Innovative microcontrollers for hypersystem implementation

Branciforte Marco
XXXV cycle
a.y. 2021-2022
Tutor: Prof. Luigi Fortuna

Index

1. Introduction

In this thesis, concepts of hypersystems and hyperneuron and the particular properties related to this class of systems are introduced. In particular focusing on the dynamics of hyper Chua's circuit, showing that higher-dimensional nonlinear systems can be generated starting from the classic Chua's circuit as a gene, whose peculiarities being inherited, including complex dynamics and synchronization properties. An hardware implementation, based on high-performance low-cost microcontrollers, is also outlined showing that the hyper Chua circuit, and in general, hypersystems, are appropriate for practical applications.

Also, the concept of hyperneuron is introduced. The main properties and dynamical characteristics of this class of hypersystems will be discussed by means of several numerical results. The intrinsic robustness of the proposed model is further investigated by proposing

a silicon implementation of the hyperneuron and reporting the corresponding experimental results.

The study is aimed at presenting a novel approach to in silico implementations of large-scale networks of neurons with increased robustness and fault tolerance.

The concept of multidimensional discrete-time maps is emerging in recent literature [Navickas et al., 2011, 2012; Bucolo et al., 2022]. This class of systems is defined starting from a scalar discrete time map by substituting the scalar variable with a square matrix of order n. This leads to higher dimensional maps, whose complex behavior hides intrinsic regularities, linked to the properties of the original scalar map. An important issue related to this class of system is assessing the conditions on initial states for the convergence of the multidimensional map to the behavior displayed by the corresponding scalar map. The case of the multidimensional logistic map with n = 2 has been studied in [Navickas et al., 2012], while in [Bucolo et al., 2022] the general behavior of multidimensional discrete maps with $n \geq 2$ has been introduced, unveiling

the mechanisms leading to the invariance of the bifurcation route to chaos with respect to the respective scalar cases.

The study is centered to the case of the multidimensional Chua's circuit, where we will refer to the term hyper Chua circuit. It should be clearly highlighted that the aim is not to search for hyperchaotic counterparts of the classic Chua's circuit, but rather to introduce the concept of multidimensional systems in the continuous-time case. Therefore, instead of dealing with nonlinear maps, the nonlinear continuous-time dynamics of Chua's circuit is used to generate multidimensional systems. This practically means to have a considerably high number of chaotic signals which are contemporaneously generated by a dynamical system.

The study presented here is referred to as Chua's circuit dynamics [Chua et al., 1986] in which the characteristic of Chua's diode is a cubic polynomial nonlinearity. This choice is due to the fact that, despite its formal simplicity, the rich dynamical behavior generated by this system outlines a complete paradigm

of complexity. Moreover, the global differentiability of the cubic nonlinearity allows to adopt appropriate mathematical tools, such as series expansion, to easily derive fundamental properties of the multidimensional dynamics. The main result that will be emphasized is the so-called invariance of the bifurcation maps, i.e. the property for which the dynamics of the classic Chua's circuit and that of the hyper Chua circuit bifurcate at the same values of the bifurcation parameter enroute to the onset of chaos, yet the hyper Chua circuit is characterized by 3×n2 state variables.

Some recent studies introduced the concept of hypersystems. The idea roots on the concept of frequency transformation [Mitra & Kuo, 2006], an effective approach to design high order selective filters starting from low-pass reference filters. While a frequency transformation is suited for linear systems, for the nonlinear case transformations in the time domain lead to the definition of hypersystems.

This class of systems is defined generalizing the variables of a scalar discrete-time maps by substituting them with square matrices of order N. This generates

higher-order maps, whose complex behavior can be described by referring to the properties of the generating scalar map.

Due to the great interest today in the area of computational neuroscience, the hypersystems theory referred to neuron models is the subject of this study. The main features of a new class of neuron models, the so-called hyperneurons, will be therefore discussed. The concept of hyperneurons, in fact, is defined by starting from a generating neuron mathematical model. The study is motivated by the emerging robust structure and fault tolerance of hyperneurons. A further important motivation is the straightforward possibility of realizing this class of systems by using low-cost silicon devices based on advanced microcontrollers.

This aspect makes the topic of particular relevance, since the use of current high performance microprocessor technology allows for the realization of clusters of boards, thus implementing physically neurodynamics emulators based on hypersystems.

The thesis is organized as follows. In Sect. 2 the concept of hypersystems and the fundamental

theoretical results are summarized, in Sect. 3 the case of continuous-time of the hyper Chua's circuit is discussed, in Sect. 4 the hyperneurons will be approached by reporting the mathematical modes adopted, numerical simulations aimed at showing the peculiar properties also in terms of robustness are discussed. The hardware implementation and the experimental results are reported in Sect. 5. The main features of the proposed hyperneurons are focused in the Conclusions drawn in Sect. 6.

2. The concepts of Hypersystems

Let us consider the dynamics of a generic nonlinear system:

$$\dot{x} = f(x,t) \qquad (1)$$

where $x \in R^{n\times 1}$, $f : R^{n\times 1} \rightarrow R^{n\times 1}$ and $t \in R^{+}$.

Let us now consider the dynamics of an hypersystem defined as:

$$\dot{X} = F(X,t) \qquad (2)$$

where $X \in R^{n\times N\times N}$, $F: R^{n\times N\times N} \rightarrow R^{n\times 1\, N\times N}$ and $t \in R^{+}$.

Therefore, an hypersystem is generated starting from a gene dynamics (1) by considering that the state vector in (1) is replaced by a vector of n square matrices each of dimension N and the vector field f is replaced by the matrix field F. This means that instead of n scalar state variables $x_i \in R$, the hypersystems dynamics (2) is described by n square matrices $X_i \in R^{N\times N}$.

In order to clarify the genesis of hypersystems, let us consider the Lorenz system that is a paradigmatic example of nonlinear dynamics, described by the following differential equations:

$$\begin{aligned} \dot{x} &= \sigma(y - x) = f_1(x, y, z) \\ \dot{y} &= \rho x - xz - y = f_2(x, y, z) \\ \dot{z} &= xy - \beta z = f_3(x, y, z) \end{aligned} \qquad (3)$$

Fixing opportunely the parameters, β, ρ and σ, the behavior of (3) depends on the initial conditions x(0), y(0), and z(0).

Let us consider now the Lorenz hypersystem with N = 2. It is described by:

$$\begin{aligned} \dot{X} &= \sigma(Y - X) = F_1(X, Y, Z) \\ \dot{Y} &= \rho X - XZ - Y = F_2(X, Y, Z) \\ \dot{Z} &= XY - \beta Z = F_3(X, Y, Z) \end{aligned} \qquad (4)$$

where

$$X = \begin{bmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{bmatrix} \; Y = \begin{bmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{bmatrix} \; Z = \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix}$$

(5)

Therefore, the dynamics of the Lorenz hypersystem with N=2, fixing opportunely the parameters, β, ρ and σ, is governed by $3\times N^2=12$ initial conditions.

Now considering $x_i(t)$ for i=1,…,n be the trajectories corresponding to the set of initial conditions $x_i(0)$, chosen inside the basin of attraction of the gene system (1). The trajectories can be expressed by Taylor expansion as:

$$x_i(t) = x_i(0) + \dot{x}_i(t)|_{t=0} t + \ddot{x}_i(t)|_{t=0} \frac{t^2}{2!} + \\ + \dddot{x}_i(t)|_{t=0} \frac{t^3}{3!} + \cdots \qquad (6)$$

By successive differentiation of Eqs. (1), it does result:

$$x_i(t) = x_i(0) + \sum_{k=1}^{\infty} P_{x_i}^{(k)}(x_1, \dots, x_n)\Big|_{(x_1(0),\dots,x_n(0))} \frac{t^k}{k!}$$

(7)

where $P_{x_i}^{(k)}(x_1, \dots, x_n)$, are n-variate polynomials.

The dynamical properties of hypersystems have been shown to be highly correlated to the dynamical properties of the corresponding gene system [Bucolo et al., 2022b]. We recall here and generalize the fundamental theorem of hypersystems.

Theorem 1. *Let us consider the dynamics of an hypersystem. The route to chaos with respect to its parameters displays the same bifurcation points of the scalar gene system.*

Proof. Let us assume $X_i(0) = T\lambda_{Xi}T^{-1}$, for i=1,..,n the initial condition of the matrix differential equations (2) where

$$\lambda_{X_i} = \begin{bmatrix} \lambda_{x_i}^{(1)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_{x_i}^{(N)} \end{bmatrix} \qquad (8)$$

being $\lambda_{x_i^{(k)}}$ with k = 1,…,N real quantities belonging to the basin of attraction S of the dynamics of the scalar gene system and $T \in R^{N\times N}$.

