Dynamic Fuzzy Cognitive Maps Embedded and Intelligent Controllers Applied in Industrial Mixer Process

Lucas Botoni de Souza
Patrick Prieto Soares
Ruan Victor Pelloso Duarte Barros
Márcio Mendonça
DAELE (Electric Academic Department)
UTFPR-CP
Cornélio Procópio, Brazil
{lucasbotoni; p.prietosoares}@hotmail.com
ruan_pelloso@yahoo.com.br
mendonca@utfpr.edu.br

Elpiniki I. Papageorgiou
Department of Computer Engineering
Technological Education Institute/ University of Applied Sciences of Central Greece
Lamia, Greece
epapageorgiou@teiste.gr

***Abstract*— This paper presents the application of certain intelligent techniques to control an industrial mixer. The controller design is based on a Hebbian modification of the Fuzzy Cognitive Maps learning mechanism. This research develops a Dynamic Fuzzy Cognitive Map (DFCM) based on Hebbian Learning algorithms. Fuzzy Classic Controller was used to help validate simulation results of an industrial mixer controlled by DFCM. Experimental analysis of simulations in this control problem was conducted. Additionally, the results were embedded using efficient algorithms into the Arduino platform to acknowledge the performance of the codes reported in this paper.**

Keywords-Fuzzy Cognitive Maps; Hebbian Learning; Arduino Microcontroller; Process Control; Fuzzy Logic; Artificial Neural Network.

I. INTRODUCTION

This work is an evolution of the article shown in [1]. In general, some of the difficulties found in acquiring knowledge in different areas of engineering (such as robotics, control or process control) are: how to recognize the processes /systems; how to identify important variables and parameters; to classify the type of physical problem; o identify the family of mathematical models that can be associated; to select the method and / or tool for the search and analysis of the model.

Indeed, the final output of modern processes is significantly influenced by the selection of the set points of the process variables, as they fundamentally impact the product quality characteristics and the process performance metrics [2]. In this context, it is possible to define the main goal of this research, to develop techniques based on knowledge for the process control of a classic problem of Fuzzy Cognitive Maps area, an industrial mixer; this work is an evolution of the previous work [3].

The article proposal is to use a different setup, in special the initial state and a comparison with a new controller using Fuzzy-Logic with ANN (artificial neural network). The motivation of this research is: developments in optimal control theory, robust control and adaptive control, expanding significantly the automation concept and, also studying the feasibility of an autonomous control in practice.

On the other hand, intelligent control techniques take control actions without depending on a complete or partial mathematical model. Otherwise, the ability of a human to find solutions to a particular problem is known as human intelligence. In short, human beings can deal with complicated processes based on inaccurate and/or approximate information. The strategy adopted by them is also of imprecise nature and usually capable of being expressed in linguistic terms. Thus, by means of Fuzzy Logic concepts, it is possible to model this type of information [4].

Some previous works that used Fuzzy techniques can be cited, such as [5], which applies a Fuzzy-Neuro predictive control tuned by Genetic Algorithms (GA) on a fermentation process. A Proportional Derivative Fuzzy Logic Controller (Fuzzy-PD) was initially used to control the process, a non-linear system with non-minimal phase, and a large accommodation time.

More recently, [6] presented a FCM used to tune PI controllers' parameters used on a non-linear system. These controllers cannot achieve satisfactory results in this type of system, by the difference of their static and dynamic properties.

There is also [7], where new types of concept and relation, not restricted to cause-effect ones, are added to the model resulting in a dynamic fuzzy cognitive map (DFCM). In this sense, a supervisory system is developed in order to control the fermentation process.

II. FUZZY COGNITIVE MAPS – BACKGROUND

Fuzzy Cognitive Maps (FCM) was introduced by Kosko's work, which added Fuzzy values to the causal relationships of Axelrod's Cognitive Maps paper. In fact, FCMs are system models represented in a graph-form, the nodes are the concepts related to the problem and the lines connecting them are the causal relationships. A FCM is a 4-tuple, as described in works as [8] and [9]. It is commonly used to study system's dynamics because of its mathematically simplicity. The relationship's influence is calculated using normalized states and matrix multiplications.

The inference of the system's dynamics might reach a steady state, a limit cycle of states or even a chaotic state [10-11]. Every concept's activation level is based on its own previous iteration and the propagated weighted values of all the concepts connected to it (it means all concepts that have influence over it).

In the literature, there are many examples of FCMs that use monotonic and symmetric weight cause-effect relationships between the concepts that might work on controlled environments but cannot be applied on the real world considering its dynamic aspects. In order to bring FCMs to a more realistic environments, there are a few techniques that can be used such as using Fuzzy rules and feedback mechanisms [12-13] or algebraic equations to define the causal relationships when the real system have been modeled by crisp relations [14].

In general, a Fuzzy Cognitive Map (FCM) is a tool for modeling the human knowledge. It can be obtained through linguistic terms, inherent to Fuzzy Systems, but with a structure like the Artificial Neural Networks (ANN), which facilitates data processing, and has capabilities for training and adaptation. FCM is a technique based on the knowledge that inherits characteristics of Cognitive Maps and Artificial Neural Networks [10-15], with applications in different areas of knowledge [16-17].

Besides the advantages and characteristics inherited from these primary techniques, FCM was originally proposed as a tool to build models or cognitive maps in various fields of knowledge. It makes the tool easier to abstract the information necessary for modeling complex systems, which are similar in the construction to the human reasoning.

Dynamic Fuzzy Cognitive Maps (DFCM) needs to be developed to a model that can manage behaviors of non-linear time-dependent systems and sometimes in real time. Examples of different variation of the classic FCMs can be found in the recent literature, e.g., [18-19].

This paper has two objectives. The first objective is the development of two controllers using an acyclic DFCM with same knowledge of a Fuzzy and Fuzzy Neural controller, and with similar heuristic, thus producing comparable simulated results. The second goal is to show an embedded DFCM in the low-cost processing microcontroller Arduino with more noise and disturbances (valve locking) to test the adaptability of the DFCM.

To succeed the goals, we initially use the similar DFCM proposed initially in [20] to control an industrial mixing tank. In contrary to [20], it is used the Hebbian algorithm to dynamically adapt the DFCM weights. In order to validate the DFCM controller, its performance was compared with a Fuzzy Logic controller. This comparison is carried out with simulated data.

III. DEVELOPMENT

To demonstrate the evolution of the proposed technique (DFCM) we will use a case study well known in the literature as seen in [3-21] and others. This case was selected to illustrate the need for refinement of a model based on FCM built exclusively with knowledge.

The process shown in Fig. 1 consists of a tank with two inlet valves for different liquids, a mixer, an outlet valve for removal of the final product and a specific gravity meter that measures the specific gravity of the produced liquid. In this research, to illustrate and exemplify the operation of the industrial mixer, the liquids are water with specific gravity 1 and soybean oil with a specific gravity of about 0.9.

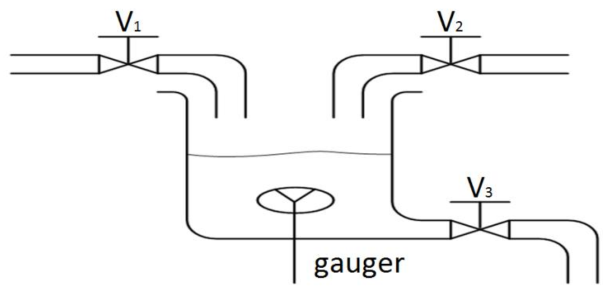

Figure 1. Mixer tank (Source: adapted from [21]).

Valves (V1) and (V2) insert two different liquids (specific gravities) in the tank. During the reaction of the two liquids, a new liquid characterized by its new specific gravity value is produced. At this time, the valve (V3) empties the tank in accordance with a campaign output flow, but the liquid mixture should match the specified levels of the volume and specific gravity.

