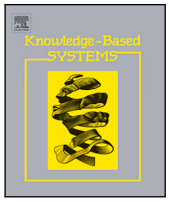
Knowledge-Based Systems 299 (2024) 112089

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



A revised cognitive mapping methodology for modeling and simulation

Gonzalo Nápoles [a], Isel Grau [b], Yamisleydi Salgueiro [c,*]

[a] Department of Cognitive Science & Artificial Intelligence, Tilburg University, The Netherlands
[b] Information Systems Group, Eindhoven University of Technology, The Netherlands
[c] Department of Industrial Engineering, Faculty of Engineering, Universidad de Talca, Campus Curicó, Chile

ARTICLE INFO

Keywords:
Fuzzy cognitive maps
Complex systems
Modeling and simulation
Recurrent neural networks

ABSTRACT

Fuzzy Cognitive Maps (FCMs) hold promise as a mathematical tool for modeling and simulating complex systems due to their transparency, flexibility to operate on prior knowledge structures and recurrent reasoning characteristics. However, they suffer from significant shortcomings that have prevented them from being more widely used. Some of these issues include discrepancies in component interpretation, saturation of neural concepts, arbitrary nonlinearities, and dynamic behaviors that are difficult to align with the problem domain. By integrating theoretical advances with practical needs, this paper proposes a revised modeling and simulation methodology termed "*neural cognitive mapping*" that addresses these issues holistically. Firstly, we redefine concepts' activation values in terms of changes rather than absolute values, ensuring a unified interpretation of the model's components. Secondly, we propose a parameterized activation function, called "*exponential normalized activator*", which allows experts to control the neurons' nonlinearities while avoiding saturation states. Furthermore, we provide a twofold reasoning rule that simultaneously computes the concepts' changes and the amounts of resources attached to problem variables. Thirdly, we introduce a framework for interpreting simulation results across various dynamic behaviors, including scenarios with unique fixed-point attractors. The simulations using both real-world case studies and synthetically generated data illustrate the superiority of our proposal compared with the traditional approach in terms of clarity, usefulness, consistency, and controllability. Moreover, the empirical studies opened new research directions to be explored in future research.

1. Introduction

Fuzzy Cognitive Maps (FCMs) [1] are knowledge-based recurrent neural networks with interpretability features. While mainstream recurrent neural networks (such as the Long Short-term Memory or the Gated Recurrent Unit) are devoted to processing sequences, FCMs represent complex systems in terms of meaningful neural concepts and causal relationships [2]. The key goal of FCM models is to capture hidden patterns that characterize the relationships between the variables describing the problem. In that sense, a recurrent reasoning rule is used to explore the pathways connecting one concept with another, thus resulting in the desired hidden patterns.

Although FCMs have been applied in machine learning settings such as pattern classification [3–5], multi-output regression [6] and time series forecasting [7–9], they were originally conceived as a scenario analysis tool enabling participatory modeling [10–13]. The simulation methodology using FCMs could be summarized as follows. Firstly, the domain experts define the problem variables to be mapped as neural concepts and the causal relationships between them. Secondly, they agree on the activation function responsible for keeping the concepts' activation values in the desired interval. Thirdly, they define what-if scenarios such that the if-part is encoded as the initial activation values of neurons. Finally, the recurrent reasoning process is performed to obtain the what-part of the hypothetical scenario being tested.

Unfortunately, fuzzy cognitive mapping has been surrounded by pitfalls and misconceptions that might have tarnished the potential of a model, as discussed in [11,14]. One of the most controversial (maybe harmless) issues concerns the "fuzzy" aspect of these recurrent neural networks. Osoba and Kosko [15] claimed the following about the fuzzy semantics of FCM models: "*Fuzzy causal edges denote partial causality. All-or-none causality can still occur but only as the endpoints of the spectrum of causal influence. The same holds for the activation of a concept node*". However, Carvalho [16] argued that "*As one can see, FCMs are not true "Fuzzy Systems", since they can be defined by a couple of matrices and can be inferred using iterative standard algebraic operations. The simple fact that a system consists of variables defined with a continuous value ranging from 0 to 1 instead of Boolean values should not be enough to call it "Fuzzy"*".

However, the FCM formalism involves more pressing issues, such as (i) the misalignment between the interpretation of causal weights

* Corresponding author.
E-mail address: ysalgueiro@utalca.cl (Y. Salgueiro).

https://doi.org/10.1016/j.knosys.2024.112089
Received 10 April 2024; Received in revised form 20 May 2024; Accepted 6 June 2024
Available online 8 June 2024

and the interpretation of concepts' activation values, (ii) the arbitrary nonlinearity imposed by the activation function, which often saturates toward the extremes of the activation interval, and (iii) the convergence to unique fixed-point attractors or non-stable states. The literature reports some promising yet isolated attempts devoted to addressing these issues.

For example, Mpelogianni et al. [17] and Vergini et al. [18] proposed a reasoning rule that operates with changes in the concepts' activation values instead of using the concepts' absolute values. In both cases, the authors assumed the model is no longer close since it receives external inputs. Koutsellis et al. [19] addressed the ambiguity in the outcomes of FCM models equipped with sigmoid and hyperbolic tangent activation functions. The rationale of their proposal consists of estimating bounds for the slope of these non-linear functions such that the concepts' values fall in a so-called "almost-linear" region. The obtained values are then normalized to ensure the concepts' activation values expand to the desired activation interval. Moreover, they performed a convergence analysis following the theoretical results by Knight et al. [20] devoted to ensuring the convergence to unique-fixed point attractors. Following an opposite path, Nápoles et al. [21] proposed a quasi-nonlinear reasoning rule coupled with a rescaled activation function that ensures that the model will never converge to a unique fixed-point attractor, which is desirable in most scenario analysis tasks.

In this paper, we take inspiration from these solutions and present a revised cognitive mapping methodology for modeling and simulation. Our proposal, termed "*neural cognitive mapping*", addresses all issues mentioned above to a large extent. Firstly, we redefine the meaning of concepts' activation values to express changes rather than absolute values and remove the misalignment between the interpretation of the model's components. This modification required integrating the reasoning rule expressed in terms of neural concepts with another equation expressed in terms of problem variables, thus aligning the mathematical model with the physical system. To our knowledge, a few attempts have been reported to map the simulation results back to the problem domain. Secondly, we introduce a parameterized activation function called "*exponential normalized function*" that allows the experts to control the model's nonlinearity, spanning from quasi-linear to exponential. Thirdly, we propose a detailed framework for interpreting the simulation results in the presence of different dynamic behaviors, including situations where the fixed-point attractor is unique.

The remainder of this paper is as follows. Section 2 goes over the classic FCM formalism for modeling and simulation, while Section 3 analyzes relevant FCM issues that motivate our research. Section 4 presents the neural cognitive mapping methodology, which is the main contribution of our paper. Section 5 conducts an empirical analysis using three case studies and Section 6 provides concluding remarks and future research endeavors.

2. Fuzzy cognitive maps

FCMs are graph-theoretic recurrent neural systems for modeling and simulating complex systems that involve interconnected variables and feedback loops [22]. Every concept C_i in the network is modeled as a neural processing entity described by its activation value in the current tth iteration. The activation vector $\mathbf{A}^{(t)} = (a_1^{(t)}, \ldots, a_i^{(t)}, \ldots, a_N^{(t)})$ characterizes the system's state in the current iteration $t \in \{1, 2, \ldots, T\}$, while the initial activation vector $\mathbf{A}^{(0)} = (a_1^{(0)}, \ldots, a_i^{(0)}, \ldots, a_N^{(0)})$ encodes the scenario to be simulated [23]. The interactions between concepts are defined through a weight matrix $\mathbf{W}_{N\times N}$, where N represents the number of concepts depicted as nodes in the cognitive digraph. Each edge is associated with a weight $w_{ij} \in [-1, 1]$ that quantifies the influence that the ith concept exerts on the jth concept.

Eq. (1) formalizes the recurrent reasoning rule used to compute the activation value of the ith concept in each iteration given an initial activation vector,

$$a_i^{(t+1)} = f\left(\sum_{j=1}^{N} a_j^{(t)} w_{ji}\right) \tag{1}$$

such that $f(\cdot)$ is the activation function [24] that ensures the concepts' activation values always lie in a desired interval, typically $[0, 1]$ or $[-1, 1]$. The activation functions addressed in this paper are described below such that $\bar{\mathbf{A}}^{(t)} = (\bar{a}_1^{(t)}, \ldots, \bar{a}_i^{(t)}, \ldots, \bar{a}_N^{(t)})$ represents the raw activation vector and $\bar{a}_i^{(t)}$ is the raw activation value of the ith concept in the current iteration.

Eq. (2) shows the generalized sigmoid activator, which produces values in the $(0, 1)$ interval,

$$f_s\left(\bar{a}_i^{(t+1)}\right) = \frac{1}{1 + e^{-\lambda\left(\bar{a}_i^{(t+1)} - h\right)}} \tag{2}$$

where $\lambda > 0$ and $h \in \mathbb{R}$ are parameters controlling the function slope and offset, respectively.

Eq. (3) shows the saturation activator, which produces values in the $[-1, 1]$ interval,

$$f_p\left(\bar{a}_i^{(t+1)}\right) = \begin{cases} -1, & \text{if } \bar{a}_i^{(t+1)} \leq -1 \\ \bar{a}_i^{(t+1)}, & \text{if } -1 < \bar{a}_i^{(t+1)} < 1 \\ 1, & \text{if } \bar{a}_i^{(t+1)} \geq 1. \end{cases} \tag{3}$$

Eq. (4) depicts the rescaled activator, which produces values in the $[-1, 1]$ interval,

$$f_r\left(\bar{a}_i^{(t+1)}\right) = \begin{cases} \frac{\bar{a}_i^{(t+1)}}{||\bar{\mathbf{A}}^{(t+1)}||_2} & \text{if } \bar{\mathbf{A}}^{(t+1)} \neq \vec{0} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Eq. (5) depicts the hyperbolic tangent activator, which outputs values in the $(-1, 1)$ interval,

$$f_h\left(\bar{a}_i^{(t+1)}\right) = \frac{e^{\bar{a}_i^{(t+1)}} - e^{-\bar{a}_i^{(t+1)}}}{e^{\bar{a}_i^{(t+1)}} + e^{-\bar{a}_i^{(t+1)}}}. \tag{5}$$

The recurrent reasoning rule in Eq. (1) will stop when either (i) the model converges to a fixed-point attractor, or (ii) a maximal number of iterations T is reached. Overall, we have three possible states:

- **Fixed point** ($\exists t_\alpha \in \{1, \ldots, (T-1)\} : a_i^{(t+1)} = a_i^{(t)}, \forall i, \forall t \geq t_\alpha$): the FCM produces the same state vector after t_α, thus $a_i^{(t_\alpha)} = a_i^{(t_\alpha+1)} = a_i^{(t_\alpha+2)} = \cdots = a_i^{(T)}$. If the fixed point is unique, the FCM model will produce invariant states (i.e., the same state vector regardless of the initial conditions).
- **Limit cycle** ($\exists t_\alpha, P \in \{1, \ldots, (T-1)\} : a_i^{(t+P)} = a_i^{(t)}, \forall i, \forall t \geq t_\alpha$): the FCM produces the same state vector periodically with period P, thus $a_i^{(t_\alpha)} = a_i^{(t_\alpha+P)} = \cdots = a_i^{(t_\alpha+jP)}$, where $t_\alpha + jP \leq T$.
- **Chaos**: the FCM produces different state vectors for successive iterations with no clear pattern.

