How to improve the convergence on sigmoid Fuzzy Cognitive Maps?
Peer-reviewed author version

# How to improve the convergence on sigmoid Fuzzy Cognitive Maps?

# How to improve the convergence on sigmoid Fuzzy Cognitive Maps?

Gonzalo Nápoles[1,2*], Rafael Bello[1] and Koen Vanhoof [2]


[1]Universidad Central "Marta Abreu" de Las Villas
Carretera a Camajuaní km. 5 1/2, Santa Clara, Villa Clara, Cuba


[2]Hasselt University
Martelarenlaan 42 BE-3500 Hasselt, Belgium


*Corresponding author: gnapoles@uclv.edu.cu (E-mail), +53 42 281515 (Tel)

**Abstract.** Fuzzy Cognitive Maps (FCM) may be defined as Recurrent Neural Networks that allow causal reasoning. According to the transformation function used for updating the activation value of concepts they can be characterized as discrete or continuous. It is remarkable that FCM having discrete neurons never exhibit chaotic states, but this premise cannot be guaranteed for FCM having continuous concepts. On the other hand, complex sigmoid FCM resulting from experts or learning algorithms often show chaotic or cyclic patterns, therefore leading to confusing interpretation of the investigated system. The first contribution of this paper is focused on explaining why most studies on FCM stability are not applicable to FCM used on classification or decision-making tasks. Next we describe a non-direct learning methodology based on Swarm Intelligence for improving the system stability once the causal weight estimation is done. The objective here is to find a specific threshold function for each map neuron simulating an external stimulus, instead of using the same transformation function for all concepts. At the end, we can compute more stable maps, so better consistency in hidden patterns is achieved.


**Keywords:** Fuzzy Cognitive Maps, Stability, Prediction, Learning Algorithm.

# 1 Introduction

The FCM theory [2] was introduced by B. Kosko as a knowledge-based methodology for modeling and simulating dynamical systems. These structures combine the reasoning strength of the connectionist approach with some elements of fuzzy logic, which are normally considered during the knowledge-engineering phase [3]. Using this methodology, a system can be modeled in terms of concepts (e.g. variables, objects or states which are equivalent to neurons in connectionist models) and causal relations among such entities. In a few words, from the structural point of view, a simple FCM may be denoted as directed graphs allowing feedback, consisting on neurons and weighted arcs.

In a simple FCM, each link takes value in the range $[-1,1]$. It denotes the causality degree between two nodes as a result of the quantification of a fuzzy linguistic variable [6]. The activation value of concepts is also fuzzy in nature and regularly takes values in the range $[0,1]$. Hence, the higher the activation value, the stronger the influence of the concept over the investigated system. In other words, FCM establish the forward and backward propagation of causality [5], admitting the knowledge base to increase when concepts and links between them are increased.

In the past decade, FCM based models have gained considerable research interest among researches, being widely used for solving several real-world problems. Some practical examples include [11]: decision-making tasks, risk analysis, prediction, text categorization, pattern recognition, management, and classification. However, estimating parameters that

characterize the whole system (e.g. the causal weight matrix) may be tedious for humans, leading to inefficient models. In order to increase the reliability of FCM-based models several learning algorithms for tuning such parameters have been introduced [10], although most of them are focused on computing causal relations between map concepts.

On the other hand, existing approaches suppose that FCM are closed systems and they do not consider external influences, while other factors such as the FCM stability are frequently ignored. As far as known, there is no existence of any learning method for enhancing the system stability once the system causality is established. For example, let us suppose a FCM resulting from experts where causal connections may be partially modified (e.g. we know the direction of causalities and an approximation of their values that should be preserved). Can we expect lineal stability in the final map? If not, how to improve the stability of the system without affecting causal connections estimated by experts?

In the literature a few researches concerning FCM convergence have been proposed. For instance, Kosko [4] developed an analytic method based on Liapounov functions for reaching stable solutions on Feedback Standard Additive Models (SAM - which share several of the FCM characteristics). Unfortunately, Kosko concluded that such conditions cannot be extended to FCM due to the large number of feedback links involved in FCM-based models. More recently, other analytical methods were introduced (see next Section 3), but we believe that such approaches are mostly useful for stabilizing FCM used in modeling tasks.

Being more specific, this paper attempts explaining why most analytical models ensuring the existence and uniqueness of a fixed-point attractor cannot be used for solving prediction or decision-making tasks. As well, we describe a non-direct algorithm based on Swarm Intelligence, whose learning goal is the system stability. Briefly, the main idea here is to find an appropriate threshold function for each map neuron, instead of using the same function for all concepts. It attempts enhancing the map stability once the estimation of the causal weight matrix is done, without affecting the inference ability of the original system. In practice this scheme could be modeled as a sigmoid FCM using excitable neurons, that is, neurons that could be influenced by external factors.

It is pertinent to remark that authors will be focused on sigmoid FCM, instead of discrete (e.g. binary o trivalent) ones. This remark is motivated by the benchmarking analysis discussed in [27] where results showed that the sigmoid function significantly outperforms the other functions, by using the same decision model. Besides, since FCM are deterministic models, a discrete map will always find a previously visited state vector (although the path length to reach this point could be exponential).