Although being relatively simple, this process is a TITO (Two Inputs and Two Outputs) type with coupled variables. To establish the quality of the control system of the produced fluid, a weighting machine placed in the tank measures the specific gravity of the liquid produced.

When the value of the measured variable G, liquid mass, reaches the range of values between the maximum and minimum [Gmin, Gmax] specified, the desired mixed liquid is ready. The removal of liquid is only possible when the volume (V) is in a specified range between the values [Vmin and Vmax]. The control consists to keep these two variables in their operating ranges, as:

$$V_{min} < V < V_{max} \tag{1}$$

$$G_{min} < G < G_{max} \tag{2}$$

In this study, it was tried to limit these values from approximately the range of 810 to 850 [mg] for the mass and approximately the range of 840 to 880 [ml] for the volume. The initial values for mass and volume are 800mg and 850ml, respectively. According to Papageorgiou and collaborators [23], through the observation and analysis of the process, it is possible for experts to define a list of key concepts related to physical quantities involved. The concepts and cognitive model are:

- Concept 1 - State of the valve 1 (closed, open or partially open);
- Concept 2 - State of the valve 2 (closed, open or partially open);

- Concept 3 - State of the valve 3 (closed, open or partially open);
- Concept 4 - quantity of mixture (volume) in the tank, which depends on the operational state of the valves V1, V2 and V3.
- Concept 5 - value measured by the G sensor for the specific gravity of the liquid.

Considering the initial proposed evolution for FCM, it is used a DFCM to control the mixer, which should maintain levels of volume and mass within specified limits.

The process model uses the mass conservation principle in incompressible fluid to derive a set of differential equations representing the process used to test the DFCM controller. As a result, the tank volume is the volume over the initial input flow of the inlet valves V1 and V2 minus the outflow valve V3, this valve V3 and the output campaign was introduced in this work to increase the original process' complexity [22].

Similarly, the mass of the tank follows the same principle as shown below. The values used for m_{e1} and m_{e2} were 1.0 and 0.9, respectively.

$$V_{tank} = V_i + V_1 + V_2 - V_3 \quad (3)$$

$$Weight_{tank} = M_i + (V_1.m_{e1}) + (V_2.m_{e2}) - M_{out} \quad (4)$$

IV. Fuzzy Controller Development

To establish a correlation and a future comparison between techniques, a Fuzzy controller was also developed. The Fuzzy rules base uses the same heuristic control strategy and conditions.

Fuzzy logic has proved being able to provide satisfactory non-linear controllers even when only the nominal plant model is available, or when plant parameters are not known with precision [24-25]. Fuzzy Control is a technique used for decades, especially in process controlling [21].

It is a motivation to validate DFCM, so in this study it was used the same approach for two controllers, with two different formalisms. It is not in the scope to discuss the development of the Fuzzy controller, but some details of the structure are pertinent: functions are triangles and trapezoidal and 6 rules are considered in its base. The Fuzzy controller surfaces are shown in Fig. 2. Moreover, the rules are symmetric and similar by two output valves; in this specific case, the surface of valve 1 is the same as in valve 2. The base rules and its respective weights are:

1. If (Level is low) then (V1 is medium) (V2 is medium)(1);
2. If (Level is medium) then (V1 is low) (V2 is low) (1);
3. If (Level is high) then (V1 is low) (V2 is low) (1);
4. If (Weight is low) then (V1 is high) (V2 is high) (1);
5. If (Weight is medium) then (V1 is low) (V2 is low) (0.5);
6. If (Weight is high) then (V1 is low) (V2 is low) (1);
7. If (ValveOut is high) then (V1 is high) (V2 is high) (0.5);
8. If (ValveOut is medium) then (V1 is medium) (V2 is medium) (0.5);
9. If (ValveOut is low) then (V1 is low) (V2 is low) (0.5).

The rules and structure of the Fuzzy Controller used on its development was based on the DFCM heuristic.

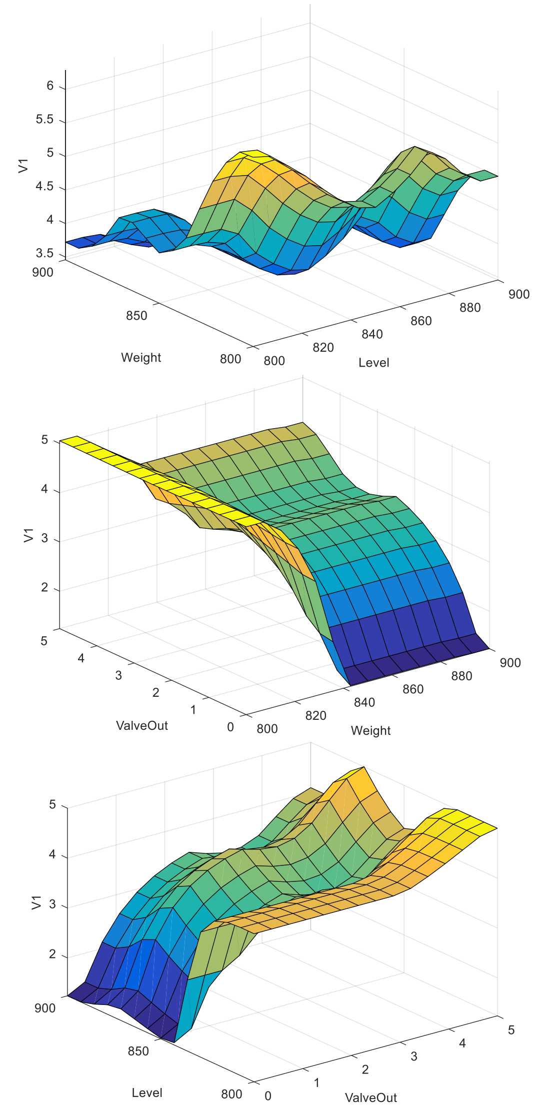

Figure 2. Fuzzy Controller Surfaces for V1 and V2

Fig. 3 shows the Fuzzy structure with same variables input and output like DFCM.

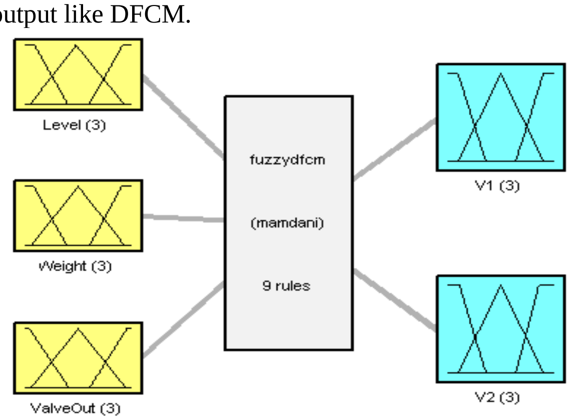

Figure 3. Fuzzy Structure

This model represents the weakest degree of possible integration between two techniques and the consistency of two subsystems connected in series. As an example, we can cite a neuro-Fuzzy model which a Fuzzy system admits inputs to a neural network as shown in Fig. 4.

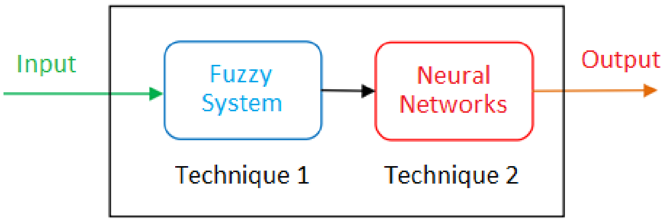

Figure 4. Sequential hybrid model

A Fuzzy-ANN cascade controller had its ANN (multilayer perceptron) trained with the output data of the Fuzzy controller. The topology was empirically chosen by observing the learning time and output error. Therefore, 200 neurons were used on its hidden layer. Moreover, there were used 6000 points from inside the control region. The results of the Fuzzy-ANN controller are shown in Figs. 21-24.