3. Motivation and challenges

In this section, we discuss relevant issues that hinder the practical usability of FCMs in modeling and simulation settings. These limitations render a potentially powerful technique obscure and fragile, often leading to misleading interpretations of the simulation results. Therefore, these issues represent theoretical challenges that motivated the model proposed in the following section.

Issue 1. The concepts' activation values in a given iteration represent absolute quantities rather than changes expressed as increments or decrements.

In order to understand the full extent of this widespread misconception, we first need to discuss how causal weights and activation

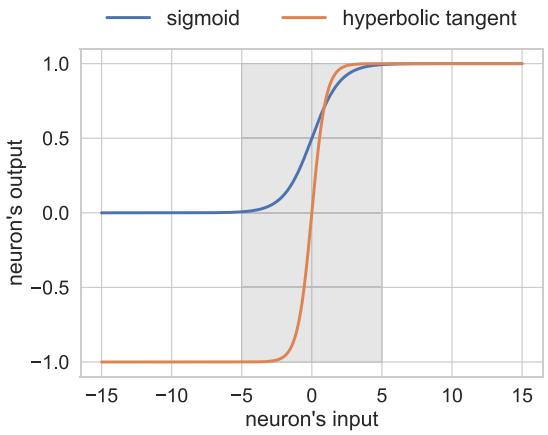


Fig. 1. Saturation issue when using the sigmoid and hyperbolic tangent functions. Significant changes in the neuron's input outside the gray region lead to nearly the same neuron's output.

values are understood in the FCM context. Let C_i and C_j denote two neural concepts using the activation function $f(x)$ to keep the concepts' activation values in the desired interval. According to a significant body of literature, the weight $w_{ij} \neq 0$ connecting C_i and C_j can be interpreted as described below.

- For $w_{ij} > 0$: Increasing (or decreasing) the activation value of C_i leads to an increase (or a decrease) in the activation value of C_j.
- For $w_{ij} < 0$: Increasing (or decreasing) the activation value of C_i leads to a decrease (or an increase) in the activation value of C_j.

While this interpretation aligns reasonably well with the semantics of dynamic systems, where simulations are often expressed in terms of changes in the variables, it seems to clash with the reasoning rule in Eq. (1). Such discordance stems from the fact that the system's states represent the absolute values of concepts in a given iteration rather than depicting the changes they underwent from the previous iteration to the current one.

Issue 2. Most activation functions quickly saturate toward the extremes of the activation interval, preventing the neuron from sensing changes.

FCM-based models are recurrent neural networks that need bounded activation functions to prevent neurons' activation values from exploding. However, this comes at the cost of producing saturated states where the quite dissimilar neuron inputs are associated with practically the same output. Fig. 1 illustrates this issue for the sigmoid and hyperbolic tangent function where values outside the gray area tend to saturate the neural concept. In the case of the sigmoid function, $f(x) \to 0$ when $x \to -\infty$ while in the hyperbolic tangent function, $f(x) \to -1$ when $x \to -\infty$, and in both cases $f(x) \to 1$ when $x \to +\infty$. However, in reality, these functions produce values close to the extremes of the activation space when inputting raw activation values with relatively small absolute values.

Another issue arises when using the sigmoid function, which is widely employed as an activation function in the literature. It yields a value of 0.5 when evaluated at $x = 0$, potentially leading to unexpected behaviors during simulations. For instance, a sigmoid neuron receiving no input from connected neurons may suddenly activate, thereby transmitting information to other neurons.

Issue 3. The model's dynamics (and consequently the simulation results) can change significantly depending on the adopted activation function.

Fig. 2 depicts an FCM model for a simplified food chain involving three concepts. The foundational knowledge encoded by this cognitive neural structure is straightforward, with components having clear

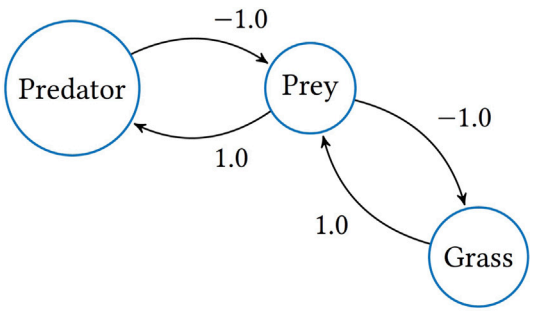


Fig. 2. Simplified food chain as an FCM model involving three neural concepts (predator, prey, and grass).

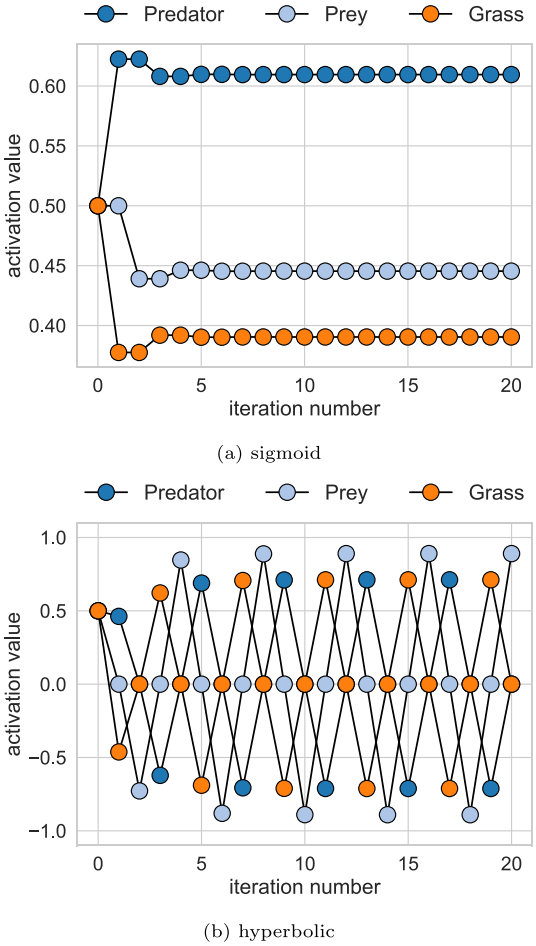


Fig. 3. Simulation results for the predator–prey-grass problem after activating all concepts with 0.5 while using (a) the sigmoid function and (b) the hyperbolic tangent function.

meanings for the problem domain. For example, predators flourish in the presence of plentiful prey, while prey face challenges from predators but benefit from abundant grass.

Let us initialize all concepts with a medium activation value ($a_i^{(0)} = 0.5$) and then perform the reasoning process using the sigmoid and hyperbolic tangent activation functions. Fig. 3 depicts the simulation results after performing 20 iterations. In this experiment, the sigmoid function leads to a fixed-point attractor while the hyperbolic tangent function yields cyclic patterns.

At first glance, the sigmoid FCM model appears to be the right choice since it exhibits stable behavior, unlike the hyperbolic tangent model, which fails to converge. However, in the sigmoid FCM model,

the fixed-point attractor is unique, meaning that the system will produce the same output regardless of the initial activation vector. The ramification of this phenomenon is that the model is not useful for performing what-if simulations.

In the hyperbolic tangent model, we observe a lagged decrease in both prey and grass as the number of predators increases, which is a logical result for the former but seemingly illogical for the latter. Conversely, as the number of predators decreases, there is an increase in both prey and grass, which is a sensible outcome for the former but apparently perplexing for the latter. However, reality unveils an intermediary component: the prey. In essence, the dynamics of this complex system unfold as follows: the presence of sufficient grass attracts prey to the area, consequently attracting predators.

From this toy example, some relevant conclusions can be formalized. Firstly, the same network employing different activation functions should be regarded as different simulation models, potentially resulting in varied outcomes. Secondly, models that fail to converge might depict the true semantics of a dynamic system in contrast to models that converge to unique fixed points. Thirdly, the choice of activation function significantly influences the model's nonlinearity and should be tailored to the problem domain. Furthermore, the activation function should provide the flexibility to represent a variety of nonlinear behaviors, spanning from linear to exponential.

4. Neural cognitive mapping

In this section, we introduce a revised cognitive mapping methodology that addresses the FCM issues discussed in the previous section to a large extent. Additionally, we will provide useful recommendations and best practices for the knowledge acquisition phase.

4.1. Interpretation of concepts' activation values

To address the first issue discussed in the previous section, we will redefine the interpretation of the concepts' activation values describing the system in each iteration to express changes instead of absolute values.

That can be done by replacing the term $a_i^{(t)}$ in Eq. (1) by the term $\Delta a_i^{(t)}$, which denotes the change in the ith concept's activation value in the tth iteration. Therefore, the change vector $\Delta\mathbf{A}^{(t)} = (\Delta a_1^{(t)}, \ldots, \Delta a_i^{(t)}, \ldots, \Delta a_N^{(t)})$ will describe the system in the t-the iteration. In line with the original methodology, the domain expert is required to encode the what-if scenario being simulated in terms of the initial changes induced to the system. In practice, this knowledge is formalized through the initial change vector $\Delta\mathbf{A}^{(0)} = (\Delta a_1^{(0)}, \ldots, \Delta a_i^{(0)}, \ldots, \Delta a_N^{(0)})$, which is employed to start the recurrent reasoning process.

