



Multistability and circuit implementation of tabu learning two-neuron model: application to secure biomedical images in IoMT

Isaac Sami Doubla^{1,3} · Zeric Tabekoueng Njitacke^{1,2} · Sone Ekonde² · Nestor Tsafack^{1,3} · J. D. D. Nkappok⁴ · Jacques Kengne¹

Received: 1 November 2020 / Accepted: 15 May 2021 / Published online: 13 June 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

In this paper, the dynamics of a non-autonomous tabu learning two-neuron model is investigated. The model is obtained by building a tabu learning two-neuron (TLTN) model with a composite hyperbolic tangent function consisting of three hyperbolic tangent functions with different offsets. The possibility to adjust the compound activation function is exploited to report the sensitivity of non-trivial equilibrium points with respect to the parameters. Analysis tools like bifurcation diagram, Lyapunov exponents, phase portraits, and basin of attraction are used to explore various windows in which the neuron model under the consideration displays the uncovered phenomenon of the coexistence of up to six disconnected stable states for the same set of system parameters in a TLTN. In addition to the multistability, nonlinear phenomena such as period-doubling bifurcation, hysteretic dynamics, and parallel bifurcation branches are found when the control parameter is tuned. The analog circuit is built in PSPICE environment, and simulations are performed to validate the obtained results as well as the correctness of the numerical methods. Finally, an encryption/decryption algorithm is designed based on a modified Julia set and confusion–diffusion operations with the sequences of the proposed TLTN model. The security performances of the built cryptosystem are analyzed in terms of computational time (CT = 1.82), encryption throughput (ET = 151.82 MBps), number of cycles (NC = 15.80), NPCR = 99.6256, UACI = 33.6512, χ^2 -values = 243.7786, global entropy = 7.9992, and local entropy = 7.9083. Note that the presented values are the optimal results. These results demonstrate that the algorithm is highly secured compared to some fastest neuron chaos-based cryptosystems and is suitable for a sensitive field like IoMT security.

Keywords Tabu learning two neurons · Multistability · Modified Julia set · Image encryption

1 Introduction

The tabu learning neuron (TLN) was proposed firstly by Beyer and Ogier [1] based on the Hopfield neural network (HNN) for solving non-convex optimization problems. Tabu learning applies the concept of tabu search [2, 3] to neural networks by continuously increasing the energy surface in a neighborhood of the current state, thus penalizing states already visited. References [4, 5] studied the dynamic behaviors of the one- and two-neuron-based tabu learning neural network to find out what the state trajectory of the tabu learning neural network looks like. Because the tabu learning method, unlike other methods, does not consist of forcing the state of the network to converge to an optimal or near-optimal solution, but rather, the network conducts a search in the solution space. The

✉ Zeric Tabekoueng Njitacke
njitacke.tabekoueng@ubuea.cm

¹ Unité de Recherche D'Automatique Et Informatique Appliquée (URAIA), Department of Electrical Engineering, IUT-FV Bandjoun, University of Dschang, Dschang, Cameroon
² Department of Electrical and Electronic Engineering, College of Technology (COT), University of Buea, P.O. Box 63, Buea, Cameroon
³ Unité de Recherche de Matière Condensée, d'Electronique Et de Traitement du Signal (URAMACETS), Department of Physics, University of Dschang, P.O. Box 67, Dschang, Cameroon
⁴ Department of Electrical Engineering and Industrial Computing, University Institute of Technology, P.O. Box 8698, Douala, Cameroon

results of their work showed periodic, chaotic solutions and the existence of Hopf bifurcation. Reference [6] demonstrated the existence of a Pitchfork bifurcation, a Flip bifurcation, and a Neimark–Sacker bifurcation in a discrete one-neuron-based TLN model, when the control parameter exceeded the critical value. Chen and Li [7] have proposed in their works the circuit design for the implementation of a single tabu learning neuron. They also proposed circuits for the case of an autonomous two-neuron-based TLN and exhibited the periodic, chaotic solutions without carrying out in-depth studies on their dynamics. Results similar to [6] were obtained by investigations conducted by Ref. [8] on a tabu learning neuron model with two delayed neurons. Recently, Ref. [9] revealed complex activities such as chaotic/periodic spiking/bursting firing patterns and coexisting bi-stable firing patterns in a case of TLN to an autonomous neuron, following digital investigations with a compound activation function. These results were confirmed during the implementation of the circuit in which an approximation of this difficult to implement activation function was performed. Subsequently, [10] showed the bi-stability in a one-neuron-based tabu learning neuron model that they were able to validate experimentally through an FPGA-based implementation.

Little work in the literature has conducted an in-depth study of the dynamics of tabu learning neuron. It is reason enough for us to carry out detailed investigations on the dynamics of this model to better understand its behavior. In this article, we will conduct a study focused on the analysis of the dynamics of a non-autonomous two-neuron-based tabu learning neural network. The interesting thing about this study is that it is accomplished by involving the compound activation function used by [9] in the study of a single-neuron TLN model.

This made it possible to obtain interesting phenomena there because the activation function plays an important role in the appearance of complex phenomena in neural networks. As such, the results of the work obtained by several authors [11–32] carried out on various models of Hopfield neural networks with two, three neurons, or even more testify to the importance that the activation function plays in the appearance of complex behaviors in neural networks in general. It will be interesting for us in this work to see the contribution that this compound activation function will have on the non-autonomous two-neuron-based tabu learning neuron model submitted to our analysis.

Nowadays, new researches are developed to propose modern encryption algorithms for confidential data including medical images [33]. A chaotic system is a major tool in this prominent research domain due to ergodicity, deterministic dynamics, unpredictable behaviors, nonlinear transformation, and sensitivity dependence of the system

[34–37]. For instance, Gao and collaborators [34] designed an encryption algorithm based on Chen hyperchaotic system. A simple diffusion-confusion encryption scheme is developed in their work. Zhou and co-workers [37] used 1-D chaotic map to establish the encryption key for both color and gray images. Analysis of the proposed scheme showed a high-security level. As strong cryptographic technics are developed, cryptanalysis is also growing. Another idea of chaos encryption is to use the discrete output of the neural network to increase the security of the process. In this line, Xing and collaborator [36] designed an encryption scheme using both the sequences of the Lorenz attractor and the discrete output of the perceptron model. Experimental results show a high-security algorithm. Lakshmi and co-workers [35] design an encryption algorithm based on HNN. The technique uses a simple diffusion confusion algorithm, and security against some existing attacks is achieved.

A variety of encryption methods can be found in the literature and classified either as spatial domain or frequency domain encryption algorithms [34–37]. The first method directly considers the pixel of the original image without any transformation. The second method applies a mathematical transformation on the original image to compute some coefficients based on image pixels. Transform domain-based algorithms seem to be more efficient and robust than the spatial domain. The above-mentioned techniques combining both chaos and neural network in cryptography rely on spatial domain algorithms. In this paper, we will use a modified Julia set and the discrete sequences of the proposed TLN to transform the pixels of the plain image. Then, the sequences of a simple TLN will be applied for encryption. Besides, the main characteristic of the model addressed in this work is summarized in Table 1. The main contributions of this work include:

- (a) Introduce a novel two-neuron-based tabu learning model,
- (b) Use nonlinear analysis tools to prove that the newly introduced model exhibits more complex behaviors than the previously presented one (see Table 1)
- (c) Provide PSPICE simulations of the electronic circuit of the proposed model to further support the presented results.
- (d) We design a robust and simple encryption/decryption algorithm using the chaotic sequence of the proposed tabu learning two-neuron model.
- (e) The performance analysis indicates that the proposed encryption scheme is secured with regard to some well-known attacks.
- (f) An idea of real-time implementation of our algorithm is provided such as an application of the

Table 1 Summary of some recent work addressed on tabu neural networks

Works	Number of neurons	Activation function type	Number of coexisting attractors	Implementation	Application
Chen and Li [7]	1	$f(u_j) = \tanh(u_j)$	0	SPICE simulation environment	No
	2	$f(u_j) = \tanh(5u_j)$			
Bao et al. [9]	1	$f(x) = \eta x \exp\left[-\left(\frac{\eta x}{\sigma}\right)^2\right]$	2	Multisim circuit simulations and hardware experiments	No
Zhu et al. [10]	2	$f(u_j) = \tanh(gu_j)$	2	Hardware experiments FPGA-based	No
This present Work	2	$f(x_j) = \eta x_j \exp\left[-\left(\frac{\eta x_j}{\sigma}\right)^2\right]$	6	PSpice circuit simulations	Yes

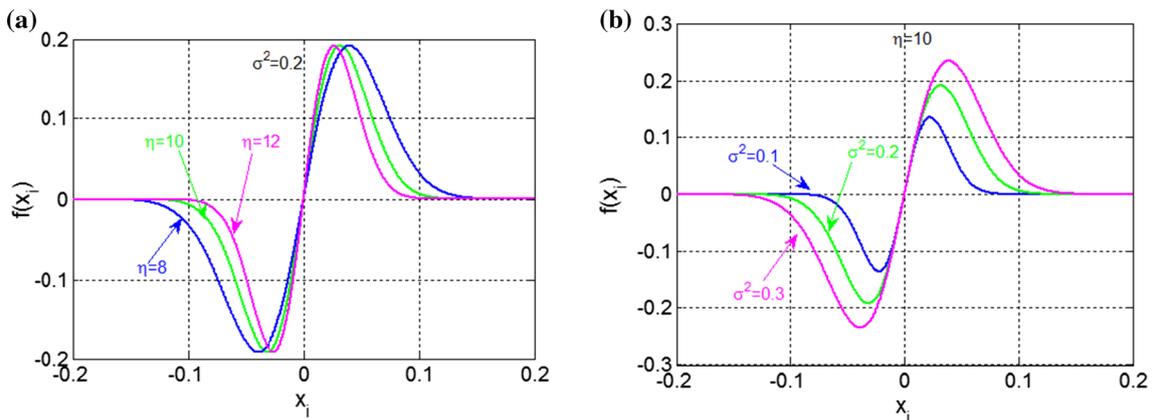


Fig. 1 Existence of different nonlinearities compound activation function, due to η and σ^2 constants in (a) and (b), respectively

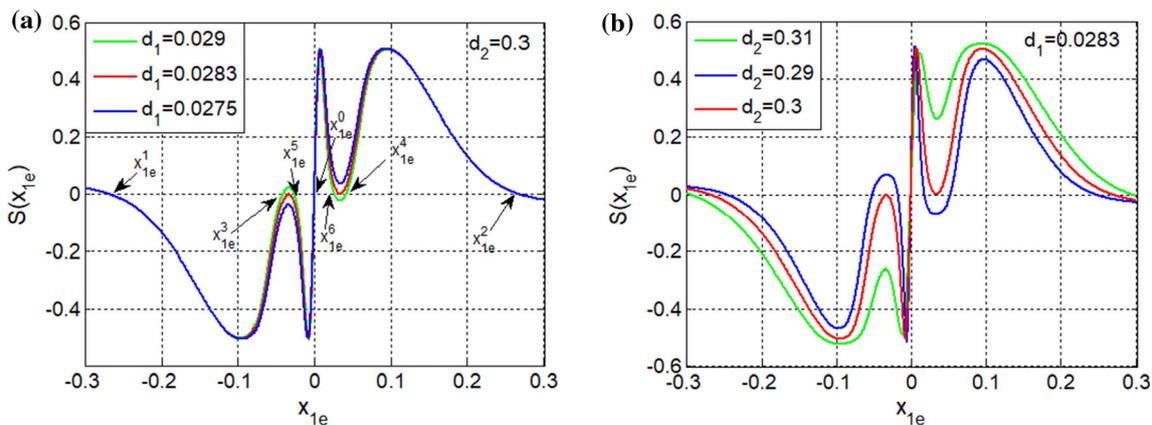


Fig. 2 Function curve described by Eq. 9 and their intersection points with respect to parameters (d_1, d_2) ; equilibrium points varying with respective parameters d_1 and d_2 in (a) and (b), respectively

Table 2 Equilibrium points states, their eigenvalues and stabilities for some values of the learning rate d_1

d_1	Equilibrium points States $P_{0,1,2,3,4,5,6}$	Eigen values $\lambda_{1,2,3,4}$	Stabilities
0.196	$P_0(0, 0, 0, 0)$	$39.1284 + 0.0000i$	Unstable saddle-focus
		$-0.2528 + 1.3851i$	
		$-0.2528 - 1.3851i$	
	$P_{1,2}(\pm 0.2687, \pm 0.0960, \mp 0, \mp 0.0288)$	$-0.0228 + 0.0000i$	Unstable saddle point
		$3.9097 + 0.0000i$	
		$-0.1000 + 0.0000i$	
	$P_{3,4}(\pm 0.0960, \pm 0.0949, \mp 0.0187, \mp 0.0314)$	$-0.1000 - 0.0000i$	Unstable saddle-focus
		$-0.0236 + 0.0000i$	
		$4.1681 + 0.0000i$	
	$P_{5,6}(\pm 0.0015, \pm 0.0944, \mp 0.0289, \mp 0.0329)$	$-0.1156 + 0.4435i$	Unstable saddle-focus
		$-0.1156 - 0.4435i$	
		$-0.0216 + 0.0000i$	
0.2089	$P_0(0, 0, 0, 0)$	$39.1284 + 0.0000i$	Unstable saddle-focus
		$-0.2527 + 1.4317i$	
		$-0.2527 - 1.4317i$	
	$P_{1,2}(\pm 0.2687, \pm 0.0960, \mp 0, \mp 0.0288)$	$-0.0229 + 0.0000i$	Unstable saddle point
		$3.9097 + 0.0000i$	
		$-0.1000 + 0.0000i$	
	$P_{3,4}(\pm 0.0969, \pm 0.0951, \mp 0.0185, \mp 0.0311)$	$-0.1000 - 0.0000i$	Unstable saddle-focus
		$-0.0236 + 0.0000i$	
		$4.1363 + 0.0000i$	
	$P_{5,6}(\pm 0.0014, \pm 0.0946, \mp 0.0288, \mp 0.0325)$	$-0.1146 + 0.4432i$	Unstable saddle-focus
		$-0.1146 - 0.4432i$	
		$-0.0217 + 0.0000i$	
0.25	$P_0(0, 0, 0, 0)$	$3.7723 + 0.0000i$	Unstable saddle-focus
		$-0.3012 + 1.5205i$	
		$-0.3012 - 1.5205i$	
	$P_{1,2}(\pm 0.2687, \pm 0.0960, \mp 0, \mp 0.0288)$	$-0.0217 + 0.0000i$	Unstable saddle point
		$39.1286 + 0.0000i$	
		$-0.2527 + 1.5710i$	
	$P_{3,4}(\pm 0.0992, \pm 0.0952, \mp 0.0180, \mp 0.0307)$	$-0.2527 - 1.5710i$	Unstable saddle-focus
		$-0.0233 + 0.0000i$	
		$3.9097 + 0.0000i$	
	$P_{5,6}(\pm 0.0011, \pm 0.0948, \mp 0.0285, \mp 0.0318)$	$-0.1000 + 0.0000i$	Unstable saddle-focus
		$-0.1000 - 0.0000i$	
		$-0.0236 + 0.0000i$	
		$4.0986 + 0.0000i$	Unstable saddle-focus
		$-0.1121 + 0.4420i$	
		$-0.1121 - 0.4420i$	
		$-0.0220 + 0.0000i$	Unstable saddle-focus
		$3.7051 + 0.0000i$	
		$-0.3090 + 1.6670i$	
		$-0.3090 - 1.6670i$	
		$-0.0220 + 0.0000i$	

Table 2 (continued)

d_1	Equilibrium points States $P_{0,1,2,3,4,5,6}$	Eigen values $\lambda_{1,2,3,4}$	Stabilities
0.28	$P_0(0, 0, 0, 0)$	$39.1287 + 0.0000i$ $-0.2527 + 1.6653i$ $-0.2527 - 1.6653i$ $-0.0234 + 0.0000i$	Unstable saddle-focus
	$P_{1,2}(\pm 0.2687, \pm 0.0960, \mp 0, \mp 0.0288)$	$3.9097 + 0.0000i$ $-0.1000 + 0.0000i$ $-0.1000 - 0.0000i$ $-0.0236 + 0.0000i$	Unstable saddle point
	$P_{3,4}(\pm 0.1007, \pm 0.0951, \mp 0.0177, \mp 0.0309)$	$4.1241 + 0.0000i$ $-0.1107 + 0.4409i$ $-0.1107 - 0.4409i$ $-0.0222 + 0.0000i$	Unstable saddle-focus
	$P_{5,6}(\pm 0.0010, \pm 0.0948, \mp 0.0283, \mp 0.0317)$	$3.7089 + 0.0000i$ $-0.3136 + 1.7651i$ $-0.3136 - 1.7651i$ $-0.0222 + 0.0000i$	Unstable saddle-focus

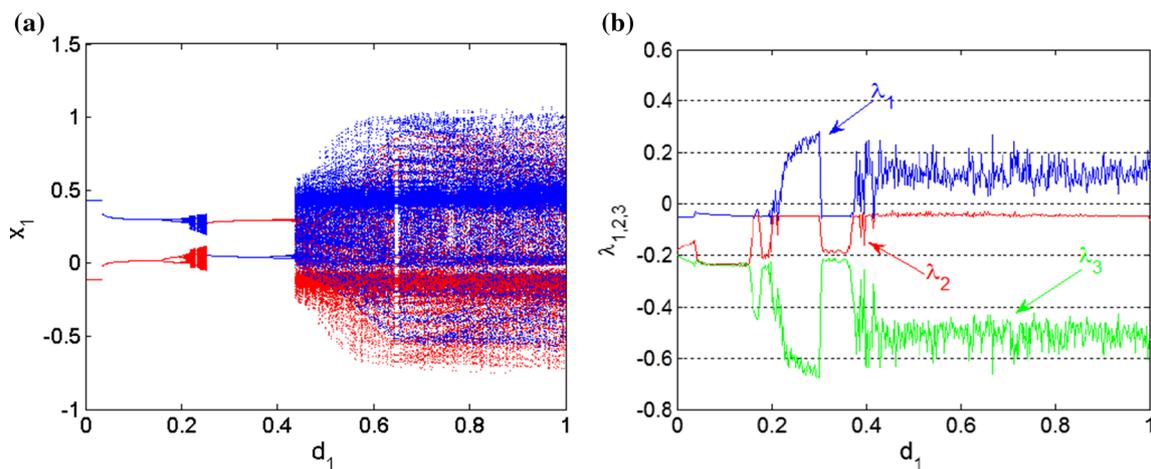


Fig. 3 Bifurcation diagrams show the local maximum of x_1 in term of the learning d_1 (rate control parameter) in (a) and his corresponding Lyapunov spectrum in (b)

proposed cryptosystem in the Internet of Medical Things (IoMT).