V. DFCM DEVELOPMENT

The structure of the DFCM controller is similar to the developed Fuzzy controller, using same heuristics, e.g., if the output valve (V3, in accordance to Fig. 1) increases its flow, the inlet valves (V1 and V2) increase too. On the other hand, in case volume and weight of the mixture increase, the inlet valves decrease. For example, the relationships W54 and W53, in the DFCM, are similar in effects or control actions of the Fuzzy controller's base rules.

The development of the DFCM is made through three distinct stages. First, the DFCM is developed as structure, concepts and causal relationships, similar to a classic FCM, where concepts and causal relationships are identified through sensors and actuators of the process. The concepts can be variables and/or control actions, as already mentioned.

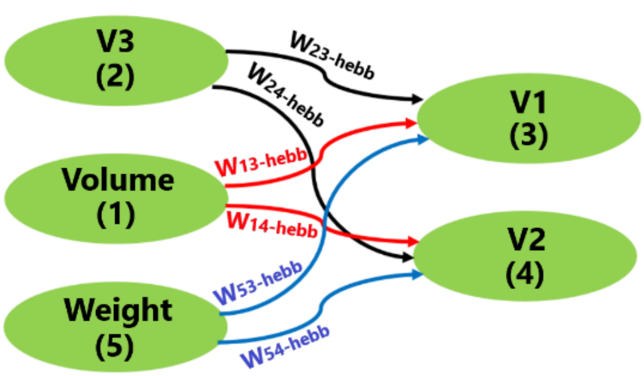

Figure 5. DFCM Controller

The output valve is defined by a positive relationship, i.e., when the campaign increases, the output flow (V3) also increases, similarly, the input valves increase too; moreover, when the mixture volume and weight increase, V1 and V2 decrease. In both cases, the flow of the valves increases or decreases proportionally. The second development stage is the well-known GA [26]. Fig. 5 shows the schematic graph of a DFCM controller.

In this research, the initial values of causal relationships are determined through offline Genetic Algorithms. The GA used is a conventional one, with a population of 30 individuals, simple crossing and approximately 1% of mutation. The chromosomes were generated by real numbers with all the DFCM weights, individuals were random and the initial method of classification was the tournament method with 3 individuals.

Finally, the fitness function, for simplicity, considers the overall error of the two desired outputs with 15 generations of the proposed GA. It stabilizes and reaches the initial solution for the opening of the valves, approximately 44% (V1) and 42% (V2), as shown in Fig. 6. In short, some of the GA parameters used in this work are:

- Recombination method: single-point crossover;
- Mutation method: randomly chosen;
- Selection method: tournament;
- Initial causal relationships: randomly chosen nearby expected values;
- Fitness function *E(i)*, given by (5):

$$Ei=\{0.44-A3k+12+0.42-A4k+12\}0.5 \quad (5)$$

- Probability of recombination: 1;
- Initial population size: 30 chromosomes.

Table I shows the initial values of the DFCM weights. Different proposals and variations of this method applied in tuning FCM can be found [26]. Fig. 6 shows the initial causal relationships' evolution by GA optimizing valve locking.

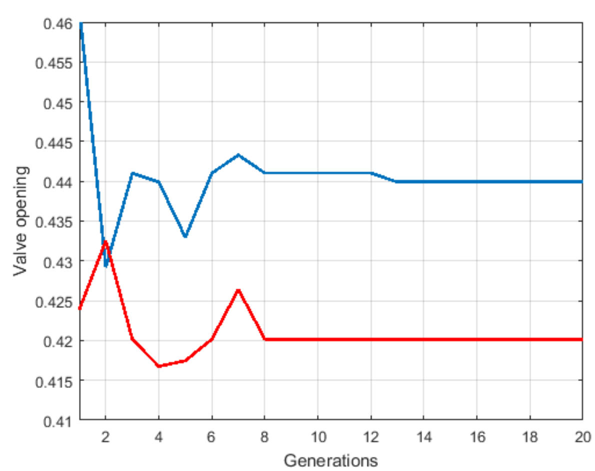

Figure 6. Initial weight's evolution by GA

TABLE I. INITIAL CAUSAL RELATIONSHIP WEIGHTS

W13	*W14*	*W23*	*W24*	*W53*	*W54*
-0.2647	-0.324	-0.2831	-0.3339	0.2648	0.2754

The third stage of the DFCM development concerns the tuning or refinement of the model for dynamic response of the controller. In this case, when a change of output set point in the campaign occurs, the weights of the causal relationships are dynamically tuned. To perform this function, a new kind of concept and relation was included in the cognitive model.

To dynamically adapt the DFCM weights it was used the Hebbian learning algorithm for FCM, which is an adaptation of the classic Hebbian method [10]. Different proposals and variations of this method applied in tuning or in learning for FCM are known in the literature, for example, [28]. In this paper, the method is used to update the intensity of causal relationships in a deterministic way according to the variation or error in the intensity of the concept or input process variable; equations (5) and (6) show this.

Specifically, the application of the Hebbian learning algorithm provides online control actions as follows: if the weight or volume of the liquid mixture increases, the inlet valves have a causal relationship negatively intensified and tend to close quicker. On the other hand, if the volume or weight mixture decreases, the inlet valves have a causal relationship positively intensified. The mathematical equation is presented in (6).

$$W_i(k) = W_{ij}(k-1) \pm \gamma \Delta A_i \qquad (6)$$

Where: ΔAi is the concept variation resulting from causal relationship, and it is given by ΔAi = Ai (k)-Ai (k-1), γ is the learning rate at iteration k.

This version of the Hebbian algorithm is an evolution of the two proposals of Matsumoto and collaborators [28].

Causal relationships with negative causality have negative sign and similarly to positive causal relationships. The equations applied in this work are adapted of the original version (7).

$$W_i(k) = k_p \,.\, (W_{ij}(k-1) - \gamma \,.\, \Delta A_i) \qquad (7)$$

Where: γ=1 for all, and **kp** is different for every weight pairs. It has their assigned values empirically by observing the dynamics of process performance, recursive method, **kp**=40 for (W14; W23), **kp**=18 for (W13; W24) and kp=2.35 for (W53; W54), with normalized values.

The DFCM inference is like Classic FCM [10], and the inference equations are shown below (equations (8) and (9)).

$$A_i = \int \left(\sum_{\substack{j=1 \\ j \neq i}}^{n} (A_j . W_{ji}) \right) \qquad (8)$$

$$f(x) = \frac{1}{1+e^{-\lambda x}} \qquad (9)$$

Fig. 7 and Fig. 10 show the results of Hebbian Learning algorithm for DFCM considering the variations ΔAi of the concepts concerning volume, weight, outlet valve state, and the weights of the causal relationships in the process, considering two campaigns. These figures also show the evolution of the weights of the causal relationships during the process within a range of [-1, 1].

These equations combined suggest stability similarly to the work [30], which shows that threshold sigmoid functions have interval previous defined and are continuous differentiable. This observation is attributed to the use of the sigmoid function, which lures the calculated values and causes their convergence to the same specific value [31].

The stability initials analyses and results have been presented by the same authors in [32], this study was done by using an appropriately defined contraction mapping theorem and the non-expansive mapping theorem. In other way, Kosko examined Associative Memories stability by identifying a Lyapunov or energy function with associative memory states [33-34]. The DFCM uses the same equations of FCM, with dynamic tune, thus experimental results show stability.

VI. SIMULATED EXPERIMENTAL RESULTS

The results of DFCM are shown in Figs. 8, 9, 11, and 12, which show the behavior of the controlled variables within the predetermined range of the volume and weight of the mixture.