The rest of the paper is organized as follows: in Section 2 mathematical specifications about FCM are provided. In Section 3 we review the most relevant studies on FCM convergence and their drawbacks, whereas in Section 4 we describe a learning algorithm based on Swarm Intelligence principles. After that, Section 5 introduces the experimental framework concerning HIV-1 drug resistance, and also provides further details about simulations. Finally, conclusions are given in Section 6.

4

## 2  A brief background on Fuzzy Cognitive Maps

Mathematically speaking, a simple FCM can be defined using a 4-tuple $(C, W, A, f)$ where $C = \{C_1, C_2, C_3, \ldots, C_M\}$ is a set of $M$ graph nodes or neurons, $W: (C_i, C_j) \rightarrow w_{ij}$ is a function which associates a causal value $w_{ij} \in [-1,1]$ to each pair of nodes $(C_i, C_j)$. It denotes the weight of the directed edge from $C_i$ to $C_j$, which also represents the causality degree between these concepts. Therefore the weight matrix $W_{M \times M}$ gathers the system causality which could be estimated by experts, or computed from historical data. Likewise, $A: (C_i) \rightarrow A_i$ associates an activation degree $A_i \in \mathbb{R}$ to each concept $C_i$ at the moment $t$ $(t = 1,2, \ldots, T)$. Finally, a threshold or transformation function $f: \mathbb{R} \rightarrow [0,1]$ is used to keep the activation value of neurons in the allowed range.

Following equation (1) shows how to update the activation vector (i.e. the activation value of neurons) using the state vector $A^0$ as the initial configuration. In the same way to other recurrent models - such as the well-known Hopfield network - this information propagation mechanism is iteratively repeated until a hidden pattern [5] is observed (which is an ideal outcome), or a maximal number of iterations $T$ is reached. In this scheme the most commonly used threshold functions are: the bivalent function, the trivalent function, and the sigmoid variants.

$$A_i^{(t+1)} = f\left(\sum_{j=1}^{M} w_{ji} A_j^{(t)}\right), i \neq j \qquad (1)$$

The effect on the selection of a specific function over the stability and inference capabilities of the FCM was explored in [1]. From this work some important remarks were concluded:

- Discrete FCM never show chaotic behavior. It means that always a fixed-point attractor or a limit cycle will be detected in the map outputs. These states have the following behavior:

  - Fixed-point attractor ($\exists t_k \in \mathbb{N} : A^{(t+1)} = A^{(t)}, \forall t > t_k$): the system produces the same output after the time $t_k$.

  - Limit cycle ($\exists t_k, P \in \mathbb{N} : A^{(t+P)} = A^{(t)}, \forall t > t_k$): the same output or state vector is regularly observed with period $P$.

- Continuous FCM may additionally exhibit chaotic states, where the FCM model continues to produce different state vectors for successive cycles. In these situations the FCM cannot stabilize, leading to confusing system responses.

During the simulation phase on a sigmoid FCM, the activation value of each map neuron is influenced by the values of the connected concepts with the appropriate weights [15]. It shows the causal effect of changes on the neurons' activation value on the whole system. More specifically, once the system reaches a fixed-point attractor, decision-makers use this information to make decisions leading to the desired solution[25]. That is why FCM theory is a very convenient approach for performing modeling and simulation tasks. However, when the system reaches a limit cycle or a chaotic behavior, decision-making is practically impossible [30]. In the following Section 3 we review some theoretical results that attempts to provide better stability features on FCM-based models.

## 3   Stability on non-discrete Fuzzy Cognitive Maps

As far as known, in the literature few researches concerning stability on continuous FCM have been proposed. In this section we revise the most relevant contributions in this field, and after that we try to explain why these approaches are not adequate for stabilizing sigmoid FCM used on classification tasks. Here the main goal of a FCM-based classifier is the mapping of an input set (initial excitation of input neurons) to a desired output vector (system response), according to the representation of the knowledge stored inside the classifier's structure.

In reference [31] the authors proved that when the weight matrix fulfills certain conditions, then the FCM will converge to a unique fixed-point attractor. For example, if the map has not input neurons (they influence but are not influenced by the other nodes) then the map will converge to the same attractor regardless the exact values of the initial concept values (see Theorem 1). Besides, they introduce an adaptive weight-estimation method that employs appropriate weight projection criteria to assure that the uniqueness of FCM solution is not compromised.

This result is only useful for modeling and cannot be used in FCM-based classifiers. In order to understand implicit limitations of this theorem, let us suppose a balanced classification problem having three values for the decision attribute (e.g. Low, Middle and High). According to Theorem 1 and supposing that convergence conditions are met, then the FCM will always converge to the same fixed-point attractor, which means that 2/3 of all instances will be erroneously classified.

***Theorem 1****: There is one and only one solution for any concept value $A_i$ of any FCM (using a sigmoid transformation function), if:*

$$\left(\sum_{i=1}^{n}\|w_i\|\right)^{1/2} < 4 \qquad (2)$$

It does not consider FCM with input concepts; although results discussed above are still valid (see Theorem 2). In these cases the equilibrium point does not depend solely on the weight set, as in the case of FCMs with no input nodes, but also depends on the values of the input neurons. Hence, different excitation values will drive the FCM to different fixed-point attractors. It means that, for each equilibrium point, we need to compute a distinct weight set ensuring the desired response, leading to ambiguous understanding of the same investigated system.