The next section of this article will be organized as follows: In Sect. 2, the mathematical model is described, the equilibrium points are illustrated graphically, and the determination of the characteristic equation from the Jacobian matrix is also presented. In Sect. 3, the dynamic behaviors associated with a control parameter under the

presence of the stimulus are revealed numerically by bifurcation diagrams and dynamic evolution in the parameter space as well as phase portraits. In Sect. 4, analog analyses are performed in Pspice to verify compliance with digital simulations. The cryptosystem based on IoMT is designed and experimented in Sect. 5. A comparative analysis in terms of encryption/decryption performance is presented in Sect. 6. Finally, in Sect. 7 the conclusions of this work are summarized.

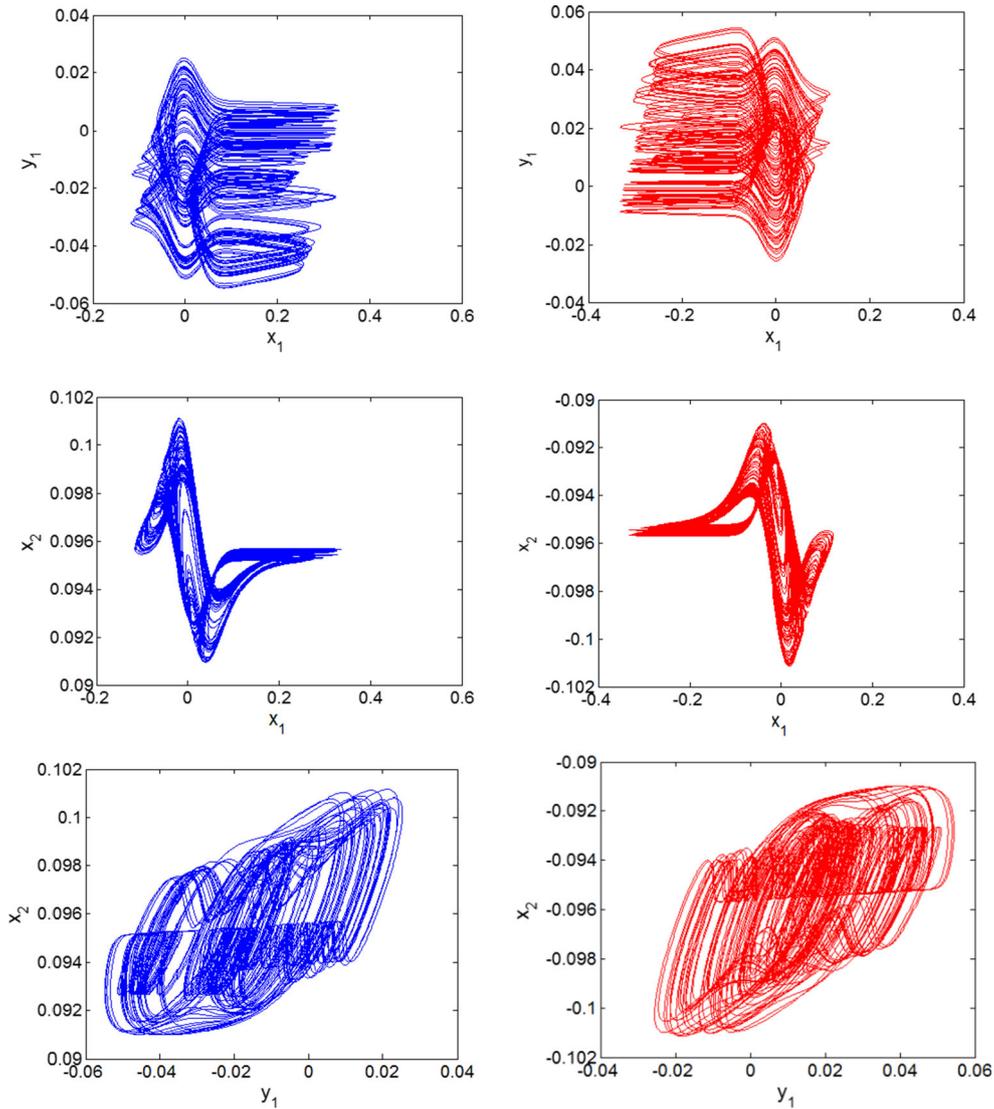


Fig. 4 Projection of symmetrical chaotic attractors on different planes. These attractors are obtained for $d_1 = 0.25$ and by changing sign of the initial conditions $(0, 0, \pm 0.1, 0)$

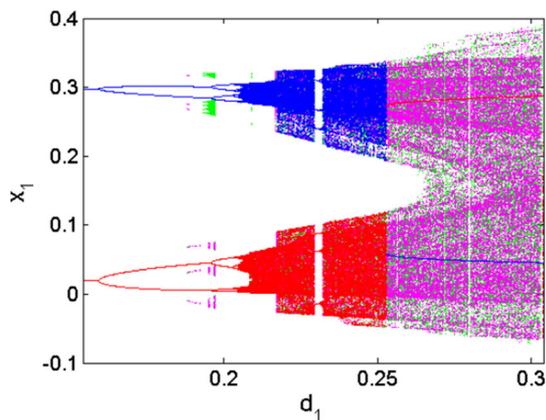


Fig. 5 Enlargement of bifurcation diagram of Fig. 3a in the range $0.154 \leq d_1 \leq 0.304$

2 Preliminary model description

2.1 Mathematical model involving composite activation function

The form of two-neuron-based HNN [38, 39] is given by Eq. (1):

$$\dot{x}_i = -a_i x_i + \sum_{j=1}^2 b_{ij} f(x_j) + I_i \tag{1}$$

where a_i are the positive constants, $x_i = [x_1, x_2]^T$ is the state vector of neurons, $f(x_j) = [f(x_1), f(x_2)]^T$ is the vector of activation function, and b_{ij} represents a synaptic weight matrix 2×2 describing the strength of the connections

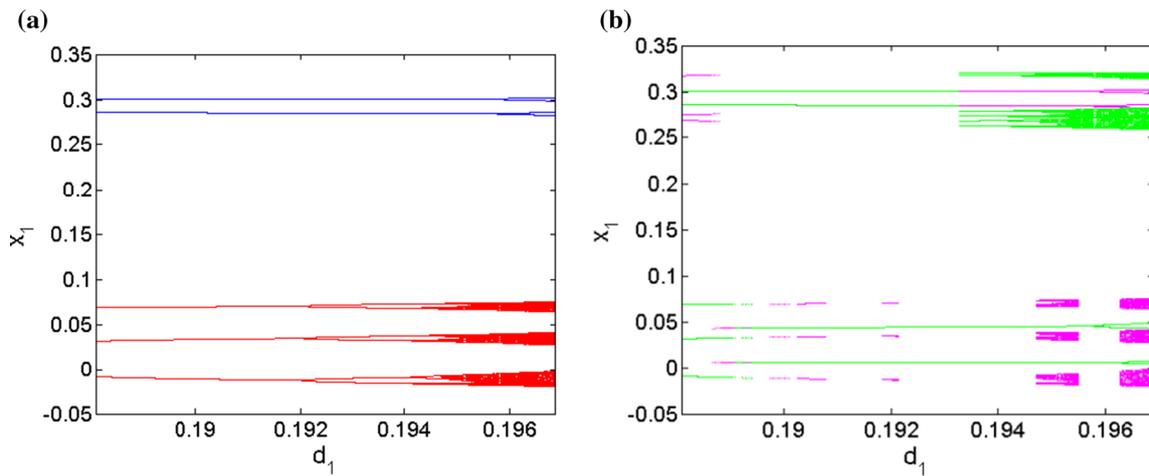


Fig. 6 Enlargement of bifurcation diagram of Fig. 5 in the range $0.1881 \leq d_1 \leq 0.1969$

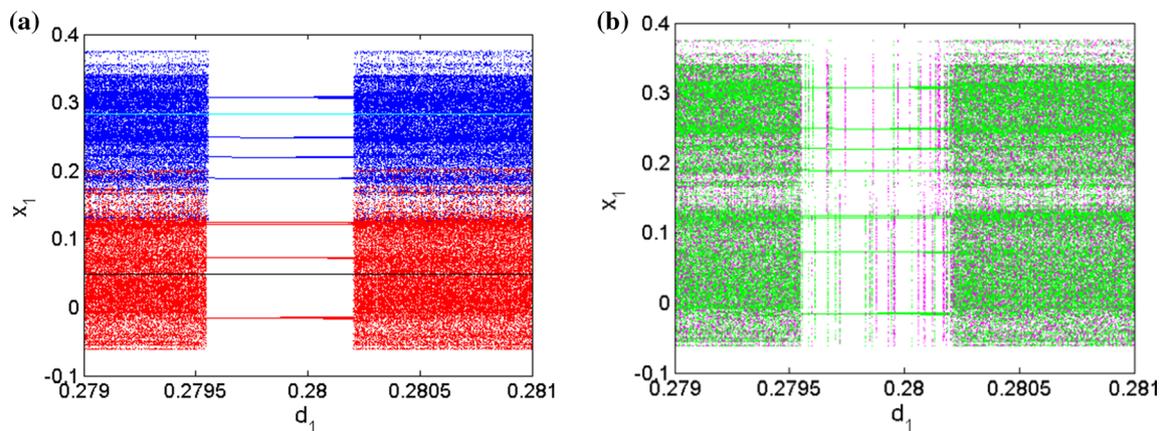


Fig. 7 Enlargement of bifurcation diagram of Fig. 5 in the range $0.279 \leq d_1 \leq 0.281$

between the two neurons of the network and $I_i = [I_1, I_2]^T$ is the stimulus can be constant or sinusoidal. The TLN model [1, 10, 40] for two neurons is described by Eq. (2):

$$\begin{cases} \dot{x}_i = -a_i x_i + \sum_{j=1}^2 b_{ij} f(x_j) + y_i + I_i \\ \dot{y}_i = -c_i y - d_i f(x_i) \end{cases} \quad (2)$$

With $y_i = [y_1, y_2]^T$ being the vector of the tabu learning neuron variables, c_i and d_i are the positive constants that represent the memory decay rate and the learning rate. The activation function considering in Eq. 1 is described in Eq. 3. This activation function was used by [9] in his work on the tabu learning neuron model with one neuron. The graphical representation of the compound activation function given in Eq. 3 is shown in Fig. 1a and b for different values of constants η and σ , respectively. In Fig. 1a for $\sigma^2 = 0.2$, the compound activation function curve becomes

flat when the value of constant η is decreases, which corresponds to a fast response of neurons. On the other hand in Fig. 1b when, the compound activation function curve is dilated, this implies an increase in response range of neurons. In other words by acting on the constants η and σ^2 , it is possible to modify the speed and the limits of the response of neurons, respectively. This makes the compound activation function (3) much more flexible than the hyperbolic tangent type activation function, of which we can only adjust the response speed of neurons by acting on its activation gradient as indicated in [11, 19].

$$f(x_j) = \eta x_j \exp \left[-\left(\frac{\eta x_j}{\sigma} \right)^2 \right] \quad (3)$$

with $\eta > 0$ and $\sigma > 0$ represented in Fig. 1. From these diagrams, it can be seen that the crossover point of the neuron activation function strongly depends on these important parameters, namely η and σ .

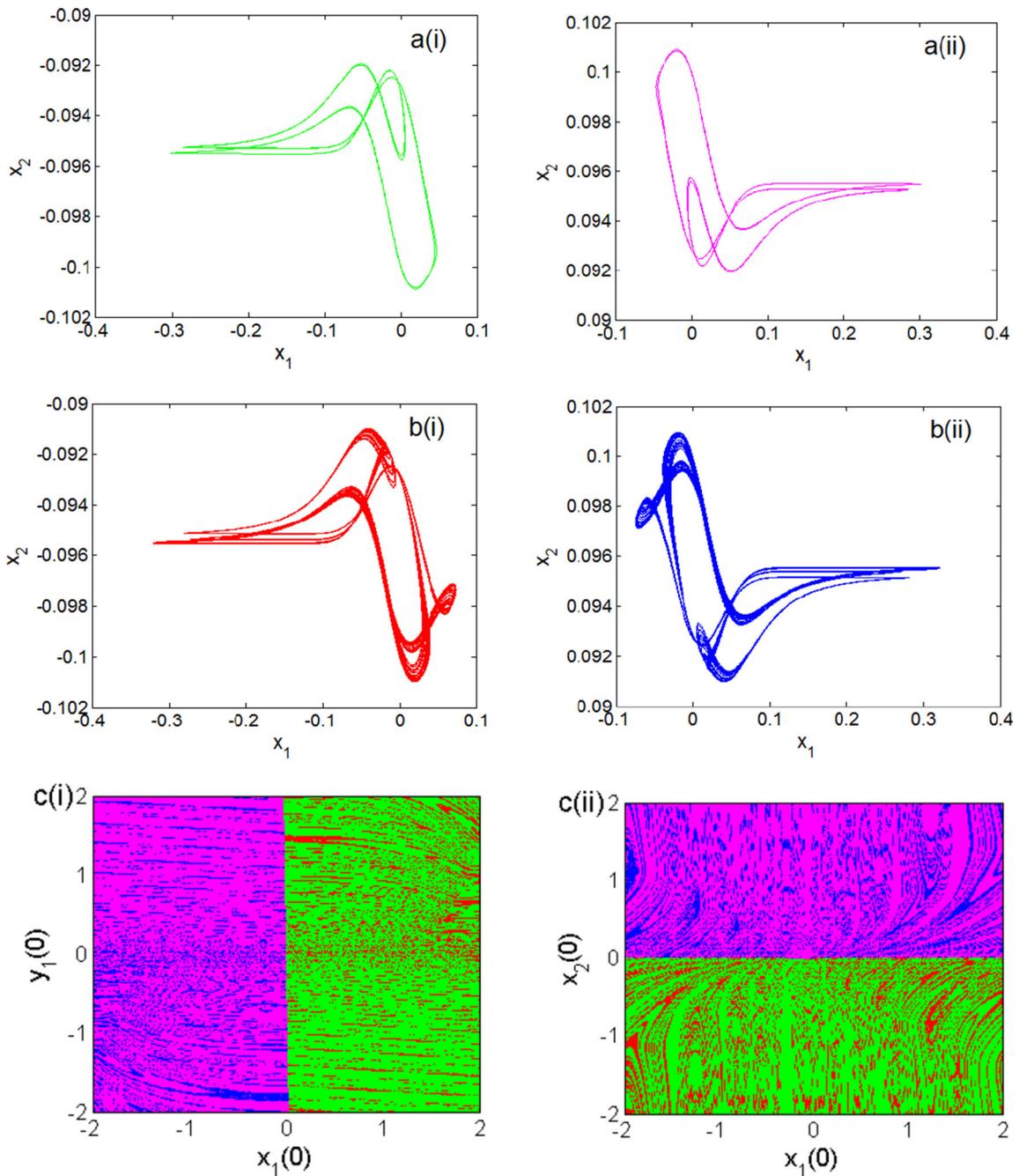


Fig. 8 Representation the phase portraits of the coexistence of four different attractors in (x_1, x_2) plane, showing: period-4 limit cycles (low and upper), chaotic attractors (low and upper), and its corresponding cross section of basin with respective colors in ©. These attractors are obtained for $d_1 = 0.196$ and for initial conditions