The Taylor expansions of Eqs. (2) leads to:

$$X_i(t) = T\lambda_{X_i}T^{-1} + + \sum_{i=1}^{\infty} T\Pi_{X_i}^{(k)}(\lambda_{X_1}, \dots, \lambda_{X_n})T^{-1}\frac{t^i}{i!} \quad (9)$$

being $\Pi_{X_i}^{(k)}$ diagonal matrices as

$$\Pi_{X_i}^{(k)} = \begin{bmatrix} \pi_{X_{i_{1,1}}}^{(k)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \pi_{X_{i_{N,N}}}^{(k)} \end{bmatrix} \quad (10)$$

where the n−tuple $(\pi_{X_{1_{j,j}}}^{(k)}, \pi_{X_{2_{j,j}}}^{(k)}, \dots, \pi_{X_{n_{j,j}}}^{(k)})$ is defined according to the polynomials $P_{x_i}^{(k)}$ in Eqs. (7) calculated in $(\lambda_{x_1}^{(k)}, \dots, \lambda_{x_n}^{(k)})$. Therefore, the N^2 trajectories of the hypersystem are the linear combinations dictated by the matrix T of n trajectories obtained from the scalar gene system.

Remark 2.1. The fact that the dynamics of the hypersystems is a linear combination of the dynamics of n scalar gene system, ensures that when the scalar system is in periodic conditions, a limit cycle with the same periodicity is obtained for the hypersystem. Similarly, when the scalar gene system shows a chaotic behavior, the dynamics of the hypersystem is chaotic [Baptista, 2021].

Remark 2.2. The hypothesis of starting from initial conditions described by matrices with the same set of eigenvectors is fundamental to ensure the fact that the trajectories of the hypersystems do not diverge.

3. The Hypersystem of Chua circuit

Let us consider the dynamics of the Chua circuit:

$$\begin{aligned} \dot{x} &= \alpha(y - c_0 x^3 - c_1 x) \\ \dot{y} &= x - y + z \\ \dot{z} &= -\beta y \end{aligned} \qquad (11)$$

with x, y, z ∈ R, and α, β, c_0, c_1 ∈ R being its parameters [Buscarino et al., 2014].

Let x(t), y(t), and z(t) be the trajectories corresponding to the set of initial conditions x(0), y(0), and z(0), chosen inside the basin of attraction of the Chua circuit. The trajectories can be expressed by Taylor expansion as:

$$x(t) = x(0) + \dot{x}(t)|_{t=0} t + \ddot{x}(t)|_{t=0} \frac{t^2}{2!} + \dddot{x}(t)|_{t=0} \frac{t^3}{3!} + \cdots$$

$$y(t) = y(0) + \dot{y}(t)|_{t=0} t + \ddot{y}(t)|_{t=0} \frac{t^2}{2!} + \dddot{y}(t)|_{t=0} \frac{t^3}{3!} + \cdots$$

$$z(t) = z(0) + \dot{z}(t)|_{t=0} t + \ddot{z}(t)|_{t=0} \frac{t^2}{2!} + \dddot{z}(t)|_{t=0} \frac{t^3}{3!} + \cdots$$

(12)

By successive differentiation of Eqs. (1), we get

$$x(t) = x(0) + \sum_{i=1}^{\infty} P_x^{(i)}(x, y, z)\Big|_{(x(0),y(0),z(0))} \frac{t^i}{i!}$$

$$y(t) = y(0) + \sum_{i=1}^{\infty} P_y^{(i)}(x, y, z)\Big|_{(x(0),y(0),z(0))} \frac{t^i}{i!}$$

$$z(t) = z(0) + \sum_{i=1}^{\infty} P_z^{(i)}(x, y, z)\Big|_{(x(0),y(0),z(0))} \frac{t^i}{i!}$$

(13)

where $P_x^{(i)}(x,y,z), P_y^{(i)}(x,y,z), P_z^{(i)}(x,y,z)$ are trivariate polynomials. The set of polynomials up to i=3 is explicitly reported.

$$P_x^{(1)}(x,y,z) = \alpha(y - c_0x^3 - c_1x)$$
$$P_y^{(1)}(x,y,z) = x - y + z$$
$$P_z^{(1)}(x,y,z) = -\beta y$$

(14)

$$P_x^{(2)}(x,y,z) = 3\alpha^2c_0^2x^5 + 4\alpha^2c_0c_1x^3 + (-3\alpha^2c_0y)x^2 + \alpha(\alpha c_1^2 + 1)x - \alpha(y - z + \alpha c_1 y)$$
$$P_y^{(2)}(x,y,z) = -\alpha c_0x^3 + (-\alpha c_1 - 1)x - z + y(\alpha - \beta + 1)$$
$$P_z^{(2)}(x,y,z) = -\beta(x - y + z)$$

(15)

$$P_x^{(3)}(x,y,z) = (-15\alpha^3 c_0^3)x^7 - 27\alpha^3 c_0^2 c_1 x^5 + 21\alpha^3 c_0^2 y x^4 - \alpha^2 c_0(1 + 3(\alpha c_1^2 + 1) + 10\alpha c_1^2)x^3 + \alpha^2 c_0(3(y - z + \alpha c_1 y) + 15\alpha c_1 y)x^2 + (-\alpha(\alpha c_1 + \alpha c_1(\alpha c_1^2 + 1) + 6\alpha^2 c_0 y^2 + 1))x + \alpha(y(\alpha - \beta + 1) - z + \alpha c_1(y - z + \alpha c_1 y))$$

$$P_y^{(3)}(x,y,z) = 3\alpha^2 c_0^2 x^5 + (4c_0 c_1 \alpha^2 + c_0\alpha)x^3 + (-3\alpha^2 c_0 y)x^2 + (\alpha c_1 - \beta + \alpha(\alpha c_1^2 + 1) + 1)x + z - \alpha(y - z + \alpha c_1 y) + \beta y - \beta z - y(\alpha - \beta + 1)$$

$$P_z^{(3)}(x,y,z) = \alpha\beta c_0 x^3 + \beta(\alpha c_1 + 1)x + \beta(z - y(\alpha - \beta + 1))$$

(16)

Let us consider the dynamics of the Chua circuit in Eqs. (11). The hyper Chua circuit of order n is given by the following matrix differential equations:

$$\dot{X} = \alpha(Y - c_0 X^3 - c_1 X)$$
$$\dot{Y} = X - Y + Z \qquad (17)$$
$$\dot{Z} = -\beta Y$$

with $X \in R^{n\times n}$, $Y \in R^{n\times n}$, and $Z \in R^{n\times n}$.

Remark 3.1. Even if the number of parameters is the same as the scalar Chua's circuit, the dynamics of the hyper Chua circuit is defined by $3\times n^2$ variables. Therefore, the dynamical behavior of the hyper Chua circuit is strongly dependent on the $3\times n^2$ initial conditions.

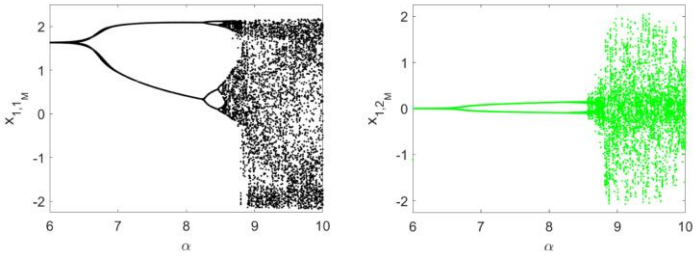
Let us consider the dynamics of the hyper Chua circuit. The bifurcation route to chaos with respect to its parameters has the same bifurcation points of the scalar Chua circuit (Theorem 1).

The fact that the dynamics of the hyper Chua circuit is a linear combination of the dynamics of n scalar Chua's circuit, ensures that when the scalar circuits are in periodic conditions, a suitable limit cycle with the same periodicity is obtained for the hyper Chua circuit (*Remark 2.1)*. Similarly, when the n scalar Chua circuit

shows chaotic behavior, the dynamics of the hyper Chua circuit is chaotic [Baptista, 2021].

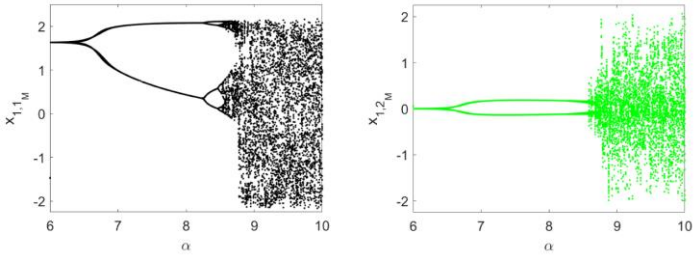
The dynamics of the hyper Chua circuit is now investigated by bifurcation diagrams obtained with respect to parameter α. The bifurcation diagrams reported in the following are realized by plotting the local extremes of the state variables x11 and x12, similar results can be obtained considering different variables. The bifurcation diagrams for the hyper Chua circuits with n = 2, n = 3, and n = 5, fixing β = 14.286, c_0 = 1/16 and c_1 = −1/6 , are reported in Figs. 1–3.