***Theorem 2****: For a sigmoid FCM with m input nodes, there is one and only one solution for any concept value $A_i$ if:*

$$\left(\sum_{i=1}^{n}\|w_{m+i}\|\right)^{1/2} < 4 \qquad (3)$$

In reference [7] a slightly different theorem concerning the steepness $\lambda$ of sigmoid threshold function is presented (see next Theorem 3). It is based on the assumption that if there were two stable fixed points then a slight change in initial conditions (which cannot be exactly quantified) may result in a totally different outcome, making the value of the map hard to justify. Though, following the same reasoning explained above it is clear that this approach seriously limits the simulation ability of sigmoid FCM when solving classification or prediction problems.

***Theorem 3****: The number of solutions of (1) depends on the size of λ:*

- *If λ > 0 is small enough then there is a unique solution. This fixed point of the sigmoidal FCM, is linearly stable.*

- *If λ > 0 is large enough there can be multiple solutions, where many of these fixed points may be linearly stable.*

For better understanding of the above affirmation let us consider a FCM concerning travel behavior analysis [24]. This model involves three kinds of variables which are translated as map neurons, and three further nodes for each transport mode (bus, bike or car). When the FCM inference process is activated, a decision concept called "Utility" is examined. It has evidence related to the user preferences with respect to each transport mode for an initial condition. Conversely, according to Theorem 3 and supposing that λ > 0 is small enough, the map will converge to the same solution independently the initial scenario, so the system will not be able to distinguish between two different decisions.

Summarizing, although some researches [21] ensure that FCM cannot be applied in classification tasks, recent studies [9,19,20,22] showed that FCM are in turn potential classifiers. However, FCM-based classifiers do not regularly consider the system stability in their learning phase, and as was discussed in this section, some of most relevant approaches seem to be inadequate to overcome this serious problem. In the next section we describe a supervised learning algorithm based on Swarm Intelligence, which attempts improving the system stability on sigmoid FCM, once the estimation of the causal weight matrix is done.

## 4   Improving the global stability on Sigmoid FCM

In the literature several supervised and unsupervised learning algorithms have been proposed, mainly focused on the transformation of the causal weight matrix. As a brief categorization, they can be gathered in three large categories [15]: Hebbian-based, population-based and also hybrid approaches. Nevertheless, most of these algorithms cannot guarantee the system stability (e.g. population-based methods). Thus, in this section we describe a non-direct learning algorithm, which is able to provide better stability on sigmoid FCM, once the causal weights matrix estimation is done. This approach attempts to reduce the system response variability over the time, but always preserving the accuracy.

Being more specific, Tsadiras [1] demonstrated that the inference ability of FCM-based models may be strongly influenced by the selection of the concepts' transformation function. On the other hand, Theorem 2 proved that the steepness $\lambda$ could be associated to the system stability. Based on these assumptions, in reference [16] a set of empirical simulations were performed with the goal of studying the effects of using several sigmoid transformation functions, over the global stability of the system response, as next equation (4) suggests. It should be stated that such experiments are equivalent to use a custom parameter $\lambda_i$ for each map neuron, instead of using the same constant factor for all functions. But which is the real

impact of this variation over the system behavior?

$$f_i(x) = \frac{1}{1 + e^{-(x-0.5)\lambda_i}} \qquad (4)$$

From empirical simulations authors concluded that variations of factor $\lambda_i$ lead to some changes on the map stability. Consequently, it seems to be reasonable to suppose that a learning algorithm could help to improve the system convergence, by solving the related real-parameter optimization problem. It is relevant to remark that normally FCM are considered as closed systems where external factors affecting the concepts are omitted. But in many real world problems this perception will be inadequate and may affect the global accuracy or the system stability.

For example, it is well-known that biological behavior on proteins not only depends on the amino acids interaction, but also is conditioned by external factors such as the chemical processes influencing the catalytic responses [16]. Precisely, such factors may be modeled using a threshold function $f_i$ for each neuron $C_i$ (instead of using the same function for all neurons), ensuring better stability. The activation value of a node $C_i$ will be now conditioned by the free interaction of the connected neurons, and also by the steepness $\lambda_i$ which involves the stimulus.

The effect of this artificial stimulus may be computed at each cycle as the difference between the map response using the same function for all nodes (i.e. $\lambda_1 = \cdots = \lambda_n$), and the system output using a proper family of sigmoid functions ensuring better stability. What is more, the model is equivalent to introduce a positive factor $\xi_i \leq 1$ to intensify or reduce the

final activation value of each neuron $C_i$. From this modification we can deduce that the parameter $\lambda_i = \lambda \xi_i$, which is the value that requires to be optimized during the supervised learning phase.

The learning goal is focused on estimating a family of sigmoid functions ensuring stability features. More explicitly, we need to find a family of functions $\{f_1(x), f_2(x), \ldots, f_M(x)\}$, where the $i$th sigmoid function will be used for updating the activation value of the $i$th concept. In practice, it implies to find the steepness $\lambda_i$ for each function. With this goal in mind, in reference [16] the authors presented a novel learning algorithm based on Swarm Intelligence principles [23], which exploits a particle swarm to solve the related optimization problem.

Particle Swarm Optimization (PSO) is a non-direct search method for solving challenging continuous problems [26]. In the standard PSO each particle (i.e. swarm agent) denotes a $M$-dimensional point in the solution space (where $M$ is the total number of map neurons). The $i$th dimension of each particle represents the steepness $\lambda_i$ associated to the $i$th threshold function. During the search, swarm agents adjust their position by using a combination of an attraction to the best positions that they individually have found, and an attraction to the best solutions that any particle has found, imitating those who have a better performance.