$(x_1(0), x_2(0), y_1(0), y_2(0)) = (0, 0, \pm 0.28, 0)$ and $(0, 0, \pm 1.44, 0)$, respectively

The mathematical model of the TLTN corresponding to the configuration of the synaptic weight matrix 2×2 is given by Eq. (4):

$$\begin{cases} \dot{x}_1 = -a_1x_1 + b_{11}f(x_1) + b_{12}f(x_2) + y_1 + I_1 \\ \dot{x}_2 = -a_2x_2 + b_{21}f(x_1) + b_{22}f(x_2) + y_2 + I_2 \\ \dot{y}_1 = -c_1y_1 - d_1f(x_1) \\ \dot{y}_2 = -c_2y_2 - d_2f(x_2) \end{cases} \quad (4)$$

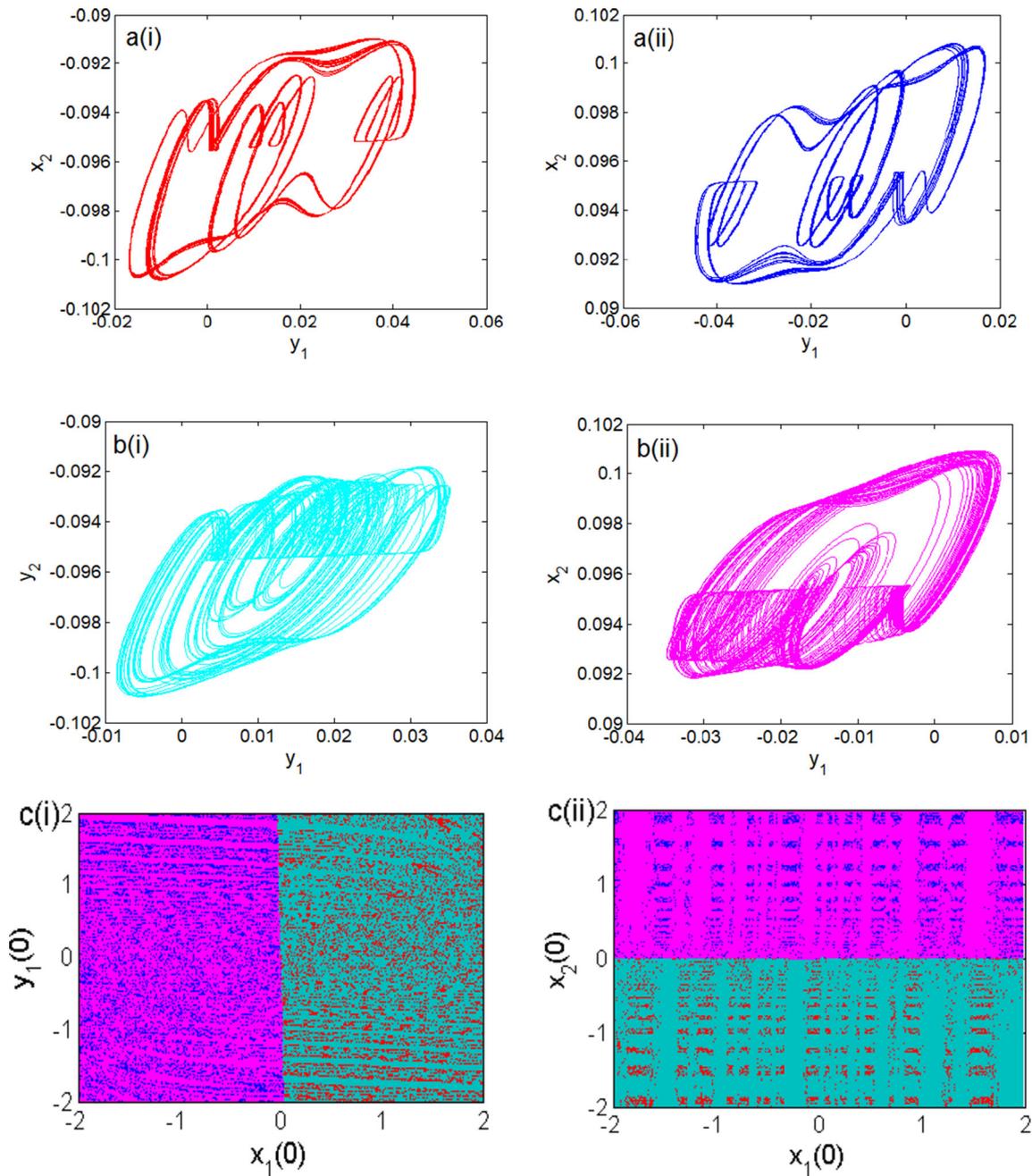


Fig. 9 Representation the phase portraits of the coexistence of four different symmetric chaotic attractors (**a**, **b**) in plane and its corresponding cross section of basin with respective colors in ©.

These attractors are obtained for $d_1 = 0.2089$ and for initial conditions $(x_1(0), x_2(0), y_1(0), y_2(0)) = (0, 0, +0.2/-0.12, 0)$ and $(0, 0, \pm 0.16, 0)$, respectively

where
$$\begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} -0.1 & 2.8 \\ -0.1 & 4 \end{bmatrix} \tag{5}$$

with $a_1 = 0.1, a_2 = 0.1, c_1 = 0.1, c_2 = 0.1, I_1 = I_m \sin(2\pi F_1 \tau), I_2 = 0, I_m = 0.2, F_1 = 0.2, \eta = 10$ and $\sigma^2 = 0.2$. (6)

2.2 Equilibrium points related to parameters

The equilibrium points of system (4) are obtained by making its left-hand member equal to zero at $\tau = 0$; this enables to solve the set of equations given in (7):

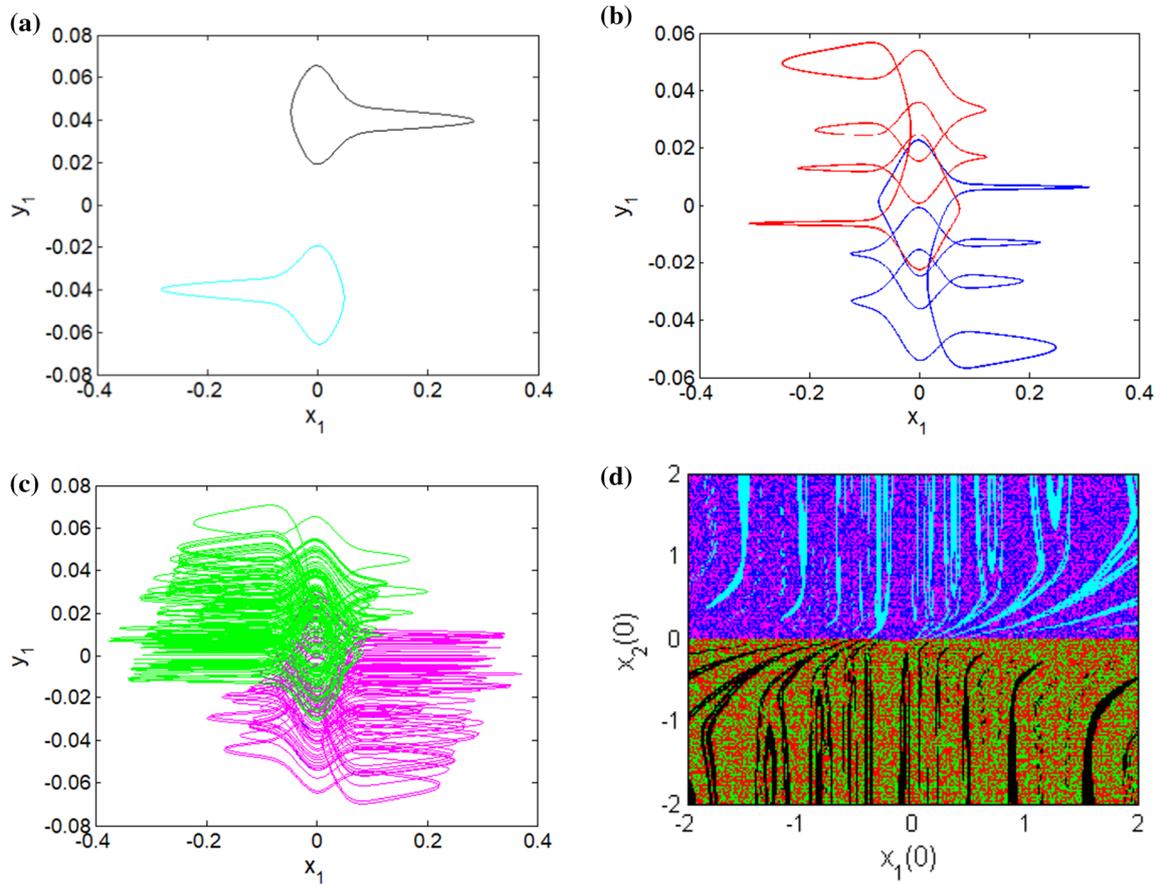


Fig. 10 Representation the phase portraits of the coexistence of six different symmetric attractors in (x_1, y_1) plane, showing: period-1 limit cycle (low and upper) in **(a)**, peiod-6 limit cycles (low and upper) and chaotic attractors (low and upper) and its corresponding

cross section of basin with respective colors in **(d)**. These attractors are obtained for $d_1 = 0.28$ and for initial conditions $(x_1(0), x_2(0), y_1(0), y_2(0)) = (0, 0, +0.04/-0.072, 0), (0, 0, \pm 0.11, 0)$ and $(0, 0, \pm 0.216, 0)$, respectively

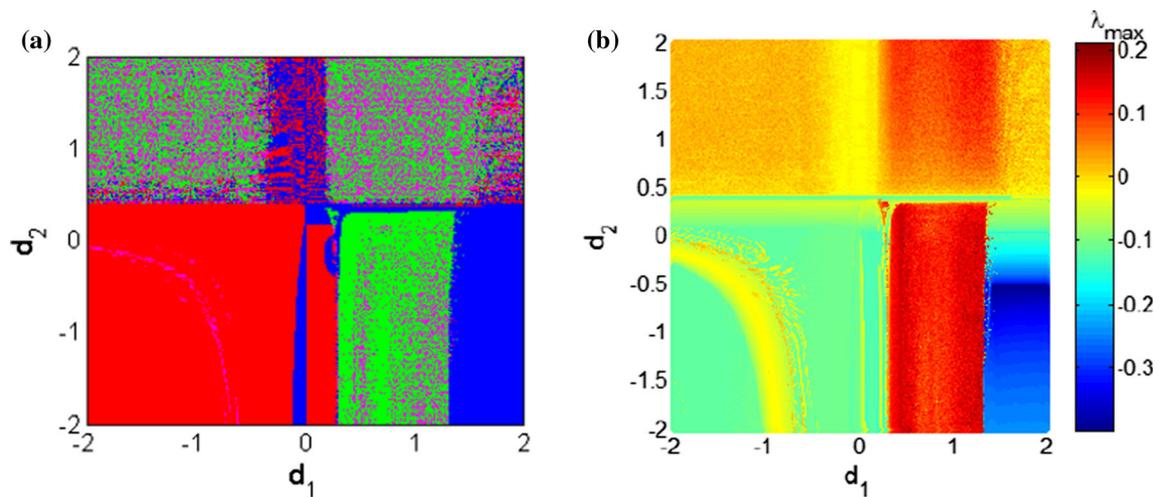


Fig. 11 Influence of the variation in two parameters (d_1, d_2) , to the dynamical behavior

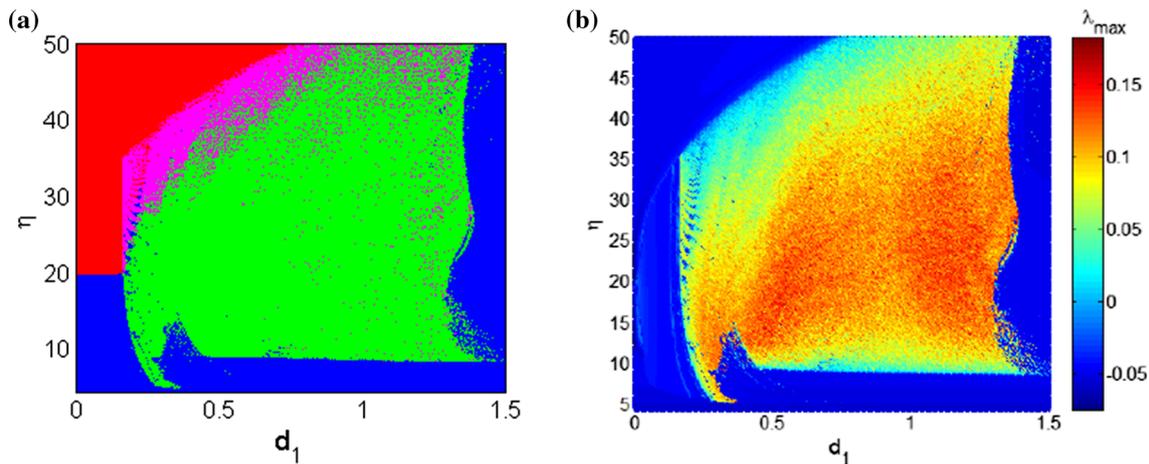


Fig. 12 Influence of the variation in two parameters (d_1, η), to the dynamical behavior

$$\begin{cases} 0 = -a_1x_1 + b_{11}f(x_1) + b_{12}f(x_2) + y_1 \\ 0 = -a_2x_2 + b_{21}f(x_1) + b_{22}f(x_2) + y_2 \\ 0 = -c_1y_1 - d_1f(x_1) \\ 0 = -c_2y_2 - d_2f(x_2) \end{cases} \quad (7)$$

After developments and arrangements of the system (7), Eq. 8 is obtained.

$$\begin{cases} x_{2e}^n = \frac{a_1}{a_2b_{12}} \left(b_{22} - \frac{d_2}{c_2} \right) x_{1e}^n - \frac{f(x_{1e}^n)}{a_2b_{12}} \\ \left[\left(b_{11} - \frac{d_1}{c_1} \right) \left(b_{22} - \frac{d_2}{c_2} \right) - b_{12}b_{21} \right] \\ y_{1e}^n = \frac{-d_1}{c_1} f(x_{1e}^n) \\ y_{2e}^n = \frac{-d_2}{c_2} f(x_{2e}^n) \end{cases} \quad (8)$$

The equilibrium points $P_n = (x_{1e}^n, x_{2e}^n, y_{1e}^n, y_{2e}^n)$ (trivial and non-trivial) are obtained by expression (9) which is graphically solved using a MATLAB function [12–14, 28]:

Remark that $n \in \mathbb{N}$ is the numbering index of equilibrium points x_{1e}^n , which corresponds to the graphical intersections with the abscissa of the solution curve represented by Eq. (9):

$$S(x_{1e}) = -a_1x_{1e} + b_{11}f(x_{1e}) + b_{12}f(x_{2e}) - \frac{d_1}{c_1}f(x_{1e}) \quad (9)$$

where

$$\begin{aligned} x_{2e} = & \frac{a_1}{a_2b_{12}} \left(b_{22} - \frac{d_2}{c_2} \right) x_{1e} \\ & - \frac{f(x_{1e})}{a_2b_{12}} \left[\left(b_{11} - \frac{d_1}{c_1} \right) \left(b_{22} - \frac{d_2}{c_2} \right) - b_{12}b_{21} \right] \end{aligned} \quad (10)$$

when the learning rate $d_2 = 0.3$, the curve of Fig. 2a shows three and seven intersections with the X-axis for $d_1 < 0.0283$ and $d_1 > 0.0283$, respectively. In Fig. 2b, we get seven intersections for the learning $d_2 < 0.3$ and three intersections for $d_2 > 0.3$ with the X-axis when we fix the learning rate $d_1 = 0.0283$. The two curves (Fig. 2a, b) have five intersections with the X-axis, for the critical values $(d_1, d_2) = (0.0283, 0.3)$. Let us specify that the number of the intersection of the solution curve with the X-axis coincides with the number of the equilibrium point. Thus, the intersection of three, five, and seven of the solution curve with the X-axis denotes the presence of three, five, and seven equilibrium points, respectively, in the model (4). From this observation, it is easy to choose which parameter should be modified to obtain a significant effect on the dynamics of the model (4).