It appears evident that the bifurcation route is the same of the scalar Chua circuit, including a series of period-doubling bifurcations leading to a single-scroll and, finally, to the classic double-scroll chaotic attractor.



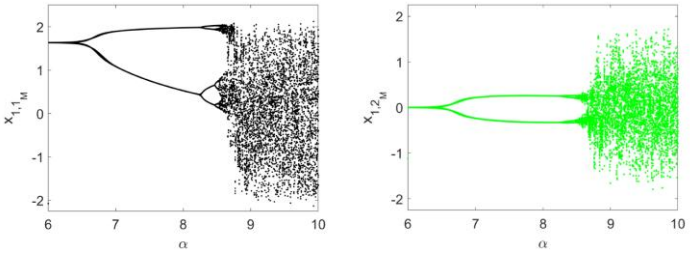
(a) *(b)*

Fig. 1. Bifurcation diagram with respect to parameter α for the hyper Chua circuit with n = 2: local maxima and minima of state variable (a) x_{11} and (b) x_{12}.



(a) *(b)*

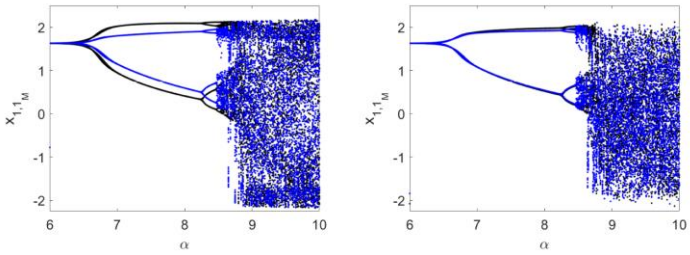
Fig. 2. Bifurcation diagram with respect to parameter α for the hyper Chua circuit with n = 3: local maxima and minima of state variable (a) x_{11} and (b) x_{12}.



(a) *(b)*

Fig. 3. Bifurcation diagram with respect to parameter α for the hyper Chua circuit with n = 5: local maxima and minima of state variable (a) x11 and (b) x12.

The shape of the attractors are, however, strongly dictated by the chosen initial conditions and this effect is more evident for lower values of n. In order to show this effect, in Fig. 4 the bifurcation diagrams obtained for two different initial conditions when n = 2 [Fig. 4(a)] and n = 5 [Fig. 4(b)] are reported.



(a) *(b)*

Fig. 4. Bifurcation diagram with respect to parameter α for the hyper Chua circuit starting from two different sets of initial conditions: (a) n = 2 and (b) n = 5. Local maxima and minima of state variable x_{11} are reported.

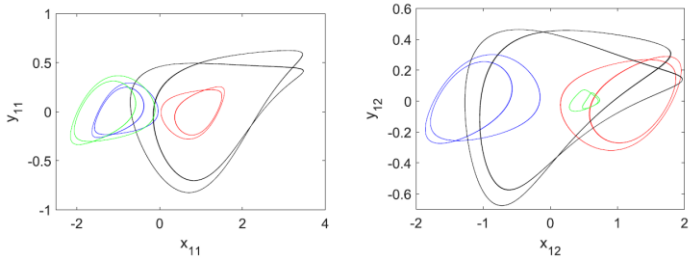
We focus now on the case n = 2 to show the different attractors which can be obtained for different initial conditions along the whole bifurcation route to chaos. Let us fix α = 8.4, thus obtaining in the scalar Chua's circuit a period-2 limit cycle. In Fig. 5, four different period-2 limit cycles are retrieved in the hyper Chua circuit with the same value of the parameter for four different initial conditions.

It can be noticed that the period-2 behavior is present either in diagonal or off-diagonal elements of the hyper Chua circuit. When the scalar gene Chua's

circuit is set for period-4 conditions, i.e. with $\alpha = 8.5$, a similar behavior can be retrieved in the dynamics of the hyper Chua circuit, as shown in Fig. 6.

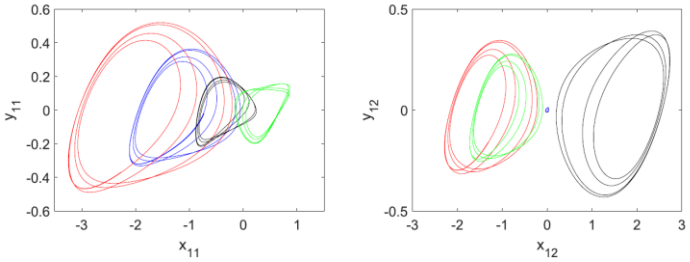
The bifurcation route to chaos leads to a first chaotic window in which the so-called single-scroll attractor appears. In this case, in the scalar Chua circuit for $\alpha = 8.6$, the shape of the single-scroll is strongly affected by the initial conditions, but it tends to maintain its geometric properties, as shown in Fig. 7.

Finally, selecting $\alpha = 9$, the double-scroll attractor appears in the scalar gene Chua's circuit and the hyper Chua circuit produces a doublescroll-like attractor in all the elements of the matrix state variables, as reported in Fig. 8. Also in this case, the complexity of the attractor is dictated by the initial conditions.



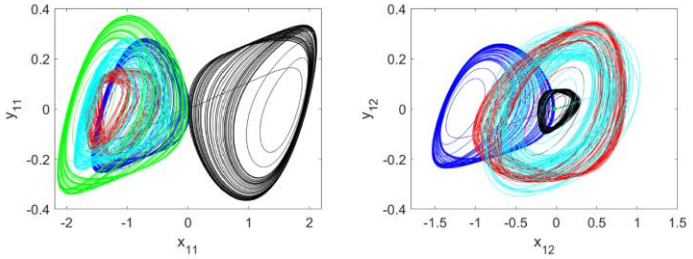
(a) *(b)*

Fig. 5. Period-2 limit cycles obtained for different initial conditions with α = 8.4: (a) x_{11}–y_{11} and (b) x_{12}–y_{12}.



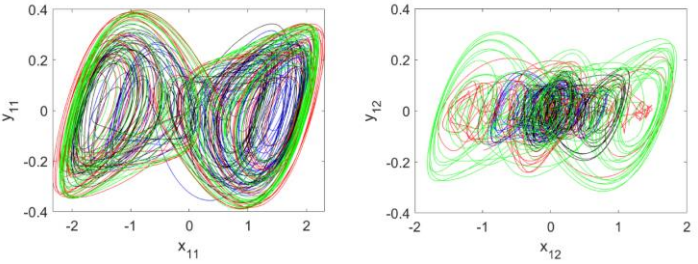
(a) *(b)*

Fig. 6. Period-4 limit cycles obtained for different initial conditions with α = 8.5: (a) x_{11}–y_{11} and (b) x_{12}–y_{12}.



(a) *(b)*

Fig. 7. Single-scroll attractors obtained for different initial conditions with α = 8.6: (a) x_{11}–y_{11} and (b) x_{12}–y_{12}.



(a) *(b)*

Fig. 8. Double-scroll attractors obtained for different initial conditions with α = 9: (a) x_{11}–y_{11} and (b) x_{12}–y_{12}.

The effect of initial matrices is further confirmed by the largest Lyapunov exponent Λmax,

reported in Fig. 9 as a function of the parameter α. Interestingly, the value of Λmax ≈ 0.3 for α = 9 is consistent with the largest Lyapunov exponent calculated for the scalar Chua's circuit for the same set of parameters [Buscarino et al., 2017].

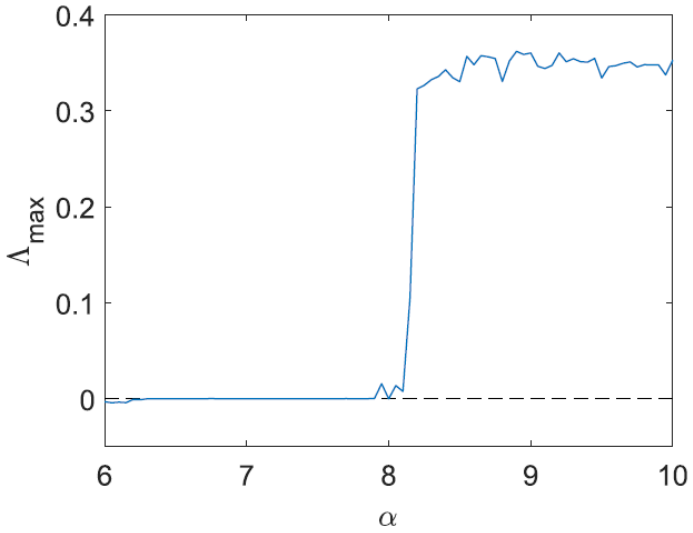


Fig. 9. Largest Lyapunov exponent Λmax evaluated for different values of α for the hyper Chua circuit with n=2. Other parameters as reported in the main text.