A relevant aspect related to the redefinition of the concepts' activation values concerns the information provided by experts during the knowledge acquisition process. The weight matrix exemplifies this knowledge, requiring experts to identify pairs of related concepts and delineate their interactions, either numerically or symbolically. Such a delineation involves defining the direction and intensity of the causal relationships. These pieces of knowledge can be captured using the interface displayed in Fig. 4 where the expert indicated that an increase on C_i will cause a decrease on C_j with an intensity of 2.5/10. Therefore, we can conclude that $w_{ij} = -0.25$ since the relationship between concepts C_i and C_j is deemed inverse.

After the experts have defined the weight matrix, it is considered good practice to add self-loops with $w_{ii} = 1.0$ to those concepts having no incoming connections. The rationale behind this practice is that such concepts would otherwise lose their activation value in the first iteration, thus contributing nothing to the reasoning process. However, if a self-loop with $w_{ii} = 1.0$ is added, the concept will maintain the same activation value through successive iterations. Finally, isolated neural concepts must be removed from the cognitive network.

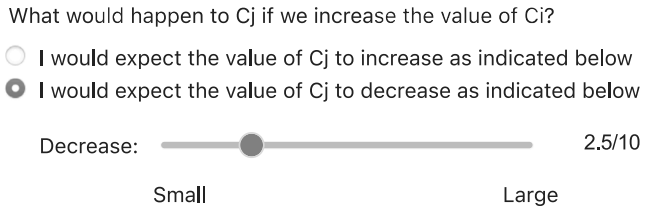


Fig. 4. Interface for domain experts to characterize the relationship between two neural concepts.

Another aspect that demands attention is the meaning of iterations in terms of time units [14]. We claim that domain experts must also provide such information during the knowledge acquisition phase. In practice, it could be formalized in a variety of ways (e.g., at the concept level or the model level). For example, domain experts can be presented with the following question: "How long does it take for concept C_i to exert an effect on concept C_j?". For simplicity, the proposed methodology will assume a synchronous approach where all neural concepts are associated with the same time unit. For example, if the domain experts decide that the model is expressed in hours, then we will need to perform 24 iterations to investigate the system's state after reaching one day.

4.2. Exponential normalized reasoning

Next, we will address the second issue discussed in the previous section, which refers to the saturation issues induced by mainstream activation functions such as the saturation, sigmoid, and hyperbolic tangent functions. Moreover, we should provide the experts with the flexibility to configure the system's nonlinearity.

More explicitly, the *exponential normalized reasoning* presented below fulfills three pivotal properties. Firstly, it accounts for the activation values of negative and positive concepts, denoting negative and positive changes, respectively. Secondly, it addresses the saturation issues where neural concepts are assigned values close to the extremes of the activation space for a wide variety of incoming information flows. Thirdly, it provides modelers with the flexibility to control the nonlinearity of simulations, varying from linear to exponential. Eq. (6) displays the reasoning rule that fulfills these properties,

$$\Delta a_i^{(t+1)} = S\left(\Delta\bar{a}_i^{(t+1)}\right)\left(\left|\Delta\bar{a}_i^{(t+1)}\right| \Big/ \sum_{j=1}^{N}|w_{ji}|\right)^{\beta_i} \tag{6}$$

such that

$$\Delta\bar{a}_i^{(t+1)} = \sum_{j=1}^{N}\Delta a_j^{(t)} w_{ji} \tag{7}$$

where $S(\cdot)$ is the sign function and $\beta_i > 0$ is an important parameter controlling the neuron's nonlinearity.

In this reasoning rule, the term $\sum_{j=1}^{N}|w_{ji}| \leq N$ represents an upper bound for the neuron's capacity [25], which is defined as the sum of the absolute weights associated with the neuron's incoming connections. Therefore, it holds that $|\Delta\bar{a}_i^{(t+1)}| \leq \sum_{j=1}^{N}|w_{ji}|$, which provides a more realistic normalization than $|\Delta\bar{a}_i^{(t+1)}|/N$. The normalization operation ensures that the neuron's absolute values lie in the $[0, 1]$ interval and that it will reach its maximum absolute value when $|\Delta\bar{a}_i^{(t+1)}| = \sum_{j=1}^{N}|w_{ji}|$, solving the saturation issues of mainstream activation functions. After the normalization operation, we perform an exponentiation step to either reduce or expand the neuron's absolute activation values, thus equipping the model with a tunable nonlinearity component controlled by the β_i parameter. Finally, we apply the sign function on the neuron's raw activation value (before the normalization and exponentiation operations) to determine whether the concept's change is positive or negative.

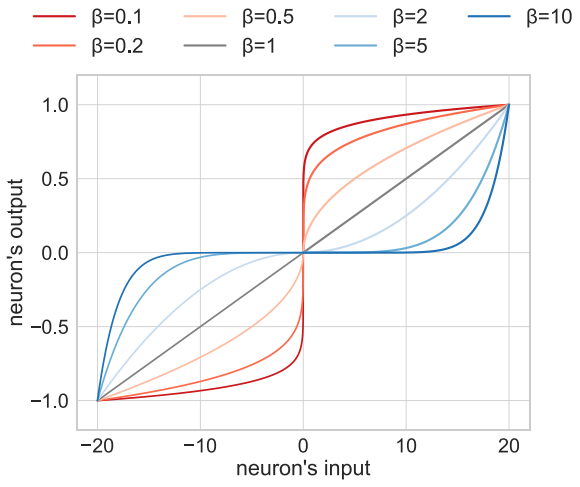


Fig. 5. Exponential normalized activation function.

Let us further discuss the role of the β_i parameter on the proposed exponential normalized function. Fig. 5 illustrates the behavior of the proposed reasoning rule using the exponential normalized activation function for a neural concept that receives information from 20 other concepts. When $\beta_i = 1$, the reasoning rule behaves linearly within the $[-20, 20]$ interval. If $\beta_i > 1$, the neuron's input values contract in comparison to the identity line. This configuration is recommended for problems characterized by densely connected cognitive networks. If $0 < \beta_i < 1$, the neuron's input values dilate. This setup is advisable for problems described by sparsely connected cognitive networks. However, the nonlinearity of each neural concept should be defined by the domain experts based on the particularities of the problem being modeled.

As a final step, we need a separate equation to map the changes in the neural concepts computed by Eq. (1) to absolute values for the problem variables that describe the system. Let $\mathbf{X}^{(0)} = (x_1^{(0)}, \ldots, x_i^{(0)}, \ldots, x_N^{(0)})$ denote a set of resources where $x_i^{(0)}$ gives the initial amount of resources for the ith concept. For example, the initial amount of resources associated with C_i = "Predators" could be quantified as the number of big cats observed in the area. We can compute the resource vector $\mathbf{X}^{(t)} = (x_1^{(t)}, \ldots, x_i^{(t)}, \ldots, x_N^{(t)})$ in the tth iteration as follows:

$$x_i^{(t+1)} = x_i^{(t)} + \Delta a_i^{(t+1)} \theta_i^{(t+1)} \tag{8}$$

such that $\theta_i^{(t+1)}$ represents the maximum amount of resources available for the ith problem variable in the $(t+1)$-th iteration. Since our simulation model is symmetric, it must be assumed that $\theta_i^{(t+1)}$ will be the maximum amount of resources to be earned or lost.

In this model, the amounts of resources can be zero, positive, or even negative, and their interpretation depends on the problem domain. As a general guideline, a zero value indicates that the available resources in the previous iteration were fully consumed, while negative and positive values signify a deficit or increase, respectively. For example, a company may incur losses before achieving a positive balance, so simulations could help determine whether the losses will persist over an extended period or how much time the company would need to generate profits. Caution is advised when interpreting negative values since their meaning is problem-dependent.

4.3. Interpreting the observed dynamic behaviors

To address the third issue, we will introduce a flow diagram that indicates how to interpret the simulation results based on the system's convergence status. Such a roadmap is based on the fact that FCMs can exhibit a spectrum of behaviors, ranging from stability to cycles and chaos, each offering distinct interpretations.

Fig. 6 shows the proposed flow diagram that operates on the simulation trace for the ith neural concept, which is given by $\Delta\mathbf{A}_i = (\Delta a_i^{(0)}, \ldots, \Delta a_i^{(t)}, \ldots, \Delta a_i^{(T)})$. However, we often need to analyze all traces simultaneously to have a holistic picture of the simulation.

When the model converges to an equilibrium point, we must investigate whether the fixed point is unique. In that case, the model will not depend on the initial conditions since it will always converge to the same final state regardless of the initial conditions. Therefore, analyzing the effect of varying the initial conditions on the system's final state has limited usability. However, this does not mean the model will always follow the same "path" to the unique fixed point [3,26]. Fig. 7 portrays a fictional example inspired by the case study in [27] where the same concept is activated with different initial conditions. It can be observed that both settings lead to the same output, but the first setting $\Delta a_i^{(0)} = 0.26$ makes the model converge faster to the solution when compared with the second setting $\Delta a_i^{(0)} = 0.16$. The authors in [27] suggested using this information for decision-making purposes, although they did not provide a specific method.

In this paper, we propose a convergence-aware measure termed *rate of convergence* that quantifies the deviation of the concept's activation values from the fixed point. It allows experts to analyze the effects of different initial conditions on the system's behavior, even when the outcome remains the same. The rate of convergence measure can be computed as formalized below:

$$\Omega(\Delta\mathbf{A}_i) = \sum_{t=1}^{T} \left| \Delta a_i^{(t)} - \Delta a_i^{(T)} \right|. \tag{9}$$

Notice that Eq. (6) converging to a unique fixed point does not necessarily translate into Eq. (8) producing the same outcome for all initial conditions, which aligns with the rationale of the proposed rate of convergence metric. This behavior will be illustrated during the empirical study conducted in the next section.

If the model governed by Eq. (6) converges to different fixed points, we can conclude that the simulations depend on the initial conditions. Aiming to interpret the results, we must inspect the model's final states. If $\Delta a_i^{(T)} > 0$, then the amounts of resources for the ith variable (as computed using Eq. (8)) will increase permanently. This happens because the model indicates that there will be constant changes in that variable. The amounts of resources for the ith variable will decrease permanently when $\Delta a_i^{(T)} < 0$. In contrast, $\Delta a_i^{(T)} = 0$ indicates an equilibrium point where the amounts of resources for the ith variable will no longer change.