The Jacobian matrix derived from Eq. (4) for equilibrium points $P_{n=0,1,2,3,4,5,6}$ is given in (11):

$$J = \begin{bmatrix} -a_1 + b_{11}g_1 & b_{12}g_2 & 1 & 0 \\ b_{21}g_1 & -a_2 + b_{22}g_2 & 0 & 1 \\ -d_1g_1 & 0 & -c_1 & 0 \\ 0 & -d_2g_2 & 0 & -c_2 \end{bmatrix} \quad (11)$$

The characteristic equation associated with (11), specified in Eq. (12), is obtained from the MATLAB software:

$$\det(J - \lambda I_4) = \alpha_4\lambda^4 + \alpha_3\lambda^3 + \alpha_2\lambda^2 + \alpha_1\lambda^1 + \alpha_0 = 0 \quad (12)$$

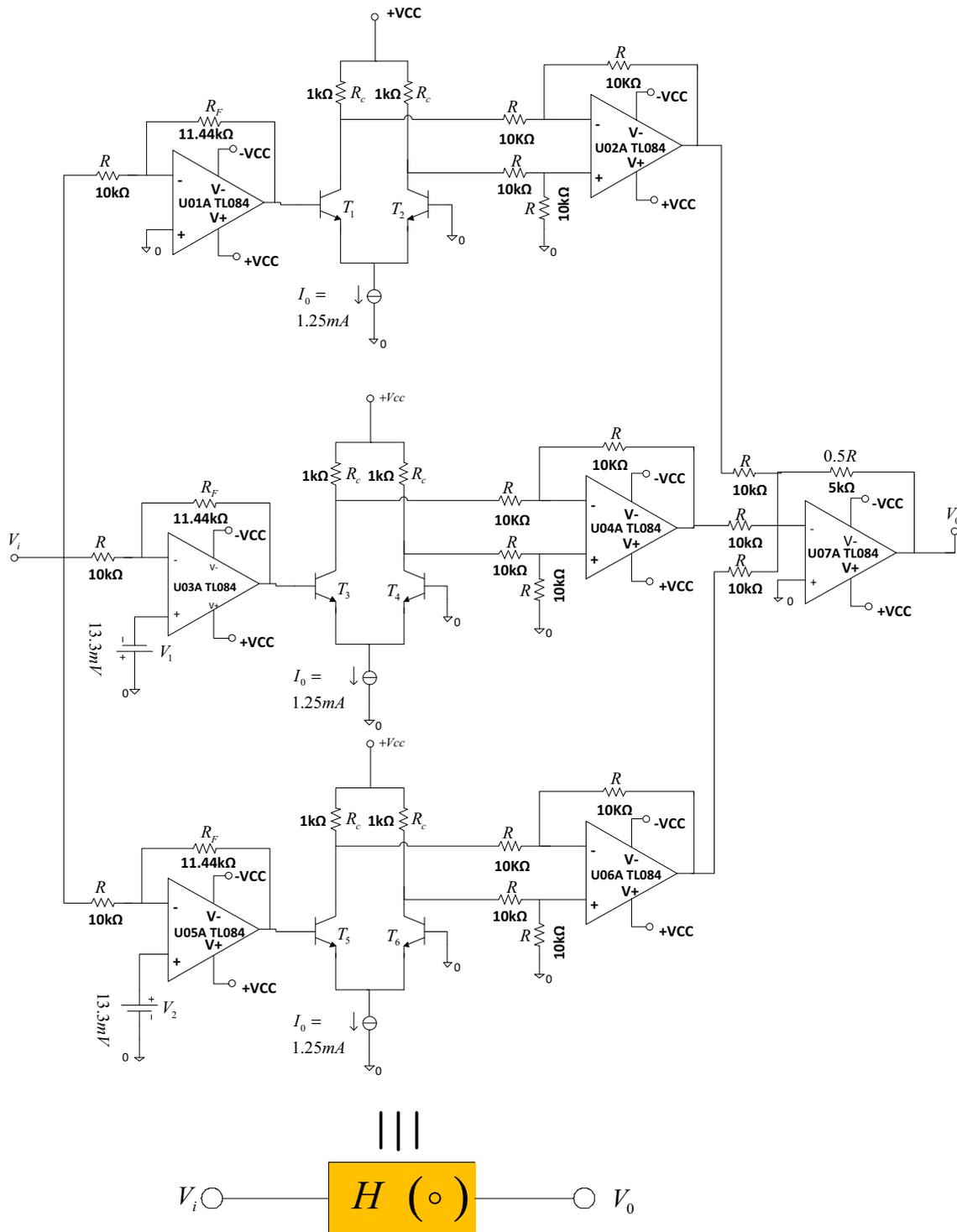


Fig. 13 Synthesized circuit of the approximate activation function using hyperbolic tangent modules

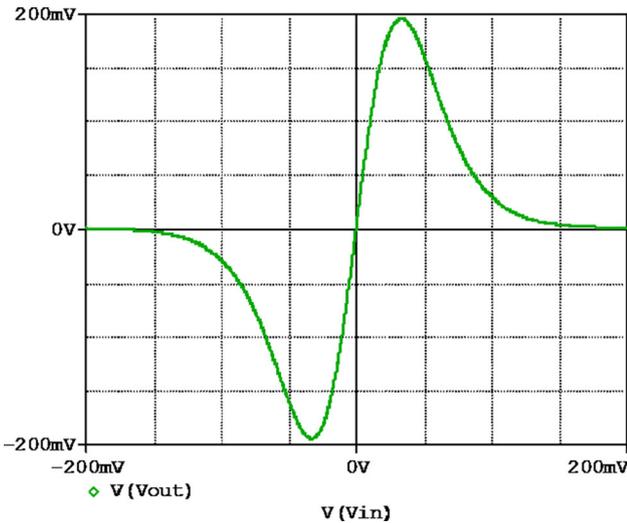


Fig. 14 Capture in Pspice of the approximation of the compound activation function given in Fig. 13, effectively validating the numerical results of Fig. 1a

$$\begin{cases}
 \alpha_4 = 1 \\
 \alpha_3 = a_1 + a_2 + c_1 + c_2 - b_{11}g_1 - b_{22}g_2 \\
 \alpha_2 = a_1a_2 + a_1c_1 + a_1c_2 + a_2c_1 + a_2c_2 + c_1c_2 + d_1g_1 + d_2g_2 \\
 \quad - a_2b_{11}g_1 - a_1b_{22}g_2 - b_{11}c_1g_1 \\
 \quad - b_{11}c_2g_1 - b_{22}c_1g_2 - b_{22}c_2g_2 + b_{11}b_{22}g_1g_2 - b_{12}b_{21}g_1g_2 \\
 \alpha_1 = a_1a_2c_1 + a_1a_2c_2 + a_1c_1c_2 + a_2c_1c_2 + a_2d_1g_1 + a_1d_2g_2 \\
 \quad + c_2d_1g_1 + c_1d_2g_2 - a_2b_{11}c_1g_1 \\
 \quad - a_2b_{11}c_2g_1 - a_1b_{22}c_1g_2 - a_1b_{22}c_2g_2 - b_{11}c_1c_2g_1 - b_{22}c_1c_2g_2 \\
 \quad - b_{11}d_2g_1g_2 + b_{22}d_1g_1g_2 \\
 \quad + b_{11}b_{22}c_1g_1g_2 - b_{12}b_{21}c_1g_1g_2 + b_{11}b_{22}c_2g_1g_2 - b_{12}b_{21}c_2g_1g_2 \\
 \alpha_0 = a_1a_2c_1c_2 + a_1c_1d_2g_2 + a_2c_2d_1g_1 + d_1d_2g_1g_2 - a_2b_{11}c_1c_2g_1 \\
 \quad - a_1b_{22}c_1c_2g_2 - b_{11}c_1d_2g_1g_2 \\
 \quad - b_{22}c_2d_1g_1g_2 + b_{11}b_{22}c_1c_2g_1g_2 - b_{12}b_{21}c_1c_2g_1g_2
 \end{cases} \tag{13}$$

where $g_j = f'(\bar{x}_j) = \left[\eta + 2\eta \left(\frac{\eta \bar{x}_j}{\sigma} \right)^2 \right] \exp \left[- \left(\frac{\eta \bar{x}_j}{\sigma} \right)^2 \right]$
 $j = [1, 2]$ (14)

The coefficients of the characteristic polynomial (12) are all nonzero. The parameters (d_1, d_2) will be $d_2 = 0.3$ and $d_1 =$ tuneable in the rest of the document unless otherwise indicated. For some discrete values of the control parameter d_1 , the equilibrium points and Eigenvalues are determined, with their stabilities in Table 2. Based on the results displayed on that table, the first condition to have self-excited dynamics from the TNL neuron model is verified since all the equilibria are unstable.

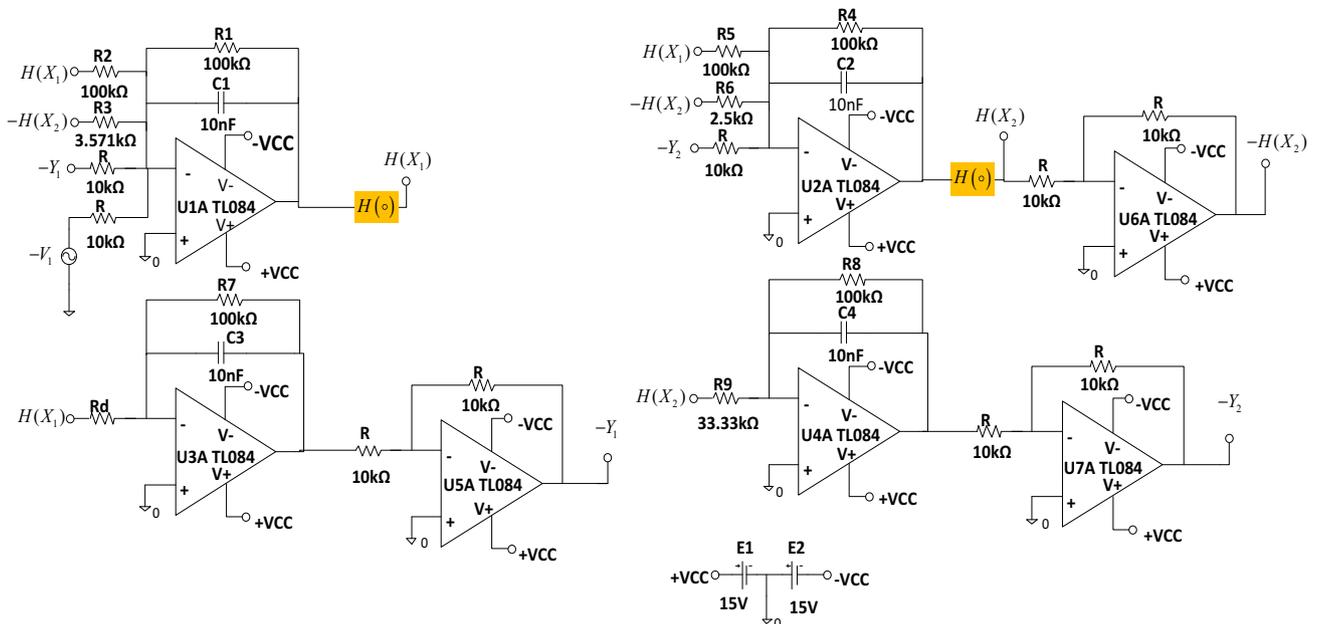


Fig. 15 Analog circuit of non-autonomous tabu learning neuron with two neurons

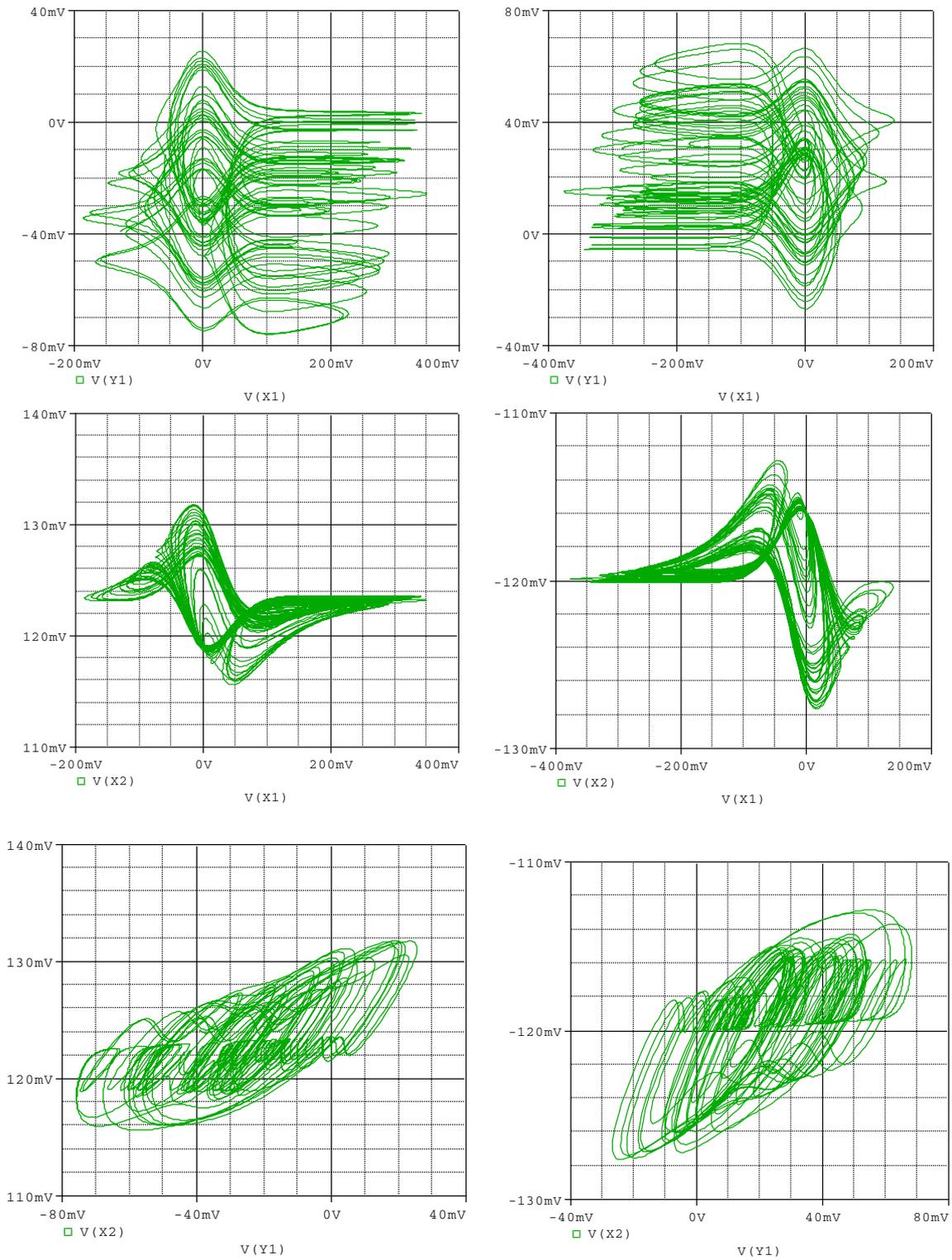


Fig. 16 Representation in Pspice of the complexity of symmetrical chaotic attractors for $R_d = 38 \text{ k}\Omega$ in different planes. These attractors were obtained for initial conditions $V(X_1(0), X_2(0), Y_1(0), Y_2(0)) = (0, 0, \pm 0.9, 0)$