Despite the strong effect of different initial conditions, the synchronization of a pair of hyper Chua circuits follows the same path of the scalar Chua's circuit. Let us consider two hyper Chua circuits coupled through a diffusive coupling acting on the first matrix differential equation as:

$$\dot{X}_1 = \alpha(Y_1 - c_0 X_1^3 - c_1 X_1) + K(X_2 - X_1)$$
$$\dot{Y}_1 = X_1 - Y_1 + Z_1$$
$$\dot{Z}_1 = -\beta Y_1$$

$$\dot{X}_2 = \alpha(Y_2 - c_0 X_2^3 - c_1 X_2) + K(X_1 - X_2)$$
$$\dot{Y}_2 = X_2 - Y_2 + Z_2$$
$$\dot{Z}_2 = -\beta Y_2$$
(18)

with $X_1 \in R^{n\times n}$, $Y_1 \in R^{n\times n}$, and $Z_1 \in R^{n\times n}$ accounting for the first hyper Chua circuit and $X_2 \in R^{n\times n}$, Y2 $\in R^{n\times n}$, and $Z_2 \in R^{n\times n}$ accounting for the second hyper Chua circuit. The scalar coefficient K modulates the coupling strength. Let us set $\alpha = 9$, $\beta = 14.286$, c0 = 1/16 and c1

$= -1/6$, so that a chaotic double-scroll attractor can be observed over each element of the two hyper Chua circuits. In Fig. 10, the synchronization errors

$$E_{x11} = \left| x_{11}^{(1)} - x_{11}^{(2)} \right|$$

$$E_{x12} = \left| x_{12}^{(1)} - x_{12}^{(2)} \right|$$

$$E_{x22} = \left| x_{22}^{(1)} - x_{2}^{(2)} \right|$$

are reported as a function of the coupling strength K. It can be observed that complete synchronization occurs for K >3.5, as for the scalar Chua's circuit [Buscarino et al., 2009].

It is interesting to observe, however, that the offdiagonal elements are synchronized even below this threshold.

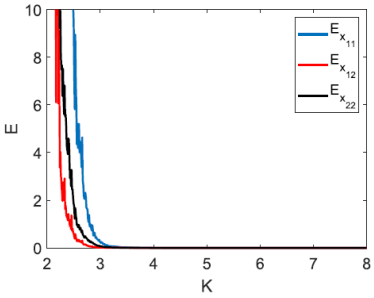


Fig. 10. Synchronization errors

4. The Hypersystem of the H-R Neuron

The task of computational neuroscience is to get focus on biologically plausible models of neurons. Computational neuroscience starts from mathematical models of neurons to understand, by simulations or experiments, the capabilities of nervous systems. Neuron models in theoretical neuroscience are finalized at deriving the essential items of the biological neurons, especially as regards their spiking behavior. Even if there is a plenty of software packages devoted to neuron simulations, in silico modeling of realistic neurons is a less explored approached. The Blue Brain project [Makram, 2006] is a fundamental example as it aims at constructing a biophysically detailed simulation of the neocortical column using a Blue Gene supercomputer.

In silico neuron implementations can be used for educational purposes [Baden et al., 2018], or as modelling tools [Litt, 2015], since they can run real-time simulation of large networks of interconnected

cells, easily interfaced with external stimuli and even with biological tissues [Bozhko et al., 2021]. Moreover, in silico neurons allow to implement neuron-based systems capable to control complex systems, such as robots.

Even if silicon neurons are intended in the sense of a very-large-scale-integration circuit emulating the electrophysiological behavior of real neurons, in this work, exploiting the concept of hypersystems, we propose an high order neuron model which can be implemented in new generation low-cost, fast microcontrollers.

This approach, in fact, implementing the neuron dynamics as electrical low voltage signals, guarantees the possibility of using it to control engineering systems, including automation devices and robots.

The signal of a single neuron has a pulse like structure. It consists of a sequence of action potentials, short spikes of membrane voltage. Neuronal spiking behavior assumes a key-role in control and in information encoding [Arena et al., 2009]. It is intrinsically clear that the exact spikes time and the

precise length of interspike intervals can be modulated in order to drive information more efficiently than using time averaged firing rates. In particular, in all the situations when the fast reaction of the system is required there is no time for temporal averaging and the single spike can be exploited. For this reason, the neuron models dynamics is often characterized in terms of the interspike interval, i.e. the time interleaving between two successive spikes [Hindmarsh & Rose, 1984].

Let us now consider the dynamics of the Hindmarsh-Rose model:

$$\dot{x} = y - ax^3 + bx^2 - z + I$$
$$\dot{y} = c - dx^2 - y$$
$$\dot{z} = r[s(x - x_R) - z]$$
(19)

where a, b, c, d, r, s, x_R, and I are system parameters. The Hindmarsh-Rose model behavior displays several peculiar neuronal activities, empirically observed in real neurons, ranging from periodic to chaotic spiking,

on the basis of the parameter values. The bifurcation diagrams reported in Fig. 11 is realized by plotting the interspike intervals (ISI) calculated from the state variables x (similar results can be obtained considering different variables).

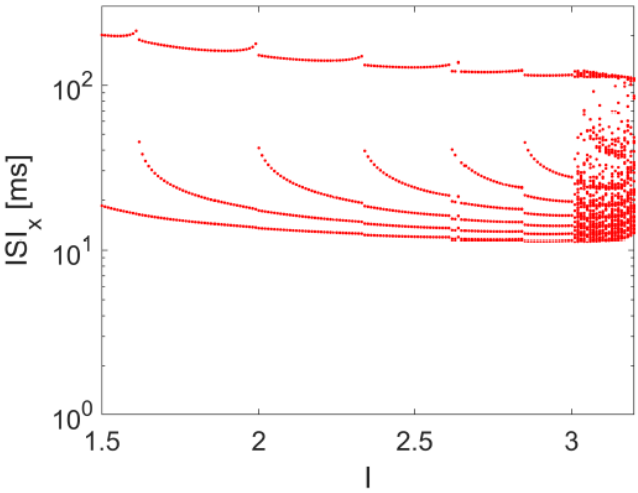


Fig. 11. Bifurcation diagram with respect to parameter I for the scalar Hindmarsh-Rose neuron: interspike interval (ISI) calculated in ms for x.

Let us now introduce the Hindmarsh-Rose hyperneuron model, proceeding as in the general definition (2):

$$\dot{X} = Y - aX^3 + bX^2 - Z + I\Upsilon_N$$
$$\dot{Y} = c\Upsilon_N - dX^2 - Y$$
$$\dot{Z} = r[s(X - x_R\Upsilon_N) - Z]$$
(20)

where $X \in R^{n\times n}$, $Y \in R^{n\times n}$, $Z \in R^{n\times n}$, and Υ_N represents the identity matrix of dimension N.

Remark 4.1. Even if the number of parameters is the same of the scalar Hindmarsh-Rose model, the dynamics of the hyperneuron is defined by 3 × N2 variables. Therefore, the dynamical behavior of the hyperneuron strongly depends on the 3 × N2 initial conditions.

The intrinsic nature of the hyperneuron defined starting from the Hindmarsh-Rose model shows the high number (i.e. N2) of interacting neural units, thus representing a compact way to model large-scale neuron networks. Moreover, the result of Theorem 1 guarantees that the bifurcation routes with respect to each hyperneuron parameter can be inferred from the corresponding route in the scalar Hindmarsh-Rose mode. Moreover, its robustness to noise and faults will

be investigated in the following to reveal non trivial properties, thus making hyperneurons ready for in silico implementations.

The dynamics of the hyperneuron generated by the Hindmarsh-Rose model is now investigated by means of the bifurcation diagrams obtained with respect to parameter I, that represents the external stimulus and drives the spiking behavior in the scalar Hindmarsh-Rose model. It is important to stress here that the dynamics of the Hindmarsh-Rose model is dimensionless, either in amplitudes or in time [Hindmarsh & Rose, 1984]. Therefore, the proper choice of the integration time step allows to fix the proper time-scales of the spiking behavior. ISIs are here reported in ms in order to be comparable to observations on biological neurons [Hindmarsh & Rose, 1984]. The bifurcation diagrams for the hyperneurons with N=2, and N=3, fixing a=1, b=3, c=1, d=5, r=0.003 s=4 and $x_R=-8/5$, are reported in Figs. 12, 13. It appears evident that the bifurcation route is the same of the scalar Hindmarsh-rose neuron model,

reported in Fig. 11, including the cascade to the chaotic attractor.