Another possible behavior is that the simulation model produces cycles that depend on the initial conditions. However, cyclic patterns in Eq. (6) might not translate into cycles in Eq. (8), which is an appealing property of our methodology. If the cycle is positive ($\exists t_\alpha, P \in \{1, \ldots, (T-1)\} : \Delta a_i^{(t+P)} = \Delta a_i^{(t)}, \Delta a_i^{(t)} > 0, \forall t \geq t_\alpha$), then the amounts of resources for the ith problem variable will increase at uneven rates. If the cycle is negative ($\exists t_\alpha, P \in \{1, \ldots, (T-1)\} : \Delta a_i^{(t+P)} = \Delta a_i^{(t)}, \Delta a_i^{(t)} < 0, \forall t \geq t_\alpha$), then the amounts of resources for the ith problem variable will decrease unevenly. If the states of the observed cycles have different signs, then the amounts of resources will show cyclic patterns.

The last possible behavior is chaos. In these cases, we could conclude that the initial conditions and the weight matrix defining the interaction between the concepts do not allow for consistent decision-making. However, it might be the case that some initial conditions cause more chaos than others, so we could resort to measuring the standard deviation of the simulation trace $\Delta\mathbf{A}_i$ to quantify the chaos degree. The reader can notice that this measure follows a similar rationale as the rate of convergence metric in Eq. (9), where the hypothetical convergence point is the mean of the simulation trace.

Fig. 6. Flow diagram indicating how to interpret the simulation results and different convergence behaviors.

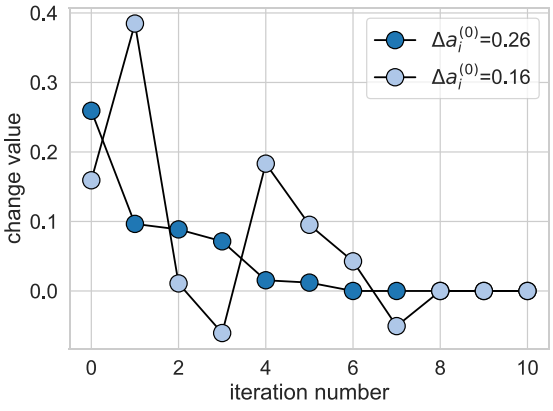


Fig. 7. Convergence to a unique fixed-point attractor using Eq. (6) for two different initial conditions.

5. Numerical simulations

In this section, we will conduct an empirical study using both real-world case studies and synthetically generated cognitive networks that will help illustrate the superiority of the proposed methodology.

5.1. Simulations using real-world case studies

In this section, we will compare our simulation model in Eq. (6) with the model $\Delta a_i^{(t+1)} = f(\sum_{j=1}^{N} \Delta a_j^{(t)} w_{ji})$ using the real-world case studies and the activation functions discussed in Section 2. The sigmoid function is excluded from our study because it does not allow for negative changes in the concepts' activation values. In all cases, we will rely on Eq. (8) to compute the amounts of resources for the problem variables. This empirical analysis is devoted to illustrating the inner workings of our model under the influence of different activation functions.

The first case study pertains to civil engineering and examines the implications of urban development on public health within a city. This

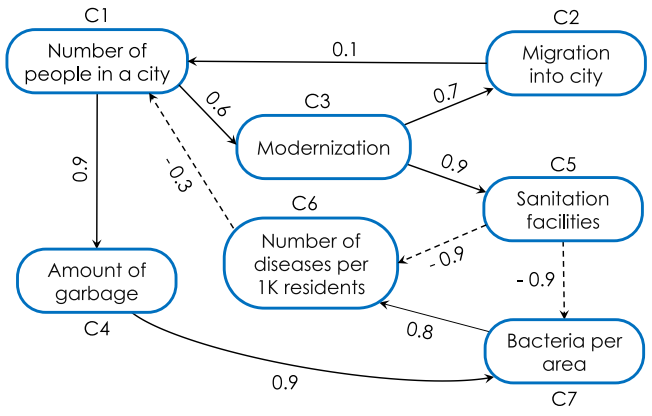


Fig. 8. FCM modeling the "public health" case study.

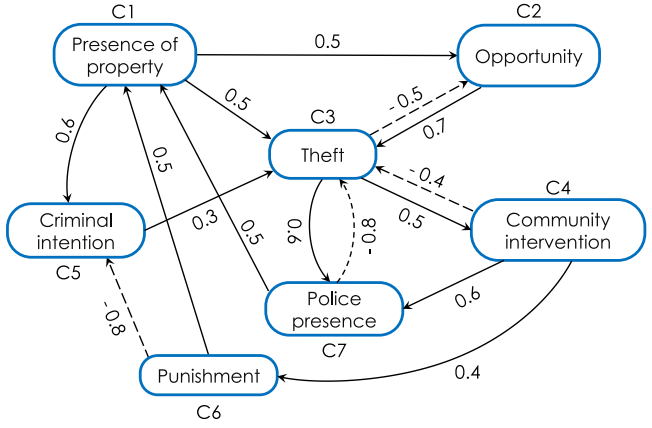


Fig. 9. FCM modeling the "crime and punishment" case study.

model was utilized in a prior study [28] to contrast the inference capabilities across binary, trivalent, and sigmoid FCM models. The concepts describing the system include city inhabitants (C_1), migration inflows (C_2), urbanization (C_3), waste generation (C_4), sanitation infrastructure (C_5), the incidence of diseases per 1000 residents (C_6), and bacterial density per area (C_7). Fig. 8 visualizes this cognitive network, with positive relationships denoted by solid lines and negative relationships depicted by dashed lines.

The second case study used in our simulations, depicted in Fig. 9, concerns the "crime and punishment" model, as outlined in [29]. It depicts the interplay between various factors influencing crime and the criminal justice system. The model incorporates concepts such as the presence of property (C_1), opportunity (C_2), theft (C_3), community intervention (C_4), criminal intention (C_5), punishment (C_6), and police presence (C_7).

Fig. 10 portrays the third case study, which concerns a complex system representing the car industry, sourced from [30]. The neural concepts describing this system include high profits (C_1), customer satisfaction (C_2), high sales (C_3), union raises (C_4), safer cars (C_5), foreign competition (C_6), and lower prices (C_7).

The reader can notice that these FCM models involve non-trivial pathways, even when they do not involve a large number of concepts or causal relationships. Due to the lack of domain knowledge, we will generate random values for the initial conditions (the initial changes and the initial amounts of resources). Specifically, the initial changes are randomly generated within the $[0, 1]$ interval, while the initial amounts of resources are uniformly distributed within the $[50, 100]$ interval. For reproducibility, we used the same Python seed in all experiments. It is important to emphasize that these values define the

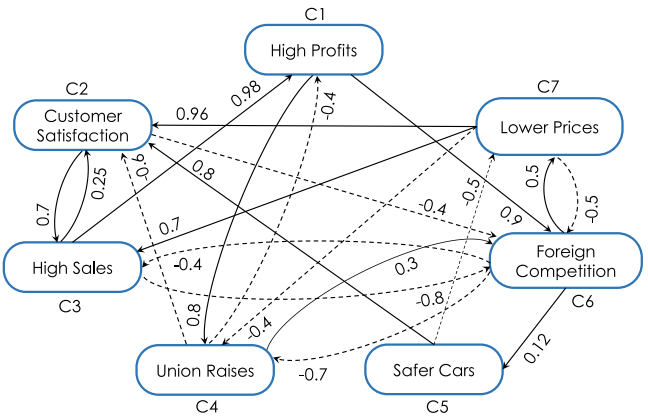


Fig. 10. FCM modeling the "car industry" case study.

simulation being performed and should be determined by a domain expert when applying the model.

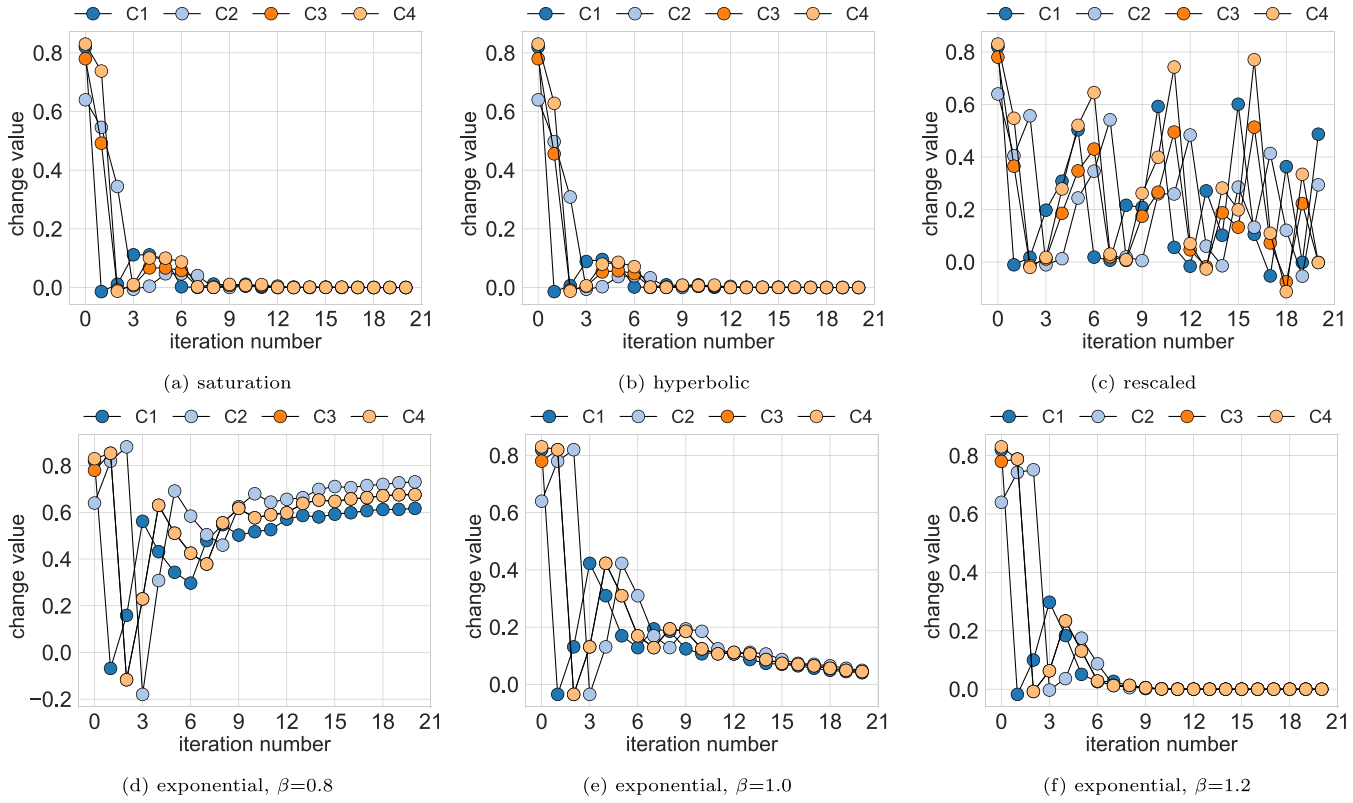
Fig. 11 and Fig. 12 depict the change values and the amounts of resources, respectively, for the "public health" case study. For simplicity, we will visualize the results concerning the first four neural concepts. In these simulations, we arbitrarily assumed that the maximal amounts of resources are determined by the initial amounts of resources associated with the variables describing the problem. This means the resources earned or lost in a single iteration will not exceed the initial quantities.

