Sistemas embebidos: Estado actual con visión al futuro
ISBN 978-607-9394-08-0, Asociación Mexicana de Mecatrónica A.C.

DESIGN OF A FUZZY C-MEANS ALGORITHM FOR AIR POLLUTION PREDICTION BASED ON AN EMBEDDED PLATFORM

Juan M. De la Cruz-Aguirre[1], Marco A. Aceves-Fernández[1], Juan Manuel Ramos-Arreguín[2], José Emilio Vargas-Soto[2]

[1] *Facultad de Informática, Universidad Autónoma de Querétaro, Querétaro, México*

[2] *Facultad de Ingeniería, Universidad Autónoma de Querétaro, Querétaro, México*

juanmca@gmail.com, marco.aceves@gmail.com

***Abstract*— The airborne particles PM10 and PM25 are some of the most dangerous pollutants for the human health due to the minimum aerodynamic diameter that they have. The design of a Fuzzy-C means algorithm for Air Pollution prediction based on VHDL represents a feasible and reliable solution, which is validated comparing its performance with a powerful existent simulation tool.**

I. INTRODUCTION

The action of breathing implies a permanent contact of the respiratory system with the environment. Although this relation is fundamental for life, it makes us also vulnerable to the pollutant materials contained in breathable air. The lungs are the entry open, invisible most of the times, for a large number of substances capable of generating breathing, cardiac, and other organs diseases. Clean air is a shared concern among scientists and institutions [1], and not only the big cities are the target, but also medium scale cities as demonstrated in a researching conducted from 2012 in Xiamen China, which corroborates a progressive and significant increasing of PM particles exposure [2].

Studies as the one mentioned in [3] have been fundamental to determine that air quality is vital for health and welfare; this quality depends on the presence in the atmosphere of pollutants in quantities superior to the allowed levels for the human being. Urban planning is also of paramount importance for air quality also, because mobility and industrial process aspects determine, in conjunction with meteorological conditions, the emission, distribution and the diffusion of atmospheric pollutants. The following are the predominant pollutants: Sulphur dioxide (SO2), nitrogen dioxide (NO2), carbon monoxide (CO), volatile organic compounds (VOC), the total suspended particles (TSP) and plumb [4].

Two of the most dangerous pollutants from the suspended particles are the PM10 and PM2.5. They have a featured aerodynamic diameter below 10 µm and 2.5 µm respectively, and they are considered riskier than other ones with bigger diameters, mainly the 2.5 particles; because of the less diameter the most capable to penetrate deeper the respiratory tract conducting to the region of air interchange, transgressing more through other organs. Additionally, they contain a greater quantity of toxins due to the compounds forming them.

The effects these particles have in the human health varies from minor symptoms like nose and throat irritations to the most severe consequences in the respiratory system, cardiovascular diseases, and even premature death, increasing the probability in the elderly as described in [3]. The key aspects contributing this pollution are the heavy industry activity, the over-pollution, and city traffic. Besides these factors, other secondary sources as the contaminant gases through the gas-to-particle conversion, and the re-suspension of particles due to human or earth movements, also serve as a source of PM2.5 which contains several metal elements. Other pollution generators are the infiltration of particles due to the combustion like the automotive smog and the waste incineration. The permissible levels are 15.0 µg/m3 (micrograms per cubic meter) although in some cases these can reach 35 µg/m3 and 50 µg/m3 in the more contaminated cities. According to the Official Mexican Law (NOM for its Spanish acronym), NOM-025-SSA1-2014 [5], the acceptable levels of PM10 particles is 75 µg/m3 for a 24 hours average, and 40 µg/m3 annual average limit. For the PM2.5 particles, this law specifies 45 µg/m3 for a 24 hours average and 12 µg/m3 annual average limit.

Therefore, air pollution controls to prevent worst measurements in the long and the short terms are needed. Several research works have been performed, modeling the space-time to prevent hourly concentrations in USA cities in [6]. The current pollutant particles prediction scheme proposes a clustering based algorithm according to the Fuzzy C-Means (FCM) technique used in [7].

In traditional clustering or other modeling systems such as [8], [9] or [10], one entity belongs to one single cluster. In the FCM technique, it is allowed for one item to belong to several clusters based on their location on the histogram and with different degrees of membership. This versatility

provides major certainty to the pollutant particles prediction.