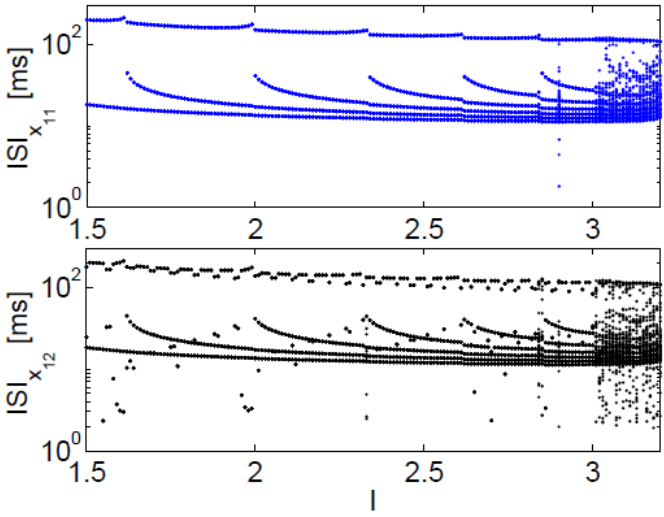


Fig. 12. Bifurcation diagram with respect to parameter I for the Hindmarsh-Rose hyperneuron with N = 2: interspike interval (ISI) calculated in ms for x11 (top) and x12 (bottom). Other parameters: a = 1, b = 3, c = 1, d = 5, r = 0.003 s = 4 and $x_R = -8/5$.

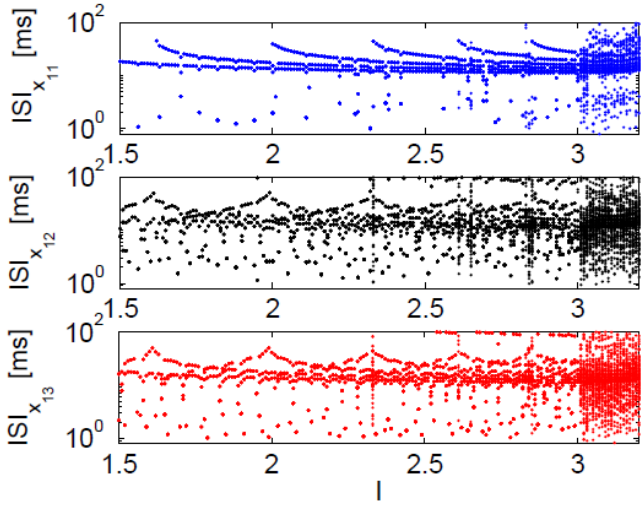


Fig. 13. Bifurcation diagram with respect to parameter I for the Hindmarsh-Rose hyperneuron with N = 3: interspike interval (ISI) calculated in ms for x11 (top), x12 (middle), and x13 (bottom). Other parameters: a = 1, b = 3, c = 1, d = 5, r = 0.003 s = 4 and xR = −8/5.

The shape of the chaotic attractor, moreover, is dictated by the chosen initial conditions. We focus now on the case N=2 to show the different chaotic attractors obtained for three sets of initial conditions when I=3.2, as reported in Fig. 14. It is evident that, while the global

structure of the Hindmarsh-Rose attractor is preserved for all initial conditions in the x_{11}–y_{11}–z_{11} space, the chaotic attractors in the space x_{12}–y_{12}–z_{12} shows significantly different shapes, yet maintaining the spiking behavior.

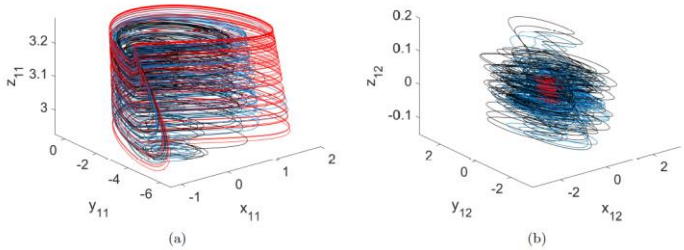


Fig. 14. Chaotic attractors obtained in the hyperneuron with N = 2 for three different initial conditions and for I = 3.2: (a) $x_{11} - y_{11} - z_{11}$ and (b) $x_{12} - y_{12} - z_{12}$. Other parameters: a = 1, b = 3, c = 1, d = 5, r = 0.003 s = 4 and $x_R = -8/5$.

Since the implementation of in silico neurons requires the capability to include in a silicon support more and more neurons and to guarantee a suitable robustness. Even if some fault does occur, thus, the neuron spiking behavior is preserved. Faults,

uncertainty and noise, in fact, are likely occurring in the fabrication process of in silico devices. The features of the hyperneuron generated by the Hindmarsh-Rose model appears interesting also in term of its robustness to noise and faults. We now focus on this peculiar aspect of hyperneurons.

In order to explore the robustness of the hyperneurons, we introduce numerical faults by following two approaches. The first approach is based on including at each step a random perturbation taken from a uniform distribution between −1 and 1 in the numerical integration of the variable y_{ij} for a given (i, j). The second approach is to include the random perturbation with a probability p, thus ideally mimicking the occurrence of a not persistent fault.

Let us focus on the chaotic attractor obtained for I = 3.2. Applying a persistent random perturbation to the integration, as outlined above, the scalar gene Hindmarsh-Rose model is driven to a not-spiking, underthreshold, behavior. On the contrary, the hyperneuron is able to adapt its behavior to the perturbation, maintaining the chaotic spiking trend. The

two cases N = 1 and N = 2 are shown in Figs. 15 and 16. The effect of a not persistent perturbation is explored in Fig. 17, where the interspike interval for I=3.2 is reported when varying the fault probability p. The interspike interval in the case N = 1 vanishes for p>0.1, since the spiking behavior disappears, while for N = 2 the spiking behavior is preserved also for higher values of p, thus confirming the resilience of the hyperneuron model.

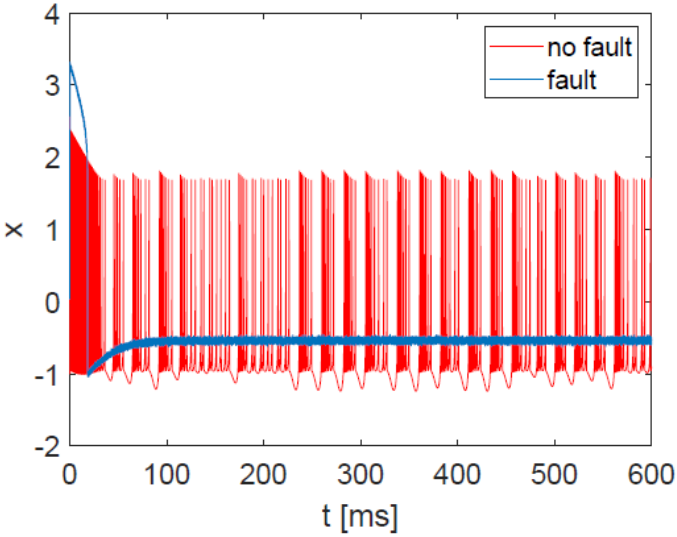


Fig. 15. Temporal behavior of state variable x of the scalar Hindmarsh-Rose model for I = 3.2 in presence of a persistent fault. Fault occurs on the variable y.

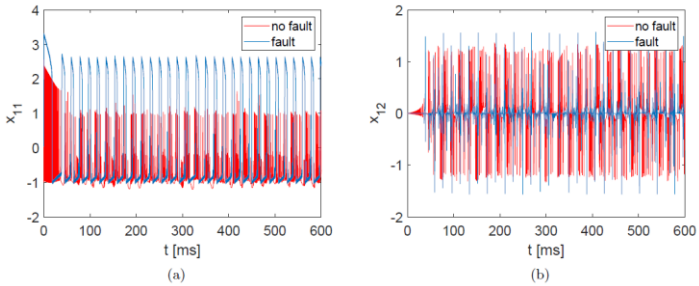


Fig. 16. Temporal behavior of state variables (a) x11 and (b) x12 of the hyperneuron based on the Hindmarsh-Rose model for N = 2 and I = 3.2 in presence of a persistent fault. Fault occurs on the variable y11. Other parameters: a = 1, b = 3, c = 1, d = 5, r = 0.003 s = 4 and xR = −8/5.