The results reveal that the saturation, hyperbolic tangent, and exponential ($\beta = 1.2$) models lead to an equilibrium point where the amounts of resources no longer change. The exponential ($\beta = 1.0$) model exhibits slower convergence compared to the saturation, hyperbolic tangent, and exponential ($\beta = 1.2$) models, but it will produce the same behavior if more iterations are performed. However, the fact that these models display similar convergence behaviors does not imply that they yield the same results, as indicated by the amounts of resources. The reader might find it surprising that the hyperbolic tangent model behaves similarly to the quasi-linear model (when using the exponential function and $\beta = 1.0$, this activator behaves linearly within the $[-\sum_{j=1}^{N}|w_{ji}|, \sum_{j=1}^{N}|w_{ji}|]$ interval). This behavior is explained in Mpelogianni et al. [17], where the authors noted that raw activation values with relatively small absolute values fall within a region where the hyperbolic function resembles a linear activator. Fig. 13 shows this phenomenon, where the gray area denotes the region containing the neurons' inputs for the randomly generated initial conditions.

The exponential model ($\beta = 0.8$) converges to a fixed point where the final states of neurons are positive values. This suggests that the quantities of resources produced by Eq. (8) will continue to increase steadily. A similar pattern is observed with the rescaled activator, which generates quasi-positive cycles (where most changes are positive). As a result, it produces occasional decreases in the amounts of resources when the cycles transition through a negative state. Both scenarios are elucidated in the diagram illustrated in Fig. 6, which provides guidelines for interpreting the simulation results.

Fig. 14 and Fig. 15 depict the change values and the amounts of resources, respectively, for the 'crime and punishment case study. Similarly to the previous case study, we visualize the simulation traces concerning the first four concepts for the sake of simplicity.

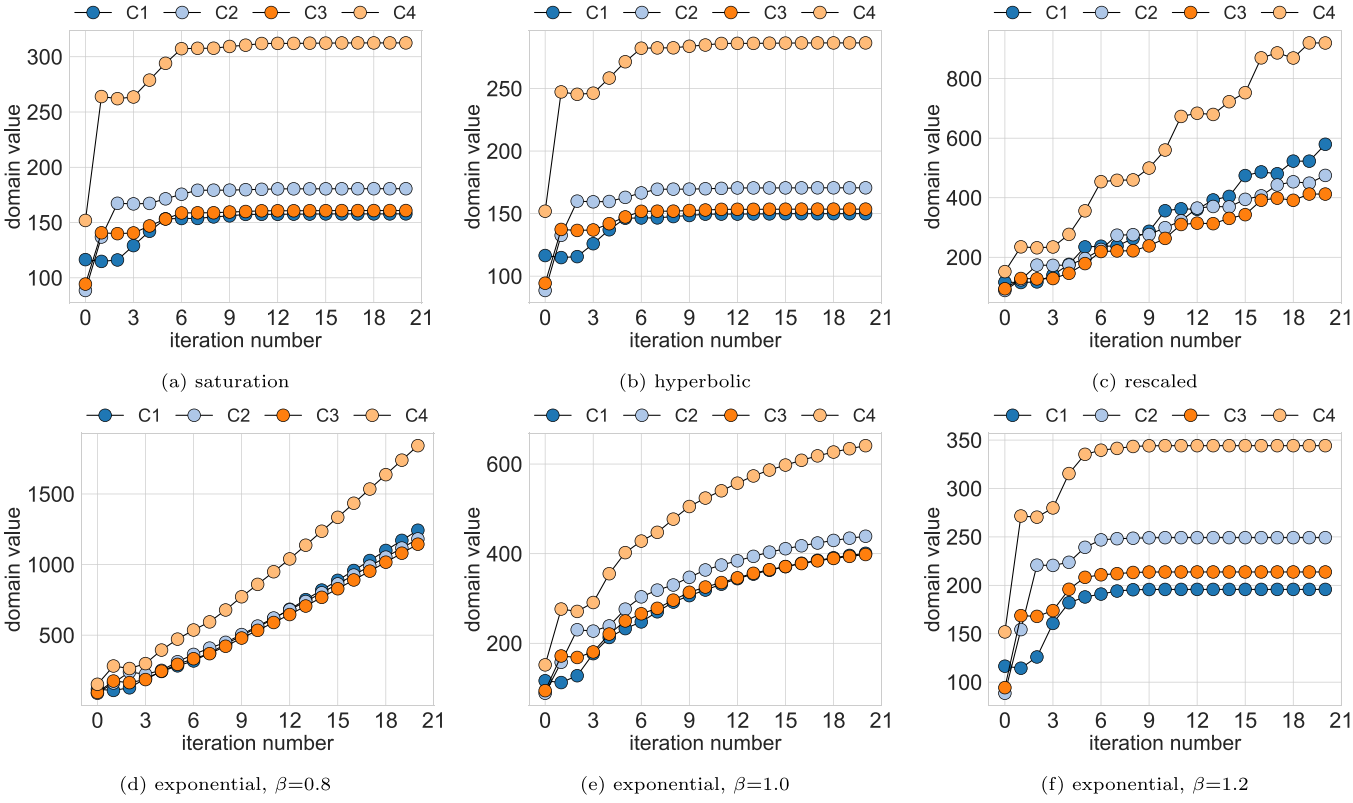
The results indicate that the saturation, hyperbolic tangent, rescaled, and exponential ($\beta = 0.8$) models produce cyclic patterns with the amounts of resources ranging from 50 to 250. Although these cycles involve both negative and positive states, no deficit in the resources is observed after performing 20 iterations. In contrast, the exponential models using $\beta = 1.0$ and $\beta = 1.2$ converge to a fixed-point attractor where the concepts' final states are zero. Therefore, the amounts of resources no longer change, which means that the forces driving change



(a) saturation (b) hyperbolic (c) rescaled

(d) exponential, β=0.8 (e) exponential, β=1.0 (f) exponential, β=1.2

Fig. 11. Change values outputted by the FCM modeling the "public health" case study when using different activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized activators).



(a) saturation (b) hyperbolic (c) rescaled

(d) exponential, β=0.8 (e) exponential, β=1.0 (f) exponential, β=1.2

Fig. 12. Amounts of resources outputted by the FCM modeling the "public health" case study when using different activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized activators).

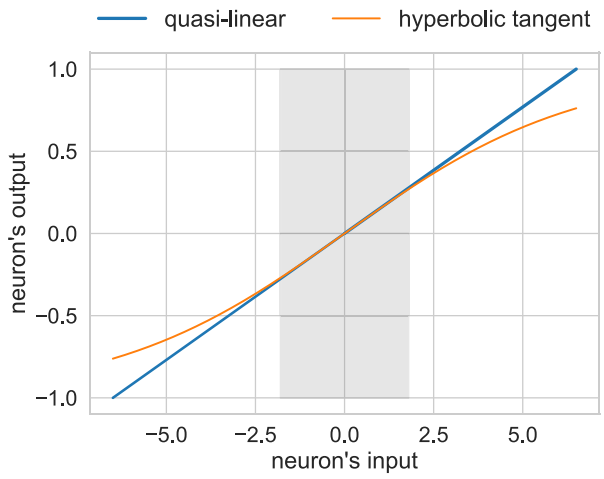


Fig. 13. Behavior of the quasi-linear and hyperbolic tangent activator for the "public health" case study.

are balanced. The hyperbolic tangent function will also exhibit this behavior if more iterations are performed.

Fig. 16 and Fig. 17 show the change values and domain values calculated using Eqs. (6) and (8), respectively, for the "car industry" case study.

For this case study, the saturation, hyperbolic tangent, rescaled, and exponential ($\beta = 0.8$) models produce cycles containing negative and positive states, with the amounts of resources varying from 100 to 400. The exponential models with $\beta = 1.0$ and $\beta = 1.2$ converge to a fixed-point attractor where no significant changes occur as more iterations are performed, meaning the amounts of resources are balanced. Notice that the hyperbolic tangent and the quasi-linear models exhibit different behaviors for this case study. This might be related to the cognitive network being slightly denser than the previous ones, which increases the neurons' capacity. Therefore, the neurons' inputs move in a wider area, and thus, the hyperbolic tangent function might produce less linear behaviors.

Overall, the empirical studies using these real-world case studies indicate that different activation functions lead to models that might exhibit distinct nonlinearities and behaviors, as highlighted in Section 3. Domain experts should ideally define the system's nonlinearity and the time unit used in the simulation. If available, this information could be extracted from historical data with little effort by investigating how variables behave in time.

5.2. Simulations using synthetically generated data

The experiments using real-world case studies allowed us to illustrate the inner workings of the proposed simulation model and how to interpret the results. Moreover, we exemplified how different activation functions lead to different simulation models. However, these results do not allow elucidating which function provides superior simulation capabilities. In this section, we will conduct additional experiments using synthetically generated data to empirically demonstrate the superiority of the exponential normalized activator compared to the other functions with respect to two desirable properties.

The first property concerns the *absence of saturation states*. As already explained, saturated neural concepts produce the same (minimal or maximal) outputs when receiving significantly different inputs. Aiming to investigate this behavior, let us start with a weight matrix where $w_{ij} = -1$ when $i \neq j$ and $w_{ii} = 0$. Afterward, we will progressively flip each weight in the matrix to $w_{ij} = 1$ while computing the norm of the final state vector. To illustrate the saturation issues, we will use a matrix as small as the ones describing the real-world case studies. The rationale of this experiment is to measure how many flipping operations are necessary to escape from the saturation state before reaching another saturation state.