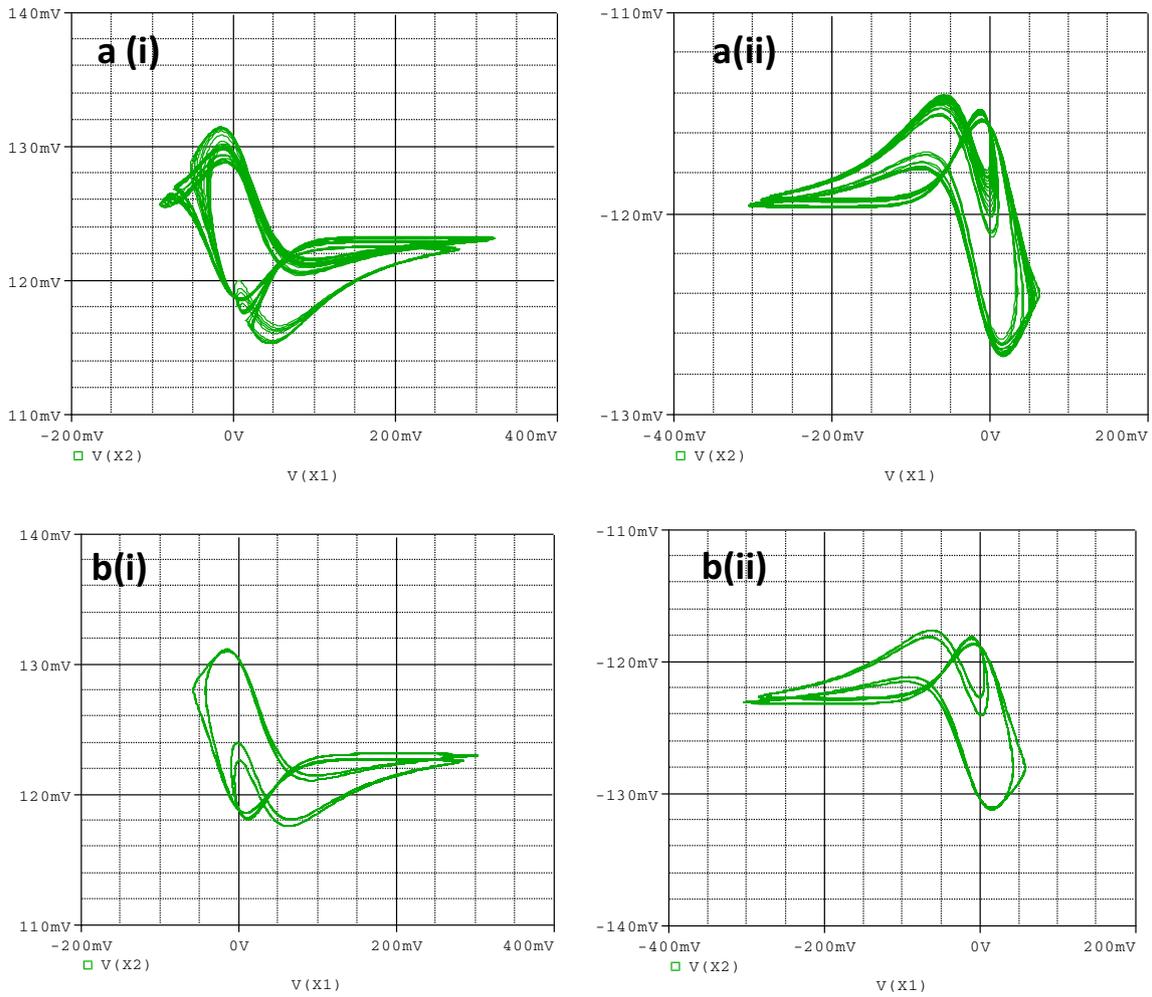


Fig. 17 Representation in Pspice of the coexistence of four symmetrical attractors for $R_d = 51\text{ k}\Omega$ in: **a** period-4 limit cycles (Low and Upper) and **b** chaotic attractors (Low and Upper). These attractors were obtained for initial conditions $V(X_1(0), X_2(0), Y_1(0), Y_2(0)) = (0, 0, \pm 0.1, 0)$ and $(0, 0, \pm 0.9, 0)$, respectively

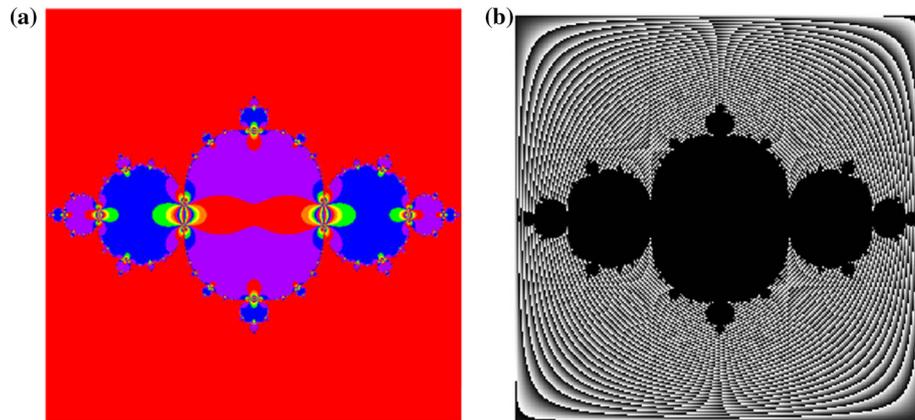


Fig. 18 Visualization of **a** the traditional Julia set for $c = -0.745429$ and **b** the modified Julia set for $c = -0.745429$ and $a = 25$

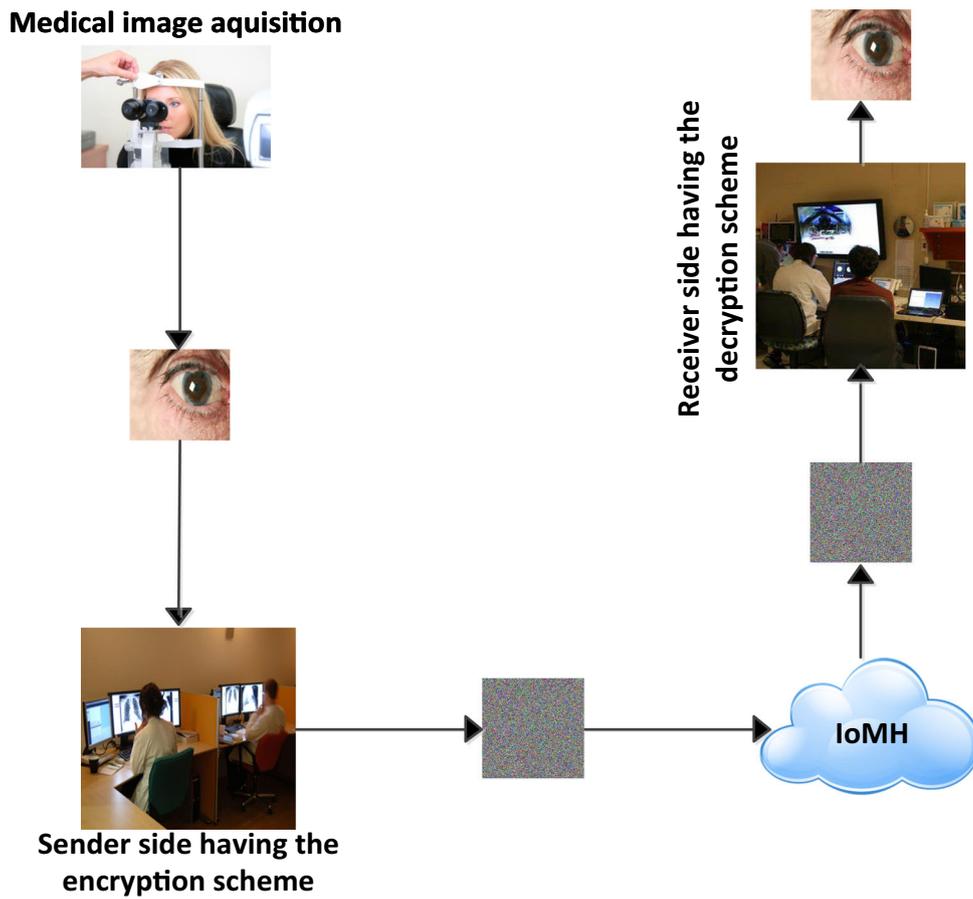


Fig. 19 Structure of the secure IoMT system

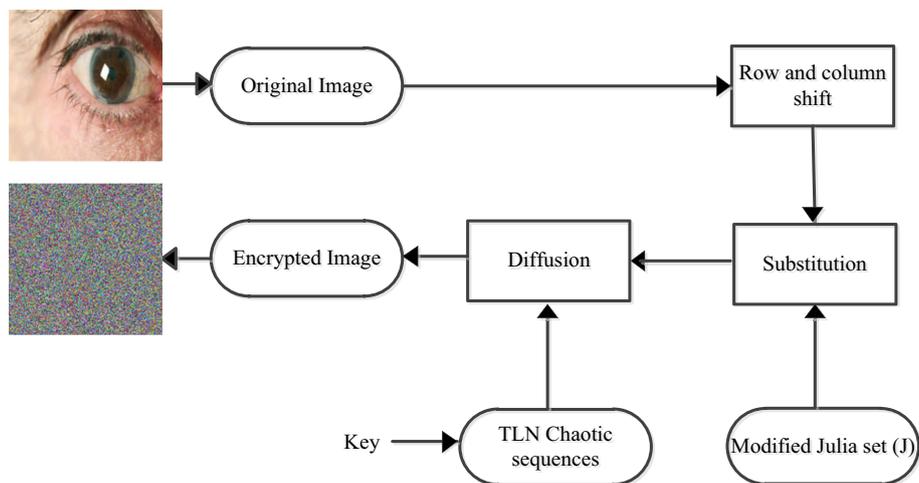


Fig. 20 Structure of the cryptosystem

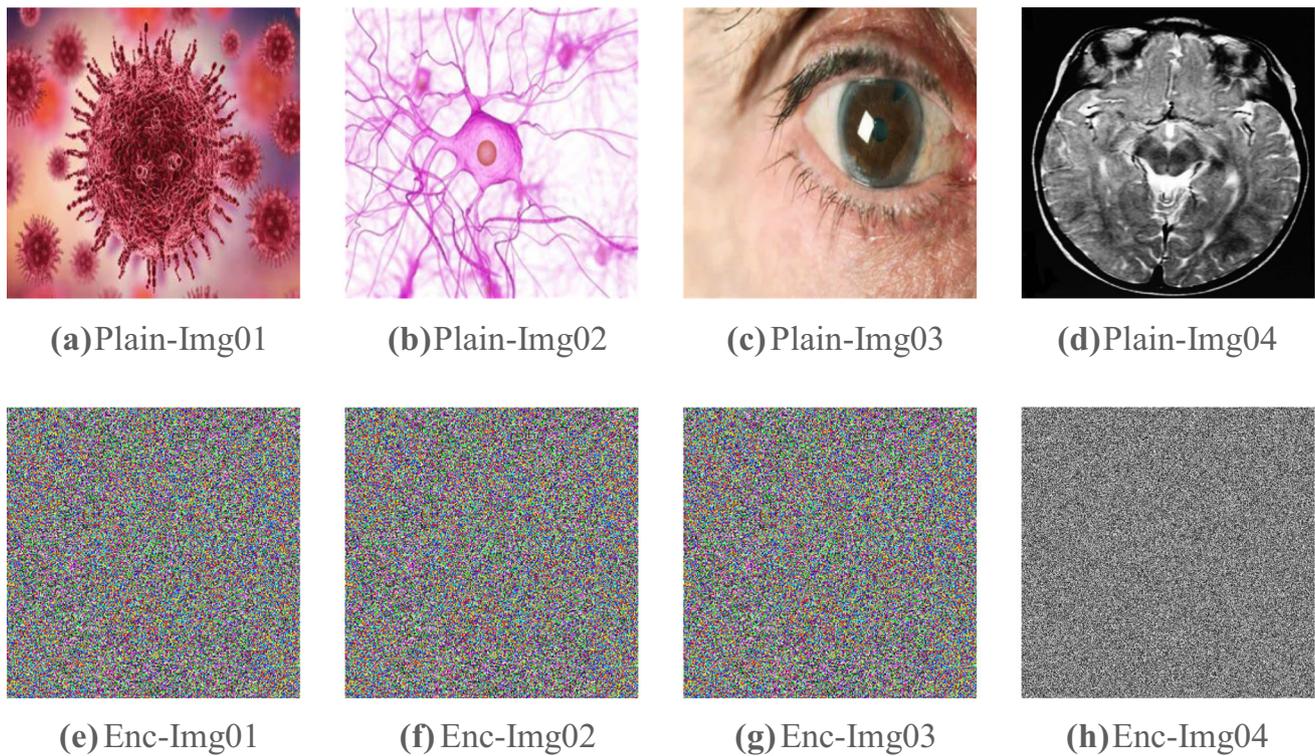


Fig. 21 Visual test of the dataset images. It is observed that the plain medical images are no more recognizable after encryption

3 Numerical results

The numerical approach is done based on the nonlinear dynamical systems analysis tools, namely bifurcation diagrams and Lyapunov exponents, the phase portrait, the basin of attraction, etc. These tools could be implemented thanks to the Turbo Pascal environment for numerical calculations and the results of these calculations represented from the MATLAB environment. The numerical integrations are based on the 4th-order Runge–Kutta algorithm for the precision and the speed of convergence which it offers with an integration time interval $0 \leq \tau \leq 5500$ choice for an integration time step $\Delta\tau = 0.005$. The bifurcation diagram plot consists of taking the local maxima of the neuron state variable when weeping the control parameter d_1 , performed on the integration time interval $5000 \leq \tau \leq 5500$ for a time step $\Delta\tau = 0.005$. The plot of the Lyapunov exponent diagram is done using the Wolf method in the time interval $0 \leq \tau \leq 5000$ with the time step $\Delta\tau = 0.005$ [41].

3.1 Bifurcation diagrams and phase portraits

The superimposition of the bifurcation diagrams in Fig. 3a shows the effect induced by the variation in the learning rate d_1 on the dynamics of the model (4) in the range

$0 \leq d_1 \leq 1$. On this diagram, there is an appearance of the zones of periodic behaviors, followed by zones of chaotic behavior and periodic windows separated by chaotic zones of behavior by complex bifurcations. Note also that the road to chaos begins through a period-doubling scenario when the learning rate d_1 increases. During his excursion, there is the appearance of the phenomenon of crisis followed by a new doubling of the period toward chaos. This crisis is due to the fact that our system suddenly switches from chaotic behavior to regular oscillations from a given value of the learning rate d_1 . There are several crisis configurations specified in [42, 43].

According to [42, 43], the crisis occurs because an unstable fixed point or an unstable limit cycle “collides” with the chaotic attractor as some system control parameters are changed. The diagrams of the Lyapunov spectrum (Fig. 3b) drawn from the Wolf algorithm [41] confirm these scenarios of bifurcation observed. The initial conditions used for these diagrams are summarized in Table 2. For some discrete values of the learning rate d_1 , some phase portraits have been represented including limit cycles and chaotic attractors in different planes. For example, in Fig. 4, the projections of the symmetric chaotic attractors are well done in the (x_1, y_1) , (x_1, x_2) and (y_1, x_2) planes for $d_1 = 0.28$ which exhibits the complexity of the system. An enlargement of the bifurcation diagram