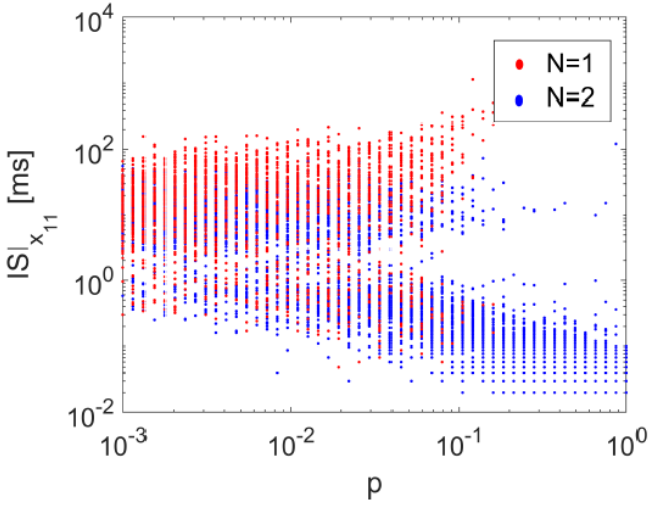


Fig. 17. Interspike interval for the scalar Hindmarsh-Rose model and for the hyperneuron with N = 2 and I = 3.2 with respect to the probability of a fault to occur. The ISI vanishes for p > 0.1 in the scalar model, thus indicating that the spiking behavior is lost.

5. The relevant role of the innovative STM32 family in the hypersystem generation

The analog implementation of a n^{th} order hyper Chua's circuit and hyperneuron based on the Hindmarsh-Rose model would involve the use of many multipliers and operational amplifiers, since the explicit dynamical equations include several variable products. Therefore a digital, low-cost implementation has been preferred.

The hardware implementation of the hyper Chua's circuit for N=2 is based on a STMicroelectronics Nucleo developing board, reported in Fig. 18, equipped with an STM32L476 microcontroller characterized by an 80MHz main clock. Regarding the realization of hyperneurons based on the Hindmarsh-Rose model for N=2 we adopted the same microcontroller (STM32L476), to generalize the implementation to n^{th} order a more powerful microcontroller was utilized, the STM32H723 and his corresponding Nucleo developing board.

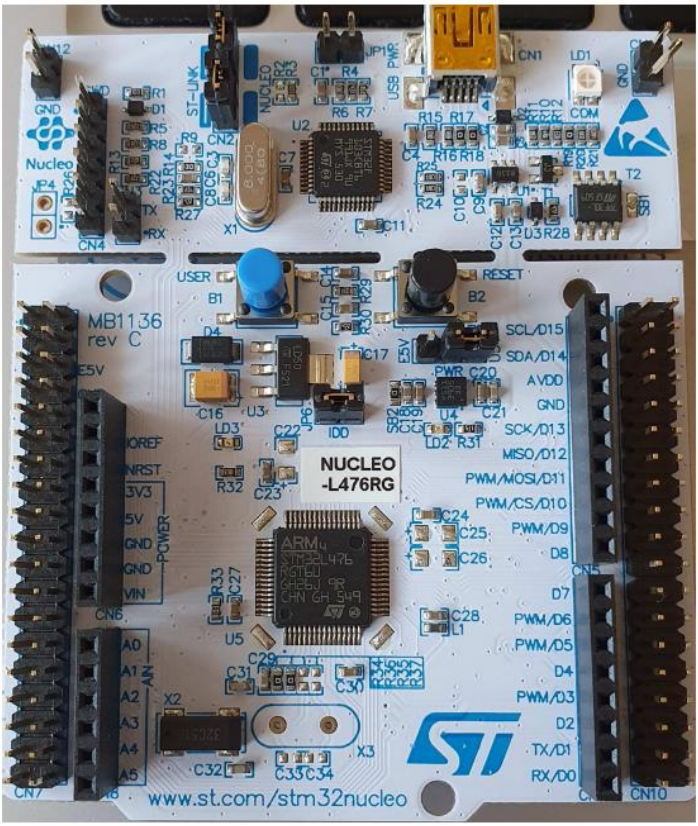


Fig. 18. STMicroelectronics Nucleo board used to implement the hyper Chua circuit.

Let us introduce some information on the microcontrollers used.

The STM32L476 device is an ultra-low-power microcontrollers based on the high-performance Arm® Cortex®-M4 32-bit RISC core operating at a frequency of up to 80 MHz. The Cortex-M4 core features a Floating point unit (FPU) single precision which supports all Arm® single-precision data-processing instructions and data types. It also implements a full set of DSP instructions. The STM32L476xx devices embed high-speed memories (Flash memory up to 1 Mbyte, up to 128 Kbyte of SRAM).

The devices offer up to two DAC channels and many general-purpose 32-bit timer and 16-bit PWM timers.

The STM32H723 device is based on the high-performance Arm® Cortex®-M7 32-bit RISC core operating at up to 550 MHz. The Cortex® -M7 core features a floating point unit (FPU) which supports Arm® double-precision (IEEE 754 compliant) and single-precision data-processing instructions and data types. The Cortex -M7 core includes 32 Kbytes of instruction cache and 32 Kbytes of data cache. STM32H723xE/G devices support a full set of DSP

instructions and a memory protection unit (MPU) to enhance application security. STM32H723xE/G devices incorporate high-speed embedded memories with up to 1 Mbyte of Flash memory, up to 564 Kbytes of RAM

The devices embed peripherals allowing mathematical/arithmetic function acceleration (CORDIC coprocessor for trigonometric functions and FMAC unit for filter functions). The devices offer two DACs, 4 general-purpose 32-bit timers, 12 general-purpose 16-bit timers including two PWM timers for motor control, five low-power timers, a true random number generator (RNG).

Many other features are implemented in these microcontrollers, we reported only the specs useful for our implementation.

Two peripherals of the microcontrollers have been used: a timer, which ensures a constant sampling time, and a Digital to Analog Converter, in order to produce an output voltage signal, allowing to visualize the state variables, with a 12-bit resolution.

The algorithm to implement the hyper Chua circuit is essentially based on the discretization and vectorization of Eqs. (17), thus obtaining a discrete map. It is adopted therefore an Euler integration method, included in the loop interrupts the routine of the timer. Using a double precision, the obtained sampling time is $\tau = 100$ μs. The use of optimized matrix manipulation routine can be adopted in order to maintain the matrix form of the hyper Chua circuit also in the algorithm, but this often leads to higher sampling times, thus reducing the frequency of the obtained circuit.

The experimental attractors produced by the digital implementation for $\alpha = 8.6$ and $\alpha = 9$ are reported in Figs. 19 and 20, thus showing the numerical stability of the dynamics of the hyper Chua circuits and the possibility to adopt a simple but effective implementation for the use of the hyper Chua circuit in engineering applications.

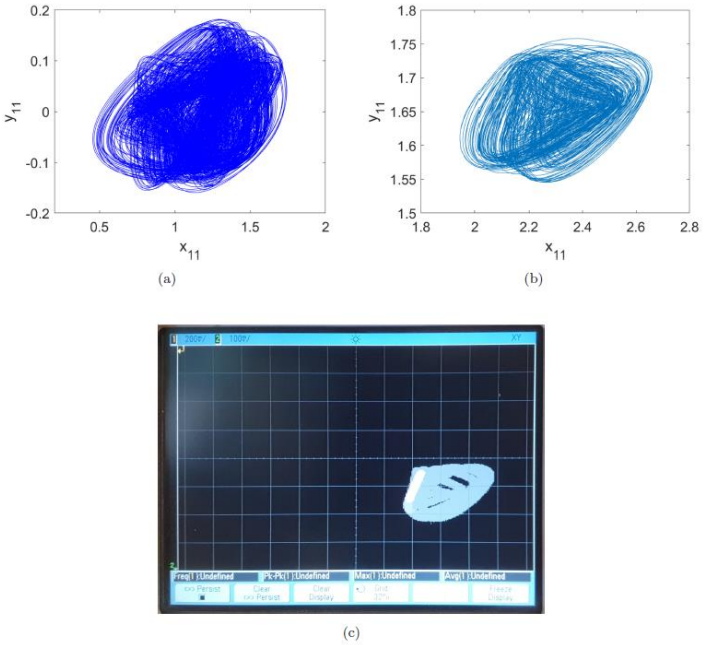


Fig. 19. Experimental results: (a) Numerical and (b) experimental attractors for α = 8.6; (c) oscilloscope trace (x-axis: 200mV/div; y-axis: 100mV/div).