PSO-based methods have proven to be quite competent for solving real-parameter optimization tasks [32]. Nevertheless, the swarm is frequently attracted to local optima, causing premature convergence states. For this reason, the authors adopted a variant called PSO with Random Sampling

in Variable Neighborhoods (RSVN) which is able to notably outperform the constricted PSO algorithm [13]. Equation (5) displays the objective function that should be minimized during the search steps.

$$\phi(f_1, f_2, f_3, \dots, f_M) = \sum_{k=1}^{K} \sum_{i=1}^{M} \sum_{t=2}^{T} \left| A_{ik}^{(t)} - A_{ik}^{(t-1)} \right| \qquad (5)$$

In the above function, $K$ is the available number of instances, $M$ denotes the number of neurons, $T$ is the maximal number of times, whereas $A_{it}^{k}$ is the activation value of the $i$th concept for the current time $t$, using the $k$th instance as initial condition. Here a single instance is a sequence of values codifying the initial conditions of the modeled system, and its corresponding response. In brief, during the learning step the algorithm attempts to reduce the variability of the system response for each input over the time. In this scheme a solution will be considered as no feasible if the system inference is negatively affected, which preserves the global accuracy. On the other hand, causal weights cannot be modified during this optimization procedure since this model is oriented to compute more stable maps once the causality estimation is done.

It should be highlighted that the main goal of our research is to describe a methodology for improving the system stability, therefore we could use any continuous metaheuristic (e.g. Evolutionary Algorithms) for solving the related optimization task. What is more, during simulations we prefer to adopt an approximate approach since population-based metaheuristics are capable to find near-optimal solutions in a reasonable execution time, ignoring analytical properties of the objective function (e.g. continuity, convexity, differentiability or gradient information) which are frequently

unknown in advanced. However, exact algorithms such as mathematical programming techniques could be used as well.

It is fair to mention that the idea of using "excitable" neurons on FCM-based models is not a contribution of this work. For example, Stylios and Groumpos [8] introduced a new FCM model where each concept has an external output (bias), which influences each node with a weight and it is taken into account at the calculation rule. The reader may easily perceive the similarity between the Stylios' proposal and the approach discussed in this section, since both models are oriented to simulate influences over the neurons. In next Section 5 we conduct several experiments that allow studying the effects of the proposed learning algorithm over the stability of six sigmoid FCM concerning drug resistance analysis.

## 5   Performance analysis

In order to validate the method discussed in the above section we use six adjusted FCM taken from [18]. Such maps describe the behavior of some HIV-1 mutations related to their resistance to existing antiviral drugs. More precisely, the authors modeled the HIV protease protein as a FCM where each sequence position is taken as a neuron, while another node for the resistance target is also defined. The protease sequence is defined by 99 amino acids, and its main function is related to the maturation of released viral particles by cleaving precursor proteins [14]. With the goal of reducing the number of concepts involved in the final modeling, they use those sequence sites associated with drug resistance. In this topology

neurons are fully connected; whereas a causal connection between each

sequence position and the resistance is established.

It is important to mention that each map denotes the protein behavior for a specific drug: Amprenavir (APV), Indinavir (IDV), Saquinavir (SQV), Nelfinavir (NFV), Ritonavir (RTV) and Atazanavir (ATV). Each drug has associated a high-quality filtered datasets taken from [29] consisting in reported mutations and their resistance value. Despite the prediction abilities of this model [22], the authors cannot ensure the system stability. Following we conduct experiments studying the effects of the algorithm described before over the stability of these maps.

In all the experiments performed in this section we use a constricted PSO Type 1 where $\omega = 0.7298$ and $c_1 = c_2 \approx 1.496$. The parameter settings of the PSO-RSVN algorithm adopted as optimizer is fixed as follows: 40 particles as swarm size, 80 generations, 10 variable neighborhoods, while the allowed number of generations without progress is set to 20. It should be mentioned that parameters concerning the PSO-RSVN algorithm were tuned as reference [17] suggests, since such values proved to be a proper configuration for several problems. Last of all, the number of times used during the inference phase is $T = 100$, whereas the activation degree of each map node is calculated as the normalized contact energy [28] of the corresponding amino acid in the input sequence.

As a first analysis, the stability of the resistance neuron for each inhibitor (using a randomly selected mutation) is measured. Figures 1-3 show the activation degree of the resistance concept ($x$ axis) over the time ($y$ axis) for two situations: the dashed line represents the response using the same function for all neurons, whereas the solid line denotes the output using

the family of sigmoid functions found by the algorithm.

From these simulations we can conclude that the algorithm improves the convergence property of each map. In this simulation, only the resistance node was monitored since it is the decision neuron, allowing predicting whether a new unclassified mutation observed in a patient infected by the virus will be susceptible to the target inhibitor or not.

The reader may observe that the system response changes for next drugs: IDV, RTV and ATV. In such cases the classification rate does not suffer any change since the resistance target for a drug is measured in a certain range instead of using a single value. This range is computed by adopting a pre-defined biological cut-off which allows classifying a new mutation in susceptible or resistant. However, we noticed that some FCM achieved better accuracy, which is an unexpected positive result.