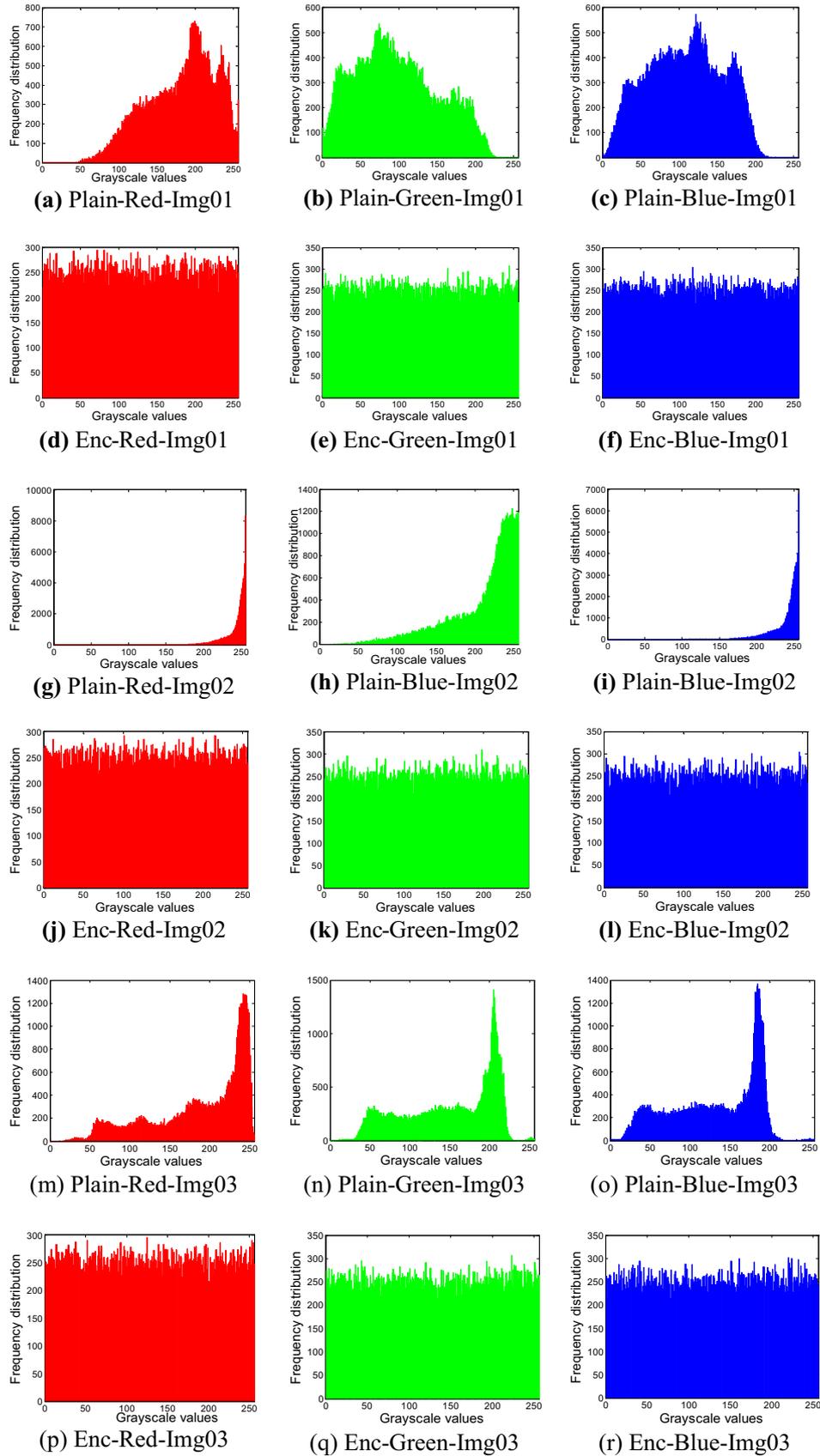
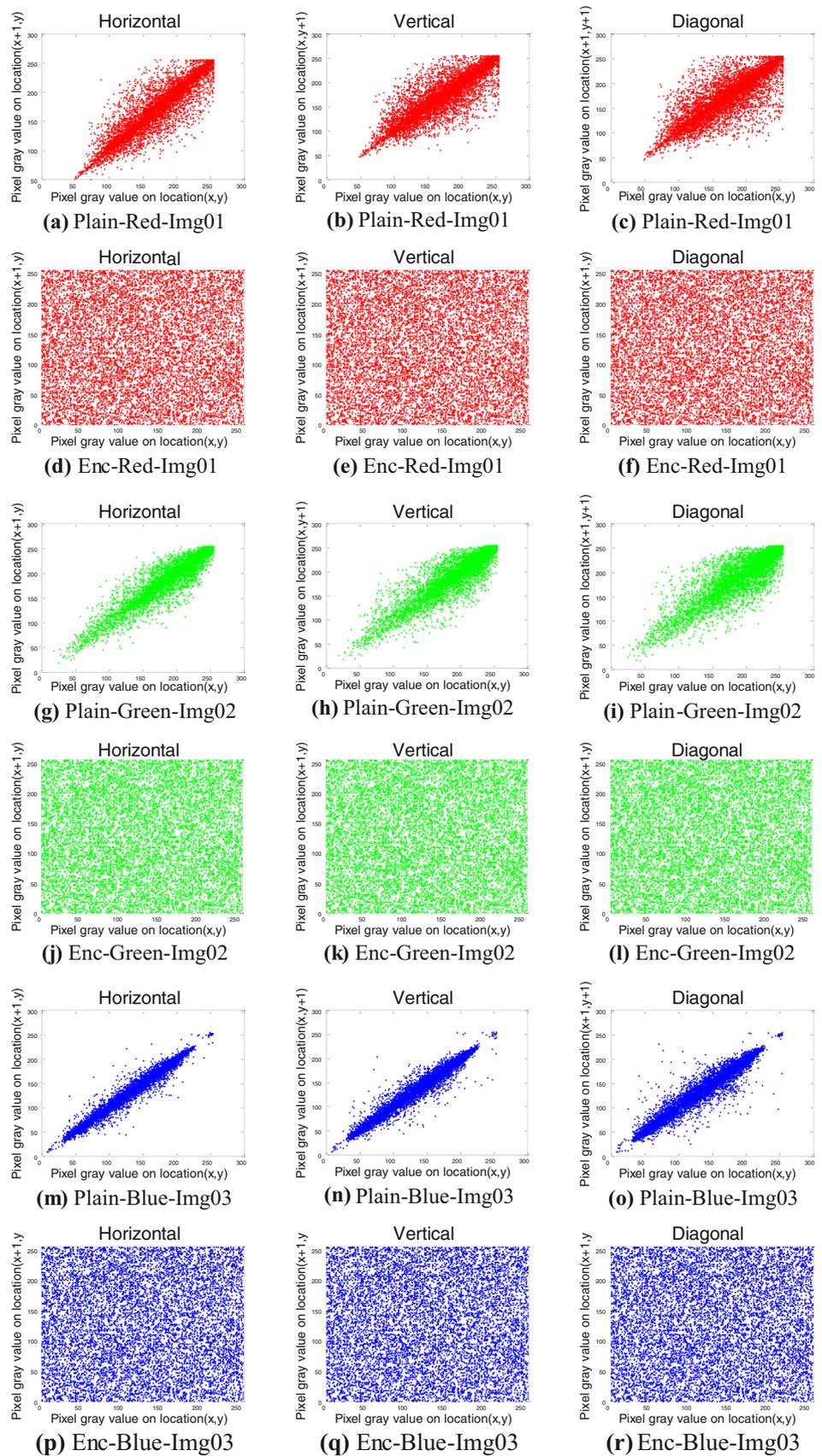


Fig. 22 Histograms for each plain data set and its corresponding cipher

Fig. 23 Distribution of correlation for plain data set and corresponding cipher



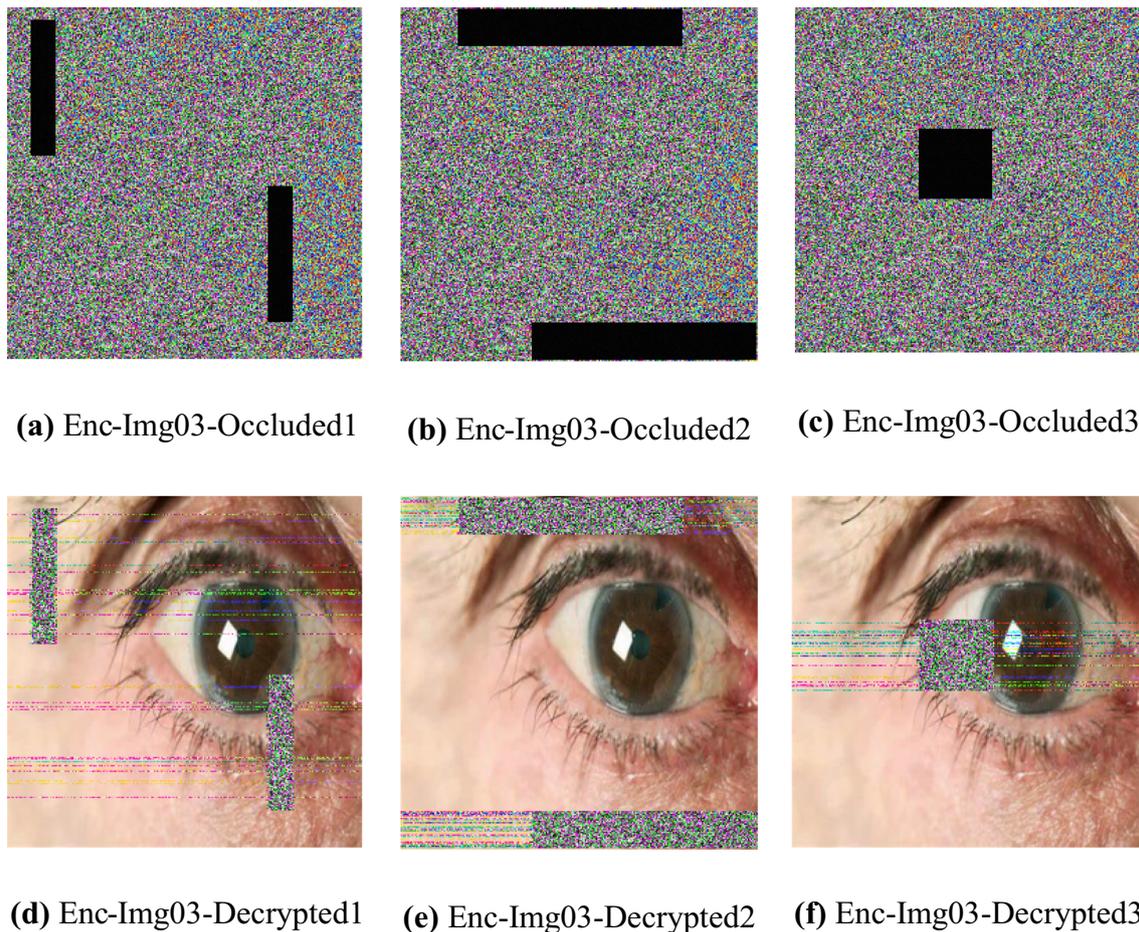


Fig. 24 Img03 is encrypted, occluded and decrypted with success to illustrate occlusion attacks

of Fig. 3a is given in Fig. 5 for the control parameter d_1 belonging to the range $0.154 \leq d_1 \leq 0.304$. In this figure, we can see the coexistence of three bifurcation diagrams, the presence of parallel branches' and the crisis phenomena. The representation of the bifurcation diagram of Fig. 5 in the range $0.1881 \leq d_1 \leq 0.1969$ in Fig. 6 showed clearly that these parallel branches' coexist four in number. The enlargement of Fig. 3 in the interval $0.279 \leq d_1 \leq 0.281$ presented in Fig. 7 shows the coexisting of other diagrams in this region. These diagrams argue the existence of the phenomenon of multistability in these ranges. The initial methods and conditions for obtaining these diagrams are given in Table 3.

3.2 Coexistence of attractors and basins of attraction

The notion of multistability or coexistence of multiple attractors is a very important phenomenon in chaotic

dynamical systems. Among others, for the flexibility, it offers to the system and its adapted applications in information engineering [12]. Multistability has caught the attention of many researchers in recent years [44], because it encompasses a diversity of many stable states in a system.

The study of the coexistence of attractors in the HNNs would allow understanding in depth its dynamical effect on the aspects of the treatment of the brain information and the cognitive function [12]. We have exhibited through the TLTN presented in (4) the coexistence of four (periodic–chaotic), four (chaotic–chaotic) and six (periodic–chaotic) attractors, considering the values of the control parameters $d_1 = 0.196$, $d_1 = 0.2089$ and $d_1 = 0.28$, respectively, presented in Figs. 8a, b, 9a, b, and 10a–c, respectively. Different types of coexisting attractors and the initial conditions making it possible to obtain them are specified in these figures. The basins of attraction corresponding to the coexistence of four (periodic–chaotic), four (chaotic–

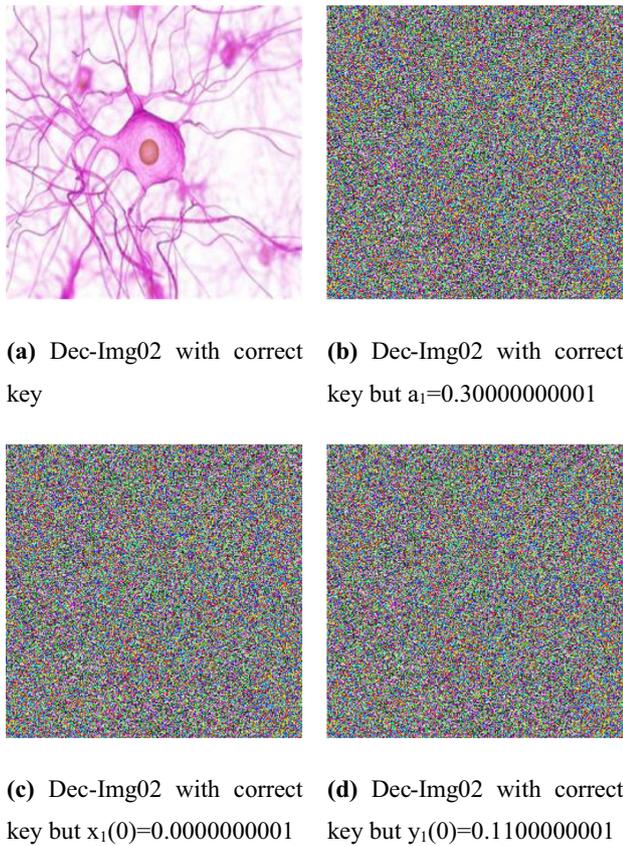


Fig. 25 Img02 is decrypted with various keys to illustrate key sensitivity

chaotic), and six (periodic-chaotic) symmetric attractors are shown in Figs. 8c, d, 9c, d and 10d, respectively. On these attraction basins, each color corresponds to a set of initial conditions that enable to obtain each coexisting attractor. These basins of attraction correspond to those of the self-excited attractors because they intercede with the open neighborhood of other equilibrium points. In contrast, hidden attractors have basins of attraction that do not cross the open neighborhood of the other points of equilibrium. Note that the coexistence of these four and six attractors is unprecedented; these numbers have nowhere been reported in the literature of tabu learning neuron, whose maximum number so far was two [9, 10]. This demonstrates to the satisfaction that this considered compound activation function makes the model (4) more complex and interesting than those already existing using other activation functions.

3.3 Dynamics of tabu learning neuron in the parameter space

Figures 11 and 12 show the effect of the variation in two different parameters particularly the learning rate on the

dynamics of model (4) exposed in the parameter space in (d_1, d_2) and (η, d_1) planes, respectively.

We can note from Figs. 11a and 12a that the dynamics of model (4) has two main regions, one periodic (in blue and red) and one another chaotic (in green and magenta), where the red and blue colors are used distinguishing periodic attractors of opposite sign. Likewise, the green and magenta colors allow distinguishing the chaotic attractors of the opposite sign. These representations enable us to confirm that model (4) is symmetrical and its dominant behavior for the composition of these parameters. The coded colors contained in Figs. 11b, 12b correspond to values of the maximum exponent of Lyapunov (λ_{\max}); these values are indicated on the graduations of the color bar of the corresponding figures. In these figures, we can appreciate the expansion and reduction in chaotic puff via transitions of the zones of periodic behaviors (where $\lambda_{\max} < 0$) toward areas of chaotic behavior (where $\lambda_{\max} > 0$) and vice versa during the variation in parameters.

4 Circuit design

In this section, the TLTN model (4) will be studied in the form of a circuit or an analog simulator in PSPICE. The analog simulator equivalent to the mathematical model (4) is set up essentially by electronic components. This rigorous and inexpensive strategy is employed because it has been used for experimental studies of some models of dynamic systems [12, 13] or to emulate other complex systems [22, 45–47].

4.1 Analog circuit synthesis for the TLTN

The nonlinear activation function involving in model (4) is an exponential and polynomial-mixed nonlinearity, which is rather complicated and difficult to implement in an analog circuit form. It is for this reason that [9] proposed an approximation function using hyperbolic tangent which is more simple to realize. The equivalent analog circuit provided for this approximation to realize the nonlinear activation function of the TLTN model (4) is given in Fig. 13, which is an equivalent of three hyperbolic tangent modules with different offsets. More details of this approximation of the hyperbolic tangent circuit design can be found in the [9, 11, 48] and his current–voltage characteristic given in Fig. 14. From this figure, it is obvious that the nonlinear characteristic is bounded below and above. The analog circuit of the TLTN model (4) is constituted by referring to [7, 9]; it consists of four integrators involving three inverters in feedback loops and two approximation functions shown in Fig. 15.