II. METHODOLOGY

A. The Fuzzy-Clustering Algorithm

This algorithm implements clustering method which allows one data fragment to be part of one or more clusters. Developed by [11] and improved by [12] in 1981, is frequently used in patterns recognition by assigning membership to each data point corresponding to each cluster center, where the summation of membership of all data points should be equal to one. The algorithm is based on the minimization of the following objective function:

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

Where C is the number of clusters, m is the fuzziness exponent expressed by a real number greater than or equal to 1, u_{ij} is the degree of membership of x_i in the cluster j, x_i is the i-th part of the measured vector data, c_j is the center of the cluster, and ||*|| is any rule expressing similarity between any data measured and the center of the cluster. Therefore, the value u_{ij} lies between 0 and 1 for every parameter in the network to each cluster center.

The fuzzy partitioning is carried out through iterative optimization of the objective function membership u_{ij} and the updated cluster center c_j as:

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

and:

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

The iterative optimization stops when the termination criteria ∂ is met, i.e.:

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

Where ∂ is the termination criteria between 0 and 1 and k is the iteration step. After cluster formation, the network is split into clusters and the CH selection process is started locally within each cluster.

B. Fuzzy-Clustering Model

The objective of the algorithm to be modeled is to predict the pollutant particles contained in the environment air, based on a Fuzzy Clustering VHDL scheme. This model starts with a real set of data divided into groups (clusters). Every one of these clusters has a center value (called centroid), and every element of the group has a specific degree of membership with the center of each group.

The first step of the model to be implemented consist in the definition of an initial threshold which will be used to measure the proximity of our model with the real data obtained, this limit will set the maximum difference that may exist between the predicted values with respect to the real captured values. This initial threshold is fundamental because the algorithm's certainty to provide the expected objective values will depend on it. Once the threshold is selected, it is required to establish the first group of centers, assigning randomly center values for each group. The chosen values should be no greater than the maximum expected value and not lesser than the minimum possible captured value (they can be assigned from four average values from the whole expected group). The first group of centers of real data is analyzed, and the first degree of membership is calculated; then the software implementation in VHDL as shown in equation 2 is processed with each of the assigned centers as an input.

When the previous step is completed, and membership data is calculated for every cluster, a new group of centers is calculated with a VHDL algorithm. Then, the separation between real data with respect to the new centers is measured. The purpose is to identify if this value is inside the threshold of the algorithm and if it is not, to re-calculate in other iteration; this sequence continues until an accurate model is obtained (inside the expected threshold). If the model cannot be adapted, then the initial threshold should be changed to adjust the algorithm.

With this model including new centers and the previously calculated memberships, new data for every group is obtained, and this is the forecasted pollution agents values. The complete algorithm is shown in figure 1.

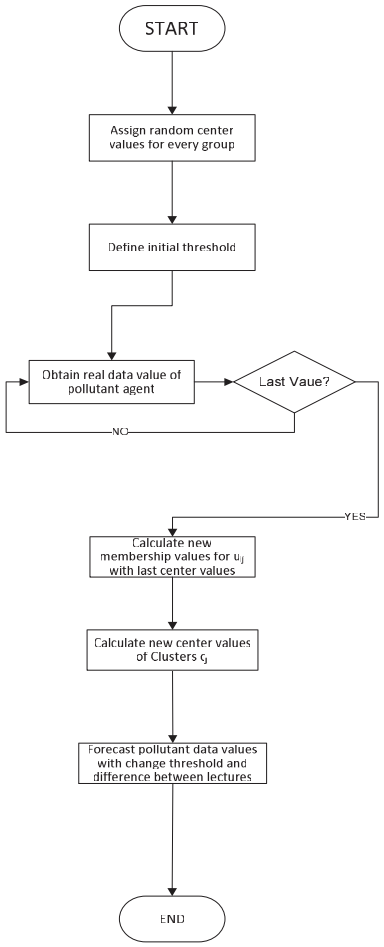

Fig. 1. Algorithm to calculate Fuzzy-C pollutant agents.

III. IMPLEMENTATION

A. System Definition

Global system forecasting pollutant agents have three main steps in which the complete functionality resides: to obtain the memberships of the current group, to calculate the center values for the prediction and to forecast the following day values.