For better understanding of this issue let us analyze the behavior of the selected mutation "FKLDVFMIIVVSVTVNML" for the map IDV. This sequence has high level of resistance for the drug IDV, which means that the higher the activation value of the resistance neuron, the better the accuracy of the model for this instance. But after applying the learning algorithm (see Figure 1a) the FCM is able of computing higher resistance value for the decision neuron. From these results four behaviors were observed: (1a-1b) the convergence rate of stable maps was improved, (2a-2b) cyclic patterns were removed, (3a) the variability response on chaotic systems was reduced, although the final system remains chaotic, and finally (3b) the chaotic behavior was corrected, leading to a perfectly

17

stable system which is the most desirable outcome.

To generalize these results we introduce a second experiment which tries to answer the next question: are the stability improvements statistically significant? By doing so, we calculate for each instance the difference between the original system response, and the average system response after applying the described learning algorithms 10 times. In practice this measure is equivalent to compute the evaluation function (4) for each mutation individually ($K = 1$). Table 1 summarizes the $p$-value resulting from the Wilcoxon signed rank test [12] associated to each inhibitor, and also the behavior of the objective function for all mutations (taking into account the original system, and the best sample solution after applying the proposed learning methodology). This test suggests rejecting the null hypothesis $H_0$ ($p$-value $< 0.05$) in all cases, confirming that there exist significant improvements on the systems stability.

Why it is desirable more stable systems? To answer this question let us analyze the inference process for the map SQV regarding the selected mutation "FKLDVFMIGVPVISTVNML". This sequence has high level of resistance to inhibitor SQV. When the same function is used for all the neurons, the activation level of the decision neuron has lower degree of resistance towards the end (iteration 82), and hence the sequence may be erroneously classified as susceptible. But using the family of sigmoid functions found by the learning algorithm the final neuron is more stable, although the biological system remains chaotic. Actually, the variability that shows this map towards the end will not affect the system response, because in this case the final class value (susceptible or resistant) will be

determined using a range instead of a single value.

Figures 4-5 show the swarm diversity behavior for all inhibitors, during the search phase. The $x$ axis denotes the current generation, whereas the $y$ axis represents the average distance among particles, normalized in the range [0,1]. Observe that the PSO-RSVN algorithm performed identical to the standard constricted PSO model for drugs IDV and RTV, since the swarm reorganization strategy remained inactive. However, for inhibitors ATV, APV, NFV and SQV the algorithm detected potential stagnation or premature convergence states, so the swarm reorganization method was activated, therefore improving the population diversity.

To conclude, we present another experiment illustrating why theoretical results concerning FCM stability (discussed in the Section 3) cannot be used in prediction tasks. With this goal in mind, we estimate the causal matrix of each map by using the weights adaptation procedure discussed in reference [31]. Observe that Theorem 2 is not applicable to the FCM described before because they have not input neurons, so all concepts are updated during the inference procedure. Similarly, we use Theorem 3 for stabilizing FCM-based systems as reference [18] suggests.

Once six maps have been stabilized, all available mutations are classified to evaluate the prediction accuracies. Table 2 shows the confusion matrix achieved for each inhibitor. Based on these simulations we can conclude that, after applying such theoretical results, all FCM are able to converge to a stable fixed-point attractor. However, since this attractor is the same regardless the initial stimulus (i.e. mutation) that they attempt to predict,

the classification quality is considerable affected.

## 6  Conclusions

Stability on sigmoid FCM-based systems plays an important role during the decision-making process. As discussed, stability means to ensure the existence of fixed-points attractors. However, most learning algorithms do not consider this issue in their weights estimation scheme, leading to systems exhibiting cyclic or chaotic patterns. To deal with this drawback some analytical methods have been developed, but they are not adequate for stabilizing FCM used in prediction or decision-making tasks. More exactly, such methods are focused on determining analytical conditions ensuring the existence and uniqueness of a stable point, however, in most cases this attractor is the same for all input sequences.

In this paper an approximate algorithm for enhancing the global stability properties of sigmoid FCM was discussed. The central proposition of this procedure is to estimate, using a Swarm Intelligence approach, a custom sigmoid function for each node instead of using the same transformation function for all neurons. It attempts to efficiently simulate the effects of external factors over the neurons, where the learning goal is the system stability. In order to validate the proposal we used six FCM concerning the Bioinformatics field. From simulations we can conclude that, after applying the learning methodology, adjusted FCM exhibit better stability without varying the system causality. The future work will be focused on studying the convergence on Sigmoid FCM, but now from perspective of

the causal links characterizing the neurons' interaction.

# References

[1]  A.K. Tsadiras, Comparing the inference capabilities of binary, trivalent and sigmoid fuzzy cognitive maps, *Information Science* **178** (2008), 3880–3894.

[2]  B. Kosko, Fuzzy Cognitive Maps, *International Journal of Man-Machine Studies* **24** (1986), 65–75.

[3]  B. Kosko, Fuzzy Engineering, Prentice-Hall, New York, 1997.

[4]  B. Kosko, Global Stability of Generalized Additive Fuzzy Systems, *IEEE Trans. on Systems, Man, and Cybernetics* **23** (1998), 65–75.

[5]  B. Kosko, Hidden patterns in combined and adaptive knowledge networks, *International Journal of Approximate Reasoning* **2** (1988), 377–393.