Table 3 Methods used to obtain coexisting bifurcation diagrams of Fig. 3 and its enlargements of Figs. 3, 4, 5, 6, 7

Figure	Control parameter range	Color diagram	Scanning direction	Initial starting condition
Figure 3	$0 \leq d_1 \leq 1$	Red	Upward	(0, 0, 1.5, 0)
		Blue	Upward	(0, 0, -1.5, 0)
Figure 5	$0.154 \leq d_1 \leq 0.304$	Red	Upward	(0, 0, 1.5, 0)
		Blue	Upward	(0, 0, -1.5, 0)
		Magenta	Downward	(0, 0, 1.5, 0) fixed
		Green	Downward	(0, 0, -1.5, 0) fixed
Figure 6	$0.1881 \leq d_1 \leq 0.1969$	Red	Upward	(0, 0, 1.5, 0)
		Blue	Upward	(0, 0, -1.5, 0)
		Magenta	Downward	(0, 0, 1.5, 0) fixed
		Green	Downward	(0, 0, -1.5, 0) fixed
Figure 7	$0.279 \leq d_1 \leq 0.281$	Red	Upward	(0, 0, 1.5, 0)
		Blue	Upward	(0, 0, -1.5, 0)
		Magenta	Downward	(0, 0, 1.5, 0) fixed
		Green	Downward	(0, 0, -1.5, 0) fixed
		Cyan	Upward	(0, 0, 0.04, 0) fixed
		Black	Downward	(0, 0, -0.072, 0) fixed

Table 4 Chi-square values for each encrypted test data

Images	χ^2 values				Decision
	R	G	B	Average	
Img01	45,603.9531	22,340.4218	28,854.7500	32,266.3750	Non-uniform
Enc-Img01	257.3984	252.0312	243.7656	251.0651	Uniform
Img02	76,521.1546	11,764.2320	57,778.7289	48,688.0385	Non-uniform
Enc-Img02	201.6640	250.4296	279.2421	243.7786	Uniform
Img03	78,081.5000	60,018.4843	68,009.5078	68,703.1640	Non-uniform
Enc-Img03	220.8046	253.7578	270.8437	248.4687	Uniform
Img04	–	–	–	1,095,300	Non-uniform
Enc-Img04	–	–	–	261.0249	Uniform

Table 5 Correlation coefficients of each encrypted color test data

Images	Plan	R	G	B
Enc-Img01	H	- 0.0051	- 0.0033	- 0.0027
	V	0.0051	0.0031	0.0045
	D	- 0.0017	- 0.0024	- 0.0005
Enc-Img02	H	- 0.0051	- 0.0026	- 0.0034
	V	0.0052	0.0038	0.0021
	D	- 0.0003	- 0.0014	- 0.0001
Enc-Img03	H	- 0.0040	- 0.0043	- 0.0035
	V	0.0024	0.0036	0.0038
	D	- 0.0009	- 0.0033	0.0002

Table 6 NPCR and UACI of each encrypted test data

Images	NPCR (%)	UACI (%)
Img01	99.6200	33.6512
Img02	99.6256	33.6147
Img03	99.6098	33.5823
Img04	99.5947	33.6019

$$\begin{cases}
 C_1 \frac{dX_1}{dt} = -\frac{X_1}{R_1} - \frac{1}{R_2}f(X_1) + \frac{1}{R_3}f(X_2) + \frac{Y_1}{R} + \frac{A_m}{R}\sin(2Pif_1t) \\
 C_2 \frac{dX_2}{dt} = -\frac{X_2}{R_4} - \frac{1}{R_5}f(X_1) + \frac{1}{R_6}f(X_2) + \frac{Y_2}{R} \\
 C_3 \frac{dY_1}{dt} = -\frac{Y_1}{R_7} - \frac{1}{R_{d_1}}f(X_1) \\
 C_4 \frac{dY_2}{dt} = -\frac{Y_2}{R_8} - \frac{1}{R_9}f(X_2)
 \end{cases}
 \tag{15}$$

Table 7 Global and local entropy of each encrypted test data

Images	Global entropy	Local entropy
Enc-Img01	7.9990	7.9051
Enc-Img02	7.9989	7.9059
Enc-Img03	7.9992	7.9083
Enc-Img04	7.9978	7.9071

where X_1, X_2, Y_1 and Y_2 denote capacitor voltages C_1, C_2, C_3 and C_4 , respectively, with $C_1 = C_2 = C_3 = C_4 = C = 10$ nF and $R = 10$ kΩ.

The set of Eq. (15) is equivalent to (4) considering the following equalities:

Table 8 Computational time (in milliseconds) for various size test images and comparison with existing works

Algorithm	Img01			Img02			Img03		
	256 × 256	512 × 512	1024 × 1024	256 × 256	512 × 512	1024 × 1024	256 × 256	512 × 512	1024 × 1024
Proposed	2.03	5.18	15.27	1.82	4.07	12.91	3.13	7.28	15.27
[63]	7.79	31.10	124.64	5.82	28.09	120.42	9.80	38.07	129.43
[64]	4.60	18.06	54.35	3.86	12.49	50.15	8.21	27.92	61.02
[65]	1270	5070	20,560	986.05	40,256	17,285	1586	8459	25,785

Table 9 ET and NC computed with 512 × 512 × 3 bytes version of Img01

Algorithm	ET (MBps)	NC
Proposed	151.82	15.80
[63]	24.06	122.85
[64]	41.52	62.00
[65]	0.14	94.60

Table 10 Comparative analysis for Img01 in terms of computational time (CT), Encryption Throughput (ET), Number of Cycles (NC), NPCR, UACI, χ^2 -values, entropy and key sequence

Algorithm	CT(ms)	ET (MBps)	NC	NPCR	UACI	Entropy		Key sequence
						Global	Local	
Proposed	5.18	151.82	15.80	99.6200	33.6512	7.9990	7.9051	TLN
[63]	31.10	24.06	122.85	99.6340	33.5800	7.9994	7.9153	Arnold map
[64]	18.06	41.52	62.00	99.6093	33.4480	7.9996	7.9073	1-D map
[63]	5070	0.14	94.60	99.6184	33.6157	7.9981	7.9027	Logistic map
[64]	NR	NR	NR	99.6500	33.4600	7.9969	NR	CNN
[65]	NR	NR	NR	99.6170	33.4360	7.997	NR	HNN

NR refers to Not Reported

$$\begin{aligned}
 t &= \tau RC, X_i, Y_i = x_i, y_i (i = 1, 2), \\
 R_1 &= \frac{R}{|a_1|} = 100 \text{ k}\Omega, R_2 = \frac{R}{|b_{11}|} = 100 \text{ k}\Omega, \\
 R_3 &= \frac{R}{|b_{12}|} = 3.571 \text{ k}\Omega, \text{ and} \\
 R_4 &= \frac{R}{|a_2|} = 100 \text{ k}\Omega, R_5 = \frac{R}{|b_{21}|} = 100 \text{ k}\Omega, \\
 R_6 &= \frac{R}{|b_{22}|} = 2.5 \text{ k}\Omega \\
 R_7 &= \frac{R}{|c_1|} = 100 \text{ k}\Omega, R_8 = \frac{R}{|c_2|} = 100 \text{ k}\Omega, \\
 R_9 &= \frac{R}{|d_2|} = 3.33 \text{ k}\Omega \\
 A_m &= 0.2V, \text{ and } f_1 = \frac{F_1}{RC} = 20 \text{ kHz}
 \end{aligned} \tag{16}$$

4.2 Validation by PSPICE analog simulation

The implementations of the circuit design of Fig. 13 and the analog circuit of Fig. 15 in PSPICE have led to the results of Fig. 14 and Figs. 16, 17, respectively. The curve in Fig. 14 is the trace of the approximation of the compound activation function whose circuit design is given in Fig. 13. This curve was represented by considering $V_i = 5 \sin(100\pi t)$. By comparing it with that of Fig. 1a obtained for $(\eta, \sigma^2) = (10, 0.2)$, we see that they coincide which validates the circuit design of Fig. 13 proposed by [9]. The symmetrical chaotic attractors shown in Fig. 16 in various planes are obtained through Fig. 15 for $R_d = 38 \text{ k}\Omega$ and the initial conditions $V(X_1(0), X_2(0), Y_1(0), Y_2(0)) = (0, 0, \pm 0.9, 0)$. These attractors are similar to those obtained during the numerical investigations in Fig. 4. The coexisting of four symmetrical chaotic attractors in Fig. 17 was obtained using the analog circuit of Fig. 15 for $R_d = 51 \text{ k}\Omega$ and the initial conditions $V(X_1(0), X_2(0), Y_1(0), Y_2(0)) = (0, 0, \pm 0.1, 0); (0, 0, \pm 0.9, 0)$. These coexisting attractors are close to that of Fig. 6. All these results were obtained by choosing the Final step: 500 ms; No-Print Delay: 480 ms; Step Ceiling: 2 μ s.

5 Application to secure biomedical images in IoMT

5.1 Modified Julia set

Julia set is a fractal of complex numbers considered as input whose output through a quadratic function $f(z) = z^2 + c$ is bounded [49]. Here c is a complex constant. The function $f(z)$ is initialized and iterated. Setting the real values of the complex number z as the x pixel index and the

imaginary values of the complex number z as the y pixel index, the Julia set can be visualized for different values of the complex constant c .

5.2 Encryption process

An issue of this visualization for the traditional complex Julia set is illustrated in Fig. 18a for $c = -0.745429$. If (x, y) is the location of pixels in Fig. 18a, the intensity values of the image representing the modified complex Julia set are computed as $I(x, y) = [a * (x/y)] \text{ mod } 256$ when the values of pixels of Fig. 18a are 0 representing the red pixel. An issue of the visualization of the image representing the modified complex Julia set is illustrated in Fig. 18b for $c = -0.745429$ and $a = 25$. Internet of things (IoT) is a new prominent research field where objects, sensors, and the internet are jointly interconnected to form a unique system [50]. The main objective of this interconnection is to solve common life problems. IoT can be used to provide a set of healthcare services and this is defined as the internet of healthcare things (IoHT) or the internet of medical things (IoMT) [51]. One of the most important services of IoHT is the communication of medical data of patients including medical images. As medical images contain very confidential data, we provide in this paper lightweight encryption/decryption technics useful to secure medical images in the IoMT. Figure 19 presents the Structure of the secure IoHT system where a medical sensor is used for medical image acquisition. The acquired image is encrypted using the structure of Fig. 20 and transmitted in the IoMT. The receiver uses the decryption scheme to recover the original image for analysis. The encryption algorithm can be detailed in the following five steps, and the decryption procedure is the reverse of the encryption scheme:

Step 1: Read the original image I at the output of the acquisition system. This is $256 \times 256 \times 3$ image or $256 \times 256 \times 1$ image. Apply row and column rotation on each red, green and blue component to obtain R, G and B matrixes.

Step 2: Using the output matrix of the Julia set J , substitute the values of each component R, G and B to obtain S_R, S_G and S_B .

Step 3: Select the initial values $(x_{10}, x_{20}, y_{10}, y_{20}, a_1, a_2, c_1, c_2, I_2, I_m, F_1, \eta, \sigma, b_{11}, b_{12}, b_{21}, b_{22})$ as indicated in Eq. 6 for iterating the presented TLTN model for $256 \times 256 \times 3$ times for color images and $256 \times 256 \times 1$ times for gray scale images to outcome four real sequences $\{X1\}, \{X2\}, \{Y1\}$ and $\{Y2\}$. Then convert the real sequence $X1$ to integer as $X = \text{fix}(X1_i \times 10^{16} \text{ mod } 256)$.

Step 4: Perform Bit-XORed-based diffusion process on the diffused matrixes S_R , S_G and S_B using the sequence X of the TLTN model.

Step 5: Perform components fusion to obtain the final encrypted image from S_R , S_G and S_B .

5.3 Performance study of the proposed cryptosystem

To validate any cryptosystem performance, analysis is required [52]. In this work, we will use some well-known metrics such as histogram, chi-square, correlation coefficients, NPCR, UACI, local and global entropy, occlusion and key sensitivity to assess the proposed cryptosystem. Experimentations are performed on a workstation equipped with Intel Core™ i7-3630QM, 16 GB RAM, and provided with MATLAB R2016b. Our dataset consists of three medical images each of size $256 \times 256 \times 3$ (see Fig. 21) obtained from various medical image sources including the COVID-CT database [53] which is the most important database of COVID-19 Computed Tomography (CT) images available for the public. Setting the initial seed for iterating the presented TLTN model as in Eq. 6, Fig. 21 shows that the input medical images are no more recognizable after encryption and can be sent to the transmission system securely. But we still need to access the security or performance of the transmitted (encrypted) images given that visual test is not a sufficient condition of security [54].

5.3.1 Histogram and Chi-square tests

Any good encryption scheme must pass the histogram and chi-square test to be able to resist the statistical intrusion of a third party [55, 56]. The histogram of a plain image is usually distributed randomly, whereas the histogram of the corresponding cipher is required to be uniform. Figure 22 presents the histograms of the plain and cipher medical images. It is obvious to observe that the histograms of the plain image are randomly distributed while the histograms of the encrypted data are flat. This flatness can be checked using the Chi-square test. Table 4 provides the issue of Chi-square values with 0.05 as the weight value. Usually, the flatness of the histogram is validated if the Chi-square value of the test sample is less than 293.2478 indicating a p -value higher than 0.5. Regarding Table 4, the histograms of various test samples are validated.

5.3.2 Correlation coefficients

If the correlation coefficients of the cipher data are very close to zero, it is sure that the proposed cryptosystem can resist all form statistical attacks [57, 58]. The correlation coefficient is computed by a random selection of 10^4 pairs

of neighboring pixels in each plan of the image. The following formula is usually applied:

$$C_{mn} = \frac{\sum_{x=1}^A (m_x - \bar{m})(n_x - \bar{n})}{\sqrt{\sum_{x=1}^A (m_x - \bar{m})^2 \sum_{x=1}^A (n_x - \bar{n})^2}} \tag{17}$$

here m_x, n_x point out the values of adjacent pixels and A points out the whole amount of nearby pixel pairs (10^4). Table 5 provides the outcome of computations of the correlation coefficients of each encrypted medical image in the red (R), green (G), and blue (B) components. We can observe that the correlation coefficients of the cipher data are very close to zero; consequently, we are sure that the proposed cryptosystem is able to resist all forms of statistical attacks. The distribution of correlation coefficients for the plain data set and corresponding cipher is also represented in Fig. 23. As linear relation is observed for plain images, it is evident that the pixels of these images are highly correlated. On the other hand, no linear relation is observed in the case of cipher images. This simply implies that there is no correlation between the pixels of the encrypted images. Consequently, no profitable information can be retrieved from these encrypted data.

5.3.3 NPCR and UACI tests

To assess the capability of an encryption algorithm to withstand differential attacks, NPCR (“Number of Pixels Change Rate”) and UACI are commonly used [59]. These metrics evaluate the rate of change in the original image on its equivalent cipher one. The numerical value of NPCR is computed as:

$$\text{NPCR} = \frac{\sum_{m,n} \text{Diff}(m,n)}{D} \times 100\%, \tag{18}$$

$$\text{Diff}(m,n) = \begin{cases} 0 & \text{if } P(m,n) = C(m,n) \\ 1 & \text{if } P(m,n) \neq C(m,n) \end{cases}$$

here D indicates to the complete pixel numbers in the image. On the other hand, numerical value of UACI is computed as:

$$\text{UACI} = \frac{100}{m \times n} \sum_1^m \sum_1^n \frac{|IC_1(m,n) - IC_2(m,n)|}{255} \tag{19}$$

where IC_1 and IC_2 are two encrypted images obtained from ciphers images different in just on pixel. m and n are the dimension of the images.

The outcomes of NPCR and UACI for the experimented dataset are displayed in Table 6. From these results, the given encryption approach has a high sensitivity to tiny pixel changes in the original image. Consequently, encrypted images are secured against any form of differential attacks.

5.3.4 Global and local entropy

To estimate the distribution of the pixel values of each plane in the image, local and global entropy is commonly used [60]. Global entropy is computed as:

$$E(Y) = - \sum_{a=1}^{2^b-1} p(y_a) \log_2(p(y_a)) \quad (20)$$

here $p(y_a)$ represents the possibility of y_a and b indicates the pixel bit level, which is equivalent to the typical entropy value 8-bit. Local entropy (which is more accurate than global entropy) can be calculated by evaluating the average value of global entropy for all non-overlapping blocks in the image. Table 7 states the outcomes of local and global entropy values for encrypted images, which values for the cipher images are very close to 8.

5.3.5 Occlusion analysis

In this work, medical images are transferred securely from one user to the other. During the transmission process, encrypted data may be easily infected by losing partial information (occlusion) [61]. The decryption algorithm at the receiver side must be able to recover the original image with minimal loss. To assess the proposed capability of the proposed cryptosystem to resist occlusion, we made a cutting block with various sizes of pixels to the ciphered Enc-Img03 and then decrypt it. Figure 24 presents the outcomes of the data loss attacks, in which the decrypted images have good visual quality.

5.3.6 Key sensitivity analysis

A good encryption algorithm is required to be highly sensitive to the encryption key [62]. To assess the key sensitivity of the proposed technics, we execute the decryption procedure for the encrypted image Enc-Img02 with several initial values, in which the outcomes are presented in Fig. 25.