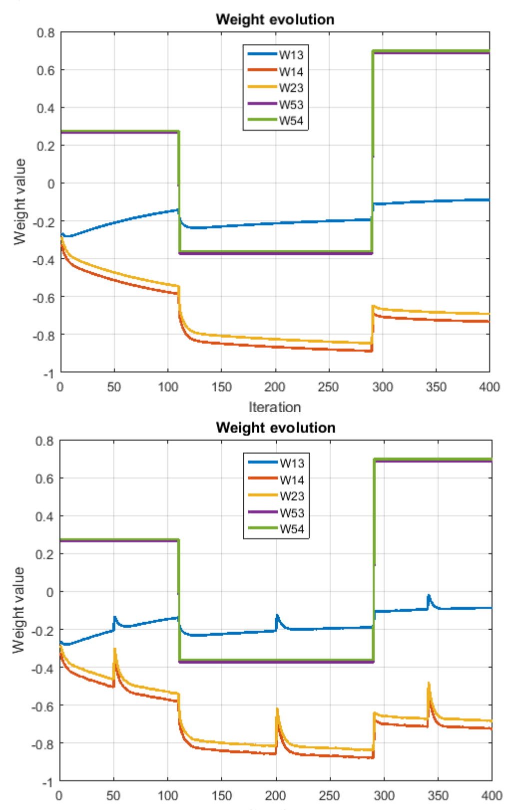

Figure 7. Weight evolution in the Hebbian Learning, 1st campaign without and with disturbances

It is noteworthy that the controller keeps the variables in the control range and pursues a trajectory according to a campaign, where the output flow is also predetermined. In this initial experiment, a campaign with a sequence of values ranging from 7.5 and 11 ml/min can be a set point output flow (outlet valve).

Similarly, the results for the first and second campaigns of the Fuzzy controller are shown in Figs. 13-16. It is observed that: the behaviors of DFCM and Fuzzy controllers were similar when the tank is empty, with a slightly advantage for the Fuzzy controller, which reached the desired result after 230 steps, while the DFCM needed 250 steps with the adaptation off.

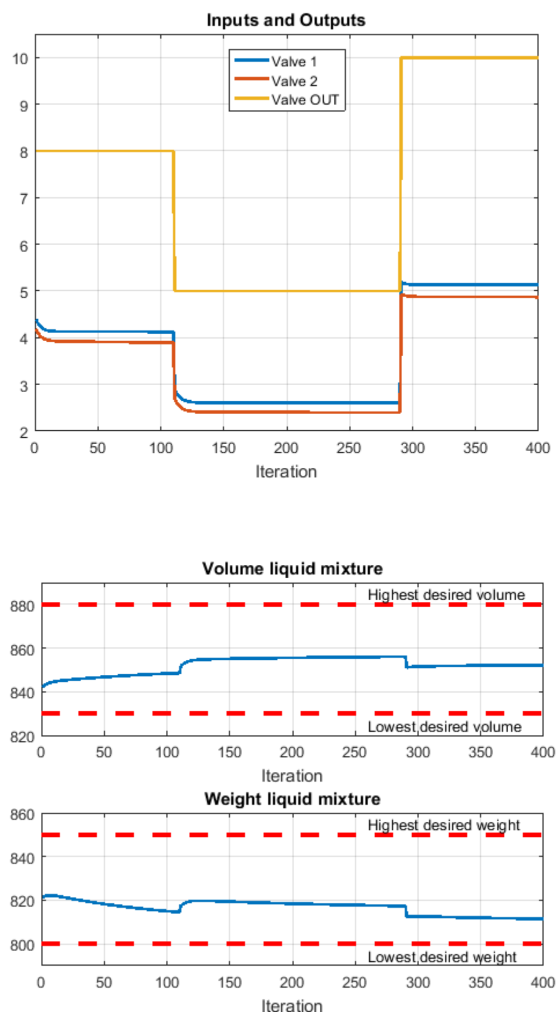

Figure 8. Valves and Results of the DFCM Controller, 1st campaign without disturbances

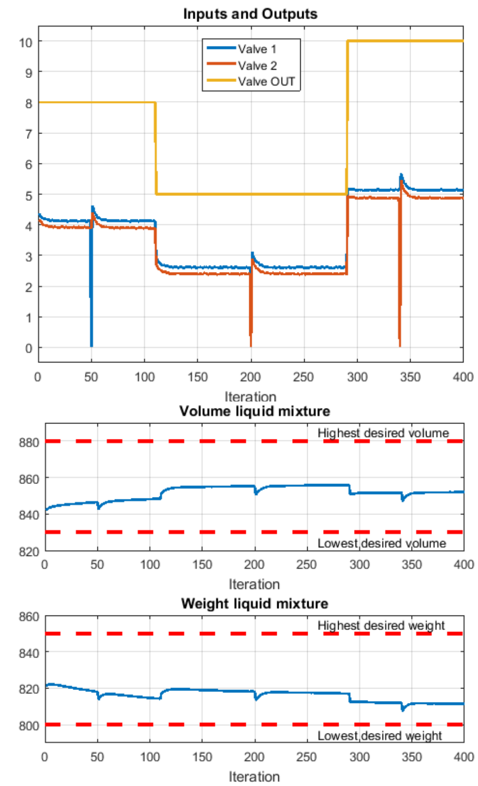

Figure 9. Valves and Results of the DFCM Controller, 1st campaign with disturbances

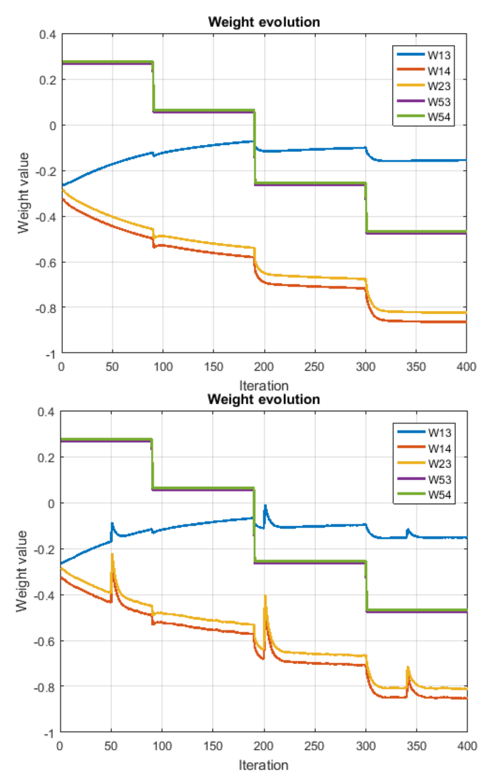

Figure 10. Weight evolution in the Hebbian Learning, 2nd campaign without and with disturbances

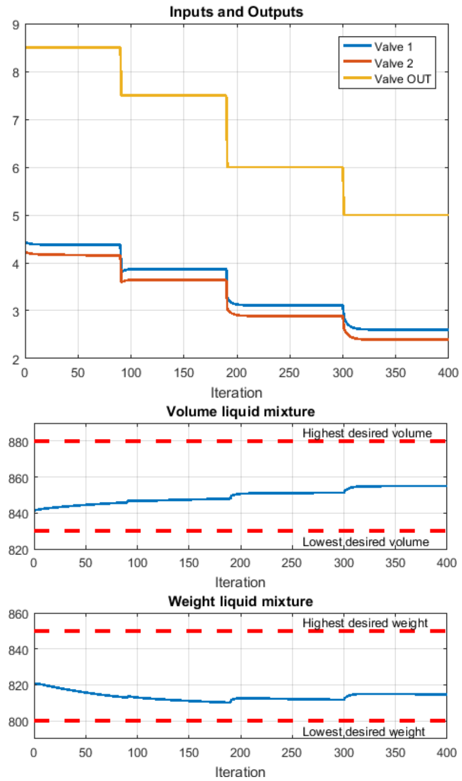

Figure 11. Valves and Results of the DFCM Controller, 2nd campaign without disturbances