Fig. 18 portrays the simulation results for $T = 20$ iterations and 20 random initial conditions. In this figure, the x-axis indicates how many flipping operations are performed while the y-axis gives the normalized norm of the state vector in the last iteration. The solid line represents the average and the shadow visualizes the standard error statistic, which is given by the standard deviation divided by the square root of the sample size.

The results clearly show that the saturation and hyperbolic tangent functions need about 15 flipping operations before escaping from the saturation state while returning to that state promptly. In contrast, the exponential models using $\beta = 0.8$ and $\beta = 1.0$ do not saturate. The exponential normalized activator using $\beta = 1.2$ and the rescaled function do not saturate either, yet they cause the network to converge to a unique fixed point.

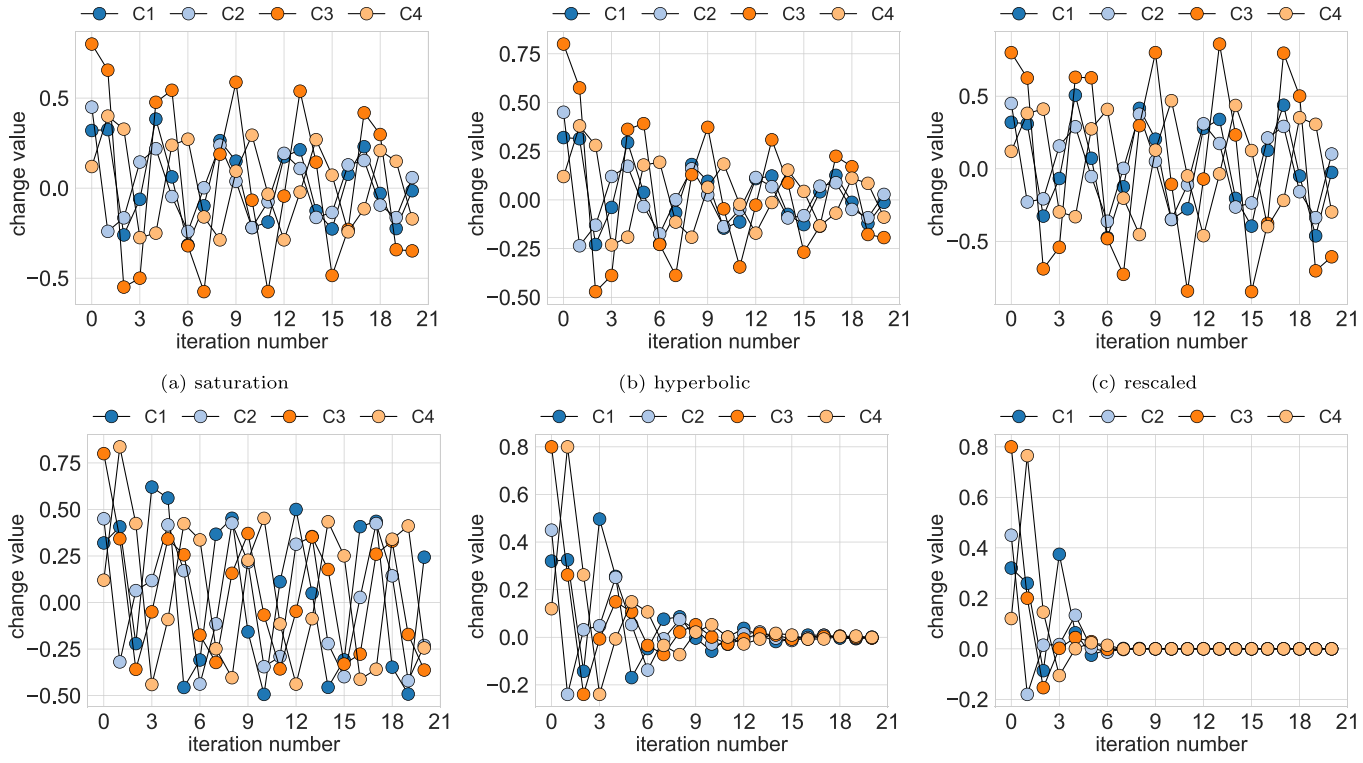
The second desirable property concerns the *system's controllability*, as defined by the model's convergence behavior. The saturation, the hyperbolic tangent, and the rescaled activation functions carry the crux of leading to models difficult to control. If no parameters are involved in these functions, then the model's convergence depends almost exclusively on the weight matrix. Several studies [21,25,31–35] have derived conditions related to the convergence of FCM-based models that involve the weight matrix. On the one hand, if these conditions are not fulfilled, then convergence cannot be ensured. On the other hand, if the convergence behaviors do not meet the simulation requirements, then we might be forced to change the weight matrix, which might not be acceptable in some domains. In this regard, the exponential normalized activator allows controlling the model's convergence to some extent through the β parameter.

Aiming to provide empirical evidence supporting the controllability of the models using the exponential normalized activator, we will generate a random weight matrix such that $w_{ij} \in [-1, 1]$ when $i \neq j$ and $w_{ii} = 0$ and 20 random initial conditions. Subsequently, we will perform the reasoning process for $T = 20$ iterations and measure the Euclidean distance between the final state vectors for different β values (ranging from 0.1 to 1.6 with a step of 0.1). The experiment aims to measure the variability in the system's responses under different parametric settings. Fig. 19 displays the results where the shadow denotes the variability in the distance values.

The results clearly indicate that the variability in the model's responses decreases as the β value increases. We can notice that $\beta \geq 1$ leads to a unique fixed-point attractor since the distance between all state vectors in the last iteration approaches zero. In contrast, $\beta < 1$ leads to system's outputs that depend on the initial conditions, which could represent multiple fixed points or unstable situations. Overall, these findings generalize well the simulation results obtained for the real-world case studies. In future research, we will attempt to prove that these findings hold for any weight matrix.

6. Concluding remarks

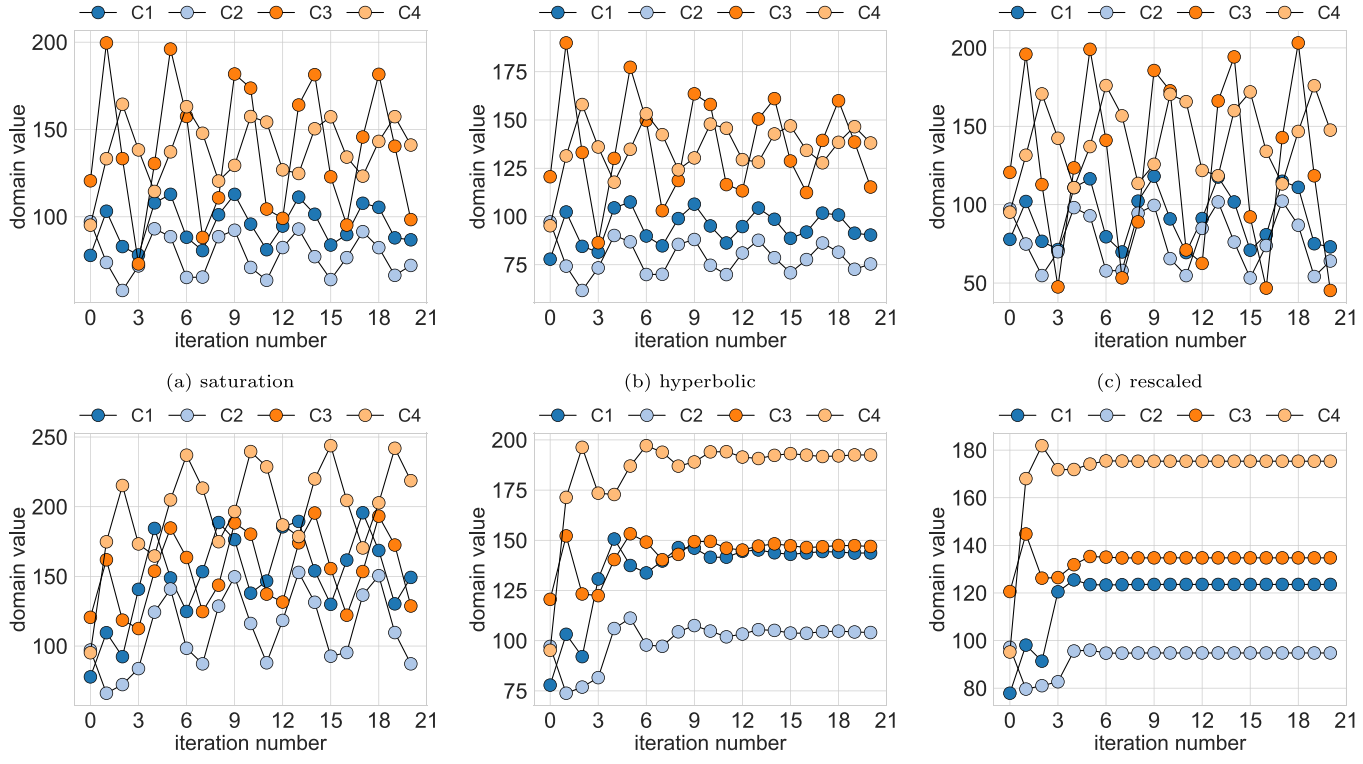
In this paper, we have proposed a modeling and simulation methodology named "neural cognitive mapping", resulting from a theory-driven revision of the FCM formalism. By redefining concepts' activation values in terms of changes, neural cognitive mapping aligns the interpretation of causal weight with the activation values of neural concepts. Moreover, the exponential normalized activation function equips the domain experts with a flexible tool to define the system's nonlinearity while preventing saturation states from happening. Finally, our proposal enables clearer insights into the model's dynamic behavior through a systematic interpretation framework, thereby enhancing its utility across various domains such as decision support systems and policy analysis.



