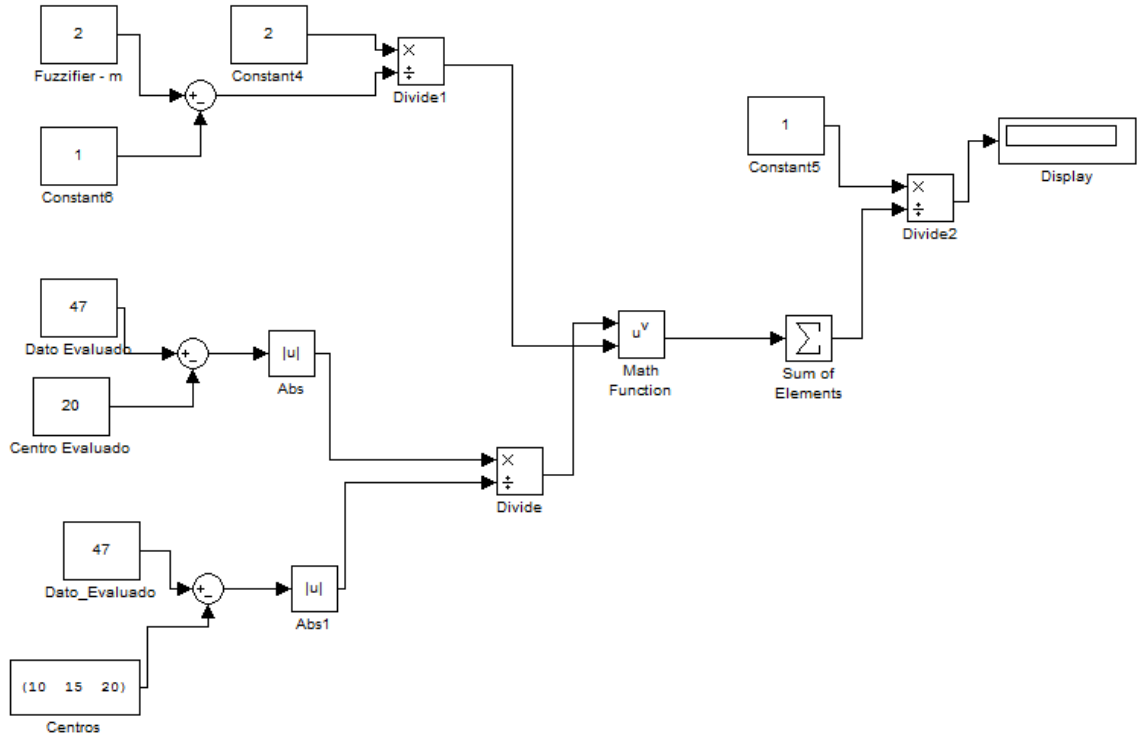


Figura 9. Algoritmo de la ecuación de membresía en Matlab.

Al establecer la simulación con esta ecuación en un valor propuesto de 47, y un centro situado en el valor 20, tal como se muestra en las figuras anteriores, el resultante de la membresía es 0.04455 tal como se muestra en la figura 10.

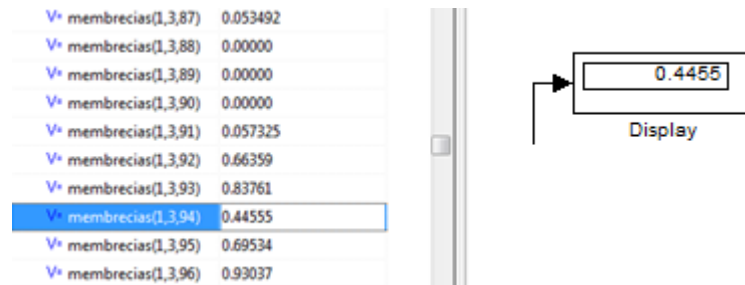


Figura 10. Valor de MEMBRECIA resultante de Fuzzy-Clustering en SIMULINK.

De esta forma, y con ambos resultados del algoritmo se comprueba que los resultados de ambas implementaciones son similares independientemente de la plataforma.

4. CONCLUSIONES

Del presente desarrollo y resultados del trabajo expuesto, se pueden deducir las siguientes conclusiones:

- Es posible implementar sobre una arquitectura VHDL un algoritmo de predicción de partículas contaminantes basado en Lógica Difusa, aún cuando las operaciones sean de programación por procedimientos más orientado hacia aplicaciones de PC o en microcontroladores.