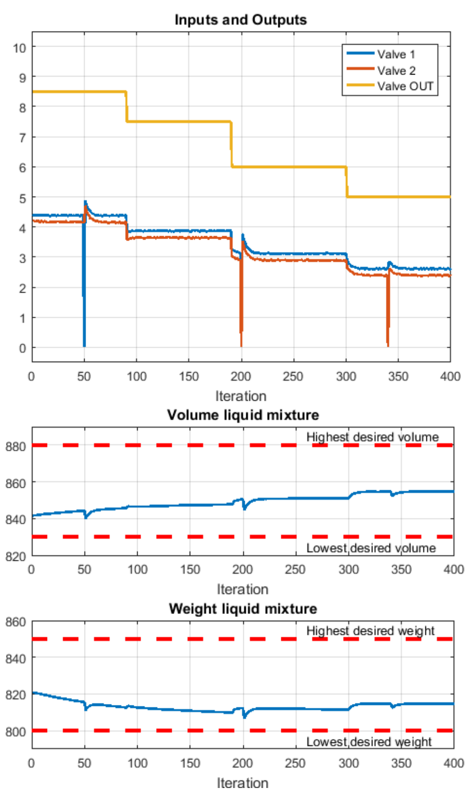

Figure 12. Valves and Results of the DFCM Controller, 2nd campaign with disturbances

Tables II and III show that the simulated numeric results of the DFCM controller had a similar performance compared to the conventional Fuzzy Logic controller, and DFCM embedded in Arduino with small difference under same conditions, with simulated noise and valve locking.

TABLE II. QUANTITATIVE RESULTS WITHOUT DISTURBANCES

	DFCM		***Fuzzy Logic***		***DFCM-Arduino***		***Fuzzy-ANN***	
	Max-min		**Max-min**		**Max-min**		**Max-min**	
Campaign	**1**	**2**	**1**	**2**	**1**	**2**	**1**	**2**
Volume mix (mL)	14.07	13.52	35.55	38.20	24.74	26.11	36.69	38.11
Weight mix (mg)	10.74	10.68	22.87	16.65	9.23	8.66	25.31	25.28

TABLE III. QUANTITATIVE RESULTS WITH DISTURBANCES

	DFCM		***Fuzzy Logic***		***DFCM-Arduino***		***Fuzzy-ANN***	
	Max-min		**Max-min**		**Max-min**		**Max-min**	
Campaign	**1**	**2**	**1**	**2**	**1**	**2**	**1**	**2**
Volume mix (mL)	13.82	14.79	35.51	38.12	24.79	26.05	36.69	38.10
Weight mix (mg)	14.69	14.31	28.02	20.64	13.05	11.49	25.28	25.29

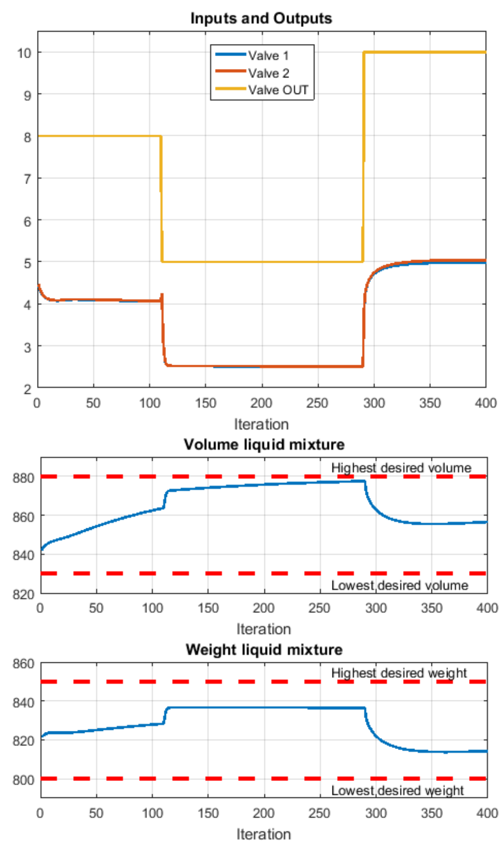

Figure 13. Valves and Results of the Fuzzy Controller, 1st campaign without disturbances

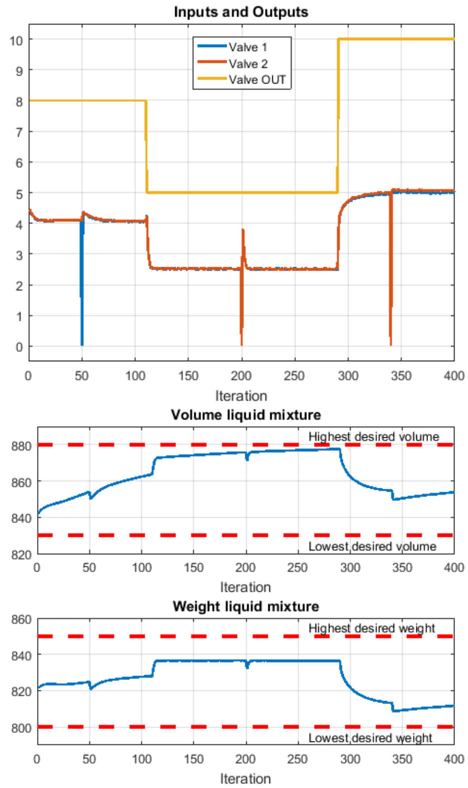

Figure 14. Valves and Results of the Fuzzy Controller, 1st campaign with disturbances

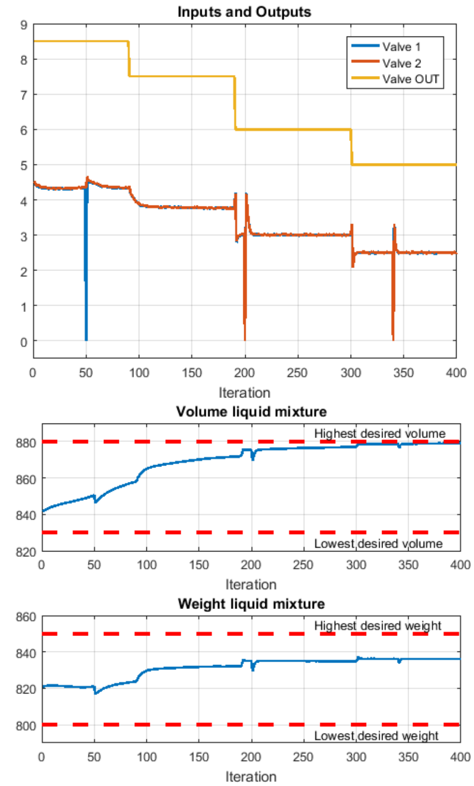

Figure 16. Valves and Results of the Fuzzy Controller, 2nd campaign with disturbances

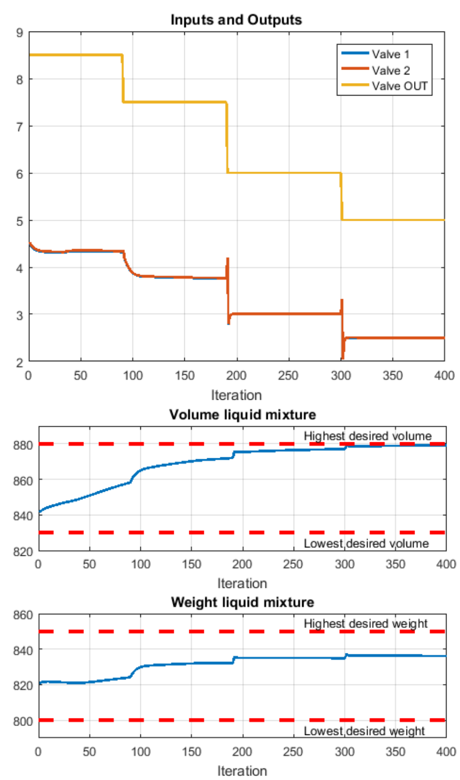

Figure 15. Valves and Results of the Fuzzy Controller, 2nd campaign without disturbances