5.3.7 Complexity analysis of the proposed technics

Complexity analysis is one of the most important measures to assess the performance of an algorithm. This complexity can be computed in terms of running speed or the Encryption Throughput (ET) and the Number of Cycles (NC) required securing one byte of the plain image. Note that the encryption time is computed using the ‘‘tic-toc’’ functions of MATLAB while ET and NC are computed as:

$$ET = \frac{\text{Size of the image (Byte)}}{\text{Encryption time (s)}} \quad (21)$$

$$NC = \frac{\text{CPU speed (Hz)}}{ET(\text{Byte/s})} \quad (22)$$

A good encryption algorithm is required to take less encryption time, less NC, and high ET to be suitable for real time implementation. Table 8 contains the running speed of the encryption algorithm while using various size of test image Img01 (example $512 \times 512 \times 3$ bytes). On the other hand, Table 8 provides the ET and the NC computed with $512 \times 512 \times 3$ bytes version of Img01. The computational workstation is characterized by a 2.4 GHz processor Intel® core TM i7-3630QM 16 GB RAM and MATLAB R2016b software. The computational time increases with respect to the size of the plain image. Note this computational time also relies on the capacity of the workstation (the processor and the RAM). Tables 8 and 9 show that an acceptable complexity is obtained, and the algorithm is competitive with some fastest chaos-based cryptosystems results of the state of the art.

6 Superiority of the encryption scheme based on the TLN chaotic attractor

In this section, a comparative analysis in terms of encryption/decryption performance and key sequence is achieved. Table 10 compares complexity, NPCR, UACI, χ^2 -values, entropy, and key sequence of the present work with some fastest chaos-based encryption/decryption algorithms from the literature. This comparative analysis shows that the proposed cryptosystem has high-security issues and is competitive with the fastest chaos-based cryptosystems.

7 Conclusion and future works

This paper was focused on the dynamics of a non-autonomous tabu learning two-neuron model with compound activation functions. For some discrete values of two parameters of this compound function, its characteristic curves have been obtained. The graphical representations of the equilibria under the variation in the learning rate of the first and the second as well as their stability have been investigated. It has been found that the adapting synapse-based neuron model displays self-excited dynamics. Using the learning rate of the first neuron as the bifurcation control parameter, some complex behaviors including periodic states, chaotic behavior, and periodic windows

separated by areas of chaotic behaviors are observed using bifurcation diagrams as well as the graph of the Lyapunov spectrum as argument. Besides, phenomena such as crisis, the parallel bifurcations branches, and the uncovered coexistence of up to six disconnected stable states have been also captured during our numerical simulations. The representations of the basins of attraction of some of those coexisting states have shown that they are all self-excited. An analog circuit was developed [7, 9, 11, 48], through its implementation in PSPICE. Some results similar to those from the numerical approach were given, and good accordance was observed between both results [17, 20, 46–48, 66, 67]. Finally, the complex sequence of random numbers obtained from the dynamical analysis of the TLTN model was used to design an encryption/decryption algorithm based on a modified Julia set and confusion-diffusion operations. The security performances of the built cryptosystem were analyzed in terms of Computational time (CT = 1.82), Encryption Throughput (ET = 151.82 MBps), Number of Cycles (NC = 15.80), NPCR = 99.6256, UACI = 33.6512, χ^2 -values = 243.7786, global entropy = 7.9992 and local entropy = 7.9083 and it was found that the algorithm was highly secured compared to some fastest neuron chaos-based cryptosystems thus suitable for IoMT security.

Cluster or clustering analysis is usual for data classification. Unfortunately, this idea is out of the scope of this work and may be taken into consideration for future works. That is, for future work we can consider an unsupervised clustering technique to extract meaningful information from complex data such as medical images and encrypt only these meaningful data to improve the encryption time. Let us mention that the aim of the present work is to derive a robust and simple encryption algorithm based on the chaotic sequences from the TLTN model useful for complex data such as medical image encryption.

Acknowledgements The authors are also grateful to the anonymous reviewers and the Associate Editor for their valuable comments, which helped improve the content of the present paper.

Funding No funding has been used.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Data availability The data used in this research work are available from the authors by reasonably request.

References

- Beyer D, Ogier RG (1991) Tabu learning: a neural network search method for solving nonconvex optimization problems. In: [Proceedings] 1991 IEEE International Joint Conference on Neural Networks, pp 953–961. IEEE
- Glover F (1990) Tabu search—part II. *ORSA J Comput* 2(1):4–32
- Glover F (1989) Tabu search—part I. *ORSA J Comput* 1(3):190–206
- Li C, Chen G, Liao X (2005) Yu, J, Chaos: Hopf bifurcation and chaos in tabu learning neuron models. *Int. J. Bifurcation Chaos* 15(08):2633–2642
- Zhou X, Wu Y, Li Y, Ye Y (2006) Hopf bifurcation analysis on a tabu learning single neuron model in the frequency domain. In: 2006 International Conference on Communications, Circuits and Systems 2006, pp 2042–2045. IEEE
- Xiao M, Cao J (2008) Bifurcation analysis on a discrete-time tabu learning model. *J Comput Appl Math* 220(1–2):725–738
- Jun C, Chun-Guang L (2011) Circuit design of tabu learning neuron models and their dynamic behavior. *Acta Phys Sin* 60(2):020502
- Li Y (2011) Hopf bifurcation analysis in a tabu learning neuron model with two delays. *ISRN Appl Math*. <https://doi.org/10.5402/2011/636732>
- Bao B, Hou L, Zhu Y, Wu H, Chen M (2020) Bifurcation analysis and circuit implementation for a tabu learning neuron model. *AEU Int J Electron Commun* 121:153235
- Zhu D, Hou L, Chen M, Bao B (2020) FPGA-based experiments for demonstrating bi-stability in tabu learning neuron model. *Circuit World*. <https://doi.org/10.1108/CW-12-2019-0189>
- Bao B, Chen C, Bao H, Zhang X, Xu Q, Chen M (2019) Dynamical effects of neuron activation gradient on Hopfield neural network: numerical analyses and hardware experiments. *Int J Bifur Chaos* 29(04):1930010
- Bao B, Qian H, Wang J, Xu Q, Chen M, Wu H, Yu Y (2017) Numerical analyses and experimental validations of coexisting multiple attractors in Hopfield neural network. *Nonlinear Dyn* 90(4):2359–2369
- Bao B, Qian H, Xu Q, Chen M, Wang J, Yu Y (2017) Coexisting behaviors of asymmetric attractors in hyperbolic-type memristor based Hopfield neural network. *Front Comput Neurosci* 11:81
- Chen C, Bao H, Chen M, Xu Q, Bao B (2019) Non-ideal memristor synapse-coupled bi-neuron Hopfield neural network: Numerical simulations and breadboard experiments. *AEU-Int J Electron Commun* 111:152894
- Chen C, Chen J, Bao H, Chen M, Bao B (2019) Coexisting multi-stable patterns in memristor synapse-coupled Hopfield neural network with two neurons. *Nonlinear Dyn* 95(4):3385–3399
- Dan Z, Huang W, Huang YJ (2010) Chaos and rigorous verification of horseshoes in a class of Hopfield neural networks. *Neural Comput Appl* 19(1):159–166
- Duan S, Dong Z, Hu X, Wang L, Li H (2016) Applications: Small-world Hopfield neural networks with weight salience priority and memristor synapses for digit recognition. *Neural Comput Appl* 27(4):837–844
- Hopfield JJ (1995) Pattern recognition computation using action potential timing for stimulus representation. *Nature* 376(6535):33–36
- Hopfield JJ (1984) Neurons with graded response have collective computational properties like those of two-state neurons. *Proc Natl Acad Sci* 81(10):3088–3092
- Lakshmi C, Thenmozhi K, Rayappan JBB, Amirtharajan R (2019) Applications: Hopfield attractor-trusted neural network:

- an attack-resistant image encryption. *Neural Comput Appl* 13:1–13
21. Mathias AC, Rech PC (2012) Hopfield neural network: the hyperbolic tangent and the piecewise-linear activation functions. *Neural Netw* 34:42–45
 22. Njitacke Z, Kengne J, Fostin H (2019) A plethora of behaviors in a memristor based Hopfield neural networks (HNNs). *Int J Dyn Control* 7(1):36–52
 23. Njitacke Z, Kengne J, Fozin TF, Leutcha B, Fotsin HB (2019) Dynamical analysis of a novel 4-neurons based Hopfield neural network: emergences of antimonotonicity and coexistence of multiple stable states. *Int J Dyn Control* 7(3):823–841
 24. Njitacke ZT, Kengne J (2018) Complex dynamics of a 4D Hopfield neural networks (HNNs) with a nonlinear synaptic weight: Coexistence of multiple attractors and remerging Feigenbaum trees. *AEU-Int J Electron Commun* 93:242–52
 25. Njitacke ZT, Kengne J (2019) Nonlinear dynamics of three-neurons-based Hopfield neural networks (HNNs): Remerging Feigenbaum trees, coexisting bifurcations and multiple attractors. *J Circuits Syst Comput* 28(07):1950121
 26. Njitacke ZT, Isaac SD, Kengne J, Negou AN, Leutcho GD (2020) Extremely rich dynamics from hyperchaotic Hopfield neural network: hysteretic dynamics, parallel bifurcation branches, coexistence of multiple stable states and its analog circuit implementation. *Eur Phys J Special Topics* 229:1133–1154
 27. Njitacke ZT, Kengne J, Fotsin HB (2020) Coexistence of multiple stable states and bursting oscillations in a 4d Hopfield neural network. *Circuits Syst Signal Process* 39(7):3424–3444
 28. Xu Q, Song Z, Bao H, Chen M, Bao B (2018) Two-neuron-based non-autonomous memristive Hopfield neural network: numerical analyses and hardware experiments. *AEU-Int J Electron Commun* 96:66–74
 29. Xu Q, Song Z, Qian H, Chen M, Wu P, Bao B (2018) Numerical analyses and breadboard experiments of twin attractors in two-neuron-based non-autonomous Hopfield neural network. *Eur Phys J Spec Top* 227(7–9):777–786
 30. Yang J, Wang L, Wang Y, Guo T (2017) A novel memristive Hopfield neural network with application in associative memory. *Neurocomputing* 227:142–14
 31. Zheng YG, Bao LJ (2014) Slow–fast dynamics of tri-neuron Hopfield neural network with two timescales. *Commun Nonlinear Sci Numer Simul* 19(5):1591–1599
 32. Bao B, Zhu Y, Li C, Bao H, Xu Q (2020) Global multistability and analog circuit implementation of an adapting synapse-based neuron model. *Nonlinear Dyn* 101:1–14
 33. Njitacke ZT, Isaac SD, Nestor T, Kengne J (2020) Window of multistability and its control in a simple 3D Hopfield neural network: application to biomedical image encryption. *Neural Comput Appl*. <https://doi.org/10.1007/s00521-020-05451-z>
 34. Gao T, Chen Z (2008) A new image encryption algorithm based on hyper-chaos. *Phys Lett A* 372(4):394–400
 35. Lakshmi C, Thenmozhi K, Rayappan JBB, Amirtharajan R (2020) Hopfield attractor-trusted neural network: an attack-resistant image encryption. *Neural Comput Appl* 32(15):11477–11489
 36. Wang X-Y, Li Z-M (2019) A color image encryption algorithm based on Hopfield chaotic neural network. *Opt Lasers Eng* 115:107–118
 37. Zhou Y, Cao W, Chen CP (2014) Image encryption using binary bitplane. *Signal Process* 100:197–207
 38. Doubla Isaac S, Njitacke ZT, Kengne J (2020) Effects of low and high neuron activation gradients on the dynamics of a simple 3D hopfield neural network. *Int J Bifur Chaos* 30(11):2050159
 39. Kamdjeu Kengne L, Njitacke ZT, Pone JM, Tagne HK (2020) The effects of a constant excitation force on the dynamics of an infinite-equilibrium chaotic system without linear terms: analysis, control and circuit simulation. *Int J Bifur Chaos* 30(15):2050234
 40. Bao B, Hou L, Zhu Y, Wu H, Chen M (2020) Bifurcation analysis and circuit implementation for a tabu learning neuron model. *AEU-Int J Electron Commun* 121:153235
 41. Wolf A, Swift JB, Swinney HL, Vastano JA (1985) Determining Lyapunov exponents from a time series. *Phys D: Nonlinear Phenom* 16(3):285–317
 42. Strogatz SH (1994) *Nonlinear dynamics and chaos*. Addison-Wesley, New York
 43. Alligood KT, Sauer TD, Yorke JA (1996) *Chaos*. Springer, New York
 44. Kengne J, Njitacke Z, Fotsin H (2016) Dynamical analysis of a simple autonomous jerk system with multiple attractors. *Nonlinear Dyn* 83(1):751–765
 45. Babloyantz A, Lourenço C (1996) Brain chaos and computation. *Int J Neural Syst* 7(04):461–471
 46. Fortuna L, Frasca M, Rizzo A (2003) Measurement: Chaotic pulse position modulation to improve the efficiency of sonar sensors. *IEEE Trans Instrum Meas* 52(6):1809–1814
 47. Filali RL, Benrejeb M, Borne P, Simulation N (2014) On observer-based secure communication design using discrete-time hyperchaotic systems. *Commun Nonlinear Sci Numer Simul* 19(5):1424–1432
 48. Duan S, Liao X (2007) An electronic implementation for Liao’s chaotic delayed neuron model with non-monotonous activation function. *Phys Lett A* 369(1–2):37–43
 49. Rani M, Kumar V (2004) Superior mandelbrot set. *J Korea Soc Math Educ Ser D Res Math Educ* 8(4):279–291
 50. Ashton K (2009) That ‘internet of things’ thing. *RFID J* 22(7):97–114
 51. Islam SR, Kwak D, Kabir MH, Hossain M, Kwak K-S (2015) The internet of things for health care: a comprehensive survey. *IEEE Access* 3:678–708
 52. Tsafack N, Kengne J, Abd-El-Atty B, Iliyasu AM, Hirota K, Abd EL-Latif AA (2020) Design and implementation of a simple dynamical 4-D chaotic circuit with applications in image encryption. *Inf Sci* 515:191–217
 53. Yang X, He X, Zhao J, Zhang Y, Zhang S, Xie P (2020) COVID-CT-dataset: a CT scan dataset about COVID-19. *arXiv preprints, arXiv: 2003.13865*
 54. Hua Z, Zhou Y, Huang H (2019) Cosine-transform-based chaotic system for image encryption. *Inf Sci* 480:403–419
 55. Kansa A, Ghebleh M (2017) An algorithm for encryption of secret images into meaningful images. *Opt Lasers Eng* 90:196–208
 56. Nestor T, De Dieu NJ, Jacques K, Yves EJ, Iliyasu AM, El-Latif A, Ahmed A (2020) A multidimensional hyperjerk oscillator: dynamics analysis, analogue and embedded systems implementation, and its application as a cryptosystem. *Sensors* 20(1):83
 57. Sneha P, Sankar S, Kumar AS (2020) A chaotic colour image encryption scheme combining Walsh-Hadamard transform and Arnold-Tent maps. *J Ambient Intell Humaniz Comput* 11(3):1289–1308
 58. Tsafack N, Sankar S, Abd-El-Atty B, Kengne J, Jithin K, Belazi A, Mehmood I, Bashir AK, Song O-Y, Abd El-Latif AA (2020) A new chaotic map with dynamic analysis and encryption application in Internet of Health Things. *IEEE Access* 8:137731–137744
 59. Wang X, Feng L, Zhao H (2019) Fast image encryption algorithm based on parallel computing system. *Inf Sci* 486:340–358
 60. Alawida M, Samsudin A, Teh JS, Alkhalaf RS (2019) A new hybrid digital chaotic system with applications in image encryption. *Signal Process* 160:45–58

61. Gong L, Qiu K, Deng C, Zhou N (2019) An image compression and encryption algorithm based on chaotic system and compressive sensing. *Opt Laser Technol* 115:257–267
62. Luo Y, Yu J, Lai W, Liu L (2019) A novel chaotic image encryption algorithm based on improved baker map and logistic map. *Multimedia Tools Appl* 78(15):22023–22043
63. Diaconu A-V (2016) Circular inter–intra pixels bit-level permutation and chaos-based image encryption. *Inf Sci* 355:314–327
64. Jithin K, Sankar S (2020) Colour image encryption algorithm combining, Arnold map, DNA sequence operation, and a Mandelbrot set. *J Inf Security Appl* 50:102428
65. Liu L, Zhang Q, Wei X (2012) A RGB image encryption algorithm based on DNA encoding and chaos map. *Comput Electr Eng* 38(5):1240–1248
66. Volos CK, Kyprianidis IM, Stouboulos IN (2013) Image encryption process based on chaotic synchronization phenomena. *Signal Processing* 93(5):1328–1340
67. Modeste Nguimdo R, Tchitnga R, Wofo P (2013) Dynamics of coupled simplest chaotic two-component electronic circuits and its potential application to random bit generation. *Chaos* 23(4):043122

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.