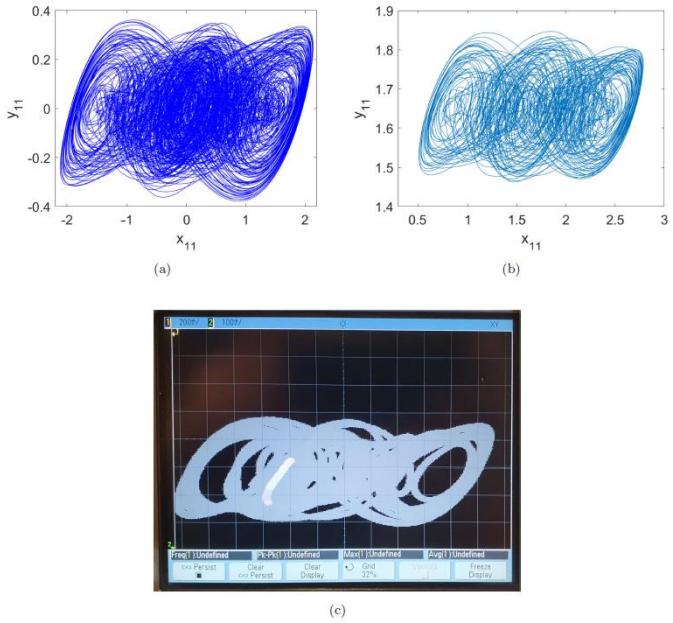


Fig. 20. Experimental results: (a) Numerical and (b) experimental attractors for α = 9; (c) oscilloscope trace (x-axis:200mV/div; y-axis: 100mV/div).

An extract of the implemented FW code is reported (variables definition and equations implementation):

```
#ifdef EQU1
double X11, X12, X21, X22, Y11, Y12,
Y21, Y22, Z11, Z21, Z12, Z22,
A, B, C0, C1, DT;
uint16_t XXX1, YYY1;
double XX11, XX21, XX12, XX22, YY11,
YY21, YY12, YY22, ZZ11, ZZ21, ZZ12, ZZ22;
#endif
```

```
#ifdef EQU1
  X11 = 1E-4;
  X12 = 0.2E-4;
  X21 = X12;
  X22 = 0.3E-5;
  Y11 = 1E-5;
  Y12 = 1E-5;
  Y21 = Y12;
  Y22 = 1E-5;
  Z11 = 1E-3;
  Z21 = 1E-3;
  Z12 = Z21;
  Z22 = 1E-3;
  A = 9;
  B = 14.286;
  C0 = ((double) (1))/((double) (16));
  C1 = -((double) (1))/((double) (6));
  DT = .001;
#endif
```

```
#ifdef EQU1
  XX11=X11 - A*DT*(C0*(X12*(X11*X21 + X21*X22) + X11*(X11*X11 + X12*X21)) - Y11 + C1*X11);
  XX21=X21 - A*DT*(C0*(X22*(X11*X21 + X21*X22) + X21*(X11*X11 + X12*X21)) - Y21 + C1*X21);
  XX12=X12 - A*DT*(C0*(X11*(X11*X12 + X12*X22) + X12*(X22*X22 + X12*X21)) - Y12 + C1*X12);
  XX22=X22 - A*DT*(C0*(X21*(X11*X12 + X12*X22) + X22*(X22*X22 + X12*X21)) - Y22 + C1*X22);
  YY11=Y11 + DT*(X11 - Y11 + Z11);
  YY21=Y21 + DT*(X21 - Y21 + Z21);
  YY12=Y12 + DT*(X12 - Y12 + Z12);
  YY22= Y22 + DT*(X22 - Y22 + Z22);
  ZZ11= Z11 - B*DT*Y11;
  ZZ21=Z21 - B*DT*Y21;
  ZZ12=Z12 - B*DT*Y12;
  ZZ22=Z22 - B*DT*Y22;
  X11=XX11;
  X21=XX21;
  X12=XX12;
  X22=XX22;
  Y11=YY11;
  Y21=YY21;
  Y12=YY12;
  Y22=YY22;
  Z11=ZZ11;
  Z21=ZZ21;
  Z12=ZZ12;
  Z22=ZZ22;

  XXX1=((uint16_t) (((X11+3)*4095)))/6;
  YYY1=((uint16_t) (((Y11+3)*4095)))/6;
#endif
```

The in silico implementation of the hyperneuron is based on STM32H723 microcontroller characterized by a 550 MHz main clock.

Also in this case the use of a timer ensures a constant sampling time, and the Digital to Analog Converter allows to produce an output voltage with a 12-bit resolution.

The algorithm to implement the hyperneuron is essentially based on the Euler discretization of Eqs. (20), thus obtaining a discrete map. It is adopted therefore an Euler integration method, included in the

loop interrupt routine of the timer. The use of optimized matrix manipulation routine is adopted in order to maintain the matrix form of the hyperneuron also in the algorithm. The clock has been adapted so that the output presents a spiking interval in the order of 10 ms, consistent with the biological observations.

The capability of the proposed approach to effectively implement high order hyperneurons essentially depends of the execution time τ_{ex} needed to complete an integration step. In Fig. 21, we report the value of τ_{ex} measured as a function of N. Therefore, on the basis of the execution time, the integration step size within the integration routine can be opportunely scaled so that the interspike interval observed in the output variables is in the desired range. The experimental bifurcation diagram with respect to I for N=2 is reported in Fig. 22 and it has been obtained calculating the interspike intervals from the output of the microcontroller. The diagram shows the same structure and bifurcation route of those obtained from numerical integration of the hyperneurons (Figs. 12 and 13).

As concerns the fault tolerance, the in silico hyperneuron confirms the capability of maintaining the spiking behavior in presence of a persistent noise, as shown by Fig. 23.

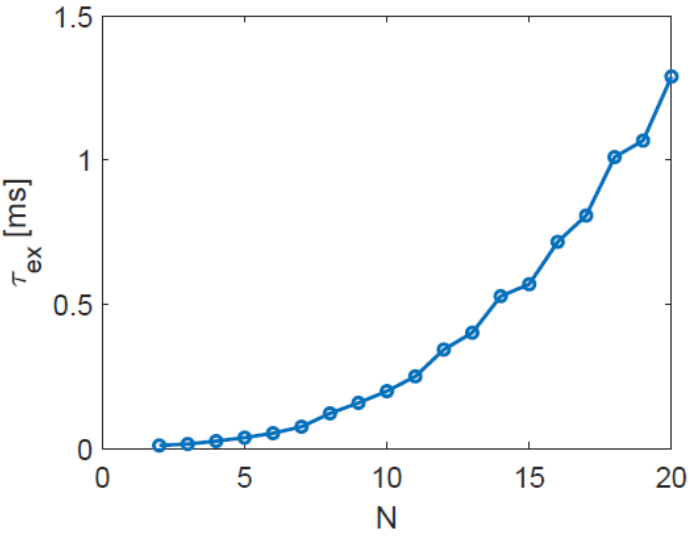


Fig. 21. Execution time τ_{ex} of a single integration step as a function of N.

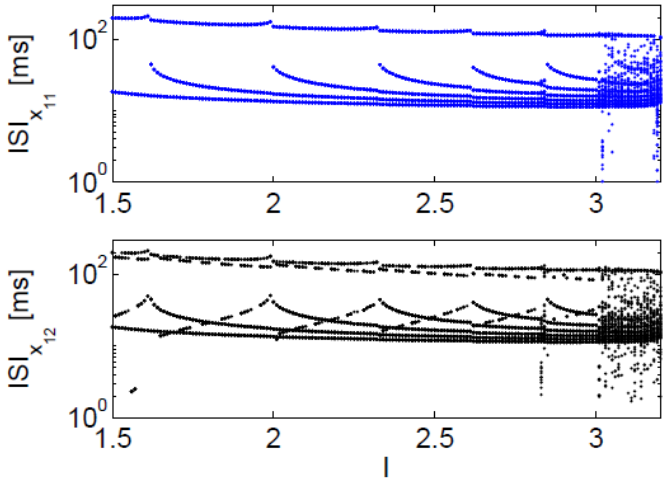


Fig. 22. Experimental bifurcation diagram with respect to parameter I for the Hindmarsh-Rose hyperneuron with N=2: interspike interval (ISI) calculated in ms for x11 (top) and x12 (bottom). Other parameters: a = 1, b = 3, c = 1, d = 5, r = 0.003 s = 4 and xR = −8/5.