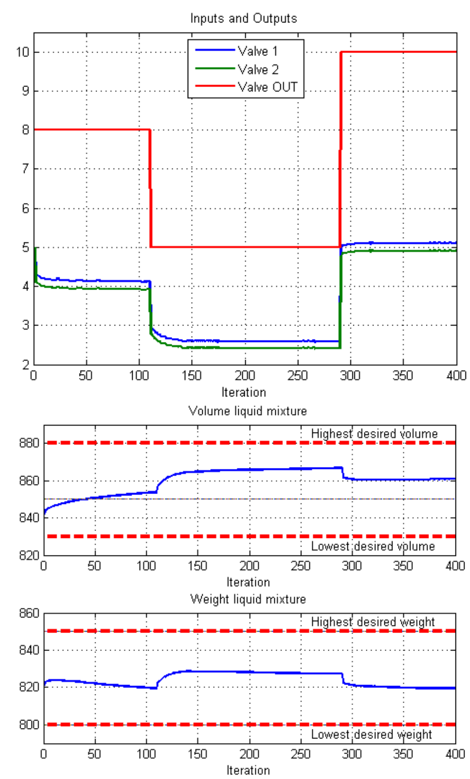

Figure 17. Valves and Results of the Arduino embedded DFCM Controller, 1st campaign without disturbances

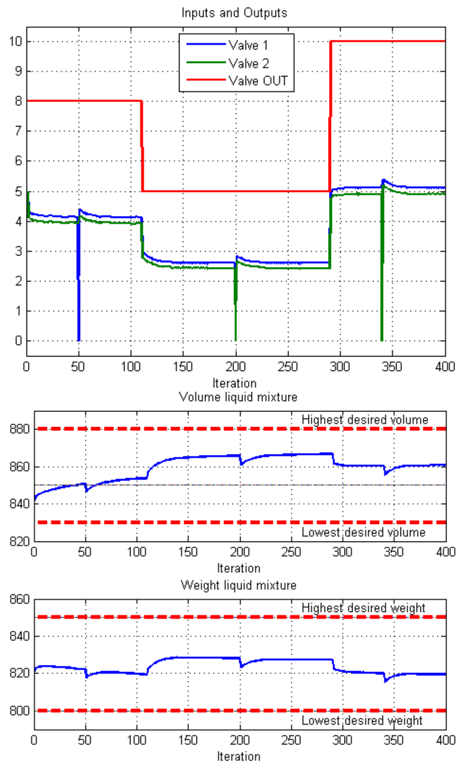

Figure 18. Valves and Results of the Arduino embedded DFCM Controller, 1st campaign with disturbances

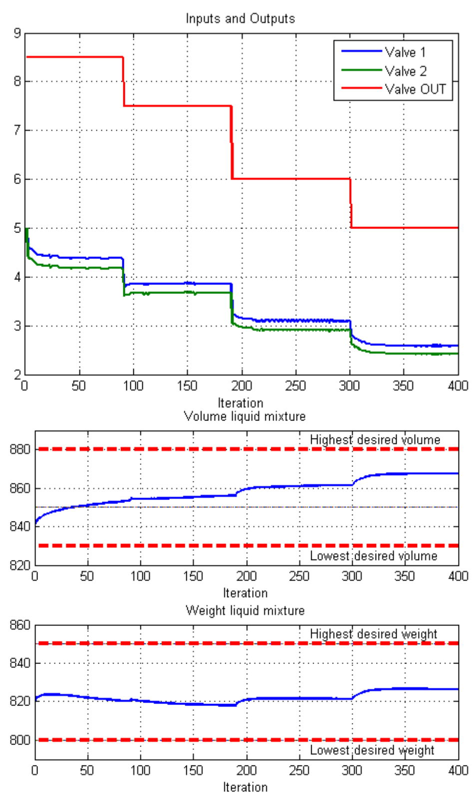

Figure 19. Valves and Results of the Arduino embedded DFCM Controller, 2nd campaign without disturbances

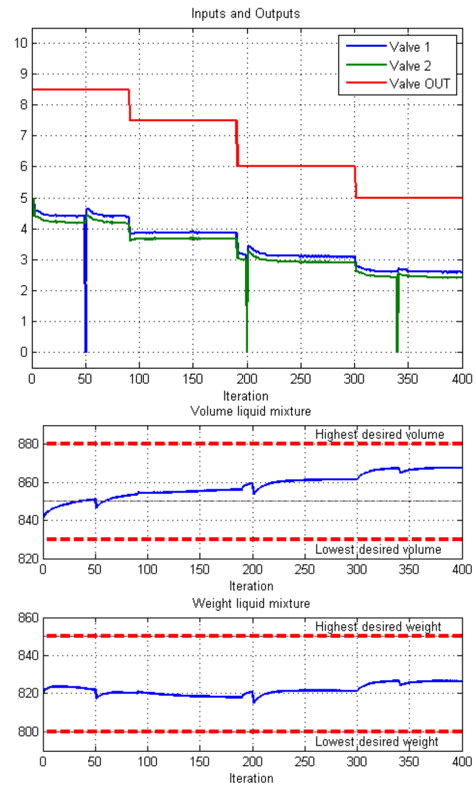

Figure 20. Valves and Results of the Arduino embedded DFCM Controller, 2nd campaign with disturbances

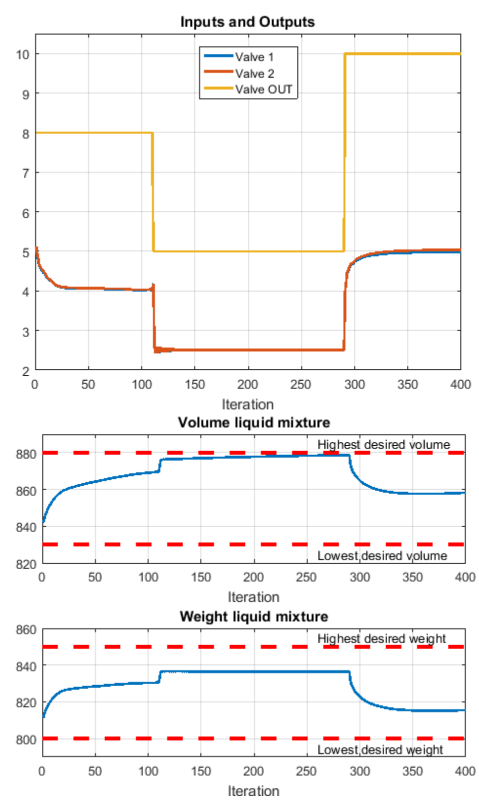

Figure 21. Valves and Results of the Fuzzy-ANN Controller, 1st campaign without disturbances

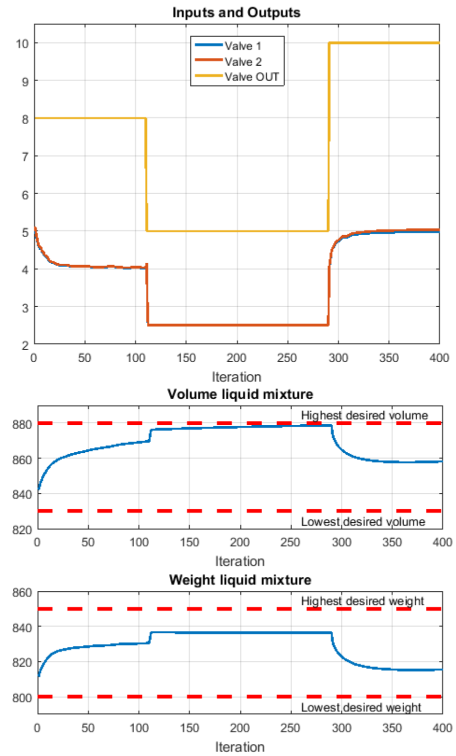

Figure 22. Valves and Results of the Fuzzy-ANN Controller, 1st campaign with disturbances