[6]  B. Kosko, Neural Networks and Fuzzy systems, a dynamic system approach to machine intelligence, Prentice-Hall, Englewood Cliffs, 1992.

[7]  C. Knigh, L. D. Lloyd, A.S. Penn, Linear and Sigmoidal Fuzzy Cognitive Maps: an Analysis of Fixed Points, Applied Soft Computing **15** (2014), 193–202.

[8]  C.D. Stylios and P.P. Groumpos, Mathematical Formulation of Fuzzy Cognitive Maps, in: Proceedings *of the Seventh Mediterranean Conf. Control and Automation*, IEEE XPress, Israel, 1999, pp. 2251–2261.

[9]  E.I. Papageorgiou and A. Kannappan, Fuzzy cognitive map ensemble learning paradigm to solve classification problems: Application to autism identification, Applied Soft Computing **12** (2012), 3798–3809.

[10] E.I. Papageorgiou, Learning Algorithms for Fuzzy Cognitive Maps—A Review Study, *IEEE Transactions on Systems, Man, and Cybernetics* **42** (2012), 150–163.

[11] E.I. Papageorgiou and J.L. Salmeron, A Review of Fuzzy Cognitive Map research at the last decade, *IEEE Transactions on Fuzzy Systems* **21** (2013), 66–79.

[12] F. Wilcoxon, Individual comparisons by ranking methods, Biometrics **1** (1945), 80–93.

[13] G. Nápoles, I. Grau, R. Bello, Constricted Particle Swarm Optimization based Algorithm for Global Optimization, Polibits **46** (2012), 5–12.

[14] G. Nápoles, I. Grau, M. León and R. Grau, Fuzzy Cognitive Maps for Modelling, Predicting and Interpreting HIV Drug Resistance, in: Advances on Soft Computing, I. Batyrshin, ed., Springer-Verlag, Berlin Heidelberg, 2013, pp. 188–199.

[15] G. Nápoles, I. Grau, R. Pérez-García, R. Bello, Learning of Fuzzy Cognitive Maps for simulation and knowledge discovery, in: Studies on Knowledge Discovery, Knowledge Management and Decision Making, R. Bello, ed., Atlantis Press, Paris, 2013, pp. 27–36.

[16] G. Nápoles, R. Bello and K. Vanhoof, Learning Stability Features on Sigmoid Fuzzy Cognitive Maps through a Swarm Intelligence Approach, in: Progress in Pattern Recognition, Image Analysis, Computer Vision and Applications, J. Ruiz-Shulcloper, ed., Springer-Verlag, Berlin Heidelberg, 2013, pp. 270–277.

[17] G. Nápoles, I. Grau, M. Bello, R. Bello, Towards swarm diversity: Random Sampling in Variable Neighborhoods procedure using a Lévy distribution, Computación y Sistemas **18** (2014), 79–95.

[18] G. Nápoles, I. Grau, R. Bello, R. Grau, Two-steps learning of Fuzzy Cognitive Maps for prediction and knowledge discovery on the HIV−1 drug resistance, Expert Systems with Applications **41** (2014), 821–830.

[19] G.A. Papakostas and D.E. Koulouriotis, Classifying Patterns Using Fuzzy Cognitive Maps, in: Fuzzy Cognitive Maps, M. Glykas, ed., Springer-Verlag, Berlin Heidelberg, 2010, pp. 291–306.

[20] G.A. Papakostas, Y.S. Boutalis, D.E. Koulouriotis, B.G. Mertzios , Fuzzy cognitive maps for pattern recognition applications, Int. Journal of Pattern Recognition and Artificial Intelligence **22** (2008), 1461–1468.

[21] H.J. Song, An Extension to Fuzzy Cognitive Maps for Classification and Prediction, IEEE Transactions on Fuzzy Systems **19** (2011), 116–135.

[22] I. Grau, G. Nápoles, and M.M. García, Predicting HIV-1 Protease and Reverse Transcriptase Drug Resistance Using Fuzzy Cognitive Maps, in: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications, J. Ruiz-Shulcloper, ed., Springer-Verlag, Berlin Heidelberg, 2013, pp. 190–197.

[23] J. Kennedy and C.E. Russell, Swarm Intelligence, Morgan Kaufmann Publishers, San Francisco, 2001.

[24] M. León, G. Nápoles, L. Mkrtchyan , B. Depaire, K. Vanhoof, Tackling Travel Behaviour: An approach based on Fuzzy Cognitive Maps, International Journal of Computational Intelligence Systems **6** (2013), 1012–1039.

[25] N. Mateou and A. Andreou, An Evolutionary Methodology to Eliminate the Limit Cycle Phenomenon in FCM-Based Models, in: Information and Communication Technologies, IEEE XPress, Georgia, 2006, pp. 1651–1656.

[26] R. Poli, J. Kennedy and T. Blackwell, Particle Swarm Optimization – An overview, IEEE Transactions on Evolutionary Computation **1** (2007), 37–57.

[27] S. Bueno and J.L. Salmeron, Benchmarking main activation functions in Fuzzy cognitive maps, *Expert Systems with Applications* **36** (2009), 5221–5229.

[28] S. Miyazawa, R.L. Jernigan, Contacts energies Self-Consistent Estimation of Inter-Residue Protein Contact Energies Based on an Equilibrium Mixture Approximation of Residues, PROTEINS: Structure, Function, and Genetics **34** (1999), 49–68.