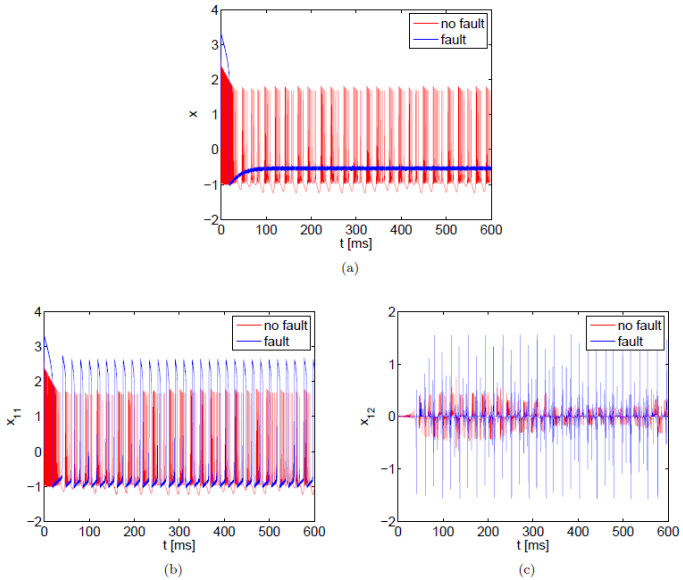


Fig. 23. Experimental robustness of the hyperneuron with respect to a persistent noise: (a) N = 1, (b)-(c) N=2. Other parameters: a=1, b=3, c=1, d=5, r = 0.003 s=4 and $x_R = -8/5$.

To implement the hyperneuron equations on the FW we used a matrix approach making use of the CMSIS DSP Software Library as shown in the following code:

```
arm_mat_mult_f32(&X, &X, &X2);
arm_mat_mult_f32(&X, &X2, &X3);
arm_mat_scale_f32(&X2, 3, &X32);
arm_mat_add_f32(&II, &Y, &A);
arm_mat_sub_f32(&A, &Z, &B);
arm_mat_add_f32(&B, &X32, &A);
arm_mat_sub_f32(&A, &X3, &B);
arm_mat_scale_f32(&B, DT, &A);
arm_mat_add_f32(&X, &A, &XX);

arm_mat_scale_f32(&X2, 5, &A);
arm_mat_sub_f32(&DIAG, &A, &B);
arm_mat_sub_f32(&B, &Y, &A);
arm_mat_scale_f32(&A, DT, &B);
arm_mat_add_f32(&B, &Y, &YY);

arm_mat_add_f32(&X, &CC, &A);
arm_mat_scale_f32(&A, 4, &B);
arm_mat_sub_f32(&B, &Z, &A);
arm_mat_scale_f32(&A, DTr, &B);
arm_mat_add_f32(&B, &Z, &ZZ);

arm_mat_mult_f32(&XX, &DIAG, &X);
arm_mat_mult_f32(&YY, &DIAG, &Y);
arm_mat_mult_f32(&ZZ, &DIAG, &Z);
```

6. Conclusion

Even if in silico neurons have been a wide subject of research since 1991 [Mahowald & Douglas, 1991] and more contributions have been presented in order to realize circuit emulating the electrical behavior of a single neuron, the topic is of wide interest today. The in silico technologies are continuously evolving as well as the microprocessor technology is proceeding towards lower costs and lower power consumption. Therefore it is appropriate to conceive networks of neurons realized on microcontroller in order to implement a high degree of parallelization.

In this contribution, the concepts of Hypersystems has been introduced focusing on the implementation of the hyper Chua circuit and hyperneurons from the Hindmarsh-Rose model, describing theoretically its main features and analyzing numerically and experimentally the emerging dynamical behavior.

Moreover, an experiment based on a new generation low cost microcontroller showing the

effectiveness of this approach in order to implement a large number of hyperneurons have been discussed.

It should be emphasized that the approach leading to the definition of the hyper Chua circuit allows the generation of $3\times n^2$ chaotic carriers. This allows to consider the hyper Chua circuit as a resilient Chua circuit. The increased number of state variables with chaotic behavior leads to more complex and nested basins of attraction within that of the original Chua's circuit. It is for this reason that the initial conditions can lead to a wide set of attractors, including that of the original Chua circuit. Notwithstanding the stronger dependence on initial conditions, synchronization strategies for two or more hyper Chua circuits can be outlined, following classical synchronization approaches. As concerns the Lyapunov spectrum, more than one positive Lyapunov exponents are retrieved for α in the chaotic region, with the largest being consistent with the positive Lyapunov exponent of the scalar Chua circuit. The proposed physical implementation has been based on low-cost microcontrollers, moreover its performance is

outstanding and allow to obtain at the same time more and more chaotic signals. Moreover, the possibility of using such device as a pseudo-random signal generator for secure communication apparatii, both in digital and analog applications, is here remarked. The hyper Chua circuit, in fact, represents a source of more chaotic encryption keys which can be generated through a single device. The set of initial conditions, in fact, representing the encryption key, leads to different chaotic behavior, even if all related to the original chaotic Chua's circuit.

The hyper Chua circuit can be investigated also to verify the occurrence of the same outstanding properties of the Chua circuit, like intermittency, stochastic resonance and so on, but generating a very high number of chaotic signals.

Finally, it is remarked that the approach to generate the hyper Chua circuit from a gene scalar Chua's circuit is general and can be applied on different continuous-time nonlinear dynamics thus representing a paradigm to generate higher-dimensional chaotic systems.

The fact that the hyperneurons defined from the Hindmarsh-Rose model allow the generation of $3 \times N^2$ chaotic spiking carriers makes them highly robust with respect to noise. The number of state variables within the hyper neuron leads to complex and nested basins of attraction, and, thus, to the generation of more coexisting chaotic attractors for the same set of system parameters.

In this study, we focused on the Hindmarsh-Rose model since it is considered a model conjugating the simplicity of the mathematical structure with the deep capability of mimicking observed neuronal behavior. Moreover, the Hindmarsh-Rose model assumed a fundamental relevance also in nonlinear dynamics as it is able to show a chaotic behavior. However, the definition of hyperneurons is independent from the specific choice of the neuron model and other, even more biologically plausible, models can be used to generate high order hyperneurons. It is important to stress that the hyperneurons guarantee also a spiking behavior in

presence of faults ore noise, differently to the more fragile dynamics of the scalar Hindmarsh-Rose model. The trade-off between the size of the hyperneuron and the computational performance is an open research problem. The proposed physical implementation based on low-cost microcontrollers allows for oustanding performance to get at the same time more and more spiking signals. This can be further improved by considering hardware devices with a higher degree of parallelization, such as networks of interconnected STM32 boards, thus proceeding forward to the realization of a dense environment of spiking neurons.

References

Arena P., Fortuna L., Frasca M., & Patanè L. [2009] “Learning anticipation via spiking networks: application to navigation control,” IEEE transactions on neural networks, 20(2), pp. 202–216.

Baden T., James B., Zimmermann M. J., Bartel P., Grijseels D., Euler T., Lagnado L. & Maravall M. [2018] “Spikeling: A low-cost hardware implementation of a spiking neuron for neuroscience teaching and outreach” PLoS biology, 16(10), art. no. e2006760.

Baptista M. S. [2021] “Chaos for communication” Nonlinear Dynamics 105 (2), pp. 1821–1841.

Bozhko D. V., Galumov G. K., Polovian A. I., Kolchanova S. M., Myrov V. O., Stelmakh V. A. & Schiöth H. B. [2021] “BCNNM: A Framework for in silico Neural Tissue Development Modeling” Frontiers in computational neuroscience, 14, art. no. 588224.

Bucolo M., Buscarino A., Fortuna L., & Gagliano S. [2022] “Multidimensional discrete chaotic maps,” Frontiers in Physics 10, art. no. 862376.

Hindmarsh J. L. & Rose R. M. [1984] “A model of neuronal bursting using three coupled first order

differential equations," Proceedings of the Royal society of London. Series B. Biological sciences, 221(1222), pp. 87–102.

Litt B. [2015] "Engineering the next generation of brain scientists," Neuron, 86(1), pp. 16–20.

Mahowald M., & Douglas R. [1991] "A silicon neuron," Nature, 354(6354), pp. 515–518.

Markram H. [2006] "The blue brain project," Nature Reviews Neuroscience, 7(2), pp. 153–160.

Mitra S. K., & Kuo Y. [2006] Digital signal processing: a computer-based approach (Vol. 2). New York: McGraw-Hill.

Navickas Z., Smidtaite R., Vainoras A., & Ragulskis M. [2011] "The logistic map of matrices," Discrete & Continuous Dynamical Systems-B 16(3), p. 927.

Navickas Z., Ragulskis M., Vainoras A., & Smidtaite, R. [2012] "The explosive divergence in iterative mapsof matrices," Communications in Nonlinear Science and Numerical Simulation 17(11), pp. 4430–4438.