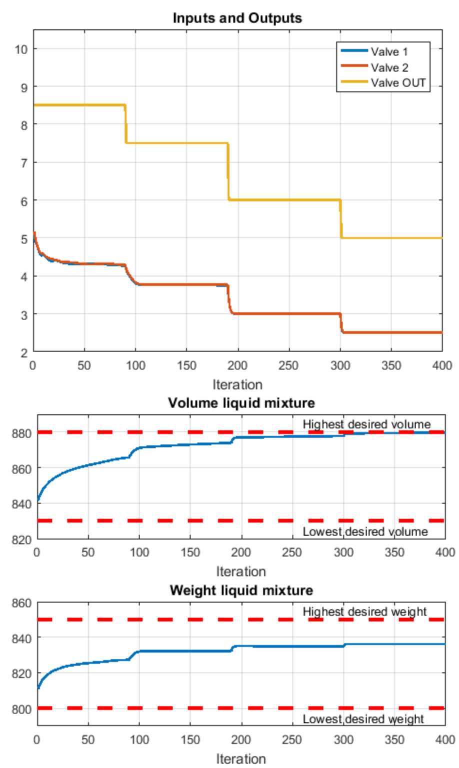

Figure 23. Valves and Results of the Fuzzy-ANN Controller, 2nd campaign without disturbances

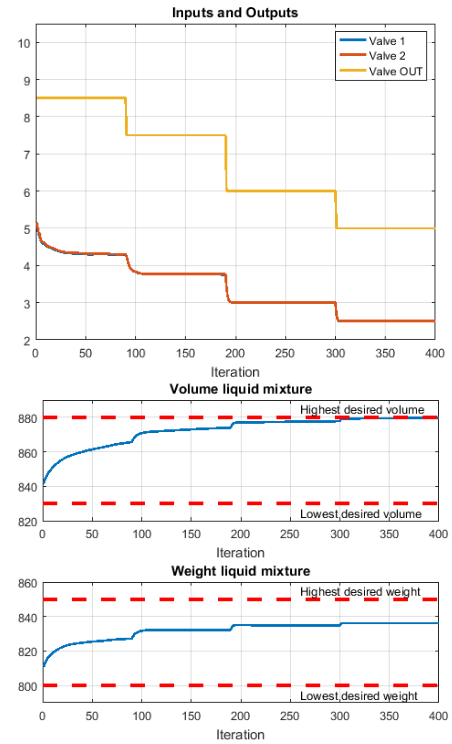

Figure 24. Valves and Results of the Fuzzy-ANN Controller, 2nd campaign with disturbances

In order to extend the applicability of this work, the developed DFCM controller is embedded into an Arduino platform, which ensures the portability of the FCM generated code. Arduino is an open-source electronic prototyping platform. Arduino was chosen because it is a cheap controller, and mainly because of its low processing capacity, to emphasize the low computational complexity of FCM [27].

Matlab, simulating the process, calculates the equations for volume and weight. Through a serial communication established with Arduino, Matlab sends the current values of volume, weight and output valve to Arduino that receives these data, calculates the values of the concept 3 (Valve 1) and concept 4 (Valve 2) and then returns these data to Matlab.

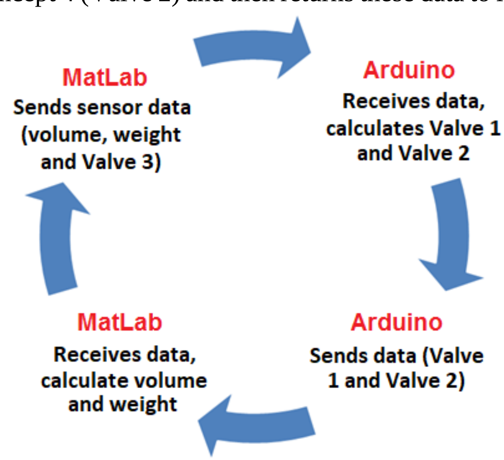

Figure 25. Matlab-Arduino comunication cycle [29]

After that, new values of volume and weight are recalculated. Details on how this technique can be used are presented in Matlab Tutorial, Matlab and Arduino codes, by accessing the link [35]. The cycle of communication between Arduino to Matlab can be checked in Fig. 25.

Figs. 17-20 show the results obtained with the Arduino platform providing data of the actuators, Valve 1 and Valve 2, with Matlab performing data acquisition. The algorithm switches the sets of causal relations that operate similarly to a DFCM simulated with noise and disturb in the Valve 1.

The noise in Figs. 18 and 20 is the sum of the real noise, observed in data transference between Arduino and Matlab, and a simulated white noise. Equation (10) shows the composition of the experiment noise. The Arduino script updates the causal relationships weights every iteration, according to (7). While the MatLab emulates the studied process and plot the results.

$$Noise_{Experiment} = Noise_{Simulated} + Noise_{Arduino-Matlab} \quad (10)$$

Some metrics aspects of the controllers were observed, such as the processing time of the simulations ran on a Intel Core I5™, 6 GB RAM computational base. The results of the Fuzzy logic and DFCM were quite the same, with a small advantage for the DFCM, due to its low computational complexity, as shown in Figs. 17 and 18.

Profile Summary
Generated 12-Sep-2017 20:38:23 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
dfcm_tanque_revista_icas_c1_comruido	1	4.911 s	1.556 s	
close	1	1.438 s	0.002 s	
close>request_close	1	1.298 s	0.028 s	
legend	2	1.252 s	0.012 s	
legend>make_legend	2	1.238 s	0.032 s	
closereq	3	1.225 s	0.850 s	

Figure 26. DFCM controller performance, 1st campaign with disturbances

Profile Summary
Generated 12-Sep-2017 20:41:09 using performance time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
fuzzy_tanque_revista_icas_c1_comruido	1	5.387 s	1.597 s	
close	1	1.307 s	0.002 s	
close>request_close	1	1.146 s	0.014 s	
closereq	3	1.099 s	0.717 s	

Figure 27. Fuzzy controller performance, 1st campaign with disturbances

In this paper, the DFCM controller is not recursive (that can be seen on equations (7) and (8)), but is compact with just 6 lines of code, as shown in Fig. 28.

The microcontroller chosen for this work was the most basic version of the Arduino software, Arduino UNO R2, with the lowest processing power; it suggests the algorithm has low computational complexity. Future works addresses the quantitative definition of the computational complexity of the algorithm.

arduino_final_12_09 §

```
 W24old=W24;
 W53old=W53;
 W54old=W54;

//*********** DFCM CONTROLLER ***********
        xx = (C1*W13)+(C2*W23)+(C5*W53);
        termo_exp = (1+exp(-xx));
        C3 = 1.0/termo_exp;

        xx = (W14*C1)+(W24*C2)+(C5*W54);
        termo_exp2 = (1+exp(-xx));
        C4 =(1.0/termo_exp2);
//****************************************
Ve1=C3*10.0; //normalize
Ve2=C4*10.0;
```

Figure 28. DFCM controller in Arduino IDE

VI. CONCLUSION

The contribution of this study focuses of Fuzzy Cognitive Maps in the embedded control area. In simulated data, the results are similar for the three controllers, with advantage for DFCM with or without Arduino, observed that DFCM controller is adaptive.

Two different campaigns (two different set-point, with and without disturbances) were used to test the algorithms, which, the results obtained from both controllers were quite the same. However, the Fuzzy-ANN did not have any significant improvement, there was a slightly reduction of the noise which can be a major factor on industrial plants.

Thus, one can emphasize the portability and the possibility of developing DFCM controllers on low cost platforms. From the data obtained from Arduino microcontroller, based on the variations of the DFCM embedded in the platform, it is observed that the controlled variables were in well-behaved ranges, which suggests that the DFCM codes have low computational complexity due to the simplicity of its inference mathematical processing. The low computational complexity can be seen through the metrics aspects observed.

Future studies will quantify the computational complexity of the DFCM, for a more general conclusion, and results with a real prototype.