[29] S.Y. Rhee, M.J. Gonzales, R. Kantor, B.J. Betts, J. Ravela, R. W. Shafer, uman immunodeficiency virus reverse transcriptase and protease sequence database, Nucleic Acids Research **31** (2003), 298–303.

[30] W.J. Grantham and B. Lee, A Chaotic Limit Cycle Paradox, Dynamics and Control **3** (1993), 157–171.

[31] Y. Boutalis, T.L. Kottas and M.C. Christodoulou, Adaptive Estimation of Fuzzy Cognitive Maps With Proven Stability and Parameter Convergence, IEEE Transactions on Fuzzy Systems **17** (2009), 874–889.

[32] Y. Wang, B. Li, T. Weise, J. Wang, B. Yuan, Q. Tian, Self-adaptive learning based particle swarm optimization, Information Sciences **181** (2011), 4515–4538.
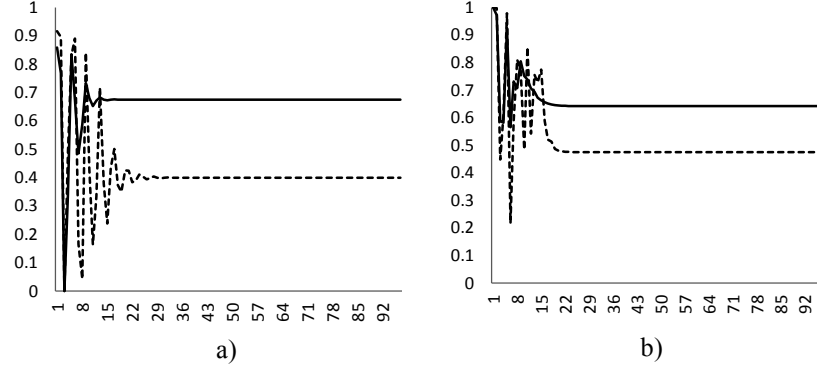
**Figure 1.** Activation value of the resistance neuron for a) drug IDV b) drug RTV where the solid line is the map response using the same function for all neurons, and the dashed line represents the map output using the family of functions estimated by the algorithm. Note that in both cases the convergence speed is improved.
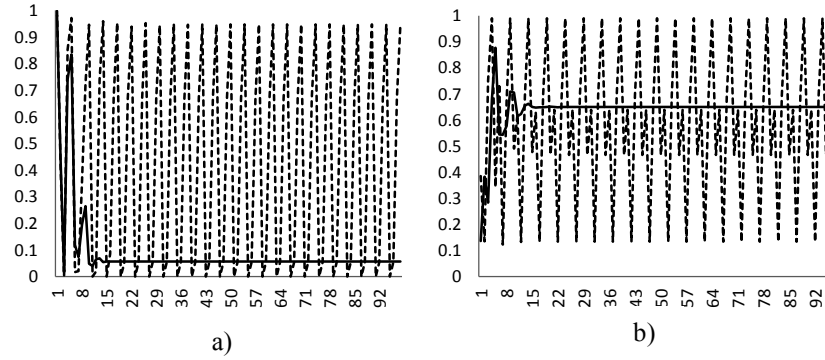


**Figure 2.** Activation value of the resistance neuron for a) drug ATV b) drug APV where the solid line is the map response using the same function for all neurons, and the dashed line is the map output using the family of functions estimated by the algorithm. In such examples the system output shows different behaviors for the same input, but after applying the learning methodology cyclic patterns are removed.
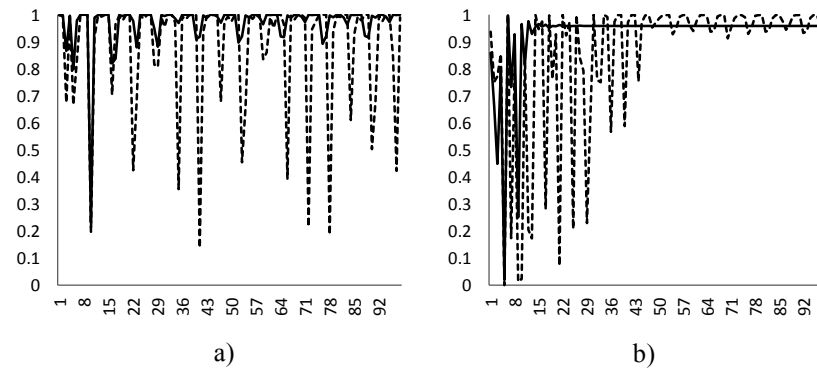