This process can be defined by the following Top-Down description:

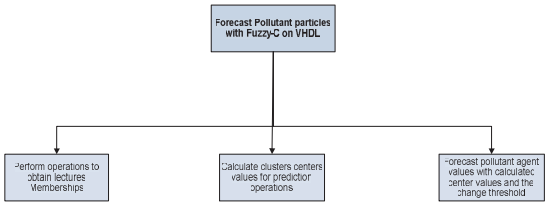

Fig. 2. Top-Down diagram of the system.

These are the main functions to be integrated into VHDL code to accomplish the prediction objective.

B. MAIN program

The proposed algorithm is executed in the MAIN program where all the functions are called. The VHDL architecture which defines the structure of the MAIN program is visualized in figure 3:

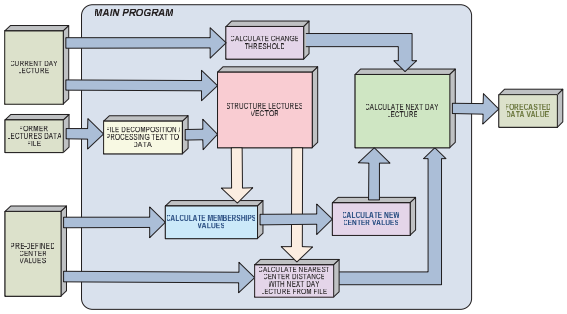

Fig. 3. Main program to calculate pollutant agents through a Fuzzy-C algorithm.

These are the main functions to be integrated into the code; their function and interaction are covered in the next sections.

C. Initial Center Values and Reading Lectures data

First step on the MAIN program is to assign preliminary values to the centers, they can be defined as explained in sub-section B from the previous chapter, from available statistical analyses, or creating a dispersion chart from the whole lectures list. The center values of the groups where more concentration of particles is perceived are selected as the initial values. After this step, the inner loop of the MAIN program is started, internal variables are initialized, and the previous lecture values are stored in a vector to manipulate them in the algorithm. The particles concentration value measured on the last available day is also saved in a variable.

D. Threshold Change variable calculation and Membership calculation

The stored lectures from previous steps are now used to calculate the Threshold Change variable; this parameter is defined as the distance from the stored read value with respect to the one that is recently captured. The first step is to identify whether the evaluated read value is greater or smaller than the stored data; then, the difference between both (stored and current values) is obtained and this value is stored as the Threshold Change variable. This process is performed to the entire vector of acquired values, and if in one case the computed difference is minor than the one stored, the new value is saved in the new Threshold Change variable replacing the previous one. This stored value is compared with the entire read values vector to obtain the minor difference to be used in the algorithm. A flag is saved also to indicate if this value should be added or decreased from the lecture on which this value will be applied.

Once this sequence is finished, the full day of values is computed and replaced with a new one using the

Threshold Change variable, the following diagram is the Top-Down figure describing this step.

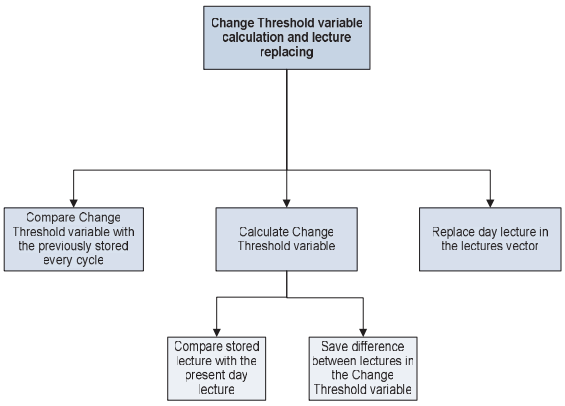

Fig. 4. Top-Down diagram for Threshold Change variable calculation.

This new group of lectures will be the input to calculate a new group of Memberships. This computation is performed with the center value for every particle using the formulas from section II.

The representation below describes how this is implemented. For every read value, there would be a degree of membership corresponding to each evaluated center.