(a) saturation (b) hyperbolic (c) rescaled

(d) exponential, β=0.8 (e) exponential, β=1.0 (f) exponential, β=1.2

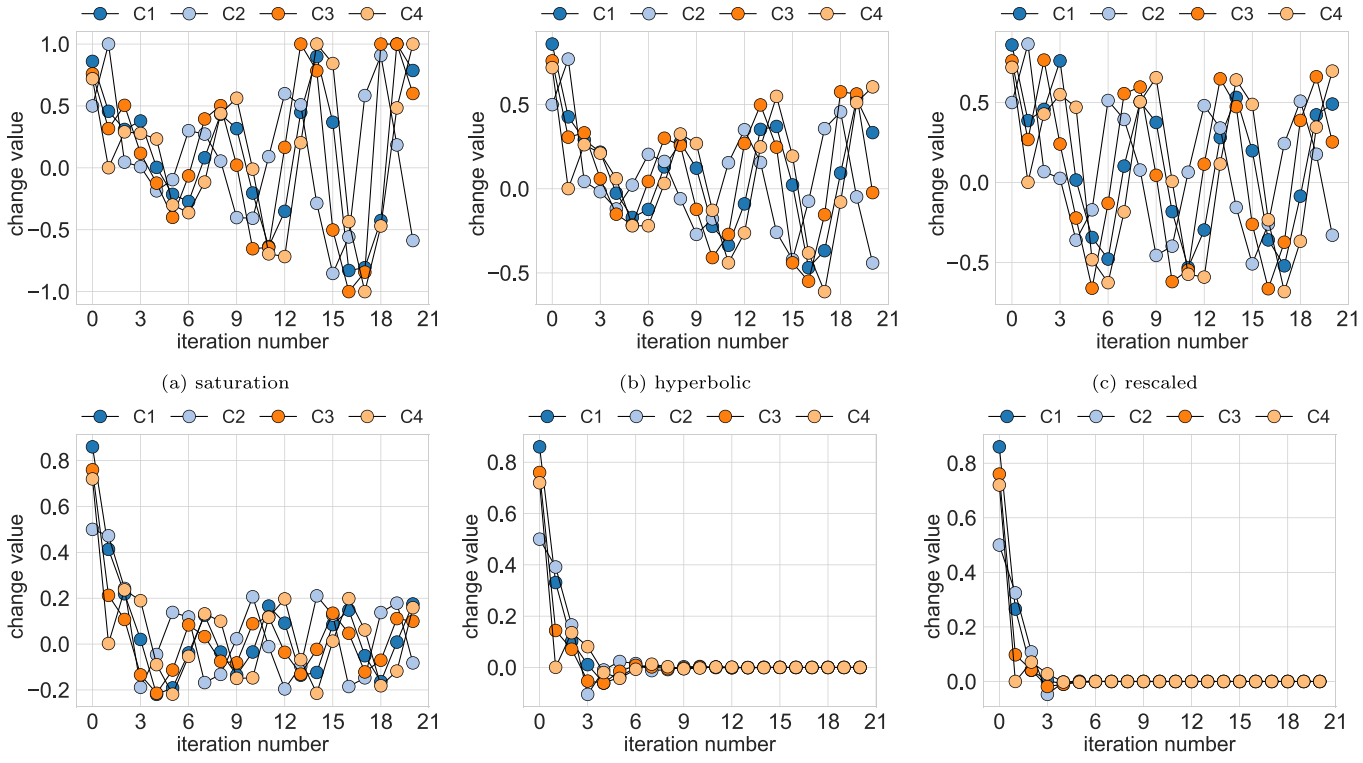
Fig. 14. Change values outputted by the FCM modeling the "crime and punishment" case study when using different activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized activators).



(a) saturation (b) hyperbolic (c) rescaled

(d) exponential, β=0.8 (e) exponential, β=1.0 (f) exponential, β=1.2

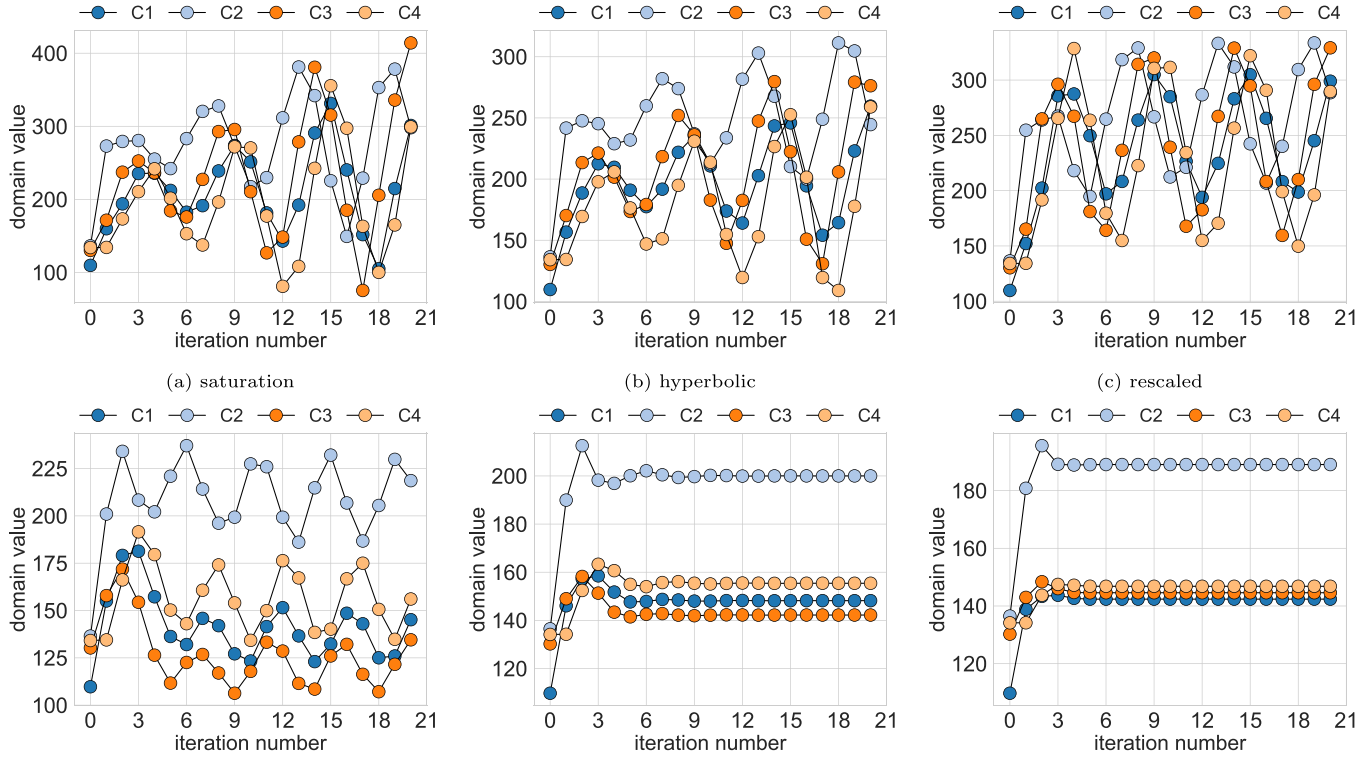
Fig. 15. Amounts of resources outputted by the FCM modeling the "crime and punishment" case study when using different activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized activators).



(a) saturation (b) hyperbolic (c) rescaled

(d) exponential, β=0.8 (e) exponential, β=1.0 (f) exponential, β=1.2

Fig. 16. Change values outputted by the FCM modeling the "car industry" case study when using different activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized activators).



(a) saturation (b) hyperbolic (c) rescaled

(d) exponential, β=0.8 (e) exponential, β=1.0 (f) exponential, β=1.2

Fig. 17. Amounts of resources outputted by the FCM modeling "car industry" case study when using different activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized activators).

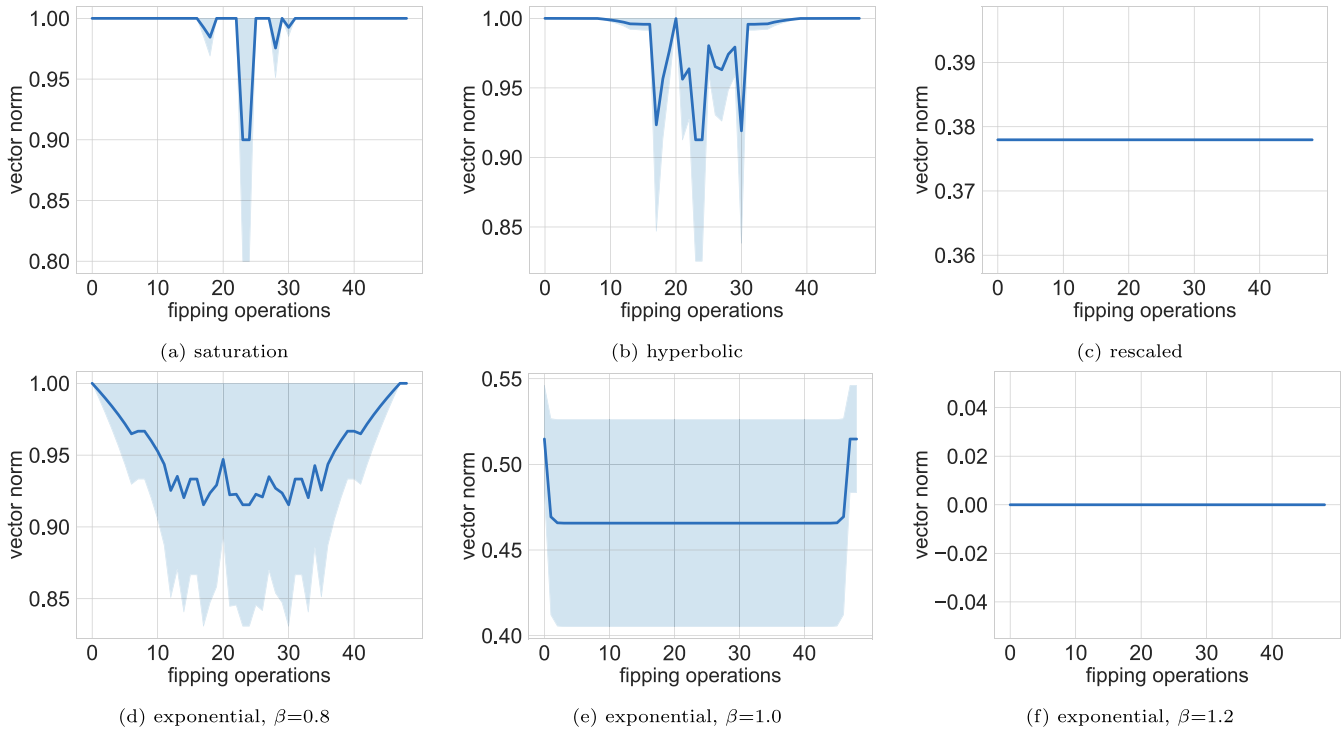


Fig. 18. Saturation behavior of activation functions (saturation, hyperbolic tangent, rescaled, and exponential normalized) for 20 randomly generated initial conditions after performing $T = 20$ iterations. The y-axis gives the norm of the state vectors in the last iteration while the x-axis gives the number of flipping operations performed on the weight matrix.