**Figure 3.** Activation value of the resistance concept for a) drug SQV b) drug NFV where the solid line is the map response using the same function for all neurons, and the dashed line represents the output using the family of functions estimated by the algorithm. In the first case the variability of the system response is notably reduced, while in the second case a perfectly stable map is computed, which evidently is the ideal outcome.
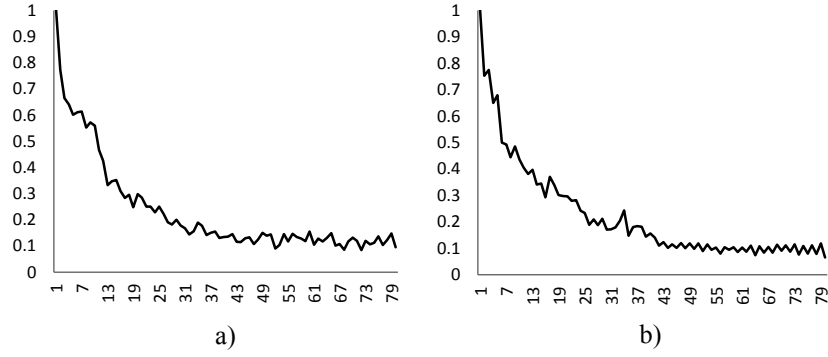
**Figure 4.** Swarm diversıty behavior for a) drug IDV b) drug RTV. In both cases the PSO-RSVN algorithm performs identical to the constricted PSO algorithm since the swarm reorganization procedure remains inactive during the search progress.
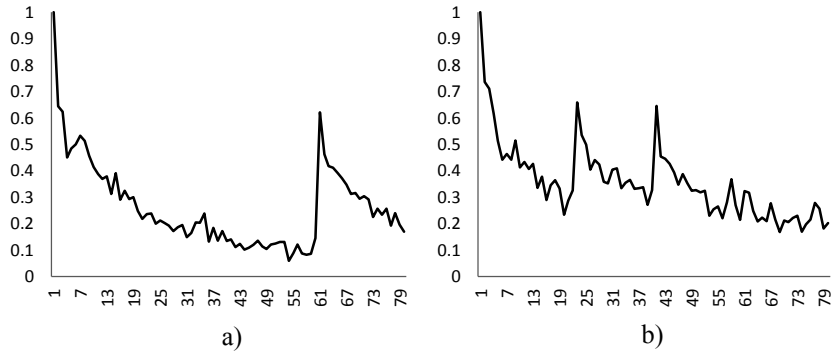


**Figure 5.** Swarm diversıty behavior for a) drug ATV b) drug APV. In the first case the swarm reorganization method is activated once, whereas in the second scenario it is activated twice. As a result, the swarm diversity is significantly improved.
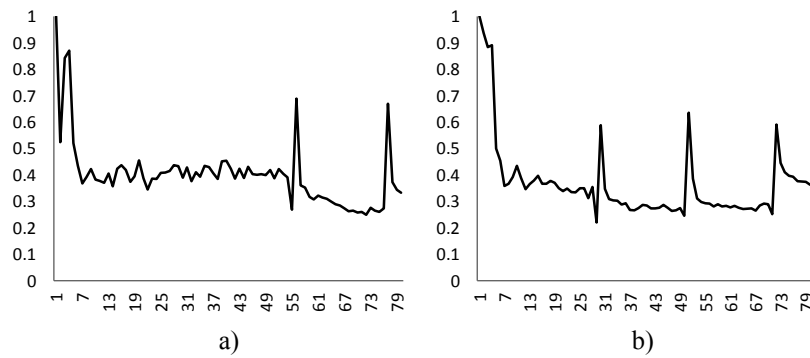


**Figure 6.** Swarm diversity behavior for a) drug SQV b) drug NFV. In the first case the swarm reorganization method is activated twice, whereas in the second example the clearing procedure is activated at iterations 30, 51 and 72.

**Table 1.** Results achieved by the Wilcoxon test (***p*-value**) for each inhibitor. Variability of the system response (***ϕ*-value**) using the original map (**BL**) and using the improved map resulting after applying the discussed learning algorithm (**AL**) for all instances.

| Inhibitor | Mutations | *p*-value | Hypothesis | *ϕ*-value (BL) | *ϕ*-value (AL) |
|---|---|---|---|---|---|
| APV | 96 | 0.000 | Rejected | 39.09962 | 1.98303 |
| ATV | 69 | 0.000 | Rejected | 13.84345 | 2.43406 |
| IDV | 137 | 0.038 | Rejected | 10.58505 | 3.78721 |
| RTV | 151 | 0.024 | Rejected | 19.19583 | 7.52529 |
| NFV | 204 | 0.000 | Rejected | 19.59037 | 2.835233 |
| SQV | 139 | 0.010 | Rejected | 13.92619 | 6.158739 |

**Table 2.** Confusion matrixes achieved for each inhibitor after applying results discussed in Section 3. Since the fixed-point atractor is the same regardless the input sequence, all mutations are always labeled as objects belonging to the same decision class.

| Inhibitor | Class | Using Theorem 1 | | Using Theorem 3 | |
|---|---|---|---|---|---|
| | | Susceptible | Resistant | Susceptible | Resistant |
| APV | Susceptible | 0 | 22 | 0 | 22 |
| | Resistant | 0 | 74 | 0 | 74 |
| ATV | Susceptible | 11 | 0 | 0 | 11 |
| | Resistant | 58 | 0 | 0 | 58 |
| IDV | Susceptible | 0 | 26 | 0 | 26 |
| | Resistant | 0 | 111 | 0 | 111 |
| RTV | Susceptible | 15 | 0 | 0 | 15 |
| | Resistant | 136 | 0 | 0 | 136 |
| NFV | Susceptible | 17 | 0 | 0 | 17 |
| | Resistant | 187 | 0 | 0 | 187 |
| SQV | Susceptible | 24 | 0 | 0 | 24 |
| | Resistant | 115 | 0 | 0 | 115 |