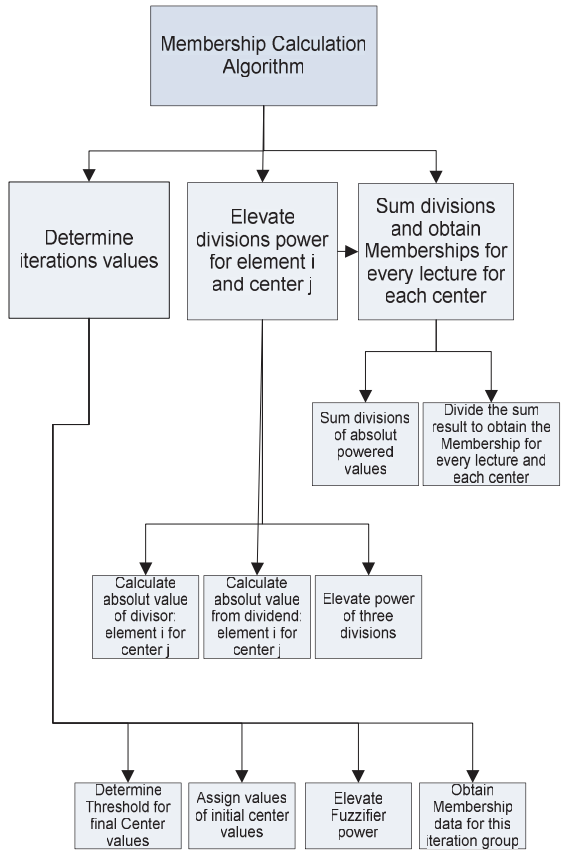

Fig. 5. Top-Down Diagram for Membership Algorithm

E. New center values calculation and Near center definition

The definition of the near center to each following-day lecture is the next step. The next-day data acquisition is obtained by taken the previous next-day value but now adding or subtracting the Threshold change value. During this process, every center is compared against the calculated next-day lecture. When all the centers are processed, the nearest center is flagged, and also another parameter is stored to indicate if this distance should be added or subtracted.

At this point, all the operations needed with the current center values have been completed and a new set should be computed with the last obtained membership values; this new centers will settle the basis for the forecasting of the pollutant elements of the next-day. The first step is to compute for each acquired data of the new group, its corresponding membership degree, by adding-up the membership of each value powered to the Fuzzifier value, and to multiply this by the evaluated acquired data as described in previous sections, figure 6 shows the Top-Down description for this sequence.

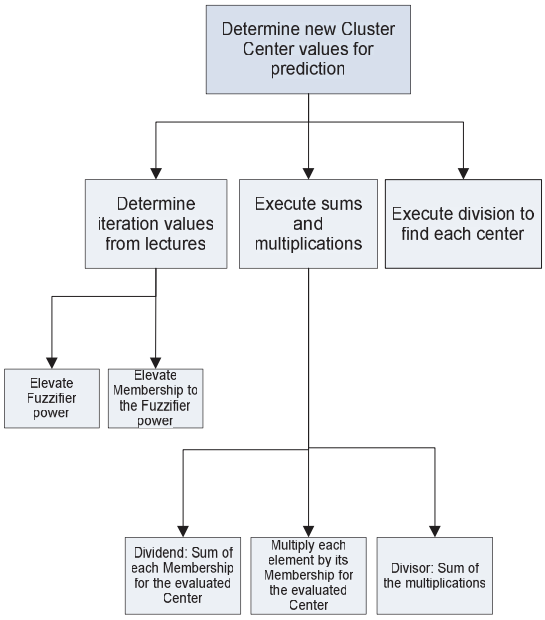

Fig. 6. Top-Down Diagram for New Centers calculations

F. Next-day lecture forecast

The membership values obtained in the first steps of the algorithm and the new center values calculated in previous sections will allow us to predict the next-day value. The Threshold Change variable flag is used, to indicate if this should be added or subtracted, and it is also used in conjunction with the flag to indicate if the nearest center distance should also be added or subtracted. These two values are applied to the stored next-day lecture for the corresponding center cluster to which this value belongs. After those operations, the forecasted next-day value is obtained, and the image below shows how this process should be performed.

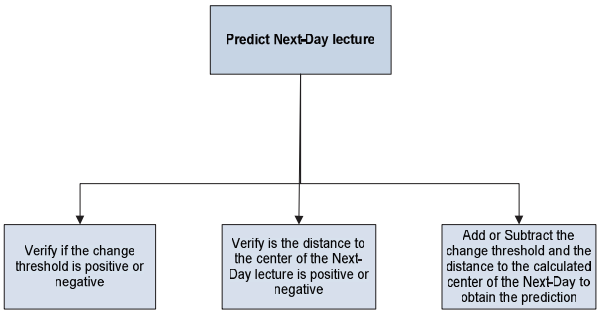

Fig. 7. Top-Down Diagram to predict the following-day lecture

During the next section, another platform (Matlab) will be used to validate the results obtained with the VHDL implementation.