- Las características de desempeño de este algoritmo en comparación con el mismo algoritmo, pero desarrollado bajo una plataforma de más recursos y costos como Matlab son prácticamente iguales, indicando una utilización de CPU de menos del 1% y memoria real ocupada de 96 MB para el enfoque VHDL y 268 MB para la ejecución en Matlab.

- La ocupación en memoria es menor aún cuando en la arquitectura VHDL se simula el vector completo de lecturas.

- No existen errores de memoria en las simulaciones por lo que la ejecución de los algoritmos tanto en VHDL como en Matlab se realiza de forma uniforme.

- Por lo anterior, ha quedado demostrando que se pueden obtener características de desempeño competentes en un proyecto VHDL Fuzzy C-Means de bajo costo, lo cual es sin duda una alternativa preponderante para las ciudades con altos índices de contaminación.

- Queda validado también que no solo la ejecución en un microprocesador pudiera ser la única opción en el campo de las soluciones embebidas incluso aún cuando el algoritmo tenga una estructura de programación por procedimientos.

- La alternativa que se presentó sería una alternativa de pocos recursos y de fácil despliegue hacia varios lugares pudiendo ser manejado o instalado por un residente con un grado aceptable de preparación.

REFERENCIAS

- [1] C. Martínez, Estudio sobre la Importancia de la calidad del Aire, Departamento de Neumología del Hospital Central de Asturias, Prensa Asturiana Media, España, 2009.
- [2] S. Zhao, L. Chen, Y. Li, Z. Xing and K. Du, Summertime Spatial Variations in Atmospheric Particulate M and Its Chemical Components in Different Functional Areas of Xiamen, Key Lab of Global Change and Marine-Atmospheric Chemistry of State Oceanic Administration, Third Institute of Oceanography, State Oceanic Administration, Xiamen, China, *Atmosphere*, 6:234–254, 2015.
- [3] D. G. Karottki, M. Spilak, M. Frederiksen, Z. J. Andersen, A. M. Madsen, M. Ketzal, A. Massling, L. Gunnarsen, P. Moller, and S. Loft, Indoor and Outdoor Exposure to Ultrafine, Fine and Microbiologically Derived Particulate Matter Related to Cardiovascular and Respiratory Effects in a Panel of Elderly Urban Citizens. Section of Environmental Health, Department of Public Health, Faculty of Health and Medical Sciences, University of Copenhagen, Copenhagen, Denmark. 12:1667–1686, 2015.
- [4] Las Palmas de Gran Canaria, Artículo sobre la Calidad del Aire, Introducción al Sistema OPANA V3 de información de la calidad del aire, 2010.
- [5] Ratama-Hernández, A. (2014) Calidad del Aire en la Ciudad México Informe 2014. Dirección de Monitoreo Atmosférico, Gobierno del Distrito Federal, México.
- [6] M. Caselli, L. Trizio, G. De Gennaro and P. Lelpo, A Simple Feedforward Neural Network for the PM10 Forecasting: Comparison with a Radial Basis Function Network and a Multivariate Linear Regression Model, *Water Air Soil Pollution*, 201:365–377, 2009.
- [7] A.B. Chelani and M. Z. Hasan, Forecasting nitrogen dioxide concentration in ambient air using artificial Neural-networks, Air Pollution Control Division, National Environmental Engineering Research Institute (NEERI), Nagpur, India, *International Journal of Environmental Studies*, 58(4):487–499, 2001.
- [8] J.C. Dunn, A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, 32–57, 1973.
- [9] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Kluwer Academic Publishers, Norwell, MA, U.S., 1981.
- [10] Kruse, R.; Dring, C.; Lesot, M.J. Fundamentals of Fuzzy Clustering. In *Advances in Fuzzy Clustering and its Applications*; John Wiley & Sons, Ltd: Hoboken, NJ, US, 2007; pp. 1–30.

- [11] D. M. Saqib Bhatti, N. Saeed, H. Nam, Fuzzy C-Means Clustering and Energy Efficient Cluster Head Selection for Cooperative Sensor Network, Department of Electronics and Communication Engineering, Hanyang University, Ansan 15588, Korea, Sensors, 16, 1459; doi:10.3390/s16091459, 2016.
- [12] A. Balamashc, W. Pedrycza, R. Al-Hmouzc and A. Morfeqc, An expansion of fuzzy information granules through successive refinements of their information content and their use to system modeling, Department of Electrical & Computer Engineering, University of Alberta, Edmonton, Canada, 2014.