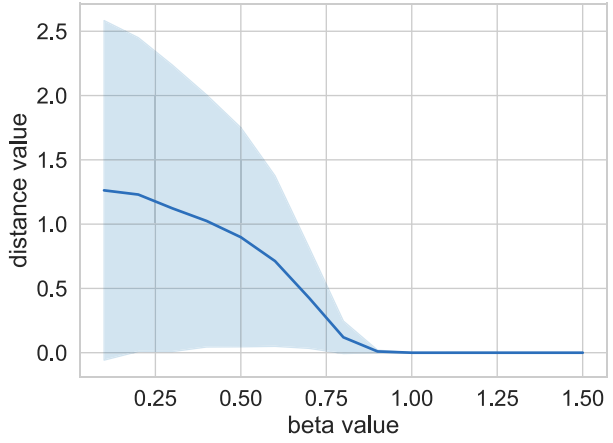


Fig. 19. Distance between the final state vector for 20 randomly generated initial conditions and different β values.

The simulations conducted on various real-world case studies illustrated the flexibility of the neural cognitive mapping methodology. Comparisons among different activation functions revealed significant differences in the convergence behaviors of the simulation models, highlighting that an arbitrarily selected activation function is not suitable for all cases. This finding underscores the value of the proposed normalized exponential activator, which allows the modeler to specify the degree of nonlinearity, enabling the creation of simulation models tailored to the system. Moreover, the alignment between the mathematical simulation model and the physical system, combined with a systematic framework for interpreting the simulation traces, introduces an enhanced feature previously unaddressed in the FCM literature. This framework enables mapping the simulation traces back to the problem domain in the presence of fixed-point attractors, cycles, or even chaos. Simulations using synthetically generated data demonstrated the effectiveness of the normalized exponential activator in avoiding saturation states while allowing for substantial control over the model's convergence.

Despite the promising results, it is reasonable to believe that our proposal needs to be tailored for specific case studies involving a paper of domain experts able to validate the correctness of simulation results. Moreover, being unaffected by saturation states opens new possibilities for designing learning algorithms able to compute the weight matrix from historical data, thus limiting the intervention of domain experts. Finally, the empirical evidence showing the model's convergence can be regulated through the β parameter cries for a mathematical proof generalizing these results for any weight matrix.

CRediT authorship contribution statement

Gonzalo Nápoles: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation, Formal analysis, Conceptualization. **Isel Grau:** Writing – review & editing, Writing – original draft, Validation, Software, Methodology, Data curation. **Yamisleydi Salgueiro:** Writing – review & editing, Writing – original draft, Visualization, Validation, Resources, Investigation, Data curation.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Yamisleydi Salgueiro reports financial support was provided by National Agency for Research and Development. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

I. Grau is supported by the European Union's HORIZON Research and Innovation Programme under grant agreement No 101120657, project ENFIELD (European Lighthouse to Manifest Trustworthy and Green AI). Y. Salgueiro would like to acknowledge the support provided by ANID Fondecyt Regular, Chile 1240293 and Basal National Center for Artificial Intelligence CENIA FB210017.

References

[1] B. Kosko, Fuzzy cognitive maps, Int. J. Man-Mach. Stud. 24 (1986) 65–75.
[2] P. Giabbanelli, C. Knox, K. Furman, A. Jetter, S. Gray, Defining and using fuzzy cognitive mapping, in: Fuzzy Cognitive Maps: Best Practices and Modern Methods, 2024, pp. 1–18.
[3] G. Nápoles, Y. Salgueiro, I. Grau, M. Espinosa, Recurrence-aware long-term cognitive network for explainable pattern classification, IEEE Trans. Cybern. 53 (2023) 6083–6094.
[4] G. Karatzinis, Y. Boutalis, Fuzzy cognitive networks with functional weights for time series and pattern recognition applications, Appl. Soft Comput. 106 (2021) 107415.
[5] J. Salmeron, I. Arévalo, Blind federated learning without initial model, J. Big Data 11 (2024) 56.
[6] G. Nápoles, A. Jastrzebska, C. Mosquera, K. Vanhoof, W. Homenda, Deterministic learning of hybrid fuzzy cognitive maps and network reduction approaches, Neural Netw. 124 (2020) 258–268.
[7] O. Orang, P. Silva, F. Guimarães, Time series forecasting using fuzzy cognitive maps: a survey, Artif. Intell. Rev. 56 (2023) 7733–7794.
[8] G. Nápoles, I. Grau, A. Jastrzebska, Y. Salgueiro, Long short-term cognitive networks, Neural Comput. Appl. 34 (2022) 16959–16971.
[9] Y. Wang, F. Yu, W. Homenda, W. Pedrycz, Y. Tang, A. Jastrzebska, F. Li, The trend-fuzzy-granulation-based adaptive fuzzy cognitive map for long-term time series forecasting, IEEE Trans. Fuzzy Syst. 30 (2022) 5166–5180.
[10] A. Jetter, Fuzzy cognitive maps for engineering and technology management: what works in practice? in: 2006 Technology Management for the Global Future-PICMET 2006 Conference, Vol. 2, 2006, pp. 498–512.
[11] A. Jetter, K. Kok, Fuzzy Cognitive Maps for futures studies—A methodological assessment of concepts and methods, Futures 61 (2014) 45–57.
[12] C. Knox, K. Furman, A. Jetter, S. Gray, P. Giabbanelli, Creating an FCM with participants in an interview or workshop setting, in: Fuzzy Cognitive Maps: Best Practices and Modern Methods, 2024, pp. 19–44.
[13] J. Salmeron, T. Mansouri, M. Moghadam, A. Mardani, Learning fuzzy cognitive maps with modified asexual reproduction optimisation algorithm, Knowl.-Based Syst. 163 (2019) 723–735.
[14] G. Nápoles, J. Salmeron, W. Froelich, R. Falcon, M. Espinosa, F. Vanhoenshoven, R. Bello, K. Vanhoof, Fuzzy cognitive modeling: Theoretical and practical considerations, Smart Innov., Syst. Technol. 142 (2019) 77–87.
[15] O. Osoba, B. Kosko, Fuzzy cognitive maps of public support for insurgency and terrorism, J. Defense Model. Simul. 14 (2017) 17–32.
[16] J. Carvalho, On the semantics and the use of fuzzy cognitive maps and dynamic cognitive maps in social sciences, Fuzzy Sets Syst. 214 (2013) 6–19.
[17] V. Mpelogianni, P. Groumpos, Re-approaching fuzzy cognitive maps to increase the knowledge of a system, Ai & Soc. 33 (2018) 175–188.
[18] E. Vergini, P. Groumpos, Modeling nearly zero energy building using new equations and separating concepts on fuzzy cognitive maps, IFAC-PapersOnLine 51 (2018) 709–714, 18th IFAC Conference on Technology, Culture and International Stability TECIS 2018.
[19] T. Koutsellis, G. Xexakis, K. Koasidis, A. Nikas, H. Doukas, Parameter analysis for sigmoid and hyperbolic transfer functions of fuzzy cognitive maps, Oper. Res. 22 (2022) 5733–5763.
[20] C. Knight, D. Lloyd, A. Penn, Linear and sigmoidal fuzzy cognitive maps: An analysis of fixed points, Appl. Soft Comput. 15 (2014) 193–202.
[21] G. Nápoles, I. Grau, L. Concepción, L. Koumeri, J. Papa, Modeling implicit bias with fuzzy cognitive maps, Neurocomputing 481 (2022) 33–45.
[22] C. Stylios, P. Groumpos, Modeling complex systems using fuzzy cognitive maps, IEEE Trans. Syst. Man Cybern. A 34 (2004) 155–162.
[23] P. Giabbanelli, G. Nápoles, Fuzzy Cognitive Maps: Best Practices and Modern Methods, Springer Nature, Switzerland, 2024.
[24] S. Bueno, J. Salmeron, Benchmarking main activation functions in fuzzy cognitive maps, Expert Syst. Appl. 36 (2009) 5221–5229.
[25] L. Concepción, G. Nápoles, R. Falcon, K. Vanhoof, R. Bello, Unveiling the dynamic behavior of fuzzy cognitive maps, IEEE Trans. Fuzzy Syst. 29 (2021) 1252–1261.
[26] G. Nápoles, N. Ranković, Y. Salgueiro, On the interpretability of fuzzy cognitive maps, Knowl.-Based Syst. 281 (2023) 111078.
[27] G. Nápoles, G. Van Houdt, M. Laghmouch, W. Goossens, Q. Moesen, B. Depaire, Fuzzy cognitive maps: A business intelligence discussion, in: Intelligent Decision Technologies 2019: Proceedings of the 11th KES International Conference on Intelligent Decision Technologies (KES-IDT 2019), Vol. 1, 2020, pp. 89–98.
[28] A. Tsadiras, Comparing the inference capabilities of binary, trivalent and sigmoid fuzzy cognitive maps, Inform. Sci. 178 (2008) 3880–3894.
[29] J. Carvalho, J. Tomè, Rule based fuzzy cognitive maps and fuzzy cognitive maps-a comparative study, in: 18th International Conference of the North American Fuzzy Information Processing Society-NAFIPS, 1999, pp. 115–119.
[30] A. Tsadiras, N. Bassiliades, Ruleml representation and simulation of fuzzy cognitive maps, Expert Syst. Appl. 40 (2013) 1413–1426.
[31] Y. Boutalis, T. Kottas, M. Christodoulou, Adaptive estimation of fuzzy cognitive maps with proven stability and parameter convergence, IEEE Trans. Fuzzy Syst. 17 (2009) 874–889.
[32] I. Harmati, L. Kóczy, On the convergence of sigmoidal fuzzy grey cognitive maps, Int. J. Appl. Math. Comput. Sci. 29 (2019) 453–466.
[33] I. Harmati, L. Kóczy, On the convergence of input-output fuzzy cognitive maps, in: Proceedings of International Joint Conference on Rough Sets, 2020, pp. 449–461.
[34] I. Harmati, L. Kóczy, Some dynamical properties of higher-order fuzzy cognitive maps, in: Computational Intelligence and Mathematics for Tackling Complex Problems, Vol. 3, 2022, pp. 149–156.
[35] I. Harmati, M. Hatwágner, L. Kóczy, Global stability of fuzzy cognitive maps, Neural Comput. Appl. 35 (2023) 7283–7295.