IV. SIMULATION AND VALIDATION

A. VHDL System simulation

A pollutant particles forecast system should be capable of interacting with cities where this is a serious pollution concern. Therefore, for the simulation and verification stages, real data is used. Previously stored data is evaluated and divided into files of 8760 samples each, which represent one year of stored information of Nitrogen Dioxide NO2 particles, and 10 nanometers of diameter PM10 particles. In order to achieve similar conditions for weather and humidity, the data is divided into four parts corresponding to every season of the year, in this way there are equal conditions for every data file being used for forecasting.

When the application is started, every iteration presents the analyzed data in vector groups, and the algorithm produces the forecasting of the most probable pollutant data that will be measured on the Next-day. The following figure shows the input vectors and the processed data when the simulation is started. In the top part of the variables list, the membership vectors which are used for the cycle execution are visualized. Also, the center values utilized for the simulation, the initial input lectures, the resultant membership values and the operation system variables for every cycle are displayed.

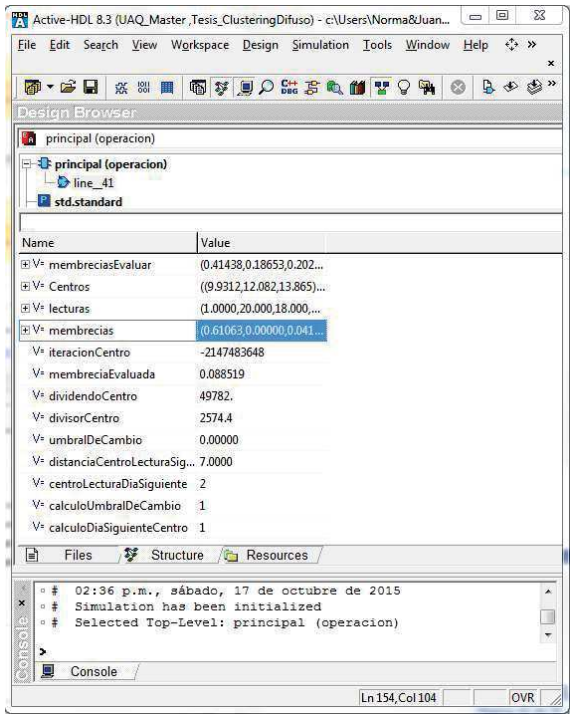

Fig. 8. Input vectors and processing of Fuzzy-Clustering algorithm on VHDL

A vector containing the Next-day forecasting results is obtained. As the iterations are executed the application continues showing the acquired values for the Next-day readings. Figure 9 shows this step:

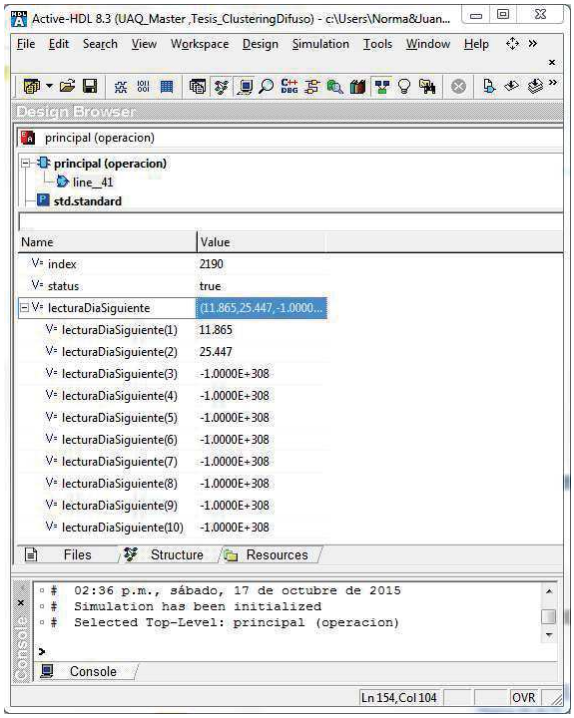

Fig. 9. Output vector including following-day prediction

B. Performance Validation

To validate the reliability of the algorithm execution performance under a VHDL implementation, the Membership equation described in Section II was developed in MATLAB.