REFERENCES

[1] E. I. Papageorgiou, M. Mendonça, R. V. P. D. Barros, P. P. Soares, L. B. de Souza, “Dynamic Fuzzy Cognitive Maps Embedded and Classical Fuzzy Controllers Applied in Industrial Process,” ICAS 2017: The Thirteenth International Conference on Autonomic and Autonomous Systems, vol. 1, pp. 54-59, May 2017.

[2] P. C. Marchal, J. G. García, J. G. Ortega, “Application of Fuzzy Cognitive Maps and Run-to-Run Control to a Decision Support System for Global Set-Point Determination,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. PP, no. 99, pp. 1-12, 2017.

[3] M. Mendonça, F. Neves Jr, L. V. R. Arruda, E. I. Papageorgiou, I. R. Chrun, “Embedded Dynamic Fuzzy Cognitive Maps for Controller in Industrial Mixer,” 8th International KES Conference on Intelligent Decision Technologies KES-IDT-16, Tenerife. KES-IDT-16, pp. 1-10, 2016.

[4] L. A. Zadeh, “An introduction to Fuzzy logic applications in intelligent systems,” Boston: Kluwer Academic Publisher, 1992.

[5] J. A. Fabro, L. V. R Arruda, “Fuzzy-neuro predictive control, tuned by genetic algorithms, applied to a fermentation process,” Proceedings of the 2003 IEEE International Symposium on Intelligent Control, Houston, TX, USA, pp. 194-199, 2003.

[6] E. Yesil, T. Kumbasar, O. Karasakal, “Selftuning interval type-2 fuzzy PID controllers based on online rule weighting,” 2013 IEEE In-ternational Conference on Fuzzy Systems (FUZZ-IEEE), Hyderabad, pp. 1-6, 2013.

[7] M. Mendonça, B. Angelico, L. V. R. Arruda, F. Neves Jr, “A dynamic fuzzy cognitive map applied to chemical process supervision,” Engineering Applications of Artificial Intelligence – Journal – Elsevier, 2012.

[8] W. Stach, L. Kurgan, W. Pedrycz M. Reformat, “Evolutionary Development of Fuzzy Cognitive Maps, The 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05., Reno, NV, pp. 619-624, 2005.

[9] L. V. R. Arruda, M. Mendonca, F. Neves-Jr, I. R. Chrun, E. I. Papageorgiou, “Artificial Life Environment Modeled by Dynamic Fuzzy Cognitive Maps,” IEEE Transactions on Cognitive and Developmental Systems, vol. PP, no. 99, pp. 1-1, 2016.

[10] B. Kosko, “Fuzzy cognitive maps,” International Journal Man-Machine Studies, vol. 24, no. 1, pp. 65-75, 1986.

[11] R. Taber, “ Fuzzy cognitive maps model social systems,” AI Expert, 1994.

[12] J. P. Carvalho, J. A. Tome, “Rule based Fuzzy cognitive maps-qualitative systems dynamics,” Proceedings 19th International Conference of the North America. Fuzzy Information Fuzzy Processing Society, 2000.

[13] J. P. Carvalho, J. A. Tome, “Rule Based Fuzzy Cognitive Maps in Socio-Economic Systems,” European Society for Fuzzy Logic and Technology Conference, 2009.

[14] J. Aguilar, “Dynamic random Fuzzy cognitive maps,” Computación y Sistemas, vol. 7, no. 4, 2004.

[15] B. Kosko, “Neural networks and fuzzy systems: a dynamical systems approach to machine intelligence,” New York: Prentice Hall, 1992.

[16] K. C. Lee, S. Lee, S., “A cognitive map simulation approach to adjusting the design factors of the electronic commerce web sites,” Expert Systems with Applications, vol. 24, no. 1, pp. 1-11, 2003.

[17] M. Mendonça, I. R. Chrun, F. Neves- Jr; L. V. R. Arruda, “A cooperative architecture for swarm robotic based on dynamic fuzzy cognitive maps,” Engineering Applications of Artificial Intelligence. vol. 59. pp. 122-132, 2017.

[18] E. I. Papageorgiou, “Fuzzy Cognitive Maps for Applied Sciences and Engineering from Fundamentals to Extensions and Learning Algorithms,” Springer, 2013.

[19] Y. Miao, Z. Q. Liu, C. K. Siew, C. Y. Miao, “Dynamical cognitive network - an Extension of fuzzy cognitive,” IEEE Trans. on Fuzzy Systems, vol. 9, no. 5, pp. 760-770, 2001.

[20] M. Mendonça, B. Angélico, L. V. R. Arruda, F. Neves-Jr, “A dynamic fuzzy cognitive map applied to chemical process supervision,” Engineering Applications of Artificial Intelligence, vol. 26, pp. 1199-1210, 2013.

[21] M. Glykas, “Fuzzy Cognitive Maps: Advances in Theory, Methodologies, Tools and Applications,” Springer-Velarg Berlin Heidelberg, 2010.

[22] C. D. Stylios, P. P. Groumpos, V. C. Georgopoulos, “An Fuzzy Cognitive Maps Approach to Process Control Systems,” J. Advanced Computational Intelligence, no. 5, pp. 1-9, 1999.

[23] E. I. Papageorgiou, K. E. Parsopoulos, C. S. Stylios, P. P. Groumpos, M. N. Vrahatis, “Fuzzy cognitive maps learning using Particle Swarm Optimization,” Journal of Intelligent Information Systems, vol. 25, pp. 95–121, 2005.

[24] T. J. Ross, “Fuzzy logic, with Engineering Aplications,” 2nd Ed., England, John Whiley & Sons, 2004.

[25] S. S. Farinwata, D. Filev, R. Langari(editors), “Fuzzy Control, Synthesis and Analysis,” West Sussex, England, John Wiley & Sons, 2000.

[26] D. E. Goldberg, “Genetic algorithms in search optimization and machine learning,” Mass: Addison-Wesley, 1989.

[27] E. I. Papageorgiou, “Learning Algorithms for Fuzzy Cognitive Maps,” IEEE Transactions on Systems and Cybernetics. Part C: Applications and Reviews, vol. 42, pp. 150-163, 2012.

[28] Y. Miao, Z. Q. Liu, C. K. Siew, C. Y. Miao, “Transformation of cognitive maps,” IEEE Transactions on Fuzzy Systems, vol. 18, no. 1, pp. 114-124, 2010.

[29] D. E. Matsumoto, M. Mendonça, L. V. R. Arruda, E. I. Papageorgiou, “Embedded Dynamic fuzzy cognitive maps applied to the control of industrial mixer,” Brazilian Symposium on Intelligent Automation – XI SBAI. 2013.

[30] Y. Boutalis, T. L. Kottas, M. Christodoulou, “Adaptive Estimation of Fuzzy Cognitive Maps With Proven Stability and Parameter Convergence,” IEEE Transactions on Fuzzy Systems, vol. 17, no. 4, pp. 874-889, Aug. 2009.

[31] V. Eleni, G. Petros, “New concerns on fuzzy cognitive maps equation and sigmoid function,” 2017 25th Mediterranean Conference on Control and Automation (MED), Valletta, pp. 1113-1118, 2017.

[32] Y. Boutalis, T. L. Kottas, M. Christodoulou, “On the existence and uniqueness of solutions for the concept values in Fuzzy Cognitive Maps,” 2008 47th IEEE Conference on Decision and Control, Cancun, pp. 98-104, 2008.

[33] B. Kosko, “Bidirectional associative memories,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 18, no. 1, pp. 49-60, Jan./Feb. 1988.

[34] A. S. Martchenko, I. L. Ermolov, P. P. Groumpos, J. V. Poduraev, C. D. Stylios, “Investigating Stability Analysis Issues for Fuzzy Cognitive Maps,” 11th Mediterranean Conference on Control and Automation - MED'03, 2003.

[35] <https://www.dropbox.com/s/2sn76n64n48qgp3/Tutorial%20 Arduino%20Matlab%20in%20English.pdf?dl=0> Last access date: 17/12/2017.