Then, following a modular strategy, the formula is divided and then integrated into one single operation. In figure 10, the first part of this process is shown, which corresponds to the description of the divisor exponent *2/(m-1)*:

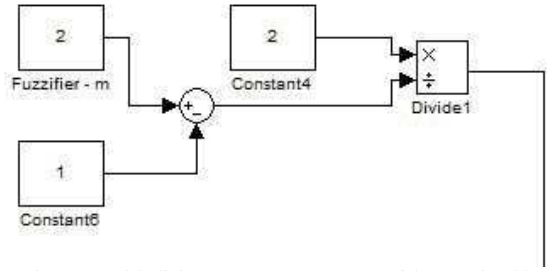

Fig. 10. Matlab divisor exponent component of the Membership equation

The next step is to develop the divisor operation element, and later to elevate its result to the power exponent obtained in the previous step, as described in figure 11:

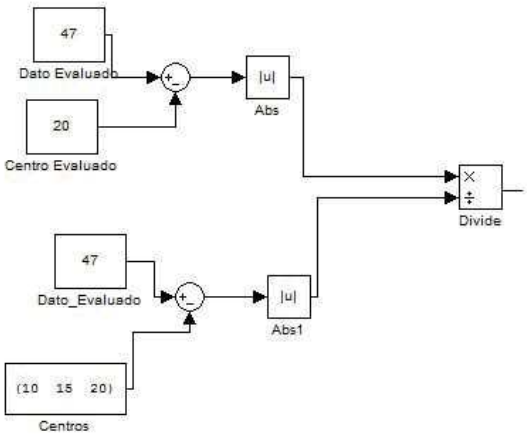

Fig. 11. Matlab divisor component of the Membership equation

For the center values case, the evaluated variables are input to the algorithm, and they are accumulated in one cluster of data, which is providing simultaneous predictions on each iteration.

Once this value is obtained it is powered to the value defined in the previous step, and every data is added to the final sum of elements, as shown in figure 12:

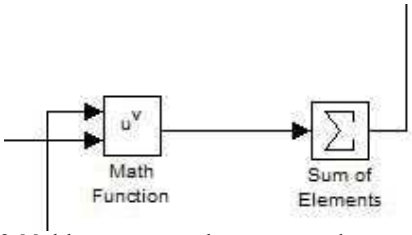

Fig. 12. Matlab component to elevate power and to sum resultant elements of Membership equation

With the divisor ready and elevated to the corresponding power, the inverse value of it is calculated to obtain the resultant membership; the corresponding image is below. The “Display” element allows to visualize the result:

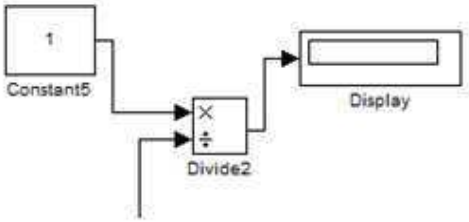

Fig. 13. Matlab Component to obtain the result of the Membership equation

The complete Matlab development is integrated into the following scheme:

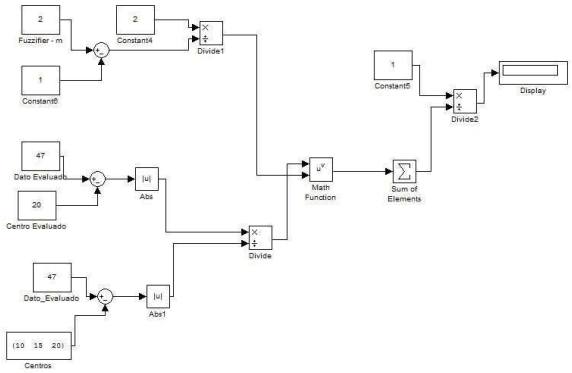

Fig. 14. Matlab algorithm of the Membership equation

For comparison purposes, a simulated data group of 47 points of one pollutant particle is tested with a center of 20, the sequence is run, and the Membership result value is 0.04455 which is visualized in the "Display" window:

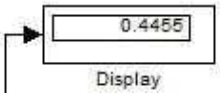

Fig. 15. Matlab Resultant Membership value of the Membership equation

Similar input values were provided to the VHDL simulation on the position 94 of the lectures vector, and in the same location could be verified on the simulation results group as shown in the image below:

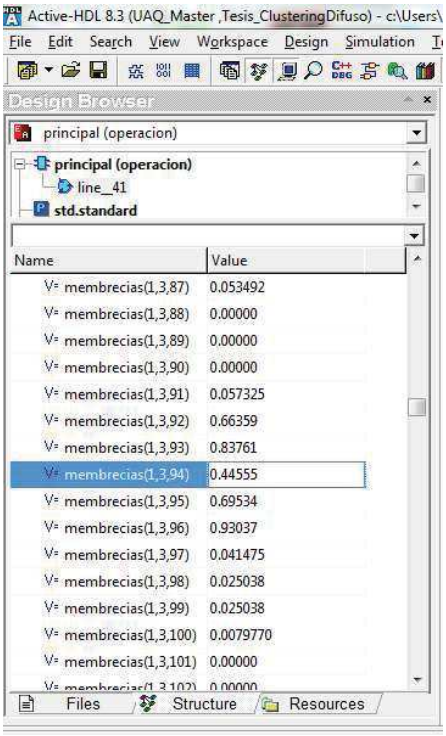

Fig. 16. VHDL results for the same input values exercised in Matlab

The similar results in both developments demonstrate that a VHDL implementation to predict pollutant agents is as reliable and capable as if it would be implemented under a powerful platform, just as Matlab.

V. Conclusions

It is not only the data acquisition what is important but also the way in which this data is processed to add value on a prediction system. On this sense, the presented development corroborates that a Fuzzy-Clustering algorithm can be implemented in VHDL for prediction objectives, just as the forecasting of pollutant agent particles.

Another deducted fact is that, even though the transition of the operations is focused on a procedural scheme, more adaptable to an embedded microcontroller based application, it can also be adjusted and worked in a FPGA-based architecture, just as the modularized algorithm described in Section II.

Therefore, a VHDL based option as an embedded solution could be considered for some specific embedded projects as a possible alternative.

Also, it was corroborated that the reliability of the results provided by a Fuzzy-Clustering algorithm realized in a VHDL implementation is the same as one developed on a platform with more resources and processing power (e.g., Matlab).

Similarly, with the same obtained results, it is defined that a prediction system is trustworthy in a solution of this type. This represents a considerable advantage over cities with limited resources, needing to predict these particles because pollution is severely affecting them.

Towards a future development following the same line of this project, the points mentioned below are possible options to continue the present work:

- For a hardware implementation, the algorithm could be tested in a DSP and for comparison purposes also in a MCU. This will allow to determine if there is any throughput, the Internal memory needed, the Pin footprint and the required size of the device.
- The algorithm can be optimized through the Possibilistic C-Means technique; this would allow managing data in case they were very dispersed that Fuzzy C-Means method was capturing them as noise, this would add certainty to the model to have more emphasis in the pollutant particles prediction.
- Improving the algorithm requires the inclusion of factors like humidity, weather, or rain to support more precise results in the forecasting.
- The usage of real data will provide the capability for a more accurate prediction, implying building a circuit able to sense the pollutant agents in the same array in real time, including an error handling and retry strategies, and creating a software component to store and use the captured samples as part of the developed algorithm.
- To have a matching verification working in parallel, a forecast system based on a Neural Network strategy would complement the prediction. In this case, additional hardware will be required depending on the decided architecture.
- To implement a similar algorithm, with different centers, for example, a procedural strategy using a microcontroller array, several escalated FPGAs, or to equip the chip to be handled by a PC, could be options to increase the capabilities of the prediction system.

The cases mentioned above represent benefit opportunities to continue the present work. Those efforts could be highlighted by presenting schematic pipelines of the implementations and also simulations showing the waveforms of calculations.

References

[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. Ketzel, 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] Official Mexican Law for Environmental Healthy, Permissible values for suspended particles PM10 and PM2.5 concentrations in the environment and their evaluation criteria, Healthy Secretary, Federal Government of Mexico, NOM-025-SSA1, 2014.

[6] S. Batterman, R. Ganguly and P. Harbin, *High Resolution Spatial and Temporal Mapping of Traffic-Related Air Pollutants*, Environmental Health Sciences, School of Public Health, University of Michigan, Ann Arbor, MI, U.S., 12:3646–3666, 2015.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] J.C. Dunn, *A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters*, 32–57, 1973.

[12] J.C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Kluwer Academic Publishers, Norwell, MA, U.S., 1981.